Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
D-6750 Kaiserslautern

SEKI - REPORT

Goal-Driven Similarity Assessment

Stefan Wess, Dietmar Janetzko & Erica Melis

SEKI Report SR-92-05 (SFB)

GOAL-DRIVEN SIMILARITY ASSESSMENT*

Stefan Wess
Dept. of Computer Science
University of Kaiserslautern
D–6750 Kaiserslautern

Dietmar Janetzko
Dept. of Cognitive Science
University of Freiburg
D–7800 Freiburg

Erica Melis
Dept. of Computer Science
University of Saarbrücken
D–6600 Saarbrücken

Abstract

While most approaches to similarity assessment are oblivious of knowledge and goals, there is ample evidence that these elements of problem solving play an important role in similarity judgements. This paper is concerned with an approach for integrating assessment of similarity into a framework of problem solving that embodies central notions of problem solving like goals, knowledge and learning.

We review empirical findings that unravel characteristics of similarity assessment most of which have not been covered by purely syntactic models of similarity. A formal account of similarity assessment that allows for the integration of central ideas of problem solving is developed. Given a goal and a domain theory, an appropriate perspective is taken that brings into focus only goal-relevant features of a problem description as input to similarity assessment.

1 Introduction

In recent years, there has been an upsurge of interest in case-based reasoning (CBR), i.e. reasoning techniques that are based on the use and reuse of previous problem solving experience [Kol91]. One of the key issues of case-based reasoning is the question how a previous case, i.e. a *source*, is selected given a current case, i.e. a *target*. This retrieval step calls for estimating similarity between source and target cases. The majority of previous approaches to similarity assessment resort to measures of similarity that have been developed within the province of categorization and clustering (e.g. in biology [Dic45]), but not within the realm of problem solving. These approaches have been termed *syntactic*, as they confine similarity assessment to the objects given in the *problem description* and refrain from using purposes or goals. In contrast, these factors on the side of the *problem-solver* are at the heart of the so-called *pragmatic* approaches (e.g. [Hol85]) to similarity.

We take the view of similarity as a genuine part of problem solving that is influenced both by syntactic characteristics of similarity judgement, e.g. number of common features, and by pragmatic factors, e.g. goals. The aim of this article is to develop a model that links pragmatic and syntactic approaches to similarity. The model we propose does not give priority to any of the two accounts on similarity assessment. It is, however, based on the assumption that similarity assessment is a *goal-driven process*. Reduced to its kernel, our model starts with a pragmatic account using a problem solving *goal* and a *domain theory*. Following this, a *perspective* (*cf.* [Str91]) is developed under which similarity of a given object to other objects can be computed in a syntactic manner.

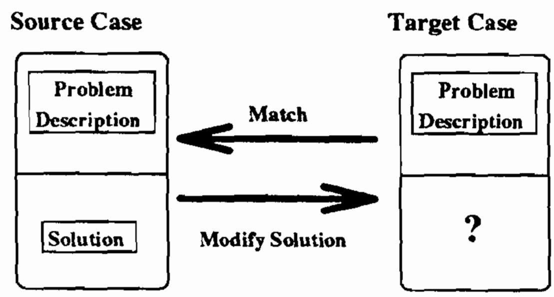

Figure 1: Basic steps of CBR

In what follows, we first discuss the characteristics of similarity that bear on problem solving, especially case-based reasoning and analogical reasoning. This is done in agreement with cognitive science findings that give rise to an enlargement of approaches to similarity developed

*This research was supported by the "Deutsche Forschungsgemeinschaft" (DFG), "Sonderforschungsbereich" (SFB) 314: "Artificial Intelligence and Knowledge-Based Systems", projects X9 and D3.

for the purpose of classification tasks. Second, we give a formal account of an approach to similarity that captures these characteristics of similarity.

2 Characteristics of similarity assessment

Whenever measures of similarity tailored to categorization tasks are used in problem-solving tasks, e.g. case-based reasoning, a general blindness of these measures towards goals, knowledge and learning is to be complained. To identify crucial characteristics of similarity in problem solving we review some cognitive science findings on similarity. The characteristics of similarity assessment in problem solving discussed below are by no means exclusive or complete. They highlight, however, characteristics of similarity in problem solving that are usually neglected in the above mentioned syntactic approaches to similarity.

Similarity as a goal–driven process

Similarity assessment has been shown to be strongly influenced by goals ([SMAR86, FR88]). For example, given a plan and a number of different goals related to that plan, e.g. developing a cheap solution, or developing an extension, the assessed similarity to other plans is assumed to vary depending on the goal.

Similarity as a knowledge–based process

Research on experts and novices demonstrates that similarity judgements depend on the availability of suitable knowledge. It has been shown repeatedly (e.g. [BFVS89]) that assessments of similarities change as a function of growing knowledge such that people become sensitive to features and dimensions that otherwise escape their attention.

