Case-Based Approaches for Diagnosing Multiple Disorders

Martin Atzmueller
Joachim Baumeister
Frank Puppe
Department of Computer Science,
University of Wuerzburg, Germany
{atzmueller, baumeister, puppe}@informatik.uni-wuerzburg.de

Wenqi Shi
John A. Barnden
School of Computer Science,
University of Birmingham,
United Kingdom
{w.shi, j.a.barnden}@cs.bham.ac.uk

Abstract

Multiple disorders are a daily problem in medical diagnosis and treatment. However, case-based diagnosis handling multiple disorders is still a challenging task. Although case-based reasoning has been applied successfully in medical domains, the handling of multiple disorders is often not sufficient. In this paper, we investigate three improved approaches for handling multiple disorders in case-based diagnosis. The methods are evaluated with a medical case base taken from a real world application. The context of our work is to supplement a medical documentation and consultation system by CBR techniques for experience management.

Introduction

The medical domain, and especially medical decision making, has attracted a lot of attention in AI research. Multiple disorders are a frequently occurring problem in daily medical diagnosis and treatment. Methods, which use complex models, e.g., causal or Bayesian networks, for diagnosing multiple disorders, e.g., the Heart Disease Program (Long, Naimi, & Criscitiello 1994), or the HEPAR$\prod$ system (Onisko, Druzdzel, & Wasyluk 2000) mainly rely on manual knowledge acquisition by experts. In contrast to these, the case-based reasoning methodology uses previous experience for current problem solving. So, CBR reduces the costs of knowledge acquisition and maintenance, and therefore has become quite popular in experience rich domains, e.g., in the medical domain.

However, handling multiple disorders is a major problem: In our domain of sonography the examination considers several partially disjunctive subdomains, e.g., liver or kidney, which results in multiple disorders, i.e., most cases contain multiple diagnoses. Our context is a medical documentation and consultation system which we extend with a component for experience management. Thus, we want to retrieve experiences such as explanations for a query case based on the presented similarity to former cases and additional information contained in these. For example, there is additional information about therapy, complications, prognosis or the treating physician as contact person for special questions. We use case-based reasoning to obtain this information.

Due to the combinatorial rules the chance of reusing a case with even 3 independent diagnoses from say 100 alternatives is roughly just one to one million. However, that many cases are rarely available. Thus, considering the fact, that our medical case base contained about 7 disorders per case on average, the naive case-based reasoning approach performed very poor. So, using this medical case base taken from a real world application and applying the naive case-based method we were only able to solve about 3% of the cases.

In this paper, we present three approaches, that significantly improve the handling of multiple disorders in a case-based diagnosis task. In the following sections we separately describe these approaches, that are evaluated on a medical case base in the subsequent section. We conclude the paper with a discussion of the presented work and pointers to promising work in the future.

Case-Based Diagnosis with Multiple Disorders

Firstly, we define necessary notions concerning our knowledge representation schema as follows:

Let Ω_D be the set of all diagnoses and Ω_A the set of all attributes. To each attribute $a \in \Omega_A$ a range $dom(a)$ of values is assigned. Further we assume Ω_F to be the (universal) set of findings $(a = v)$, where $a \in \Omega_A$ is an attribute and $v \in dom(a)$ is an assignable value. Let CB be the case base containing all available cases that have been solved previously. A case $c \in CB$ is defined as a tuple

$$c = (\mathcal{F}_c, \mathcal{D}_c, I_c), \qquad (1)$$

where $\mathcal{F}_c \subseteq \Omega_F$ is the set of findings observed in the case c. In CBR-problems these findings are commonly called *problem description*. The set $\mathcal{D}_c \subseteq \Omega_D$ is the set of diagnoses describing the *solution* for this case. I_c contains additional information like therapy advices or prognostic hints.

To assess the similarity between a query case c and another case c', we apply a similarity measure, which is an adaptation of the *Hamming distance*, with attribute weights and partial similarities between attribute values, if available. The naive case-based method, also referred to in the following as *standard CBR*, uses this plain similarity measure to retrieve a most similar case to the query case.

