Context-Based Similarity Applied to Retrieval of Relevant Cases*

Igor Jurišica†
juris@cs.utoronto.ca
Department of Computer Science
University of Toronto
Toronto, Ontario M5S 1A4, Canada

December 21, 1994

Abstract

Retrieving relevant cases is a crucial component of case-based reasoning systems. The task is to use user-defined query to retrieve useful information, i.e., exact matches or partial matches which are close to query-defined request according to certain measures. The difficulty stems from the fact that it may not be easy (or it may be even impossible) to specify query requests precisely and completely - resulting in a situation known as a fuzzy-querying. It is usually not a problem for small domains, but for a large repositories which store various information (multifunctional information bases or a federated databases), a request specification becomes a bottleneck. Thus, a flexible retrieval algorithm is required, allowing for imprecise query specification and for changing the viewpoint.

Efficient database techniques exists for locating exact matches. Finding relevant partial matches might be a problem. This document proposes a context-based similarity as a basis for flexible retrieval. Historical bacground on research in similarity assessment is presented and is used as a motivation for formal definition of context-based similarity. We also describe a similarity-based retrieval system for multifunctinal information bases.

Keywords: similarity assessment, context, relevance, case-based reasoning, information retrieval, multifunctional information bases

*An abridged version of the paper appeared in AAAI Fall Symposium Series: I. Jurišica. How to Retrieve Relevant Information?. In Russell Greiner (Ed.): Proceedings of the AAAI Fall Symposium Series on Relevance. New Orleans, Louisiana, November 4-6, 1994

†Supported by an ITRC Research Grant. I also gratefully acknowledge receiving many helpful comments and suggestions from Janice Glasgow, John Mylopoulos and David Lauzon.

Contents

1 Introduction **1**

1.1 Tasks to be Addressed by a Context-Based Similarity 1

1.2 Outline 2

2 Background on Similarity **2**

2.1 Philosophical Background 2

2.2 Cognitive Psychology Background 4

2.2.1 Surface Similarity 5

2.2.2 Deep Similarity 5

2.2.3 Psychological Similarity 6

2.2.4 Analogical Similarity 7

2.2.5 Conclusions and Motivation 8

2.3 Computer Science Approach 8

2.4 Flexible Similarity Assessment 9

2.5 The Role of Context 10

2.5.1 Context Dependency 11

2.5.2 Context and Relevance 13

2.5.3 Harmful Context 14

3 A Theory for Context-Based Similarity **15**

3.1 Definitions 15

3.2 Similarity Relation 16

3.2.1 Comparison Function 17

3.2.2 Retrieval Function 17

3.2.3 Evaluation Function 18

3.2.4 Analysis Function 18

3.2.5 Extensions 18

3.3 Features of Context-Based Similarity 19

3.3.1 Monotonicity 19

3.3.2 Symmetry 20

3.3.3 Transitivity 20

4 Context-Based Retrieval **22**

4.1 Similarity in Context - Partial Order of Similarities 22

4.2 Retrieval System 25

4.3 Context Relaxation 26

4.4 The Similarity Query Language - SimQL 28

5 Discussion **30**

Bibliography **32**

1 Introduction

Databases, knowledge bases, case bases, software repositories, etc., all store information as data or knowledge. Thus, we might refer to them as information bases. In general, an information base contains a set of items, being it either database records or cases or other kind of information. There are two essential tasks to be addressed by an information system - to store and to retrieve information. We assume items are represented as attribute/value pairs. This paper deals with the retrieval task; namely, how to retrieve relevant information.

Intuitively, relevant information is knowledge or data which is useful to a specific problem solving task. If we know what is relevant, we may retrieve appropriate information by posing a specific query. However, in some situations we may not have enough information available to specify a query completely. In such a case, we can pose only approximate request. The task for the system is to produce relevant answer even if the query is not completely specified. This is handled by use of similarity-based retrieval algorithm. Even though this is most relevant to case-based reasoning systems other information-based systems might benefit from the algorithm as well.

Similarity-based retrieval allows for assessing similarity between a repository item and an externally defined item in the query. A query specifies required item (possibly not completely) and the criteria for matching. The system retrieves all similar items.

From everyday experience we know that what is considered similar in one situation may not be similar in another. Thus, systems which do not take context into consideration may return correct answer in some situations whereas they might give an unreasonable response in other cases. To prevent this, the proposed similarity assessment theory represents constraints on similarity matching (context) explicitly. Context allows us to specify what parts of information representation to compare and what kind of matching criteria to use. This allows for excluding similar but irrelevant items. Context also allows us to constrain retrieved information in such a way that only relevant information is obtained. Thus, context can be used as a basis of relevancy measure, i.e., items are considered relevant if they are similar with respect to the current context.

To assess similarity in different situations we need to be able to specify criteria for matching flexibly. To accomplish this, we define context-based similarity, where context is a set of attributes with associated constraints on the attribute values. Context allows us to specify how the matching should be performed giving us local-based matching criteria (Kolodner, 1993).

1.1 Tasks to be Addressed by a Context-Based Similarity

Real world problems require different tasks to be applied during a problem solving process. We show how context-based similarity can be applied in various situations. The tasks to be addressed by a context-based similarity are as follows:

1. **Comparing items.** The easiest task in similarity assessment is to determine if particular items are similar. This may include comparing two items or comparing a set of items. We show that context plays a crucial role in this process and we define comparison function (see Section 3.2.1).

2. **Retrieving items.** In this case, the task is to retrieve all items, similar according to a specified criteria. We show how context allows us to specify matching criteria flexibly and we

define a retrieval function (see Section 3.2.2).

3. **Finding a context.** This task is a reverse one to the retrieval task - having a set of items, the system evaluates possible contexts in which given items are similar and returns the most restrictive of them. We show mechanisms for finding the most restrictive context for a given set of items and we define an evaluation function (see Section 3.2.3).

4. **Knowledge mining.** Combining retrieval and evaluation functions gives us an analysis function. The algorithm analyses possible combinations of set of items and contexts. The largest possible set of items and the most restrictive context is returned (see Section 3.2.4).

1.2 Outline

The main goal of our research is to define a general theory for similarity based on context and applicable to the above mentioned tasks in the domain of information bases. Even though similarity functions are defined for all four tasks, only retrieval function is discussed in detail.

Section 2 presents a selected history of research in the area of similarity assessment. This is used as a motivation for introduction of context-based similarity.

Section 3 is devoted to theoretical results. Information base, item, context and similarity functions for different tasks are formally defined. In addition, it is shown that context-based similarity in retrieval is monotonic, symmetric and transitive.

In Section 4 a similarity-based retrieval system based on our definition of similarity in retrieval is introduced. Transformations of context are discussed and a query language for retrieval function is presented.

In Section 5 related work is discussed and achievements of our research are highlighted. Section also motivates further work in the area.

2 Background on Similarity

This section discusses major studies of similarity assessment in three research areas - philosophy, cognitive psychology and computer science. This partial history of similarity theories is used as a motivation for a novel theory of similarity - context-based similarity - proposed in later sections.

2.1 Philosophical Background

Similarity was first studied by philosophers (Hume, 1947).[1] David Hume foresaw that if objects are similar in appearance then they will be attended with similar effects; thus, from causes which appear similar we expect similar effects.[2]

Hume (Hume, 1947) considers surface similarity (see Section 2.2.1) to be the only existing similarity. He believed that when assessing similarity, it is sufficient to consider only simple sensory attributes of objects and he did not consider different perception of these attributes by different subjects or by the same subjects but in different context. This approach can be called *common*

[1] In Hume's studies, the term resemblance was used instead of similarity, but this makes no real difference.