Similarity as a selective process

Experts when asked to pick up two similar descriptions of problems from a set of descriptions, tend to base their similarity assessment on a subset of the set of features and ignore others. In a series of experiments, Holyoak & Koh [HK87] and Chi, Feltovich & Glaser [CFG81] demonstrated that experts prefer structural features, i.e. features that play a causal role in generating a problem solution in order to establish the similarity between two objects. In contrast, novices tend to use surface features, i.e. features that play no causal role in problem solving, to assess the similarity between two objects.

Similarity as a constructive process

Polya [Pol45] was among the first to advocate the idea of similarity assessment as a constructive process. For example, he suggested that a problem solver can address a three-dimensional geometrical problem by transforming it into a two-dimensional one. As a result, it is often much easier to find a similar two-dimensional geometrical problem, retrieve the corresponding solution and adapt it to the problem that triggered this cycle of analogical reasoning. Additional empirical evidence that assessing similarity assessment involves construction processes on the side of the target was provided by Clement [Cle82] who analyzed protocols of problem solvers dealing with physics problems.

Similarity as a context–sensitive process

The claim that similarity assessment varies across contexts is in line with empirical results obtained e.g. by Tversky [Tve77] and Barsalou [Bar82]. In one of Barsalou's experiments the assessed similarity between pairs of animals, e.g. raccoon and snake, has been shown to be greater within no context condition than in a context of pets.

Taken together, cognitive science studies of similarity assessment provide convincing evidence that purely syntactic approaches fall short of capturing basic characteristics of similarity.

3 A computational framework

In the sequel, we give a formal account of basic terminology used in case-based reasoning. Additionally, we introduce concepts that have not been used in case-based reasoning but that are deemed necessary for our purpose. Finally, we present a sketch of a model to compute goal-driven similarity.

3.1 Basic definitions

In order to explicate and further elaborate on the ideas introduced above we settle on a first-order language $\mathcal{L}$ for knowledge representation. $\mathcal{L}$ has a finite number of symbols for constants, predicates and functions, is closed under negation and is our basis for describing the underlying domain.

Definition 1 (Description) *Let $\mathcal{L}$ be a first order language for knowledge representation. A finite, consistent subset $D \subseteq \mathcal{L}$ of literals (without free variables) is called a description.*

Descriptions are possible on different levels of detail. We have to distinguish between *complete* and *partial* descriptions.

Definition 2 (Complete Description) *A description $D \subseteq \mathcal{L}$ is called a complete description if there is no consistent extension possible, i.e. for all descriptions $D' \subseteq \mathcal{L}$, $D \subseteq D' \Longrightarrow D' = D$ holds. A description which is not complete is called a partial description.*

The concept of complete descriptions is comparable to the notion of complete state descriptions which are used in diagnosis or planning.

One of the basic concepts of CBR is the notion of a case. Seen from a cognitive science point of view, cases are abstractions of problem solving behavior that occured in a specific situation. In this sense, cases include implicit problem solving heuristics which can be interpreted with respect to different purposes.

Definition 3 (Case) *Let $\mathcal{L}_P \subseteq \mathcal{L}$ and $\mathcal{L}_S \subseteq \mathcal{L}$ be first order languages for knowledge representation with $\mathcal{L}_P \cap \mathcal{L}_S = \emptyset$ and let $\Sigma \subseteq \mathcal{L}_P \cup \mathcal{L}_S$ be a domain theory. A case is defined as an ordered pair $C = (P, S)$, where $P \subseteq \mathcal{L}_P, S \subseteq \mathcal{L}_S$ are descriptions and $\Sigma \cup P \cup S$ is consistent. We restrict our considerations for theories Σ that fullfill $\Sigma, P \models S$ for every complete description P.*

With regard to the notions of case-based reasoning, P is a description of a problem and S is a description of the corresponding solution. Normally, the descriptions are partial. But if we have a complete problem description P then there exists a case $C = (P, S)$ such that for all cases $C_i = (P, S_i)$ it is $S_i \subseteq S$.

If the languages for P and S are not disjoint, $\mathcal{L}_P$ and $\mathcal{L}_S$ are kept separated for practical reasons. This can be achieved by copying the symbols a_i common to both languages to a_{iS}, a_{iP} and by extending Σ by schemata expressing the equivalence, e.g. $a_{iS} \rightleftharpoons a_{iP}$.

Definition 4 (Consistency of Cases) *Two cases $C_i = (P_i, S_i)$ and $C_j = (P_j, S_j)$ with $P_i, P_j \subseteq \mathcal{L}_P$ and $S_i, S_j \subseteq \mathcal{L}_S$ are called consistent cases if for $P_i \subseteq P_j$ or $P_j \subset P_i$: $\Sigma \cup S_i \cup S_j$ is consistent.*

Previous cases are collected in order to profit from problem solving experiences.