In the following, we present three approaches for case-based diagnosis handling multiple disorders.

The Set-Covering Approach

In this section we will briefly describe the use of set-covering models for the case-based diagnosis task.

Set-covering models (Reggia, Nau, & Wang 1983) are a prominent approach for handling multiple disorders in diagnostic reasoning. A *set-covering model* contains set-covering relations, that describe relations like: *A diagnosis D predicts that the finding F is observed in $freq_{F,D}$ percent of all cases."* We denote a set-covering relation r by

$$r = D \rightarrow F\left[freq_{D,F}\right],$$

where $D \in \Omega_D$ and $F \in \Omega_F$. As described in (Baumeister, Seipel, & Puppe 2003) set-covering models can be augmented for handling case-based knowledge like similarities and weights. In (Baumeister, Atzmueller, & Puppe 2002) inductive methods are presented for learning set-covering models, similarities, and weights. The set-covering models are constructed from diagnostic profiles which contain typical findings for a diagnosis. We additionally apply dependency analysis utilizing the χ^2-*test for independence* to isolate the findings which are dependent on the diagnosis, as described in (Atzmueller, Baumeister, & Puppe 2003). This results in more compact set-covering models with an improved quality, since the set of findings of a set-covering model is reduced to the set of typical findings, which are dependent on the diagnosis.

Given a query case with a new problem description, set-covering models are used to generate a hypothesis $H \subseteq \Omega_D$, i.e., a set of diagnoses representing a (reasonable) explanation for the observed problem description. For case-based retrieval, we compute the k best hypotheses for a given problem description. Systematic approaches for the generation of set-covering hypotheses were introduced, e.g., in (Peng & Reggia 1990). Based on these hypotheses we generate a set of candidate cases: For each hypothesis H we combine cases $c \in CB$, so that their joined solution parts receive a high coverage with the computed hypothesis H.

In the following, we discuss the concept of candidate cases in more detail.

Definition 1 (Candidate Case). *A candidate case*

$$ca = (\mathcal{F}_{ca}, \mathcal{D}_{ca}, C_{ca})$$

consists of a set of subcases $C_{ca} \subseteq CB$ of a given case base CB and a set of diagnoses $D_{ca} \subseteq \Omega_D$. The set of findings $\mathcal{F}_{ca} \subseteq \Omega_F$ of the candidate case is created by joining the findings of the subcases as described below. $\mathcal{D}_{ca}$ is defined as the union of the subcases' diagnoses, i.e., $\mathcal{D}_{ca} = \bigcup_{c \in C_{ca}} \mathcal{D}_c$.

To create candidate cases we have to combine the problem descriptions included in the subcases of the candidate case. However, conflicts can arise if two cases contain different values for the same attribute. For the creation of candidate cases we apply the following procedure:

1. Joining sets of the solutions contained in the subcases.

2. Likewise, joining their problem descriptions:
   $\mathcal{F}_{ca} = \bigcup_{c \in C_{ca}} \mathcal{F}_c$

3. Conflict resolution: If F_{ca} contains more than one value for an attribute:

(a) We use background knowledge if available (see below).

(b) Otherwise we try to select an attribute value based on the finding in the query case:

   i. We choose the value contained in the query case if included in one subcase.

   ii. Alternatively, we select the value which is most similar to the value included in the query case.

   iii. Otherwise, if the value is not included in the problem description of the query case, then we randomly pick a value from the set of the attribute's values of the subcases.

It is obvious, that we want to obtain a candidate case explaining the findings in the query case. Thus, if no additional knowledge is available in the conflict resolution step, we base the decision of choosing an attribute value on the value which is contained in the query case. If the query case does not contain the conflicting attribute, then we do not know which value is relevant for the candidate case. Therefore we pick one randomly.