[2] There are, however, also problems where this does not work this way.

attribute view of similarity. His approach equates potential (surface) similarity with psychological similarity and thus neglects perceptual capacities of the organisms and assumes common environmental properties.

Hume's views are limited when considered deeply. According to him, the degree of similarity of two composite ideas depends on the number of simple ideas they have in common. However, Hume assumed that the similarity between simple ideas is given to us directly. The reasons for the failure of this definitions are presented in (Wallach, 1958). Wallach's work is considered to be the first modern study of similarity (see Section 2.2).

Hume also observed, that all arguments from experience are founded on the similarity which we discover among natural objects. According to Hume and Külpe (Külpe, 1895), two objects can be treated similarly because they are similar to each other (according to Külpe, there is only a slight difference between objects) or because both are similar to a third object. Külpe, moreover, defines another class of similarity - partial identity.

Bain (Bain, 1855) was a clear supporter of the similarity as an important psychological problem. He introduced a "Law of Similarity" which he states as follows:

> *The tendency to be reminded of past occurrences and thoughts of every kind, through their resemblance to something present, is here termed the Law of Principle of Similarity.*

In Bain's work, resemblance is used as an undefined primitive term to define similarity. Similarity is used as one of two principles to explain learning (the other one is contiguity). He proposes that classifications are assembled by the notion of similarity. Bain considers identity to be the limiting factor of similarity.

Similar to Bain's views on identity and similarity are ideas presented by Fullerton (Fullerton, 1890). His work leaved the way open to scale similarity over the range from identity to complete dissimilarity. In his words *similarity in the ordinary sense of the word is "identity in diversity" which implies a recognition of two things as two.* This can be thought of as a variant of partial identity similarity.

James (James, 1890) proposes that two things are resembling each other because of the absolute identity in respects to some attribute or attributes, combined with the absolute identity of the rest of their being. His views are useful when one considers similarity in context (see Section 2.5).

Goodman (Goodman, 1951) considers similarity as a four-place predicate, noting the non-transitivity of similarity matches. An important aspect of his work is that he understood the importance of the question "how similar are the compared objects?"; he states:

> *In keeping with its [similarity] ordinary use we cannot properly ask whether two qualities are similar but only how similar they are or whether they are more or less similar than some other pair.*

According to (Carnap, 1967) a resemblance relation is called a similarity relation if it is symmetrical and reflexive, and an equivalence if it is also transitive (see Section 3.3 for other opinions). Carnap also introduces *similarity circles* which are a primitive topological notion about classes of similar things.

2.2 Cognitive Psychology Background

Substantial work on similarity has been carried out by psychologists and cognitive scientists (e.g., (Wallach, 1958; Tversky, 1977; Gick and Holyoak, 1980; Gentner, 1983; Holyoak, 1985; Thagard and Holyoak, 1989; Vosniadou and Ortony, 1989; Nosofsky, 1990a; Gentner and Forbus, 1991; Keane, Ledgeway and Duff, 1991; Markham and Gentner, 1991; Wharton et al., 1992; Gentner and Bowdle, 1994)). In this section, we briefly review some of the more relevant literature in this area.

Hume's views (see Section 2.1) are limited when considered deeply. The reasons for the failure of his definition of similarity are presented in (Wallach, 1958). Wallach's work is considered to be the first modern study of similarity.

Tversky (Tversky, 1977) presents several examples where he describes features of similarity. However, his formalism does not allow him to express all of them (e.g., context) which resulted in the invalidation of symmetry and transitivity theorems. He also did not treat features of dissimilarity to a sufficient extent and considered it only as a negation of similarity.[3] For more detailed discussion of Tversky's approach see Section 2.5.

In (Nosofsky, 1990a) a distinction between independent-cue and interactive-cue (context model) models is highlighted and it is concluded that the latter model is superior to the former one. There are two main reasons for the superiority:

1. it is an exemplar-based model (as opposed to prototype-based model); and
2. an interactive (multiplicative) similarity rule is used (as opposed to additive similarity rule).

There are number of different situations where similarity is used and thus it is believable that there are different kinds of similarity. One of the most used distinctions is made between *surface similarity* and *deep similarity*. The former one is based on easily perceivable attributes of objects; the later is based on not so readily accessible attributes. It should be noted, however, that the distinction between surface and deep similarity is not clear cut; what is surface similarity in one context (or in one domain) can easily be deep similarity in another context (or in another domain). Moreover, according to (Medin and Ortony, 1989), there is a causal relation between deep and surface similarities; the former one constrains and sometimes even generates the later one. There is another problematic issue - objects can be described by different set of attributes. There are two reason for this:

1. not all people perceive objects the same way (e.g., color-blind person will perceive color as shades of grey); and
2. with changed situation (changed context) needs vary as well (e.g., describing apple as a food will require different set of attributes as opposed to the situation when apple is used as an artistic object).

In the next sections we present useful measurements of similarities which have shown to be of interest to the research community.

[3]According to Tversky, $similarity = 1 - dissimilarity$.

2.2.1 Surface Similarity

Wallach (Wallach, 1958) in his effort to revise the earlier work by Hume (see Section 2.1) defines *potential similarity* as the measure of the number of common environmental features that two objects or events are found to display.

According to (Medin and Ortony, 1989), *surface similarity* is based on readily accessible components of objects. A similar definition is presented in (Davies and Thomson, 1988, page 292), where an *euphoric similarity* is defined as a similarity of the perceptual or conceptual characteristics of an item and the characteristics of some memory representation.

Gentner (Gentner, 1983; Gentner, 1989), treats surface similarity as similarity in object attributes (one-place predicates characterizing simple descriptions of objects) as opposed to the similarity in relations (complex, relational properties of objects).

In (Rips, 1989) surface similarity is called *perceptual similarity* and is defined as cognitively primitive and well defined. It is argued that because of these features of perceptual similarity, it can be used as a construct to explain other psychological functions (e.g., categorization).

Vosniadou (Vosniadou, 1989), suggests using *salient similarity* instead of surface similarity to refer to similarity grounded in easily retrievable aspects of representations. It does not need to be descriptive attributes. Rather, it can be similarity in relational or conceptual attributes, as long as these attributes are easily accessible. Salient similarity allows for retrieval of inter-domain analogies. She also suggests that the surface/deep similarity distinction is a dynamic one because over time the representations may change and thus may change the salience of attributes.

It is also noted (see (Vosniadou, 1989)) that even though a property can be described as an object attribute, it does not necessarily mean that it is also represented in people's representation of the particular object or that it is an easily accessible property.

2.2.2 Deep Similarity

According to (Medin and Ortony, 1989), *deep similarity* is based on more central, core properties of objects.[4] This is in line with the experimental results presented in (Brewer, 1989). Ross observed that even though surface similarity is used more often, structural similarities are more successful in enabling the students to solve a new problem.

Gentner (Gentner, 1983; Gentner, 1989) considers deep similarity as *structural similarity*, similarity based on the relational structure (two-or-more-place predicates characterizing relations). (Holyoak and Thagard, 1989) propose that what constitutes structural similarity are the identities that influence goal attainment. Vosniadou (Vosniadou, 1989) notices that relational properties of objects may be particularly salient even for young children. This shows not only that surface similarity can be salient, but that relational properties can also be easily accessible. Moreover, the salience of attributes may change over time.

It was observed (see (Chi, Feltovich and Glaser, 1981)) that experts are able to perceive the underlying similarities in problems, whereas novices are not capable to see the same similarities mainly because the difference in surface attributes. A nice example showing this differences is presented in (Vosniadou, 1989). Subjects were asked to interpret statements such as: "Clouds

[4]It should always be considered in the light of a particular knowledge representation because, as was already stated, objects do not have attributes; the attributes are assigned to them in the representation.

are like a sponge." The experimental results show that children used mainly surface features (attributional interpretation) and explained the sentence as "Both are round and fluffy", whereas adults produced mainly relational interpretations and explained the sentence as "Both can hold water and later give it back".

