Fachbereich Informatik
Universität Kaiserslautern
Postfach 3049
D-6750 Kaiserslautern

SEKI - REPORT

System and Processing View in Similarity Assessment

Stefan Wess, Dietmar Janetzko, Erica Melis

SEKI Report SR-92-17 (SFB)

SYSTEM AND PROCESSING VIEW IN SIMILARITY ASSESSMENT*

Stefan Wess
University of Kaiserslautern
W–6750 Kaiserslautern

Dietmar Janetzko
University of Freiburg
W–7800 Freiburg

Erica Melis
University of Saarbrücken
W–6600 Saarbrücken

Abstract

Work on similarity can be shown to follow either a system view or a processing view with the former paying more attention to architectures of similiarity assessing systems and the latter concentrating on similarity metrics. As similarity depends on a number of characteristics (e.g. goals, knowledge, context, common features) both view have their own merits when assessing similarity. In this paper, we present a framework of multistage similarity assessment that provides a linkage for a modeling of similarity according to the system and processing view. In so doing, the stages of the system can be evaluated according to both the characteristics of similarity being modeled and the errors possibly made.

1 Introduction

During the past several years, a flurry of interest in similarity has been touched off by research done in information retrieval *(IR)*, analogical *(AR)* and case-based reasoning *(CBR)* (e.g. Vosniadou & Ortony, 1989). While puzzling out principles of similarity assessment in cognitive science and artificial intelligence two different approaches have been pursued: Investigations of similarity adopting the *processing view* strive at developing a condensed formal account of similarity intended to be used independently of the pecularities of a system's architecture. To put it another way, the core idea of the processing view has been to uncover principles of similarity as basic as possible to obtain a coverage as broad as possible. A well-known proponent of the processing view is (Tversky 1977) and his contrast model.

Conversely, research indebted to the *system view* concentrates on specifying architectural constraints on similarity assessment. That is, according to the credo of the system view charcteristics of similarity may be captured by choosing an appropriate architecture of a system. Following this line, in case-based reasoning a number of models of computing similarity start with a great number of computational cheap similarity assessments; only

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

cases that yield a high score are taken over to the second stage to be assessed again with computationally expensive methods used to select the best scoring cases (e.g. Gentner & Forbus, 1991).

Pointing out to differences between a processing and system view is not supposed to pass unchallenged. At least when it comes down to actually building a system, so a possible caveat might go, the distinction between the two views seems to be more a difference in emphasis than in substance. Our objection to this argument is that there is quite a variety of characteristics of similarity assessment (Janetzko, Wess & Melis 1992), some of which are best modeled either according to the system view as to the processing view.

For example, the dependency of similarity assessment on the number of common and distinguishing attributes is probably best captured by the processing view. In contrast, the dependency of similarity assessment on goals, knowledge, context, resources invested like time or memory are issues covered best by the system view. Thus, differentiating between these two views is more but a funny curiosity in the zoo of models of similarity as it can be used to guide modeling of characteristics of similarity according to the appropriate view.

The present paper is devoted to an analysis of the costs and benefits of similarity assessment according to the processing and the system view. First, the notion of process and system view is stepwisely fleshed out to gain further understanding of the possibilities given by each of both views. Second, we discuss errors that may occur within multistage similarity assessment that links the processing and the system view. Third, we introduce a three-stage model of similarity assessment and present an evaluation along with the criteria established before. Finally, we discuss relationships towards other models of similarity assessment.

2 Linking the processing and the system view on similarity

As foreshadowed by the preceeding discussion, distinguishing between the processing and the system view and differentiating among various characteristics of similarity leads to a desirable goal when building models of similarity assessment: Similarity may be best modeled in accordance to the possibilities of the two different views, i.e. on different levels. This first tenet may be called *"The principle of preferred levels of modeling"*.

Linked to that principle is another one that has to be fulfilled to make similarity assessment flexible. This second principle is referred to as *"The pinciple of graceful degradation"* (Norman & Bobrow, 1975). By this, we mean that similarity assessment should show a smooth decline rather than an all-or-none behavior when faced with difficulties, e.g. low-quality data or the like. This is deemed important if resources (e.g. time, memory) are limited or if the system itself does intend to limit resources (e.g. to perform a preselection) in order to invest resources in an economical fashion.