In addition, the conflict resolution step above can be improved using additional knowledge, i.e., *abnormalities*. Abnormality knowledge specifies which findings represent a normal or an abnormal state of their corresponding attribute (e.g., *pain=none* is normal, whereas *pain=high* is abnormal). If abnormalities are defined, then we take the value with the highest abnormality. This heuristic is motivated by the following example: If a patient has two (independent) diagnoses, then it is reasonable that the more severe finding will be observed.

The generation of candidate cases can be restricted by the number of included cases.

Definition 2 (Restricted Candidate Case). *We define a* restricted candidate case $ca_{|n|}$ *of size n as follows*

$$ca_{|n|} = (\mathcal{F}_{ca}, \mathcal{D}_{ca}, C_{ca}),$$

where $\mathcal{F}_{ca} \subset \Omega_F$, $\mathcal{D}_{ca} \subseteq \Omega_D$, and $C_{ca} \subseteq CB$. A restricted candidate case $ca_{|n|}$ of size n contains not more than n subcases, i.e., $|C_{ca}| = n$.

By generating restricted candidate cases with a fixed n, we can limit the number of joined cases, i.e., the number of subcases included in a candidate case. The quality of a generated candidate case ca is determined by computing the intersection coverage between the joined diagnoses $\mathcal{D}_{ca}$ of the candidate case and the diagnoses of the hypothesis H.

Definition 3 (Intersection Coverage). *For a candidate case the* intersection coverage *reflects the degree of coverage between its set of diagnoses $\mathcal{D}_{ca} \subseteq \Omega_D$ and a given hypothesis $H \subseteq \Omega_D$. The intersection coverage ic is defined by*

$$ic(D_{ca}, H) = \frac{1}{2} \cdot \left(\frac{|\mathcal{D}_{ca} \cap H|}{|H|} + \frac{|\mathcal{D}_{ca} \cap H|}{|\mathcal{D}_{ca}|} \right),$$

where $\mathcal{D}_{ca}$ is the set of diagnoses contained in the candidate case, and H is the set of diagnoses representing a hypothesis.

A candidate case $ca = (\mathcal{F}_{ca}, \mathcal{D}_{ca}, C_{ca})$ is *optimal* with respect to a given hypothesis H, iff $ic(\mathcal{D}_{ca}, H) = 1$. It is easy to see, that if we rank all candidate cases according to the intersection coverage, then the first m candidate cases represent the best coverage of the given hypothesis. Therefore, we compare the new case with the first m candidate cases for retrieving a suitable solution in the case-based diagnosis task. We summarize the presented approach by the following outline:

1. We use the set-covering models to compute the k best hypotheses. If required, set-covering models can be inductively learned.

2. Given the k hypotheses, we generate a set of *candidate cases*. The generation process is restricted by a threshold value n limiting the maximum number of joined cases.

3. We rank the generated candidate cases given their corresponding hypotheses using the intersection coverage (ic) metric.

4. The m best candidate cases are returned, i.e., the cases with a maximum intersection coverage for a given hypothesis.

5. The best matching candidate cases are used for case comparison.

Related Work In the past, several approaches for the combination of case-based reasoning and abductive or set-covering models have been presented. The systems CASEY (Koton 1988) and ADAPtER (Portinale & Torasso 1995) are prominent examples. These systems apply abductive knowledge for the adaptation of old cases, and give a verbose explanation for the adaptation. In our work, we use the reverse approach, when using abductive reasoning for guiding the search of how to combine cases.

The Compositional Case Adaptation Approach

Compositional case adaptation was originally developed for configuration tasks. First it decomposes problems into sub-problems. Then, it retrieves those sub-problems in the case base and combines different parts of the solution of similar cases. This results in the solution of a new problem (Wilke & Bergmann 1998). The main idea of compositional case adaptation is to retrieve a group of the most similar cases instead of only reusing the most similar case. Then, a compositional strategy is applied to adapt the solutions of the group of most similar cases to get the final solution.