As for the connection between surface and deep similarities, it is claimed in (Medin and Ortony, 1989) that the surface similarity serves as a heuristic for where to look for deeper attributes. Another use of surface similarity for deep attributes is to constraint on the predicates that compose the mental representation.

2.2.3 Psychological Similarity

The first modern study of *psychological similarity* is presented in (Wallach, 1958). The reasons for definition of this kind of similarity come from the criticism of Hume's work on resemblance (Hume, 1947). Wallach pointed out two main reasons why Hume's views on similarity are limited:

1. failure to note the full range of attributes in terms of which objects and events can be grouped; and
2. failure to separate potential similarity from psychological similarity.

Wallach studied the psychological aspects of similarity: the presence of a particular attribute is neither a necessary nor sufficient condition for the similarity judgments. On the one hand, a particular attribute may be present and yet the similarity judgment may not be made. On the other hand, a particular attribute may be lacking and yet the items may be judged similar.

In (Wallach, 1958), four different definitions of psychological similarity are presented:[5]

1. Psychological similarity defined in terms of *common environmental properties* (Hume, 1947): In this situation, one can change the similarity by varying the common values of attributes.
2. Psychological similarity defined in terms of *common responses*: The ground for this definition is based on observation that there are situations where attribute-value pair is present in two items and yet different response may exist to them; or the attribute-value pair may be absent and yet a common response be made. In other words, if a person responded the same way to two different situations, then those situations are considered psychologically similar. However, according to several experiments, the presence or absence of common responses does not guarantee the presence or absence of psychological similarity.
3. Psychological similarity defined in terms of *primary stimulation gradients*: This similarity definition is based on the measurement of the initial stimulus distance from the old stimulus: the farther they are the less similar are items to each other. One of the best known studies in this area is in (Pavlov, 1927).
4. Psychological similarity defined in terms of assigning items to a common categories. A particular set of attributes allows for discriminating items and assigning them to the classes.

[5]If the context is considered (see Section 2.5), the applicability problem of individual definitions is changed, since the context specifies what should be compared when similarity is assessed.

Varying either the nature of the instances or the classification bias predictable influences psychological similarity.

2.2.4 Analogical Similarity

Analogical reasoning involves the transfer of relational information from the source domain to the target domain. The basic classification of analogies is based on the distinction between *intra-domain analogies* and *inter-domain analogies*. It was observed that for these two types of analogical reasoning different similarity assessments should be deployed. According to (Vosniadou and Ortony, 1989), for intra-domain analogies, the likehood of retrieving an analog increases when the surface similarity between the source and the target is increased. For inter-domain analogies, the retrieval of analogs can be based on salient similarity. Because, some relational knowledge may be involved for the retrieval of inter-domain analogies, it was observed (e.g., (Brown, 1989; Gentner, 1989; Holyoak and Thagard, 1989)) that experts and adults have better access to intra-domain analogs than novices and children.

According to (Gentner, 1989; Vosniadou, 1989), intra-domain analogies are considered as *literal similarity* rather than analogies. In (Gentner and Forbus, 1991) structural and superficial similarities are considered as literal similarity (mundane), where superficial similarity (also dumb similarity) is strictly based on surface similarities. The reason presented is than intra-domain analogies are similar in many simple, descriptive, and non-relational properties, in contrast to inter-domain analogies, where relational properties play a central role.

According to (Holyoak and Thagard, 1989), even though both structural and surface similarities have comparable effects on spontaneous analogical transfer, only the former has a significant impact on total analogical transfer once a hint was provided. Thus, surface similarity has a greater impact on retrieval of a source analog than on application of an analog (Holyoak and Thagard, 1989; Vosniadou, 1989). However, because inter-domain analogies do not share similarity in surface attributes, they can be accessed only through similarity in their relational attributes. This also explains the difficulty to use inter-domain analogies since people are more sensitive to similarity in descriptive properties than to similarity in structural aspects (Vosniadou, 1989).

It is believed (see (Vosniadou and Ortony, 1989)) that surface similarity may often enhance the probability of using analogical reminding by helping to notice that problems belong to the same problem type. Similarity in salient attributes between the source and the target can lead to discovery of a productive analogy (Vosniadou, 1989).

According to (Ross, 1989; Vosniadou, 1989; Spiro et al., 1989), reasoning by similarity and analogy itself can make the knowledge base more flexible and facilitate the learning of general rules and the acquisition of new schemata; thus, analogical reasoning allows for restructuring the knowledge base. However, reasoning by analogy may fail if the representational structures on which the analogical mechanism operates is not appropriate (Vosniadou, 1989).

Even though the analogical mapping may not require the presence of the underlying structure in the target domain (see (Vosniadou, 1989)), it is required that enough is known about the target domain to make such a mapping feasible.

2.2.5 Conclusions and Motivation

From the presented review of literature on similarity, the following can be concluded:

- People's representation of the particular object vary and there is also a change in reasoning if the environment changes. Thus, a flexible similarity assessment algorithm should be used in similarity-based reasoning systems. The flexible mechanism will allow for attention focusing and for changing similarity measure by including context in the similarity definition.
- It is possible to diminish the difference between similarity in object attributes and similarity in object relations by using a special representation language, as suggested in (Mylopoulos et al., 1990). Here, relations and objects are treated equally.
- As will be shown in latter sections (see Section 4.1) context-based similarity has two levels. The first level - an equivalence of items - can be called a surface similarity. The second level - similarity between contexts - can be called deep similarity.
- Analogical similarity studies show that the useful similarity-based system should allow for applying constraints on values and for attribute mapping.

2.3 Computer Science Approach

To date in computer science, not much attention has been given to the theoretical foundation of similarity. Rather, the focus has been on implementing psychologically plausible theories of similarity. Previously defined theory was usually tweaked to meet system's requirements (e.g., (Watanabe, 1985; Russell, 1986; Ashley, 1989; Bareiss and King, 1989; Collins and Michalski, 1989; Golding and Rosenbloom, 1989; Porter, 1989; Russell, 1989; Whitaker, Wiggins and Klein, 1989; Cheng, 1991; Jagadish, 1991; Leake, 1991; Brown, 1992; Leake, 1992; Skalak, 1992; Suzuki, Ohnishi and Shigemasu, 1992; Lauzon and Rose, 1994; Michalski, 1993; Kashyap and Sheth, 1993; Sun, 1995)).

Watanabe (Watanabe, 1985) presented similarity assessment in the area of pattern recognition. Even though he realized the importance of context in similarity, his formalism was limited only to a distance measures of similarity (this approach is used extensively in many other systems (Fertig and Gelertner, 1991; Pankakoski et al., 1991; Brown, 1992; Tanaka, Hattori and Sueda, 1992; Féret and Glasgow, 1993; Skalak, 1993)). In our framework, the distance measure can be viewed as a limited version of similarity in comparison: the distance is a predefined numerical value, specifying the similarity between a current and the most restrictive contexts. Thus, one could say that the apple and pear are similar to the degree of 0.7. Even though this is sufficiently limited by not providing a semantical measure of similarity, there is another limitation; namely, systems based on distance measures do not include context in similarity assessment, which makes apple and pear being equally similar in different situations. One possibility to overcome this drawback is to use multiple (or local) contexts (Kolodner, 1993).

The problem of similarity has also been studied for spatial databases (Jagadish, 1991). This author proposes an organization for a database of objects that permits an efficient retrieval of objects with a shape similar to an input shape. For similarity judgments, an area-based similarity is used: two shapes are similar if the area where they do not match (error area) is smaller than error margin when one shape is placed on top of the other. The error margin is not constant and is used to control the amount of retrieved items.