Finally, the principle just mentioned implies a third one. This is called *"The principle of continually available output"* (Norman & Bobrow, 1975; Russell & Zilberstein, 1991). To spell out this principle is to specify the principle of graceful degradation. As a consequence, it should be possible to stop processes of similarity assessment, e.g. by retracting resources needed, and obtain results that are usable by the system although suboptimal when compared to the results acquired without stopping similarity assessment.

The ideas in this paper rely on the conjecture that modeling of similarity assessments according to the three principles mentioned above is only possible by linking the processing

and the system view. When put into practice, the principle-guided linkage of the two views amounts to a multistage similarity assessment with characteristics of similarity brought into focus by each view distributed on different stages. The framework of such an architecture provides a number of advantages: Modelling of characteristics of similarity can and should be done on different stages according to the principle of preferred levels of modeling. Depending on the stage of processing reached there is a smooth decline in the quality of the system's output, which obeys to the principle of graceful degradation. Finally, an architecture of multistage similarity assessment allows for a good approximation to the principle of continually available output as each stage is a kind of exit-point. The quality of the similarity assessment reached at each exit-point is a function of the resources invested.

3 Demands on the assessment of similarity

In what follows, we characterize two basic requirements to be fulfilled when assessing similarity. This is done along with a discussion of how to put the ideas of this paper into practice when building a system and an eye towards related work in information retrieval, analogical and case-based reasoning. In so doing, we will find further evidence for a multistage similarity assessment, which is spelled out in subsequent sections.

3.1 Efficiency

Analysing the process of similarity assessment from a efficiency point of view results in the demand of low computational costs of the retrieval. Since all items of the knowledge base are involved in the first step of the process, it is reasonable to require the first step to work very quickly on each item. The next step which works already on a set of preselected cases may have higher relative costs.

A similar goal is aimed at by open hashing in databases: The hash function makes it possible to access - a list of items very fast; the search within this list, being as short as possible, has higher relative costs.

In database research a lot of other retrieval approaches has been developed that are computationally cheap e.g. multidimensional associative binary trees, called *k-d Trees* (Bently, 1978), close match retrieval (Friedman, Bently & Finkel, 1977), incremental nearest-neigbour search (Broder, 1990), best-match retrieval based on Voronoi-Diagramms (*c.f.* Mehlhorn, 1984) or hypercubes (Stolter, Henke & King, 1989).

These techniques are able to retrieve a best-match based on a set of surface features in logarithmic expected time $O(log(n))$ where n is the number of stored items in the database.

The now commercial available case-based reasoning shell REMIND (Cognitive Systems, 1991) developed by Cognitive Systems an enterprise founded by R.C. Schank uses this kind of rapid retrieval algorithms for case-based reasoning.

Other approaches to a computationally cheap search of similar cases use the assessment of similarity on the basis of the dot product over feature vectors (Medin & Schaffer, 1978), connectionist models of learning (Rummelhart & McClelland, 1986), the PATDEX-approach (Wess, 1991; Richter & Wess, 1991) or the memory-based reasoning approach (Stanfil & Waltz, 1986), which relies on a massive parallel search on a connection machine.