The application of the compositional case adaptation method is motivated by the multiple disorder situation. In a multiple disorder case base a high variance of disorder combinations is possible, which cause a high variance of problem descriptions. Then, the retrieved case which is most similar to the query case might only cover parts of the solutions of the query case. Therefore, the combination of multiple case solutions should generate more desirable results. However, we do not perform decomposition before composition like (Wilke & Bergmann 1998). In our situation, there are hundreds of observed findings available and

without help from an expert, they cannot explicitly be divided automatically into different groups corresponding to different diagnoses.

We assume that in the multiple disorder situation not all the diagnoses in the solutions of the k most similar cases will be suggested as the current diagnosis. Only the diagnoses with a high occurrence among the k most similar cases have a high probability to appear in the final solution of the query case. The underlying meaning of this assumption is that those findings in the k similar cases which are similar to the query case will contribute to those desired diagnoses with a high occurrence in the similar case solutions.

At the same time, we assume that the more similar the retrieved case is to the query case, the higher the probability that the diagnoses in this retrieved case will appear in the final solution. Thus, we add weights to the occurrences of diagnoses in the set of retrieved cases. The weights are determined according to the similarity between the retrieved case the diagnoses occur in and the query case.

Definition 4 (Similarity-Weighted Frequency). *The similarity-weighted frequency of a diagnosis D is the weighted frequency of D within the* k *most similar cases.*

$$\mathrm{FQC}(D) = \sum_{i=1}^{k} W_i \delta(C_i, D),$$

where $D \in \Omega_D$ is a diagnosis; $C_i \in CB$ is the ith *most similar case to the query case. $\delta(C_i, D)$ is 1, if D occurs in the ith most similar case C_i, and 0 otherwise. W_i represents the associated weight, where we used the squared relative similarity between C_i and the query case. Therefore, the weight is proportional to the similarity.*

After calculating the similarity-weighted frequency of the diagnoses appearing in the k most similar cases, we generate a candidate solution defined as follows:

Definition 5 (Candidate Solution). *A candidate solution*
$$CS = \{D \in \Omega_D : FQC(D) \geq \epsilon\},$$
is the set of diagnoses with a similarity-weighted frequency above a dynamic threshold $\epsilon = \alpha \cdot \max_{D \in \Omega_D} FQC(D)$. This threshold has a linear relationship with the frequency value of the diagnosis occurring most frequently in the k *most similar cases. We used the default value $\alpha = 0.55$.*

Thus, we only include a diagnosis of the k most similar cases into the candidate solution, if the similarity-weighted frequency of the diagnosis is greater or equal than the threshold defined by ϵ. The diagnoses that do not appear in the k most similar cases are not considered. We summarize our compositional adaptation approach as follows:

1. Retrieve the k most similar cases to the query case.

2. Compute the similarity-weighted frequency for each diagnosis appearing in the k most similar cases.

3. Create a candidate solution as the hypothesized solution for the query case.

4. Generate a candidate case as introduced in Definition 1. The subcases consist of the k most similar cases. The contained diagnoses are then defined as the set of diagnoses, that are included in the candidate solution.

Related Work Compositional case adaptation has been investigated in the medical domain in recent years. In (Vorobieva, Gierl, & Schmidt 2002), when calculating the dose for an endocrinopathic patient, the most similar cases with the best therapy results are taken into account. The system TeCoMed (Schmidt & Gierl 2001) splits the set of the most similar cases into two sets according to whether a warning was appropriate to predict an influenza wave. Then, the solution is adapted according to a voting between these two sets. Our approach computes a weighted frequency of each diagnosis in the similar cases. Using this measure, we then select these diagnoses, which are accounted for best by the set of k most similar cases.

The Partition Class Approach

The partition class approach takes advantage of the fact that many domains can be divided into rather independent subdomains, e.g., in the medical domain partitions are representing the different organ systems. For this approach, the expert provides *partition class* knowledge describing how to divide the set of diagnoses and attributes into partially disjunctive subsets, i.e., partitions. These subsets correspond to certain problem areas of the application domain. For example, in the medical domain of sonography, we have subsets corresponding to problem areas like *liver*, *pancreas*, *kidney*, *stomach*, and *intestine*.