Michalski (Michalski, 1993) uses similarity as one of the possible transmutations - a form of analogical inference. He defines similarity with respect to context (either implicitly or explicitly defined). However, he does not define features of similarity and dissimilarity. A way of using similarity and dissimilarity relations for inductive and deductive inferences is also provided.

Veloso and Carbonell (Veloso and Carbonell, 1993) define analogical similarity as a substitution σ, such that the equivalence between literal l and l' is defined as follows: $l = \sigma(l')$. In their work, they considered only similarity which we described as similarity in analysis. They provide two versions of this similarity - a direct similarity and a foot-print similarity. The former one is a general notion of similarity in analysis and the latter one is a specific variant of the former one, being tied to the goals and preconditions according to the derivational trace of the solution.

Kashyap and Sheth (Kashyap and Sheth, 1993) present an approach to resolve schematic differences among semantically related objects in multi-database systems. They define semantic proximity as an attempt to characterize the degree of semantic similarity between two objects using the real world semantics. The key part of their definition of semantic similarity is explicitly represented context. Another use of their approach is to represent uncertain information and to resolve data value incompatibility in multi-database system.

Sun (Sun, 1995) suggests combining rule-based reasoning and similarity-based reasoning. The need for similarity in rule-based systems comes up from the existence of situations where there is no directly applicable knowledge. In such a case, the system may refer to similar concepts, propositions or situations if they occur in the same context. Following (Collins and Michalski, 1989), it is suggested that similarity be implemented as rules since rule application and similarity matching need to be performed at the same time. Similarly to the work described elsewhere (Rubin, 1992; Stanfill and Waltz, 1986; Waltz, 1989), it is suggested that similarity be computed in parallel.

2.4 Flexible Similarity Assessment

Another way to classify similarity assessment approaches is to compare their implementation as algorithms. Two possible approaches to implementing similarity-based retrieval systems are:

1. Similarity relations among items are predefined. According to our definition of similarity (see Definition 3.4 in Section 3), this approach can be characterized as a limited similarity in retrieval since the context is fixed or only a limited flexibility is allowed. In (Kolodner, 1993) a similar approach is called global since all cases are retrieved and ranked based on a global matching strategy.

2. Similar items are located by defining similarity relations at query time, allowing for flexibly changing context and criteria for matching. Such relations are defined as similarity in retrieval. In general, similarity is a relation with three parameters: a set of relevant items, a context), and an information base (for formal definition see Section 3). Similarity in retrieval is used when the context is specified and the task is to retrieve all relevant items from the information base; thus, an information base and a context is input and a set of items is an output.

   This approach is similar to the local matching criteria (Kolodner, 1993) since several possible strategies can be used for matching. The difference is that most of the previous approaches use numerical measures of similarity and thus a semantic explanation of similarity may not be available.

The first approach is obviously not flexible and is used to answer the question "are the items similar?" and not the question "how are they similar?" (e.g., in which context are they similar). Because the similarity relation among items is embedded into the system, the context is static and thus the system provides a sound inference only in this built-in context.

The second approach is more favorable because it allows for changing context and gives us additional semantic information about compared items in the form context used for actual similarity assessment. Thus, it allows for answering questions of the form "how similar are the items?" or "in which context are the items similar?". Flexibility and additional semantic information are crucial for reasoning since what is relevant in one situation may not be relevant in another situation. Thus, we need flexible similarity assessment and we need to know on what grounds the items are similar. Knowing how the items are similar and exploiting the monotonicity of similarity with context (see Section 4) allows for automatic control over the number of retrieved items.

A central problem in similarity-based retrieval systems is assessing the relevance of the retrieved information to the current goal, specified through the query. This problem becomes more difficult to solve when automatic retrieval tools are used. The difficulty is caused by automatic fine-tuning of the initial (user-defined) query to obtain the best result (see Section 4.1 for more details on the proposed way to assess the relevance of retrieved items to the initial query). Similarity theories developed so far do not allow for flexible relevance assessment of the system's response.

2.5 The Role of Context

In various research areas context is defined differently (sometimes different terminology is used, e.g., views in databases or aspects in pattern recognition). The motivation for considering context is its ability to bring additional knowledge to the reasoning process and thus focus attention on relevant details (Light and Butterworth, 1993). When retrieving items, we might want to consider all attributes used to describe them or only certain subsets of them (in different situations different subsets may be relevant).

The problem of context in analogical reasoning was presented in (Collins and Burstein, 1989) as "an important test of any of these (analogical) computer models":

> *[An important test] is whether they can select two different correspondence mappings from a source domain depending on what aspects of the source domain are relevant to the target domain. None of the models has, as yet, addressed this central problem directly.*

Everyday experience also suggests that context brings additional information to the reasoning process. Even though the role of context is accepted by the research community, most of the current systems use for determining similarity only a predefined context (or a closed set of contexts).

As a simple example, consider an information base consisting of people's records. Let us suppose that the information base is used by a head-hunter agency. It is relatively straightforward to specify what kind of features should be represented in the information base. However, it is also apparent, that if one wants to find a person for a programming job in a prestigious company, important features for retrieval of relevant information (items that correspond to perspective job candidates) would be skills such as: experience in programming, ability to work in a team, etc. Physical appearance is usually not an issue. If a new model is needed for a local fashion store, however, then the most important feature might be physical appearance (more often than not programming skills of the candidate would not count as a plus). Thus, even though in both cases the general task is the same - to find a person right for a job - different criteria is applied in order to retrieve the desired information. In other words, only certain parts of the representation are used during individual retrievals. Context is used as a mechanism to filter out all irrelevant pieces of information and to specify the criteria for matching.

Similarity judgments are made with respect to representations of entities, not with respect to the entities themselves (Medin and Ortony, 1989). Thus, having a changeable representation (i.e., having an option to add and remove attributes of an object representation flexibly), one can make any two items similar according to some criteria (see Example 3.1). To prevent this, a context may be used as a way to focus the similarity assessment only on a certain features of the representation.

Context also allows for excluding irrelevant information. This feature is particularly important in unusual situations where similarity may be affected by "usual" context, e.g., context in which the items appear most of the time.

2.5.1 Context Dependency

Tversky (Tversky, 1977) discussed dependency of similarity on context and frame of reference, however, his formalism did not capture this dependency. Tversky did not consider changed or implicitly-assumed context as a phenomenon affecting similarity judgment.

Another illustration of context playing a central role in determining similarity is the example presented in (Tversky and Gati, 1982). Here, faces with fewer features are rated as more similar to faces with many features. The easy explanation of this experimentally proved phenomenon is that in the former case, subjects considered all features to be the context, whereas in the latter case, there was no specification of the context and one could not use all features to determine similarity.

In (Barsalou, 1989, page 77), a nice example supporting the context dependency of similarity assessment is presented.[6] It is found that "something heavy" is a better cue than "something with a nice sound" for retrieving the sentence "The man lifted the piano." In contrast, "something with a nice sound" is a better cue than "something heavy" for retrieving the sentence "The man tuned the piano." This shows how different attributes of the concept piano are important in different tasks, which can be represented by changed context. The author believes that certain knowledge about a category - context-dependent information - becomes active only if relevant in the current context. Moreover, it is believed that category representations vary across contexts and the properties composing these representations also vary. We are not convinced about this, since the knowledge representation scheme should be rich enough to allow for the representation of objects in different contexts. However, we agree that the similarity judgment should be dynamic and only a relevant portion of the representation should be considered in a particular context.

It would be safer to consider the "context-independent" information as information which automatically activates a specific context for a certain object each time this object is accessed, regardless

[6] The author presents the example to explain the instability of similarity judgments.