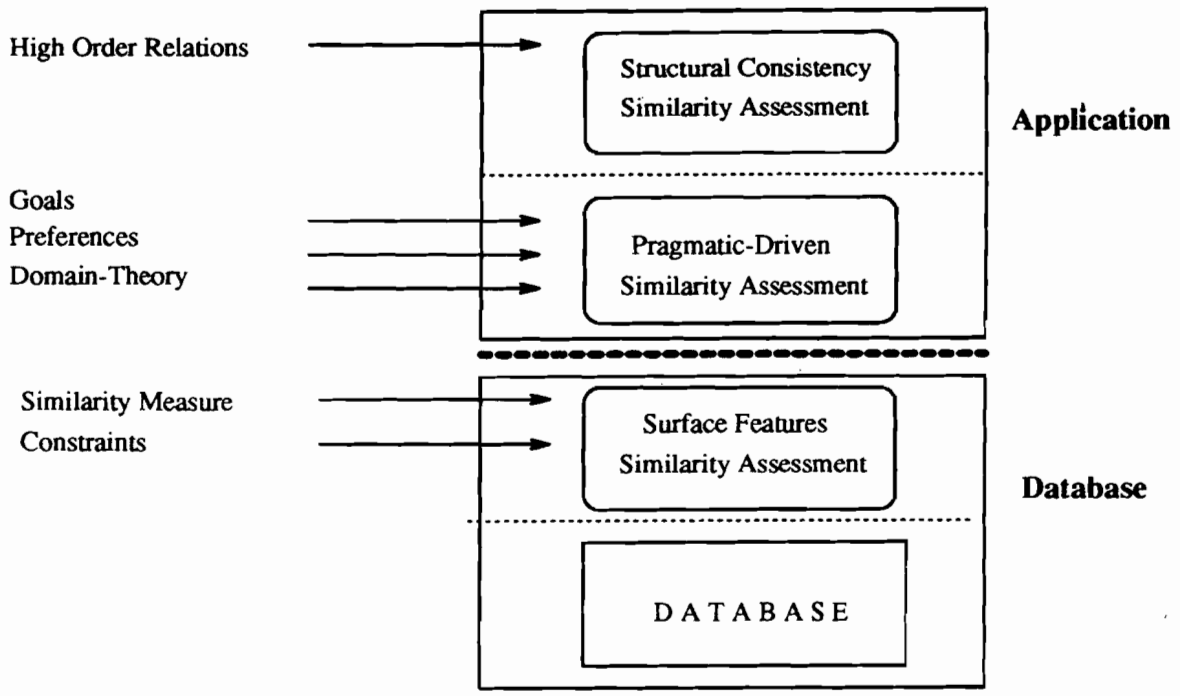

Figure 1: The system view

3.2 Reliability

The analysis of the process from a reliability point of view yields characteristics of the kind of errors occuring at the consecutive steps of the process. These types of errors are well known in statistics as they give an account of the errors that can be made whenever a hypothesis is accepted or rejected (e.g. Bock, 1975). We use the notion of α-error and β-error to classify possible errors to be made when assessing the similarity between two cases.

Definition 1 (α-Error) *If a previous case being useful to solve a problem at hand is part of the case base but not selected, the error is called α-error.*

Definition 2 (β-Error) *If a previous case not being useful to solve the current problem is part of the case base but selected, the error is called beta-error.*

Each model of selecting cases has to account for both kinds of errors. The selection of similar items (e.g. cases, concepts, entries in a database) is guided by selection criteria. α- and β-errors depend on the selection criteria applied to find similar items. Selection criteria causing no α-error are necessary criteria, and selection criteria causing no β-error are sufficient criteria.

As well known (Mitchell, Keller & Kedar-Cabelli, 1986), explanation-based generalization *(EBG)* provides sufficient descriptions. The goal-driven similarity assessment in (Janetzko, Wess & Melis, 1992) using the EBG-method provides sufficient criteria and tends to keep the β-error low.

The ideas that form the basis of α– and β-errors are closely related to the version space method introduced by (Mitchell, 1982). The description to follow shows how the version space technique can be applied to find a selection criteria that keep α– and β-error at the lowest level possible.

Let the example space be a set of pairs of items. The criteria space $CRIT$ is taken to mean a space of formulae representing selection criteria, i.e. the analogue to Mitchell's concept space. The partial order on $CRIT$ (*more specific, more general*) can be defined analogously to the hierarchy of generalizations in the version space. In agreement with the version space method the search space $CRIT$ is reduced from top and from bottom introducing G (as the set of *most general criteria* selecting all known positive examples and rejecting all known negative examples) and S (as the set of *most specific criteria* selecting all known positive examples and rejecting all known negative examples).

The criteria from G keep the α– and the criteria from S keep the β-errors at the lowest level possible. Following Mitchell's model, if $G = S$ the concept is learned and no α– or β–errors occurs.