The idea of using partition class knowledge is to split the original case base into several case bases containing partial, decomposed cases corresponding to different partition classes. This decomposition is static, and therefore can be precomputed at compile-time.

Definition 6 (Partition Class). *A partition class pc is defined as a tuple*

$$pc = (\mathcal{D}_{pc}, \mathcal{F}_{pc}),$$

where $\mathcal{D}_{pc} \subseteq \Omega_D$ and $\mathcal{F}_{pc} \subseteq \Omega_F$. For a partition class pc the sets $\mathcal{D}_{pc}$ and $\mathcal{F}_{pc}$ refer to the same problem area of the application domain. All diagnoses are covered by the different partition classes pc_i, i.e., $\Omega_D = \bigcup_i \mathcal{D}_{pc_i}$.

According to the given subsets of attributes and diagnoses of each partition class we can split cases into subcases, i.e., partial cases, where a case is divided by forming sets of attributes and diagnoses for each partition class. Since the partition classes are only partially disjunctive we apply a refinement step to the partial cases to prune the irrelevant findings. This refinement step is motivated by the fact, that findings which occur in many partition classes and which are not accounted for by a diagnosis may be too general to be meaningful in the partial case.

Let $\mathcal{PC}_e$ denote the set of partition classes the element e is contained in, where e can be a diagnosis or an attribute. Then, we check for each attribute $a \in \Omega_A$ contained in the partial case if there exists a diagnosis $D \in \Omega_D$ of the partial case such that $\mathcal{PC}_a \subseteq \mathcal{PC}_D$. If the check fails, then we apply the χ^2-*test for indecence* for the attribute and each diagnosis of the partitioned case. If the attribute is not dependent on any diagnosis, then we remove the attribute.

A partial case may even contain only a single diagnosis. However, we have to make sure that the decomposed cases are still meaningful. Firstly, they must contain a minimum number of attributes to guarantee a certain support for the diagnoses contained in the case. Secondly, cases should still contain diagnostic information, i.e., they should contain at least one diagnosis. If a generated subcase does not fulfill these requirements, then it is not considered and removed.

By recombining partial cases to candidate cases as introduced in Definition 1, we find possible solutions, i.e., most similar cases for the query case. In summary, we first decompose the query case into several subcases. Then, we apply CBR for each query subcase in order to retrieve the most similar partial cases, and finally recombine the retrieved partial cases. We outline the process in the following:

1. Precomputed: Divide the original case base into several 'partition class' case bases.

2. Decompose the query case into a set of partial cases according to the given partition classes.

3. For each partial case, apply standard CBR using the respective partitioned case base storing the set of most similar cases for each partial case in the result sets.

4. Generate candidate cases by combining the result sets: Construct a set of subcases, for which one case from each result set is drawn. Create a candidate case using these subcases.

5. Use the generated candidate cases for case comparison.

Related Work In the literature there exist several approaches based on case decomposition techniques. (Watson & Perera 1998) presented a system, which retrieves decomposed cases from hierarchically structured cases, and adapts multiple subcases into a solution. The *CADSYN* (Maher & Zhang 1991) system recombines sub-problems formed of decomposed cases into a solution taking the contexts of such partial cases into account. (Smyth, Keane, & Cunningham 2001) presented case-based reasoning on hierarchical case bases, which allows complex problems to be solved by reusing multiple cases at various levels of abstraction.

These approaches mainly apply techniques from case-based planning, i.e., utilizing hierarchical relations either in the CBR retrieve or reuse step, or both. In contrast, our candidate case generation strategy using partition class knowledge is a knowledge intensive approach, which uses the explicit partitioning information as background knowledge to decompose cases.