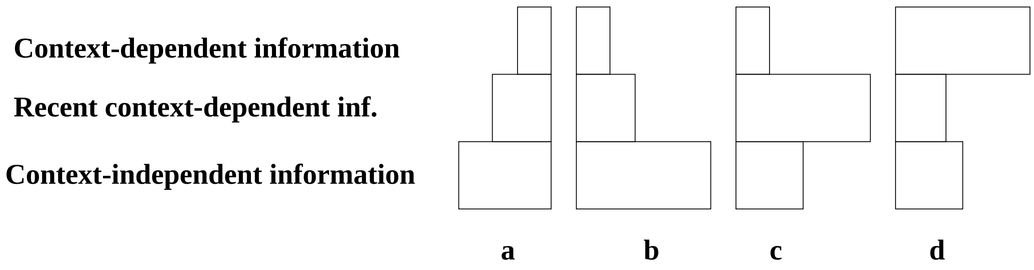

Figure 1: *Effects of different experiences on context-dependent, context-independent, and recent context-dependent informations.* **a** *represents the situation* **never owned a dog**. **b** *represents the situation* **owned a dog a long time ago**. **c** *represents the situation* **recently acquired first dog**. **d** *represents the situation* **never owned a dog, but thinking about dogs from the Eskimo point of view**.

of the current context. Thus, "context-dependent" information would not activate a specific context and would be dependent on the current context.

As an example, consider context-dependency implied in the common saying: *One cannot compare apples to oranges.* This statement is used many times in everyday life with the meaning that the two things are incomparable (because they are different). However, it is a false statement in general, unless one assumes the implicit context. If it is specified that we want to compare furniture, apples, oranges and car with respect to the attribute specifying edibility – apples and oranges are not only comparable but the are similar in this context.

Certain information automatically activates a specific context for a certain object each time this object is accessed, regardless of the current context. This is the result of stimulus bias (Barsalou, 1989; Nosofsky, 1990b). For example, if one mentions the word, skunk, it is most often the case that the notion of an unpleasant smell is activated automatically without any reference to the current context, what ever it may be. This is possible since the particular attribute has a high accessibility. Note also that the stimulus bias varies between individuals and can also change over time. This is the reason why similarity assessment needs to be flexible and why it does not suffice to have predefined similarity relations.

Context-dependent information can be either represented with the object or can be inferred. In the later case, when reading about zebras one may infer that zebras have ears from the knowledge that zebras are mammals, and because all mammals have ears, zebras have ears.

Besides using context-dependent and context-independent information Barsalou (Barsalou, 1989) also defined *recent context-dependent information.* According to psychological studies, context-dependent information can become context-independent, once activated, but without subsequent reinforcement it will loose this status. Barsalou (Barsalou, 1989) proposes that the similarity of two objects depends on the amount of overlap between their context-independent, context-dependent, and recent context-dependent information. The relative weighting of the information may vary based on contexts (see also Figure 1, an example originally presented in (Barsalou, 1989)). In this example, amount of present context (in)dependent information is shown for four different situations. It shows, how experience can affect information dependency on context.

Consider the comparison of a child to its parents. Usually it is considered odd to say that the parents are similar to the child. The reason is that usually we know parents earlier than children (obviously, if the situation is opposite, i.e., we know children earlier than parents, then odd comparison would seem to be to relate parents to children). Again, the reason is that one assumes an implicit context. If the context is stated explicitly, one can compare the similarity of parents to child as well - e.g., we can state that son is similar to his father because both have brown eyes and black hair (later we will describe features of similarity, e.g., symmetry and transitivity; see Section 3.3).

Considering above mentioned examples, we may define three kinds of dependencies of similarity assessment on context: permanently and temporally context-independent similarities and context-dependent similarity (for formal definitions of similarity relation and context see Section 3).

Definition 2.1 (Permanently context-independent similarity) *Similarity assessment becomes permanently context-independent if regardless of changing the explicit context, similarity depends only on the implicitly assumed context.*

Definition 2.2 (Temporally context-independent similarity) *Similarity assessment becomes temporally context-independent if similarity is assessed for the same objects during the short interval τ. Then, there is a psychologically explainable tendency to use the old context in a new situation. The length of τ depends on the strength of the relation, frequency of the repetition and other psychological factors. After enough occurrences of temporal independence it may become a permanent context independence.*

Definition 2.3 (Context-dependent similarity) *Similarity assessment is context-dependent if the similarity changes with changed (explicitly stated) context.*

2.5.2 Context and Relevance

As was mentioned earlier, context allows for flexible information retrieval. Retrieving of useful information is accomplished as follows:

- by attention focusing on relevant parts of knowledge;
- by locating both exact matches and similar items.

There is little commonalities between definitions of relevance used in different systems (Greiner, 1994). The most common ground is reached when relevance is defined as something useful. Our definition of relevance is as follows:

Definition 2.4 (Relevance) *Retrieved items are considered relevant if they are similar to the query with respect to current context.*

In other words, Definition 2.4 states that context is used as a relevance measure. Moreover, even though similarity is neither symmetric nor transitive in general, context allows for judging when similarity is transitive, symmetric and monotonic (see Section 3.3). These properties can advantageously be used during relevance assessment.[7]

[7] The same properties can be stated for dissimilarity. Even though dissimilarity (as a measure of irrelevance) is not symmetric in general (Subramanian and Genesereth, 1987), our approach allows for determining the circumstances where dissimilarity is symmetric. Moreover, it has been shown that dissimilarity is not reflexive, is not transitive, and is monotonic (Jurišica and Lauzon, 1994).

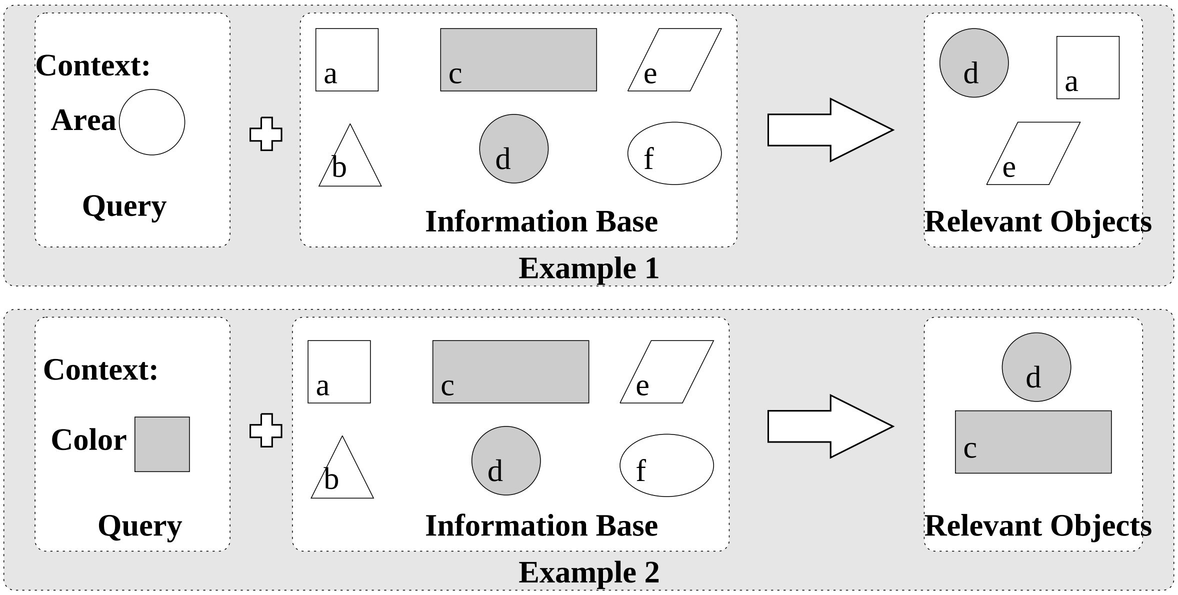

Figure 2: *An effect of changed context on similarity.*

Another motivation for context to be used as a part of similarity is that context usually changes for different situations or when a different view is used (Acker and Porter, 1992; Kashyap and Sheth, 1993) (see also Section 3). Using a finite representation of an item (a realistic assumption), a changing context may change the similarity results (see Figure 2). The query specifies context and the goal is to retrieve all relevant items from an information base, i.e., all items which are similar in the specified context. On the one hand, if the context is the *area*, as specified by the circle, then items a, e, and d are similar (see Example 1). On the other hand, if the context is changed to $A\ color : dark$ then items c and d are considered similar (see Example 2). Even though this is an artificial example, similar situations occur in the real life. The criteria for similarity measure often change with changed tasks or reasoning situations.