It is desirable that in the *first step* of case selection no (or almost no) α-error occurs, that is, no useful previous case is excluded from further processing. It is also desirable that in the *last step* cases being not useful are excluded.

There are several possibilities to meet this demand: If during the first step of the process no α-error occurs, and there are useful items in the database then there remains a nonempty set C of items. During the next steps from C items may be selected the computational costs of adaptation are lowest for. Items with computational costs of adaptation that exceed those to be expected can be eliminated.

There are several approaches including such a usability assessment. For example, the goal-driven similarity assessment (Janetzko, Wess & Melis, 1992), similarity conserving transformations (SCT's Koton, 1988) evaluate certain similarities and dissimilarities as relevant or irrelevant.

4 Stages of similarity assessment

As noted earlier, a multistage similarity model has been deemed necessary to cover a number of issues involved in similarity assessment. Among the most important of those issues are the possibility to combine various models of similarity assessments according to different characteristics of similarity. In this way, it is feasible to control the impact of each of those characteristics. Additionally, multistage similarity assessment allows for specifying constraints on errors such that the α–error should be low in the *first* and the β–error should be low in the *last* stage. During the stages the number of items considered decreases and the computational costs per item increase.

Stage I - Syntactic features: Multistage similarity assessment begins by using a syntactic measure of similarity which is based on features that form an explicit part of the representation of the items being compared. Measures deriving similarity from the number of common and different features that may or may not be combined with weigths can be used at this stage (Tversky, 1977). Alternatively, models of similarity assessment mentioned in 3.1 like k-d trees may be employed for that purpose. At this stage, similarity assessment is not dependent on the representation of the domain theory. No knowledge but that encoded in the items (cases, entries of database) is used explicitly. As this stage usually is computational cheap it is well suited to be used as preselecting items.

Stage II - Pragmatic relevance: A pure syntactic approach is not sufficient for similarity assessment. First, a difference with regard to only one feature results in a high statistical

similarity score but may be based only on a high agreement with regard to unimportant features. Vice versa, a great number of differences between two cases leads to a poor statistical similarity score but may camouflage an agreement with regard to important features. For that matter, the next stage proceeds by allowing for the influence of pragmatic determinants (e.g. goals and knowledge) on similarity assessment. In goal-driven similarity assessment (Janetzko, Wess, & Melis, 1992), for example, a set of features is computed by using EBG to single out those features that are of pragmatic relevance according to a goal and a domain theory. At this stage, similarity assessment makes use of the representation of the domain theory and pragmatic determinants like goals or purposes. This stage is computationally more expensive than the first one.

Stage III - Consistency: For economical reasons, the kind of knowledge used in multi-stage similarity assessment is distributed on three stages. Knowledge that can be used as a test to rule out similarity of items has not been employed in the preceeding stages. This kind of knowledge is taken to reject items that are definitely dissimilar when compared to the input item. This stage is extremely dependent on the domain theory and on the application. As a result, there are various possibilities to perform consistency tests. For example a diagnostic application consistency may be definend by a model–based diagnosis approach *c.f.* (Koton, 1988). Depending on the respective application this consistency check may be very expensive. Hence, this procedure is left for the last stage of similarity assessment.

5 Conclusions

Although up to now there is not a clear division into demands for knowledge-based steps of retrieval of cases and others, empirical results show a correspondence of knowledge-based and not-knowledge-based preselections respectively with the selection of cases by experts and novices respectively. Novick (1988) has found differences betweeen the retrieval cues available for the retrieval process by experts and novices: Novices almost exclusive rely on salient surface features of the target. Experts, however, will be able to use both surface and structural features. For common domains Holyoak and Koh (1987) established that retrieval of analogues relies more on surface similarity and less on structural similarity (than mapping). This might be simulated in the retrieval included in CBR by a pure statistical preselection followed up by a more knowledge-based final selection step. An attempt to capture the novice pheneomenon is done by Gentner and Forbus (1991). They use as a first stage a matcher that works as follows: Each case is stored with a content vector (vector of number of occurences of predicates, functions, and connectives) The content vector of each case is compared with the computed content vector of an entered probe. Hence, this stage consists of a purely statistical syntactic comparison. Afterwards a matcher calculating literal similarity is applied to the output of the first stage.