Evaluation

For the evaluation we applied cases from the knowledge-based documentation and consultation system for sonography SONOCONSULT (Huettig *et al.* 2004). The quality of the derived diagnoses is very good as checked by medical experts in a medical evaluation (cf. (Huettig *et al.* 2004)). Our evaluation case base consists of 744 cases. The case base contains an overall number of 221 diagnoses and 556 attributes, with a mean $M_D = 6.71 \pm 04.4$ of diagnoses per case and a mean $M_F = 48.93 \pm 17.9$ of relevant findings per case. To evaluate the three presented approaches, we used a leave-one-out cross-validation method.

We adopted the intersection accuracy, proposed by (Thompson & Mooney 1994), as a measure for multiple disorder problems. Intersection accuracy is derived by the two standard measures: *sensitivity* and *precision*.

Definition 7 (Intersection Accuracy). *The intersection accuracy $\mathcal{IA}(c, c')$ is defined as*

$$\mathcal{IA}(c, c') = \frac{1}{2} \cdot \left(\frac{|\mathcal{D}_c \cap \mathcal{D}_{c'}|}{|\mathcal{D}_c|} + \frac{|\mathcal{D}_c \cap \mathcal{D}_{c'}|}{|\mathcal{D}_{c'}|} \right) \quad (2)$$

where c and c' are two cases, $\mathcal{D}_c \subseteq \Omega_D$ is the set of diagnoses of case c, and $\mathcal{D}_{c'} \subseteq \Omega_D$ is the set of diagnoses contained in case c' likewise.

Essentially, the intersection accuracy metric is equal to the intersection coverage defined in Definition 3. However, since the semantics are slightly different, we define intersection accuracy as a metric for comparing solutions of cases while intersection coverage relates a solution of a case and a hypothesis.

Experimental Results and Discussion Initially, we performed an analysis to determine, how many cases could be solved by a "perfect" case-based reasoning method. We compared each query case in the case base with the case that solved the query case best, i.e. using the case with the highest intersection accuracy of its diagnoses with the diagnoses of the query case. It turned out, that in principle all 744 cases are solvable with a mean intersection accuracy of about 90%. However, due to the multiple disorder characteristic of our case base, intersection accuracy did not necessarily correlate with similarity. This can be explained by the special characteristic of our case base, which shows a high count of diagnosis combinations per case and rare repetitions of diagnosis combinations in the case base. The high variance of diagnosis combinations on the other hand causes a high variance of possible problem descriptions. Thus, it is possible to assess a high intersection accuracy between two cases with a low similarity, because the diagnoses which are not common to both cases can distinguish the respective problem descriptions quite significantly.

For the different approaches, we employ two rating measures to assess a case as solved, or not solvable. For the set-covering and the partition class method, we employ a similarity threshold $\mathcal{T}_{CBR}$. We say that a compare case c' *solves* a query case c, iff $sim(c, c') \geq \mathcal{T}_{CBR}$, i.e., if the cases are sufficiently similar. Cases below this threshold were withdrawn and marked as *not solvable*. The compositional case adaptation approach generates a set of suggested diagnoses, which are returned as the solution. These are packaged into a candidate case, together with the set of cases the diagnoses were derived from. Let CS denote the candidate solution, i.e., hypothesized diagnoses, of the k most similar cases, and D_k denote all diagnoses of the k most similar cases. We say that a case is solved, if the maximum frequency of a diagnosis $D \in CS$ exceeds a certain minimal support threshold, and if $\sum_{D \in CS} freq(D) / \sum_{D \in D_k} freq(D)$ exceeds a certain threshold as well. We present the results of comparing the three approaches in Table 1.