Definition 5 (Case Base) *Given Σ, $\mathcal{L}_P$, $\mathcal{L}_S$, then a finite set of consistent cases is called a case base $CB = \{C_1, C_2, \ldots, C_n\}$. The elements C_i of CB are called source cases.*

Definition 6 (Target Case) *A case without a solution $C_T = (P, \emptyset)$ is called a target case.*

Often, it is not necessary or even misleading to judge the similarity between complete case descriptions. In what follows, we introduce the notion of *aspects* as parts of the case description.

Definition 7 (Aspect) *Let C be a set of cases and $\mathcal{L}$ be the underlying language. An aspect A is a partial function from C into the powerset of $\mathcal{L}$-literals $Lit(\mathcal{L})$. $A: C \to \wp(Lit(\mathcal{L}))$.*

Defining aspects as partial *functions* implies that values of aspects may vary between different cases. Whenever a previously unknown value of an aspect is acquired, the *partial* function can be extended.

Definition 8 (Aspect of a Case) *Let A be an aspect with all its values from the problem descriptions P of cases or all its values from the solution descriptions S of cases, respectively. Let $C = (P, S)$ be a case and $d \in \{P, S\}$. The value $A(C) \subseteq d$ of A is called a d-aspect of C, written as $A_d(C)$.*

In what follows, if there is no danger of confusion, only the notion *aspect* is used for the function and their values.

Depending on their role in problem solving, two special types of aspects, *goals* and *perspectives*, may be distinguished, both of which will be introduced in subsequent sections.

3.2 Goals

One of the concepts that have not been used in similarity assessment up to now is the notion of a *goal.* Goals refer to the reasons for a specific kind of problem solving. In addition and more relevant to the present concerns, goals imply which part of the description of a problem is actually used for goal-achievement. Within the framework of our model of similarity assessment the notion of a goal serves two purposes:

- First, it provides a means to express that similarity assessment is hardly ever done without a special purpose.
- Second, by virtue of this capacity it constrains the vast amount of possibilities that arise when comparing two objects in order to estimate the similarity between them.

In our model goals are aspects of the solution description.

Definition 9 (Goal) *A goal $\mathcal{G}$ is a particular aspect A_S. Its values are S-aspects of cases, written as $A_S(C_i)$.*

Once a goal is adopted, it places specific restrictions on the kind of features of the problem description P that are taken into consideration. We use the term *perspective* in reference to features used for problem solving that involves similarity assessment.

3.3 Perspectives

Given a goal the problem-solver is committed to, the set of features by which a problem is described is reduced to a consistent and finite subset of literals. In our model, perspectives are aspects of the problem description.

Definition 10 (Perspective) *A perspective $\mathcal{P}$ is a particular aspect A_P. Its values are P-aspects of cases, written as $A_P(C_i)$.*

$\mathcal{P}$ is made up of features that bear on the goal. To make similarity assessment goal-driven is to find an appropriate perspective, which boils down to single out only goal-relevant literals. A necessary requirement to do this is a domain theory that highlights relationships between parts of the case and the goals within a domain. Depending on the goal pursued, we end up with different perspectives. Thus, given a goal $\mathcal{G}$ the perspective that is based on this goal is written as $\mathcal{P}_{\mathcal{G}}$.

3.4 Similarity

Our notion of similarity assessment will be developed in two steps. First, we start by introducing an intuitive and desirable approach to similarity assessment. But this approach is faced with difficulties when actually applied to case-based reasoning. This is the reason why, second, a different computational approach to similarity assessment is introduced that is tailored to the specific demands of case-based reasoning and problem solving.

Intuitive approach to similarity assessment

This first view of similarity is motivated by the fact that for problem solving we are interested in solutions of previous cases which are easy to transform according to the current problem. The underlying similarity relation $\sim$ could be easily defined by the costs of modification which are necessary to transform a solution S_i into a solution S_j.

Given a similarity relation $\sim$ for solutions, then two cases $C_i = (P_i, S_i)$ and $C_j = (P_j, S_j)$ are said to be similar if the corresponding solutions S_i and S_j are similar with respect to $\sim$.

However, in case-based reasoning this intuitive approach to similarity assessment cannot be pursued in a direct way since a new problem P_k that lacks a solution S_k is to be solved. According to the view provided above, similarity becomes an a posteriori criterion, because it is only after having determined the solution S_k that we can judge whether the underlying cases and therefore the problems P_i, P_k are similar.

To assess similarity for retrieval in CBR we have to look for a definition of a similarity relation which compares the problem descriptions P_i, P_k directly instead of S_i, S_j and captures the spirit of this approach.