Using example in Figure 2, we argue that systems with predefined similarity relations, i.e., the systems with context stated implicitly, work properly only in the implicit context. Even though this inflexibility may not be a problem in some domains, where the context stays unchanged during the reasoning process, it is a limiting factor for some applications, such as software repositories. Here, context might be required to change because user needs changed or because repository evolved.

2.5.3 Harmful Context

Even though we presented several reasons why and when context is helpful in problem solving, there are also situation when the use of context may result in misperception. In (Davies and Thomson, 1988, page 294-5), several psychological experiments are described. It was observed, that when a new person was introduced in a previously seen context, the false recognition was increased. Another interesting observation revealed that adults are less faulty in the recognition (33% wrong) than youth (50% wrong), which can be explained by a higher amount of knowledge possessed by adults. These experiments lead to the following conclusion:

1. context is implicated in the mechanism which generates false recognition;
2. the interactive effects of context and retention interval operate differently for false recognition and correct recognition; and
3. the effect of context on false recognition over time seems to depend on the type of context.

Because of the harmful effect of the context in some situations, it is sometimes required to use the process of decontextualization (Davies and Thomson, 1988). The need for this process is apparent from the existence of two kinds of memory:

1. **episodic** memory is based upon memories of specific experience; and
2. **semantic** memory is primarily abstract knowledge.

Abstract knowledge is derived from the specific experience via decontextualization. The contextually referenced knowledge is changed into semantically referenced knowledge. This process may also be referred to as generalization since it involves the process of extending learning to situations beyond the original learning context. The major problem here is the fact that episodic memory depends upon environmental context; thus, decontextualization must overcome these obstacles.

The generalization operation may create analogs - a problem's isomorphs. The utilization of such knowledge is not straightforward since a context-free object has a limited value if the utilization is not context-free (Lockhart, 1988).

3 A Theory for Context-Based Similarity

This section describes a novel theory of similarity - context-based similarity. As was previously argued, context plays a central role in determining similarity. We will formally define necessary parts of similarity definition, define similarity relations for different tasks and show features of similarity, such as symmetry, monotonicity and transitivity.

Context-based similarity allows for retrieving not only exact but close matches as well. Context specifies how close retrieved items are, i.e., it can be perceived as a measure of usefulness. Thus, context in context-based similarity is useful during the process of judging how relevant the returned answer is to the current goal.

3.1 Definitions

Definition 3.1 (A source item) *A source item (A) is a representation of a real world object in a particular representation scheme. An item will be represented as a finite set of attribute/value pairs:*

$$A \equiv \{a_0 : V_0\}, \{a_1 : V_1\}, \cdots, \{a_n : V_n\} \equiv \{a_i : V_i\}_0^n \equiv \{A_i\}_0^n,$$

where $\{a : V\}$ is an attribute/value pair (A), a is an attribute, and V is a value (possibly consisting of a list of values).

Definition 3.2 (Information base) *The symbol Δ will represent an information base - a search space of source items.*

Definition 3.3 (Context) *The symbol Ω will denote a context. In general, context is defined as a finite set of attributes with associated constraints on the attribute values. The constraints on attribute values are specified either as a "allowed" values (e.g., values which should be present if the attribute matching is to occur) or as a "prohibited" values (e.g., values considered which should not be present if the attribute matching is to occur). Formally:*

$$\Omega \ \equiv \{a_i : V_i\}_0^p,$$

where: $V \equiv \{v_i\}_0^m$ or $V \equiv \{V' \ |!\{v_i\}_0^m\}$, v is the value which must match between the source and the target items ("allowed" value) and $!v$ is the value which must not match between the source and the target items ("prohibited" value). Note that a valid context may be specified only by a set of attributes without values - no constraints are imposed. The meaning of $\{V' \ |!\{v_i\}_0^m\}$ is that a particular attribute can have values from a set V', except values $\{v_i\}_0^m$.

Having this flexible definition, context allows for specifying a variety of situations, ranging from completely ignoring the value of an attribute, through considering only certain ("allowed") values for an attribute, to considering a certain set of "allowed" values except some set of "prohibited" values. In the next sections we try to demonstrate that this allows for flexible retrieval specification and for a simple method for fine-tuning queries.

An item A viewed in an explicit context Ω will be denoted A_Ω. Using this notation, we can express example from Figure 5 as follows:

$$Item \ = \ RWO_{\Omega_{IC}}$$

$$View \ = \ Item_{\Omega_{EC}}$$

where RWO is a real world object, $Item$ is a representation of a real world object, Ω_{IC} is an initial context imposed by the representation scheme and $View$ is a view of the item in an explicitly stated context Ω_{EC}.

Definition 3.4 (Similarity Relation) *Similarity is defined as a relation of three parameters: set of items (SI), a context (Ω) and an information base ($\Delta \supseteq SI$).*

$$Similarity\ relation: \ sim(SI, \Omega, \Delta).$$

In the following section, similarity relation will be modified for different tasks. To allow for this, two parameters (SI and Ω) will be either constant or variable. A constant parameter means that it is fully specified in the query - it is an input. (Note that information base is an input.) A variable parameter means that it is used as an output.

3.2 Similarity Relation

In this section we present a similarity relation defined for four different tasks: comparison, retrieval, evaluation and analysis. Considering Definition 3.4, parameters Ω and SI can be either constant

Task	*Similarity Relation*	*Function*
Comparison	$sim_c(SI, \Omega, \Delta)$	$SI \subseteq \Delta \times \Omega \rightarrow T \mid F$
Retrieval	$sim_r(var\_SI, \Omega, \Delta)$	$\Omega \times \Delta \rightarrow SI$
Evaluation	$sim_e(SI, var\_\Omega, \Delta)$	$SI \subseteq \Delta \rightarrow \Omega$
Analysis	$sim_a(var\_SI, var\_\Omega, \Delta)$	$\Delta \rightarrow \{SI, \Omega\}$

Table 1: Similarity in different tasks.

or variable, depending on the task they are used in. This definition is flexible enough to be widely applicable (see Table 1, where **task** specifies activities in which a particular similarity may be used, **similarity relation** defines input and output operators, and **function** defines a domain and a range of similarity operators). In the next section we define all similarity functions, but the most emphasize will be given to retrieval function.

3.2.1 Comparison Function

For similarity in comparison (sim_c) both context and the set of items are specified. An example of this may be to ask if apples are similar to pears when considering optimal climate for growing them. Formally, we define the comparison function as follows:

Definition 3.5 (Similarity in comparison) *Given an information base Δ, a context Ω and a set of items $\{X_0, X_1, \ldots, X_n\} = SI \subseteq \Delta$, $sim_c(SI, \Omega, \Delta)$ will return T iff $\forall X_i \in SI : X_{i_\Omega} \equiv X_{j_\Omega}; i, j = 0, \ldots, n$ and $i \neq j$. The function will return F iff $\exists X_i \in SI : X_{i_\Omega} \not\equiv X_{j_\Omega}; i, j = 0, \ldots, n$.*

Extensions to the comparison function will be discussed after other similarity functions are defined (see Section 3.2.5).