744 Cases from the SonoConsult Case Base			
Approach	*rating threshold(s)*	*solved cases*	*mean acc*
Standard CBR	0.90	20 (3%)	0.66
Set-Covering	0.43	502 (67%)	0.70
Comp. Adaptation	3, 0.28	534 (72%)	0.70
Partition Class	0.30	624 (84%)	0.73

Table 1: Comparison of the approaches, using 744 cases

The results in the first line show, that the standard CBR method is performing poor for cases with multiple disorders. Standard CBR utilizing no additional background knowledge can only solve 3% of the cases in the case base, which is obviously insufficient. The set-covering approach performs promising, since it returns 502 cases as solved, i.e., it can solve about 67% of the cases in the case base with a mean accuracy of 70%. This means a significant improvement compared to the standard CBR method performed on cases with multiple disorders. The compositional adaptation method solves 534, i.e., 72% of the cases in the case base, with a mean accuracy of 70%. This demonstrates the relevance of this method in the multiple disorder situation. The compositional adaptation method is slightly better than the set-covering approach. This is probably due to two issues: The set-covering approach returns candidate cases in terms of cases with all their solutions, and therefore no sophisticated adaptation step is applied. Secondly, refining the set-covering models based on quality measures would probably increase their predictive accuracy.

The knowledge-intensive method using partition class knowledge performs best, since it can solve about 84% of the cases in the case base with a mean accuracy of 73%. In summary, the partition class based strategy can deal with the multiple disorder problem quite well.

However, one of the the main differences between the three method needs always to be considered: The partition class strategy uses special background knowledge, while the other approaches do not require background knowledge, and so can be applied in arbitrary situations, e.g., when no partitioning knowledge is available.

Conclusion and Outlook

Our context was to supplement a medical documentation and consultation system by CBR techniques to enable the extended retrieval of experiences. Here, naive CBR showed to be not appropriate for handling multiple disorders. In this paper, we presented three improved approaches for handling multiple disorders in case-based reasoning. One approach uses set-covering knowledge to create hypotheses and candidate cases. The second approach also reuses existing cases to explain the query case. It applies a compositional adaptation method to generate a set of suggested diagnosis. The third approach uses special background knowledge, i.e., partition class knowledge, to decompose and recombine cases and produced very promising results.

All three approaches improved the handling of multiple disorders significantly, but among these, the partition class

method performs best. For the partition class approach, the decomposition of cases can be precomputed in advance at compile-time, which is more efficient concerning run-time than the two other methods. The advantage of the set-covering approach and the compositional case adaptation approach is their simple applicability, since they do not require additional background knowledge.

In the future, we are interested in trying to apply refined partition class knowledge and learn partition classes automatically. Formal concept analysis for CBR, as described in (Díaz-Agudo, Gervás, & González-Calero 2003) seems a promising direction for this. Furthermore, concerning the construction of set-covering models, fine-tuning the knowledge elements seems very interesting as well. Looking at the combine step of candidate cases more closely, techniques that evaluate the quality of candidate cases may prove essential to ensure a certain quality of the generated candidate cases. So, methods which can evaluate combinations of cases, e.g., using Bayesian networks (Hennessy, Buchanan, & Rosenberg 2002), are a promising direction. This also holds for the compositional case adaptation methods to verify the generated candidate solutions more extensively.

An integration of methods, which do not need additional background knowledge, i.e., the set-covering method and the compositional case adaptation approach, with the partition class strategy could be another worthwhile approach. Cases could be decomposed using partitioning knowledge in a first step. Then, the other methods could be applied in a second step to produce solutions for the subproblems which can then be recombined.

References

Atzmueller, M.; Baumeister, J.; and Puppe, F. 2003. Inductive Learning of Simple Diagnostic Scores. In *Medical Data Analysis*, volume 2868 of *LNCS*, 23–30. Berlin, Germany: Springer. Proceedings of the 4th International Symposium on Medical Data Analysis (ISMDA 2003).

Baumeister, J.; Atzmueller, M.; and Puppe, F. 2002. Inductive Learning for Case-Based Diagnosis with Multiple Faults. In *Advances in Case-Based Reasoning*, volume 2416 of *LNAI*, 28–42. Proceedings of the 6th European Conference on Case-Based Reasoning (ECCBR 2002).