Computational approach

The goal-driven approach is intended to close the gap between similarity of solutions and similarity of problem descriptions. The main point of our approach is to determine which perspective we have to choose, so that similarity between problem descriptions is useful for deriving a solution for the target case.

In what follows, $\sim$ denotes similarity as to be defined by a syntactic measure of similarity (e.g. [RW91], [Tve77]) that is used in combination with our pragmatic model.

Equality is the most obvious kind of similarity. Interpreting $\sim$ as identity transforms similarity assessment into a test of part-identity (*cf.* [Smi89]). Similarity is then represented as equality on an abstract level.

Definition 11 (Similarity in Aspects)
Let $\sim$ be a similarity relation on sets of literals. Two cases $\mathcal{C}_i$ and $\mathcal{C}_j$ are said to be similar *with respect to an aspect A, expressed by $\mathcal{C}_i \sim_A \mathcal{C}_j$, if $A(\mathcal{C}_i) \sim A(\mathcal{C}_j)$.*

3.5 Connections

A part of the domain theory Σ that is used to find out the relevant features on the basis of a specific goal is formulated by means of *connections* [Mel90]. A connection represents knowledge about the – sometimes vague – causal or the like relation between two aspects of a system. Formally, a connection is an ordered pair of aspects $[A_l, A_k]$.

Definition 12 (Connections) *Given a similarity relation $\sim$, an aspect A_l is called connected to an aspect A_k with respect to $\sim$, written as $[A_l, A_k]$, if for almost all cases $\mathcal{C}_i$ and $\mathcal{C}_j$ of CB, which are similar with respect to the aspect A_l, i.e. $\mathcal{C}_i \sim_{A_l} \mathcal{C}_j$, $\mathcal{C}_i$ and $\mathcal{C}_j$ are similar with respect to the aspect A_k, i.e. $\mathcal{C}_i \sim_{A_k} \mathcal{C}_j$.*

In general, connections are not laws in a strong domain theory but default knowledge about relations between aspects. Connections do not guarantee correct inferences but capture the heuristic and experimental nature of this kind of knowledge (*cf.* Russell's *determinations* [Rus89]). Implications may be expressed as a strong kind of connection. Examples of well known connections are `[function, structure]`, `[cause, effect]`, `[situation, behaviour]`. Russell's determinations for example are connections of an implicational type.

3.6 A model of goal-driven similarity assessment

Putting things together, goal-driven similarity assessment starts on the basis of a domain-theory Σ containing connections, a set of *goals* $\mathcal{G}_i$, a *target case* $\mathcal{C}_T$ and a *case-base* CB with source cases $\mathcal{C}_j$. Given a target case $\mathcal{C}_T$, a goal $\mathcal{G}$, and a connection $[\mathcal{P}_\mathcal{G}, \mathcal{G}]$, the specific perspective $\mathcal{P}_\mathcal{G}$ can be chosen under which the similarity to different source cases $\mathcal{C}_j \in CB$ is assessed.

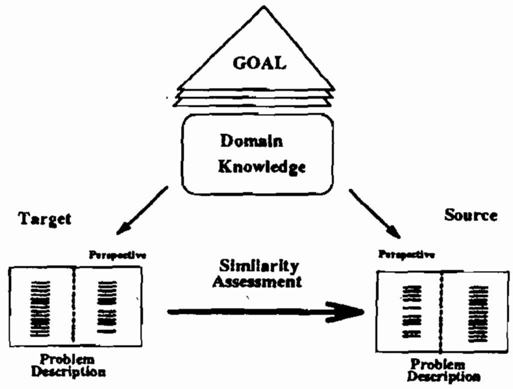

Figure 2: Goal-driven similarity assessment

In general, however, the specific connection $[\mathcal{P}_\mathcal{G}, \mathcal{G}]$ is unknown. To derive the connection $[\mathcal{P}_\mathcal{G}, \mathcal{G}]$ we use explanation-based generalisation (EBG) (*cf.* [MKKC86]) to bridge the gap between the goal $\mathcal{G}$ and the perspective $\mathcal{P}_\mathcal{G}$; i.e. techniques stemming from EBG are applied to select a feature set that pertains to a goal $\mathcal{G}$. In this way, similarity assessment becomes a knowledge-based process. That is, knowledge specified in a domain theory is used to arrive at an explanation why a set of features is required to accomplish a goal. The elements of EBG are used in our model as follows:

Goal: Existing goal $\mathcal{G}$ in the problem solving process which should be achieved.

Example: Description P_T of the target case $\mathcal{C}_T$.

Domain Theory: Σ with knowledge about relationships between the objects in the domain, e.g. connections.

Operationality Criterion: Goal $\mathcal{G}$ must be expressed in terms of features which are already used in the description of the problem P_T of the target case $\mathcal{C}_T$.