3.2.2 Retrieval Function

Similarity in retrieval (sim_r) is used when the context is specified and the task is to retrieve all relevant items from the information base; thus, the set of items is a variable. Formally, we define retrieval function as follows:

Definition 3.6 (Similarity in retrieval) *Given an information base Δ and a context Ω, $sim_r(var\_SI, \Omega, \Delta)$ implies that $\forall X \in \Delta$, $X \in var\_SI$ iff $\exists X_\Omega$.*

A possible use of similarity in retrieval is when one has to locate all relevant items in an information base and it is impossible to locate them directly. In such a situation, it is necessary to specify a certain view (context) which should be used to locate relevant (similar) items.

Consider the following example (based on Example 3.1): suppose one wants to retrieve all similar countries to Cuba. If a geographical context is specified, Jamaica would be one of the retrieved countries; if a political affinity is the context, Russia would be one of the retrieved countries. It is apparent that applying different contexts to the same knowledge base may result in retrieving different items.

The complexity of the retrieval task is characterized by the number of required comparisons, which is, in the worst case, $\mid \Omega \mid \times \mid \Delta \mid$. Classification of cases may improve the efficiency of performing the task by limiting the search space of the information base.

3.2.3 Evaluation Function

The purpose of similarity in evaluation (sim_e) is to evaluate the similarity between given items by finding possible contexts in which they are deemed similar. Hence, context is a variable.

In many real world applications the objective is to find the most restrictive context which satisfies a user specified threshold (since there is an infinite number of possibilities). Formally, we define the evaluation function as follows:

Definition 3.7 (Similarity in evaluation) *Given an information base* Δ *and a set of items* $\{X_0, X_1, \ldots, X_n\} = SI \subseteq \Delta$, *then* $sim_e(SI, var\_\Omega, \Delta)$ *implies that* $\Omega' \in var\_\Omega$ *iff* $\forall X_i \in SI : X_{i_{\Omega'}} \equiv X_{j_{\Omega'}}; i, j = 0, \ldots, n$ *and* $i \neq j$.

Considering the Example 3.1 one could ask how is Cuba similar to Russia. The returned answer would be that both are similar according to political affinity.

The complexity of similarity in evaluation depends on the number of considered concepts and number of features used to describe them; $\mid X \mid \times \mid SI \mid$ comparisons is required, where $\mid X \mid$ represents the number of attributes of an item.

3.2.4 Analysis Function

Similarity in analysis (sim_a) can be considered as a way of knowledge mining, i.e., a way of finding interesting relations within an information base (Cook and Holder, 1994; Milosavljevic and Jurka, 1993; Arikawa et al., 1993; Parsaye et al., 1991; Matheus, Chan and Piatetsky-Shapiro, 1993; Frawley and Piatetsky-Shapiro, 1991; Langley, Simon and Bradshaw, 1990; Hamilton, 1990). Formally, we define analysis function as follows:

Definition 3.8 (Similarity in analysis) *Given an information base* Δ, *then* $sim_a(var\_SI, var\_\Omega, \Delta)$ *implies that* $\Omega' \in var\_\Omega$ *iff* $\forall X_i \in var\_SI : X_{i_{\Omega'}} \equiv X_{j_{\Omega'}}$; $i, j = 0, \ldots, n$ *and* $i \neq j$ *and* $\forall \Omega_k : \Omega_k \subseteq \Omega'$; $k = 0, \ldots, p$.

This similarity computation is the hardest one since both the whole knowledge base and all features used to describe cases need to be considered; $\mid X \mid \times \mid \Delta \mid$ comparisons is required, where $\mid X \mid$ represents the number of attributes of an item. Extensions to the analysis function are discussed in Section 3.2.5.

3.2.5 Extensions

Assume the comparison function evaluates to false. The system may automatically apply either retrieval or evaluation similarity. In the first case, the system would return all items from the original set of items which satisfy the current context. In the latter case, the system would return relaxed context. In both cases, the process may be iterative - either excluding items from the set of items, or excluding constraints from the context.

Making retrieval function more flexible involves automatic context modification - either constraining or relaxing. These issues will be discussed theoretically (see Section 3.3.1) and practically (see Sections 4.1 and 4.3).

Evaluation function can be extended by allowing removing the "problematic" items from the set of items. We consider an item to be "problematic" if it is a cause of not finding a context.

Extending the analysis function to the most general case will generate a response consisting of a set of items ($\{SI_0, SI_1, \ldots, SI_m\}$) with corresponding most restrictive contexts ($\{\Omega_0, \Omega_1, \ldots, \Omega_m\}$). Formally:

$$\{(SI, \Omega)_0,\ (SI, \Omega)_1,\ \ldots,\ (SI, \Omega)_m\}.$$

3.3 Features of Context-Based Similarity

To support the claim of generality of our approach to similarity-based retrieval, we briefly introduce some important features of similarity in context - monotonicity, symmetry and reflexivity.

In (Jurišica and Lauzon, 1994) we present a more detailed analysis of similarity. Previous similarity theories are discussed and compared. We characterize existing kinds of similarity. New theory for similarity assessment is proposed and using this theory we prove that similarity is an equivalence relation.

3.3.1 Monotonicity

Monotonicity is a useful basis for a flexible retrieval - the more constraints we put in the query the less items would satisfy it. Consider two extremes. On the one hand, if the context is empty, then all items will satisfy it. On the other hand, if the context characterize precisely one item (assuming there are no duplications in the information base) then only one item will be retrieved. This suggests that context-based similarity is monotonic with an inverse monotonicity.

Theorem 3.1 (Monotonicity) *Context-based similarity is monotonic.*

> **Proof sketch:** Given an information base Δ, assume $\Omega_i \subset \Omega_j$ (Ω_j is more restrictive context when compared to Ω_i). Then, using Definition 3.6, $sim_r(var\_SI_i, \Omega_i, \Delta)$ and $sim_r(var\_SI_j, \Omega_j, \Delta)$ imply $var\_SI_i \supseteq var\_SI_j$. Restrictive context implies that more attribute/value pairs are required to match; thus, less items will satisfy it. Assume attribute a_i is specified in the context which results in k retrievals. It is obvious that if we add attribute a_j to the context, smaller or equal amount of items (compared to k) will be retrieved.
>
> For a relaxed context, the prove is just an opposite, instead of adding attributes, they are removed. Assume $\Omega_i \supset \Omega_j$ (Ω_j is more relaxed context when compared to Ω_i). Then $sim_r(var\_SI_i, \Omega_i, \Delta)$ and $sim_r(var\_SI_j, \Omega_j, \Delta)$ imply $var\_SI_i \subset SI_j$. Relaxed context requires less attribute/value pairs to match, thus, more items will qualify. A graphical proof is presented in Figure 3.$\square$

The usefulness of monotonicity feature during the retrieval process was reported in (Lauzon and Rose, 1994; Rissland et al., 1993). On the one hand, if not enough items have been retrieved

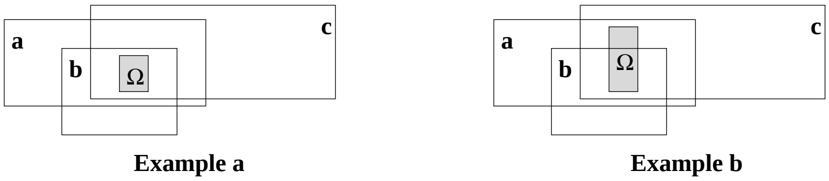

Figure 3: *Monotonicity. In the example* **a**, *all three rectangles are similar in context* Ω. *However, in the example* **b**, *when the context is more restrictive, only rectangles* **a** *and* **c** *are similar.*

from the knowledge base, the initial context may be relaxed to retrieve some "less" similar items. On the other hand, if too many items are retrieved, the initial context may be restricted to lower the number of retrieved items (see Figure 3).