This does not mean that knowledge-based similarity assessment in general provides only sufficient selection criteria. On the contrary, the domain theory can provide necessary criteria, too.

Depending on the pecularities of the domain there is the possibility to introduce knowledge-based modifications, e.g. of a pure statistic preselection by the contrast rule (Tversky & Gati 1982). This may be reasonable if the domain under study provides features or com-

binations of features which make usability probable or which rule out useability.

6 References

Bently, J.L. (1975). Multidimensional Binary Search Trees used for Associative Searching. Communications of the ACM, 18(9), 509- 517.

Bock, R. D. (1975). Multivariate statistical methods in behavioral research. New York.

Broder, A.J. (1990). Strategies for efficient incremental nearest neighbour Search. Pattern Recognition, 23(1), 171-178.

Cognitive Systems (1991). REMIND Solutions from prior experience. Technical Report, Cognitive Systems, Inc., MA, USA.

Friedman, J.H. Bently, J.L. & Finkel, A.F. (1977). An Algorithm for Finding Best Matches in Logarithmic Expected Time, ACM Transactions on Mathematical Software, 3(3), 209-226.

Gentner, D. & Forbus, K.D. (1991). MAC/FAC: A model of similarity-based retrieval. Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society, Chicago, Ill, 504-509.

Holyoak, K.J. & Koh, K. (1987). Surface and structural similarity in analogical transfer. Memory & Cognition 115, 332-340.

Janetzko, D. Wess, S. & Melis, E. (1992). Goal-Driven Similarity Assessment, In: Ohlbach (Ed.), German Workshop of Artificial Intelligence 92, Springer Verlag..

Koton, P. (1988). Reasoning about evidence in causal explanations. In: J. Kolodner (Ed) Proc. of a Workshop on Case-Based Reasoning, Florida. Morgan Kaufmann, 260-270.

Medin, D.L. & Schaffer, M.M. (1978). Context theory of classification learning. Psychological Review 85 (1978), 207-238.

Mehlhorn, K. (1984). Multi-dimensional Searching and Computational Geometry, Springer.

Mitchell, T.M. (1982). Generalization as search. Artificial Intelligence 18, 203-226.

Mitchell, T.M. Keller, R.M. & Kedar-Cabelli, S.T. (1986). Explanation-based generalization: A unifying view. Machine Learning 1, 47-80.

Norman, D.A., & Bobrow, D.G. (1975. On data-limited and resource-limited processes. Cognitive Psychology,7, 44-64.

Novick, L.R. (1988). Analogical transfer, problem similarity and expertise. Journal of Experimental Psychology, Learning, Memory , and Cognition 14, 510-520.

Puppe, F. & Goos, K. (1991). Improving case based classification with expert knowledge. In: Th. Christaller (Ed.), German Workshop of Artificial Intelligence 91, 196-205, Springer Verlag.

Richter, M.M. & Wess, S. (1991). Similarity, Uncertainty and Case-Based Reasoning in PATDEX. Kluwer Academics.

Rummelhart, D. E., & McClelland, J. L. (1986). Parallel distributed processing. Cambridge, MA: MIT Press.

Russel, S.J. & Zilberstein, S. (1991). Composing Real-Time Systems. Proc. IJCAI-91, 212-217.

Stanfil, C. & Waltz, D. (1986). Towards memory-based reasoning. Communications of the ACM 29, 1213-1229.

Stolter R.H. Henke A.L. & King J.A. (1989). Rapid Retrieval Algorithms for Case-Based Reasoning, Proc. IJCAI-89.

Tversky, A. (1977). Features of similarity. Psychological Review, 84, 327-352.

Tversky, A. & Gati, I. (1982). Similarity, separability, and the triangle equality. Psychological Review 89, 123-154.

Vosniadou, S. & Ortony, A. (1989). Similarity and analogical reasoning. Cambridge University Press.