Determine: A set of features $\mathcal{P}_\mathcal{G}$ of the target case $\mathcal{C}_T$ which is sufficient to accomplish the goal $\mathcal{G}$ with a solution S of a source case $C = (P, S)$ are to be singled out.

This is done by looking successively for precon-

ditions of the goal (*goal regression, cf.* [Wal77]) until the operationality criterion is met. In contrast to the original work of Mitchell, Keller and Kedar–Cabelli our domain-theory contains connections, i.e. experience, as well as facts and rules.

If the target case C_T and a source case C_i are similar with respect to the perspective $\mathcal{P}_\mathcal{G}$, the goal $\mathcal{G}_i$ may be achieved in the target case C_T by using the solution S_i of the source case C_i.

Given: $\Sigma, \mathcal{G}, C_T, CB$
Searching: S_i to achieve $\mathcal{G}$ in C_T

1. Derive the goal-dependent perspective $\mathcal{P}_\mathcal{G}$ by using the domain theory Σ and $\mathcal{G}$.
2. Use the perspective $P_\mathcal{G}$ to assess similarity between the source cases $C_i \in CB$ and the target case C_T, i.e. compute whether $\mathcal{P}_\mathcal{G}(C_T) \sim \mathcal{P}_\mathcal{G}(C_i)$.
3. Look for the *most similar* source case(s) $C_i = (P_i, S_i) \in CB$ with respect to $\mathcal{P}_\mathcal{G}$.
4. Use the solution S_i in particular $\mathcal{G}(C_i)$ to achieve the goal $\mathcal{G}$ for the target case C_T, i.e determine $\mathcal{G}(C_T)$.

3.7 Combination of goals

As discussed above, a single goal provides the basis for focusing similarity assessment. In general, however, the overall goal of a task may be decomposed into an ordered set of subgoals. As a consequence, our model has to be extended in order to make similarity assessment goal-driven when a multitude of goals is given. Given a set of goals $\{\mathcal{G}_1, \mathcal{G}_2, \ldots \mathcal{G}_m\}$, and a task–dependent odering $\preceq$ over goals, a straightforward approach is to compute similarity stepwise for each $\mathcal{G}_i$.

Starting with $i = 1$ and the whole case base CB, cases which are *most similar* according to the current goal $\mathcal{G}_i$ are selected. This set of cases is carried over to the subsequent run of similarity assessment according to the next goal $\mathcal{G}_{i+1}$ according to the given odering $\preceq$. This procedure continues until each goal $\mathcal{G}_i$ is used for similarity assessment or the set of selected cases contains no more elements. The best scoring cases of the last run are accepted as the result of the retrieval process.

The ordering $\preceq$ of goals is highly dependent on the specific task and the domain. In planning, the difficulty of achievement and costs of modification of a goal $\mathcal{G}_i$ respectively are appropriate criteria for establishing the ordering $\preceq$.

4 An example

To demonstrate the central notions of goal-driven similarity assessment let us discuss a short example of case-based reasoning strategies that can be used for finding a workplan for rotational parts in mechanical engineering. For simplicity, we concentrate on those parts of an example of a real-world application, which are necessary to flesh out our model of goal-driven similarity. In the sequel, the notion *feature* applies to details of the application, whereas the term *literal* refers to the corresponding logical representation.

The overall task is to generate a process plan for manufacturing a workpiece by using data provided by a CAD (Computer-Aided Design) system. In practice, in most mechanical engineering planning tasks human experts try to reuse old plans by adapting them to a new situation (*cf.* [SBKS91]). This is not surprising because planning from first principles is very difficult in a complex real-world domain such as production planning. Plans which are constructed by human experts are for the most part based on specific problem solving experiences. Thus, case-based reasoning is an adequate problem solving paradigm to reflect this common practice.

However, retrieving appropriate cases for complex tasks like planning is a crucial step in CBR because *similarity* can be assessed with regard to a number of perspectives. In this domain, e.g.

- similarity concerning necessary resources (e.g. machines, tools, fixtures)
- similarity concerning the material used
- similarity concerning necessary basic operations (e.g. cutting, drilling)
- similarity concerning the outline of the workpieces

Each *perspective* - and as a consequence each *similarity assessment* - is tied to a special *goal* of the overall planning process, e.g. finding a fixture to clamp the workpiece. As part of their domain-theory or from experience, experts know *connections*, e.g. "Similarity in the outline of the workpieces entails using similar fixtures."