Baumeister, J.; Seipel, D.; and Puppe, F. 2003. Incremental Development of Diagnostic Set-Covering Models with Therapy Effects. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 11(Suppl. Issue):25–49.

Díaz-Agudo, B.; Gervás, P.; and González-Calero, P. A. 2003. Adaptation Guided Retrieval Based on Formal Concept Analysis. In *Case-Based Reasoning Research and Development*, volume 2689 of *LNAI*. Springer. 131–145. Proceedings of the 5th International Conference on Case-Based Reasoning (ICCBR 2003).

Hennessy, D. N.; Buchanan, B. G.; and Rosenberg, J. M. 2002. Bayesian Case Reconstruction. In *Advances in Case-Based Reasoning*, volume 2416 of *LNAI*. Berlin: Springer. 148–158. Proceedings of the 6th European Conference on Case-Based Reasoning (ECCBR-2002).

Huettig, M.; Buscher, G.; Menzel, T.; Scheppach, W.; Puppe, F.; and Buscher, H.-P. 2004. A Diagnostic Expert System for Structured Reports, Quality Assessment, and Training of Residents in Sonography. *Medizinische Klinik* In press.

Koton, P. 1988. Reasoning about Evidence in Causal Explanations. In *Proceedings of the Seventh National Conference on Artificial Intelligence*, 256–261.

Long, W. J.; Naimi, S.; and Criscitiello, M. G. 1994. Evaluation of a New Method for Cardiovascular Reasoning. *Journal of the American Medical Informatics Association* 1:127–141.

Maher, M., and Zhang, D. 1991. CADSYN: Using Case and Decomposition Knowledge for Design Synthesis. *Artificial intelligence in Design*.

Onisko, A.; Druzdzel, M. J.; and Wasyluk, H. 2000. Extension of the HEPAR$\prod$ Model to Multiple-Disorder Diagnosis. In Klopotek, M.; Michalewicz, M.; and Wierzchon, S. T., eds., *Intelligent Information Systems*, 303–313. Physica-Verlag.

Peng, Y., and Reggia, J. A. 1990. *Abductive Inference Models for Diagnostic Problem-Solving*. Berlin: Springer.

Portinale, L., and Torasso, P. 1995. ADAPtER: An Integrated Diagnostic System Combining Case-Based and Abductive Reasoning. In *Proceedings of the ICCBR 1995*, 277–288.

Reggia, J. A.; Nau, D. S.; and Wang, P. Y. 1983. Diagnostic Expert Systems Based on a Set Covering Model. *Journal of Man-Machine Studies* 19(5):437–460.

Schmidt, R., and Gierl, L. 2001. Case-Based Reasoning for Antibiotics Therapy Advice: an Investigation of Retrieval Algorithms. *Artificial Intelligence in Medicine* 23:171–186.

Smyth, B.; Keane, M. T.; and Cunningham, P. 2001. Hierarchical Case-Based Reasoning Integrating Case-Based and Decompositional Problem-Solving Techniques for Plant-Control Software Design . *Transactions on Knowledge and Data Engineering* 13(5):793–812.

Thompson, C. A., and Mooney, R. J. 1994. Inductive Learning for Abductive Diagnosis. In *Proceedings of the AAAI-94, Vol. 1*, 664–669.

Vorobieva, O.; Gierl, L.; and Schmidt, R. 2002. Case-based Adaptations in Medicine Focusing on Hypothyroidism. In *7th UKCBR*.

Watson, I., and Perera, S. 1998. A Hierarchical Case Representation using Context Guided Retrieval. *The Knowledge Based Systems Journal Vol. 11* 285–292.

Wilke, W., and Bergmann, R. 1998. Techniques and Knowledge used for Adaptation during Case-Based Problem Solving. In *Tasks and Methods in Applied Artificial Intelligence*, volume 1416 of *LNCS*. Berlin: Springer. 497–506. Proceedings of the IEA-98-AIE Conference (Vol. 2).