The target case

Suppose, we want to build a process plan for manufacturing the workpiece given in figure 3. The workpiece is described by a set of features which may be extracted from an object-oriented CAD system (Fig. 4). The problem solving process is made up of several steps. One of the goals which must be achieved during the planning process is to determine a fixture to clamp the workpiece and to prepare it for the cutting process to follow. In our example, we focus on this specific goal: $\mathcal{G} :=$ `fixture(X)`

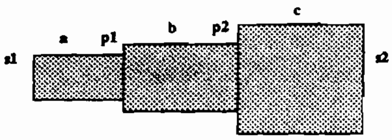

Figure 3: A primitive workpiece

Selection of the perspective

As mentioned earlier, human experts know as part of their knowledge and experience that the kind of fixtures which shall be used to clamp a workpiece depends on its outline. The fixtures are similar if the outlines of workpieces are similar. This may be formalized as a connection `[outline,fixture]` which is part of the experts domain theory. In addition, experts know a lot of technical details about the working process, the tools and the machines they use.

```
% PROBLEM: description of the workpiece
 name(workpiece2).
% Geometry:
 circular_area([s1,p1,p2,s2]).
 cylinder([a,b,c]).
 greater_diameter([[b,a],[c,b]]).
 connected([[s1,a],[a,p1,b],
           [b,p2,c],[c,s2]])
% Technology:
 material([[all,c47]]).
 surface_quality([[a,15],[b,10],[c,13]]).
 tolerance([[a,2]]).
...
```

Figure 4: The problem description

In our model, applying experience and knowledge to solve a current problem is viewed as an explanation-based process. Using the domain theory (Fig. 5) and the operationality criterion, we derive a perspective $\mathcal{P}_G$. If source and target-case are similar under $\mathcal{P}_G$ the goal $\mathcal{G}$ may be achieved by using the solution given in the source case.

```
fixture(F)     :- fixture_fc21(F).
fixture(F)     :- fixture_fc23(F).
...

fixture_fc21(F) :- shoulder(X,P,Y),
                   cylinder(Y),               *
                   quality_ok(Y),
                   F = fc21.

shoulder(X,P,Y) :- connected(X,P,Y),          *
                   greater_diameter(Y,X),     *
                   connected(Y,S),            *
                   circular_area(S).          *

quality_ok(X) :- surface_quality(X,Q1),       *
                 Q1 > 7,
                 tolerance(X,Q2),             *
                 Q2 > 5.
```

Figure 5: Some parts of the domain theory

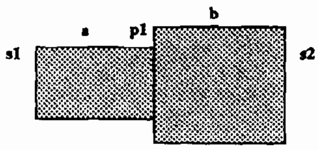

Figure 6: The workpiece of the source case

The result of the explanation-based process is a set of features of the source case C_i which are sufficient to use the same fixture in the target case C_T. This set of features is provided in the following perspective $\mathcal{P}_{\texttt{fixture(X)}}$ describing parts of the `outline` of the workpiece (in Fig. 5 marked with a $*$):

```
1) cylinder([Y]).
2) circular_area([S]).
3) connected([[X,Z,Y],[Y,S]]).
4) greater_diameter([Y,X]).
5) surface_quality([Y,Q1]).
6) tolerance([Y,Q2]).
```

Explanation: For clamping we need a cylinder at the end of the workpiece (1,2,3). The workpiece should

```
% PROBLEM: Description of the workpiece
name(workpiece1)
% Geometry:
 circular_area([s1,p1,s2])
 cylinder([a,b])
 greater_diameter([[b,a]])
 connected([[s1,a],[a,p1,b],[b,s2]])
% Technology:
 material([[all,c45]])
 surface_quality([[a,10],[b,15]])
 tolerance([[a,11],[b,10]])
 ...
% SOLUTION: Workplan
 use_machine(m44)
 chuck([b],fixture(fc21))
  change_tool(t1)
   cut([s1,a,p1],roughing)
 chuck([a],fixture(fc21))
  change_tool(t2)
   cut([s2,b],roughing)
 unchuck
```

Figure 7: Description of the source case

be fixed at the cylinder with the greatest diameter because of the transmission of the rotational force (4). The surface quality of the part where the workpiece is fixed should not be too high as clamping a workpiece destroyes high surface quality (5,6).

An example of a similar source case containing an executable workplan for manufacturing the workpiece given in figure 6 is depicted in figure 7. To clamp the workpiece given in the target case C_T (Fig. 3) we can use the fixture fc21 provided in the workplan of the source case (Fig. 7).

Retrieval of cases

In our example, there is just one source case given. Usually, there is a great number of different source cases available in the case base. Having determined the relevant features a feature-oriented similarity model like the *contrast-* or *ratio-model* proposed by Tversky [Tve77] must be applied to look for the best fitting source case according to the current goal given.

5 Related work

The ideas introduced in this paper are closely related to Kedar–Cabelli's model of purpose-directed analogy [KC85]. Kedar-Cabelli aims at integrating the influence of pragmatics, e.g. purposes, into the generation of analogies. Although purpose-directed analogy shares with goal-driven similarity the intuition of pragmatic factors to be important for similarity, a comparison shows striking differences: Kedar-Cabelli's work is rooted in the framework of analogical reasoning, thereby focusing on analogical mapping and concept formation as a result of analogical reasoning. Additionally, purpose–directed analogy reconstructs the target in terms of the source. In contrast, goal–driven similarity singles out features that are deemed necessary to be taken into consideration when assessing similarity.

Our model concentrates on similarity assessment as to be used in various forms of reasoning. In a word, Kedar–Cabelli elaborates on pragmatic-driven analogical mapping and we concentrate on pragmatic-driven retrieval.

In an attempt to improve indexing in CBR, Barletta & Mark [BM88] use *explanation-based learning* (EBL) to determine features that play a causal role in finding a solution to a target case. Based on the domain theory, the problem specification and the solution to that problem the system aims at explaining the goal concept, i.e. one or a sequence of actions that lead to a solution. The explanation of the goal concept is guided by a hypothesis tree that is provided by the domain theory.

The differences to our own work results from the fact that Barletta & Mark's approach is exclusively concerned with indexing cases that enter the case library. The featural description of cases they use is made up of the description of a problem and its solution. On this account, the assessment of similarity of a target to a source and the use of goals instead of solutions is not touched by their work.

Cain, Pazzani & Silverstein [CPS91] describe an approach to integrate domain knowledge in the assessment of similarity between source and target cases. This is accomplished by using explanation-based learning as a means to judge the relevance of features. Their measure of similarity combines the *nearest–neighbour technique* that counts the number of identical features with a measure that counts the number of matching relevant features according to EBL. In this way, similarity between two cases will be deemed high

if they share a great number of common features or a great number of relevant features. If EBL does not arrive at an explanation for the solution of a case, this measure of similarity boils down to the nearest-neighbour technique.

Contrary to our study, Cain et al. do not use concepts like goal or perspective when assessing similarity. They take EBL to explain the features that are required to reach a solution and do not use goals as we do. Thus, the model of Cain et al. starts with by preselecting cases based on a pure feature-overlap measure of similarity. Then EBL is applied to determine features relevant to reach a solution. EBL is limited to source cases since - by definition - only they have a *known* solution.

6 Conclusions and future work

The work introduced in this paper has two related foci: First, we discuss cognitive science findings that show why human similarity assessment is both a powerful and flexible capability. In addition, we present a formal model that accounts for most of the chararcteristics in human similarity assessment we discussed.

At the most general level, our model is an example in which way empirical findings can be used as a starting point to contribute to the development of formal models that may be used as building blocks in AI systems. Among the four characteristics of similarity discussed above, there are three that are supported by our model of goal–driven similarity assessment: First of all, our model exploits the notion of *goals* when assessing similarity. Additionally, by using a domain theory to focus on goal–relevant aspects our model has been proven to be a *knowledge-based* one. Finally, because of its capacity to develop connections and corresponding perspectives the process of goal–driven similarity assessment may be referred to as *constructive*. To make similarity assessment *context-sensitive* remains as a possible extension of the work described in this paper. Thus, our model gives a fairly good account as far as cognitive modelling of basic characteristics of similarity assessment is concerned.

Mention ought to be made, however, of several issues that as yet remain open. On a formal account, our model is restricted to literals; we strive for an extension to formulae. In section 3.7 a schema has been introduced that allows for similarity assessment if a multitude of goals is given. This schema, however, is not fully satisfying. The difficulty with this approach is that a cut-off value determining which subset of cases is used when assessing the subsequent goal has to be supplied in a hand-coded way. Currently, we concentrate on an extension of our model to a multitude of goals that can do without this shortcoming.

The present version of our model focuses on goal-relevant aspects which may be referred to as abstraction by reduction. By using hierarchies of aspects abstraction may be achieved by substituting an aspect by a more abstract one. In this way, goal-driven similarity assessment becomes independent of specific instantiations since similarity assessment is performed on a more abstract level.

Apart from open questions just mentioned, goal-driven similarity assessment comes up with some issues we consider as strengths of our model. More specifically, by incorporating goals our model offers four advantages that go beyond models of similarity assessment that are oblivious of pragmatic factors like goals:

First, similarity assessment and retrieval is improved. This is achieved by considering only those features of a case that pertain to a goal. Distorting similarity assessment due to an overlap of aspects that do not pertain to a goal is avoided because of a more focused similarity assessment. As a result, the search space to be traversed in order to find an appropriate case can be reduced substantially.

Second, similarity assessment is tied to the goal of a problem-solver and may vary along with a change of goals.

Third, goal–driven similarity assessment allows for a multiple use of cases which depends on a variation of goals or an improvement of the domain theory. For example, a case-based reasoner in toxicology that is equipped with a device for goal-driven similarity assessment, is able to use knowledge represented in cases in a variety of ways. Again, this is done by performing a specific similarity assessment according to different goals like *determine the toxin* or *work out a therapy*.

Fourth, in the case where no explicit goals are given, a failure when applying EBG, or a defective or totally missing domain theory occurs, goal–driven similarity boils down to the syntac-

tic approach to similarity assessment that is used in linkage to our model.

7 Acknowledgements

This work was done while Dietmar was a visiting scientist at the University of Kaiserslautern. The authors are indebted to Michael M. Richter, Jörg H. Siekmann and Gerhard Strube.

References

[Bar82] L. W. Barsalou. Context-independent and context-dependent information in concepts. *Memory & Cognition*, 10:82–93, 1982.

[BFVS89] J. D. Bransford, J. F. Franks, N. J. Vye, and R. D Sherwood. New approaches to instruction: because wisdom can't be told. In S. Vosniadou and A. Ortony, editors, *Similarity and Analogical Reasoning*, pages 470–497. Cambridge University Press, 1989.

[BM88] R. Barletta and W. Mark. Explanation-based indexing of cases. In *Proceedings of 7.th National Conference on Artificial Intelligence*. Minneapolis, 1988.

[CFG81] M. T. Chi, P. J. Feltovich, and R. Glaser. Categorization and representation of physics problems by experts and novices. *Cognitive Science*, 5:121–152, 1981.

[Cle82] A. Clement. Spontaneous analogies in problem solving: The progressive construction of mental models. In *Proc. Meeting of the American Education Research Association*, New York, 1982.

[CPS91] T. Cain, M. J. Pazzani, and G. Silverstein. Using domain knowledge to influence similarity judgements. In R. Bareiss, editor, *Proc. of the 1991 Workshop on CBR*, Washington, D.C., May 1991. Morgan Kaufmann Publishers.

[Dic45] L. R. Dice. Measures of the amount of ecologic association between species. *Journal of Ecology*, 26:297–302, 1945.

[FR88] J. Fanes and B. Reiser. Access and use of previous solutions in a problem solving situation. In *Proc. of the tenth Annual Conference of the Cognitive Science Society*, pages 433–439, Montreal, 1988.

[HK87] K. J. Holyoak and K. Koh. Surface and structural similarity in analogical transfer. *Memory & Cognition*, 15:332–340, 1987.

[Hol85] K.J. Holyoak. The pragmatics of analogical transfer. In G. Bower, editor, *The psychology of learning and motivation*. Academic Press, New York, NY, 1985.

[KC85] S. Kedar-Cabelli. Purpose-directed analogy. In *Proc. of the Cognitive Science Society*, Irvine, CA, August 1985.

[Kol91] J. L. Kolodner. Improving human decision making through case-based decision aiding. *AI Magazine*, 91(2):52–68, 1991.

[Mel90] E. Melis. Study of modes of analogical reasoning. Tasso-Report Nr. 5, Gesellschaft für Mathematik und Datenverarbeitung mbH (GMD), 1990.

[MKKC86] T. M. Mitchell, R. M. Keller, and S. T. Kedar-Cabelli. Explanation-based generalization: A unifying view. *Machine Learning*, 1(1), 1986.

[Pol45] G. Polya. *How to solve it*. Princeton University Press, Princeton, NJ, 1945.

[Rus89] S. J. Russell. *The use of Knowledge in Analogy and Induction*. Pitman Publishing, London, 1989.

[RW91] M. M. Richter and S. Wess. Similarity, uncertainty and case-based reasoning in PATDEX. In R. S. Boyer, editor, *Automated Reasoning*, Essays in Honor of Woody Bledsoe, pages 249–265. Kluwer Academic Publishing, 1991.

[SBKS91] F. Schmalhofer, R. Bergmann, O. Kühn, and G. Schmidt. Using integrated knowledge acquisition to prepare sophisticated expert plans for their reuse in novel situations. In *Proc. KAW-91*, Banff, Canada, 1991.

[SMAR86] C. Seifert, G. McKoon, R. Abelson, and R. Ratcliff. Memory connections between thematically similar episodes. *J. Exp. Psych. Learning Memory Cognition*, 12:220–231, 1986.

[Smi89] L. B. Smith. From global similarities to kinds of similarities: the construction of dimensions in development. In S. Vosniadou and A. Ortony, editors, *Similarity and Analogical Reasoning*, pages 146–178. Cambridge University Press, 1989.

[Str91] G. Strube. Dynamic perspective in distributed representations. *Zeitschrift fuer Psychologie*, 199(4/91):289–298, 1991.

[Tve77] A. Tversky. Features of similarity. *Psychological Review*, 84:327 – 352, 1977.

[Wal77] R. Waldinger. Achieving several goals simultaneously. *Machine Intelligence*, 8:94–136, 1977.