3.3.2 Symmetry

In (Tversky, 1977; Smith, 1989; Gentner and Bowdle, 1994; Ohnishi, Suzuki and Shigemasu, 1994) it is argued that not all similarities are symmetric. Our claim is that similarity is symmetric if the context is preserved, regardless of direction of reasoning. All examples presented by Tversky have implicit context which is changed if subject and referent are exchanged.

Example 3.1 *Jamaica is similar to Cuba (because of geographical proximity); Cuba is similar to Russia (because of their political affinity); but Jamaica and Russia are not similar at all.*

Using the Example 3.1, Cuba is similar to Russia because of political affinity and Russia is similar to Cuba in the same context. There is no difference in these two similarities, however, there is an implicit association that Cuba is similar to Russia since Cuba followed the political system developed in Russia. To exclude this implicit association, it is more neutral to say that political systems in Cuba and Russia are similar.

Theorem 3.2 (Symmetry) *Context-based similarity is symmetric.*

> **Proof sketch:** Assume an information base Δ, a context Ω and a set of items $SI_1 \equiv \{A, B\}$, $SI_1 \subseteq \Delta$. By using Definition 3.5, $sim_c(SI_1, \Omega, \Delta)$ returns T *iff* $A_\Omega \equiv B_\Omega$.
> It is obvious from the definition, that changing SI_1 to $SI_2 \equiv \{B, A\}$, $SI_2 \subseteq \Delta$ makes no difference since $sim_c(SI_2, \Omega, \Delta)$ returns T *iff* $B_\Omega \equiv A_\Omega$.
> Assume attribute/value pair (a_i, v_i) being defined in the context Ω. If item A is similar to item B considering Ω, then (a_i, v_i) has to be both in A and in B. It makes no difference if we compare item A to B or item B to A - they are considered similar in the specified context. $\square$

3.3.3 Transitivity

Transitivity seems to be the most controversial feature of similarity. (Goodman, 1951; Tversky, 1977; Smith, 1989) are only few of the many who argue that similarity is not transitive. However, we have already shown that the context plays a central role in similarity and thus affects transitivity (see Section 2.5 and Example 3.1). Using context-based similarity, we shall show that similarity is transitive if the context is preserved (see Figure 4).

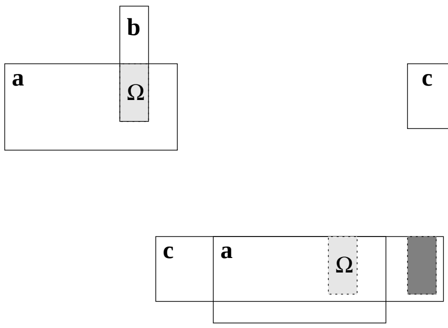

Figure 4: *Transitivity. Item* **a** *and item* **b** *are similar in context* Ω*; item* **b** *is similar to item* **c** *in context* Ω*. If the context is preserved, item* **a** *is similar to item* **c**. *It should be noted that if context* Ω *is placed outside the overlapping area, then the items are not similar; thus, there is no reason to deal with transitivity.*

An interesting class of problems, usually used to support the claim that similarity is not transitive, is presented as Example 3.2.

Example 3.2 *A widely used example to show non-transitivity of similarity is comparing two countries and claiming their similarity based on the fact that they have a common border (e.g., Canada - U.S.A.). The similarity of the U.S.A. and Mexico can be grounded on the same context - a common border. However, on the first thought, similarity is not transitive since Canada and Mexico does not share the border and thus cannot be similar in this context.*

The problem is with the example, not with the similarity. In the first part of the transitivity relation, context is common border. In the second part of the transitivity relation, the context is again the common border. However, the final part of the transitivity relation does not share the same context and thus transitivity cannot hold (for more details on context see Section 2.5).

Theorem 3.3 (Transitivity) *Context-based similarity is transitive.*

> **Proof sketch:** Assume an information base Δ, a context Ω and three sets of items $SI_1 \equiv \{A, B\}$, $SI_1 \subseteq \Delta$, $SI_2 \equiv \{B, C\}$, $SI_2 \subseteq \Delta$ and $SI_3 \equiv \{A, C\}$, $SI_3 \subseteq \Delta$. By using Definition 3.5,
>
> - $sim_c(SI_1, \Omega, \Delta)$ will evaluate to T *iff* $A_\Omega \equiv B_\Omega$;
> - $sim_c(SI_2, \Omega, \Delta)$ will evaluate to T *iff* $B_\Omega \equiv C_\Omega$;
> - $sim_c(SI_3, \Omega, \Delta)$ will evaluate to T *iff* $A_\Omega \equiv C_\Omega$.
>
> Thus, transitivity of context-based similarity can be formalized as follows:
>
> $$(A_\Omega \equiv B_\Omega) \wedge (B_\Omega \equiv C_\Omega) \Rightarrow (A_\Omega \equiv C_\Omega).$$
>
> □

Based on different types of context, there are different approaches to transitivity, which are captured in Corollaries 3.1 and 3.2.

Corollary 3.1 *Given three items A, B and C we can formally express transitivity of similarity as follows:*

$$(A \sim B) \wedge (B \sim C) \Rightarrow (A \sim C).$$

Assume context-based similarity and that the context holds for A, B and C individually, formally $(A_\Omega \sim B_\Omega) \wedge (B_\Omega \sim C_\Omega)$. *Then, it is not necessary to check if the right side of the transitivity relation holds for the context, it is guaranteed that* $A_\Omega \sim C_\Omega$.

Corollary 3.2 *Given three items A, B and C we can formally express transitivity of similarity as follows:*

$$(A \sim B) \wedge (B \sim C) \Rightarrow (A \sim C).$$

Assume context-based similarity and that the context does not hold for A, B and C individually but holds for the relations between these two pairs of items, formally $(A \sim_\Omega B) \wedge (B \sim_\Omega C)$. *Then, the transitivity relation is guaranteed only if the context holds for the relation on the right side of the transitivity relation,* $A \sim_\Omega C$.

The Corollaries 3.1 and 3.2 do have an obvious use. The first one allows to skip the context test in the transitivity relation while still guaranteeing a sound inference. The latter corollary protects us from making a false inference when the context changes.

4 Context-Based Retrieval

This section describes a similarity-based retrieval system based on context-based similarity definition. Transformations of context - relaxing and constraining - are discussed and a query language for retrieval function is presented.

4.1 Similarity in Context - Partial Order of Similarities

During the construction of an information base, a designer decides on a representation scheme. Since it is not necessarily feasible to represent all attributes of a real world object, the repository item is constrained by a limited number of information it carries. We can see this constraint as a context applied to a real world object - an implicit context (see Figure 5). During the reasoning process, an explicit context may be defined.[8] Thus, implicit context allows us to map real world object into a particular representation (an item) and explicit context allows us to define a specific view of the item in an information base. Moreover, explicit context allows for defining mapping between items defined in different schemas.

Implicit context is embedded into the system and restricts its reasoning capabilities. During the reasoning process, an explicit context may be defined to specify the reasoning goal. Even though there are no restrictions on how the explicit context is represented, the objective is to define it in the way so it is a subset of the implicit context; otherwise it will not be useful for retrieval. This modification may, however, be done by the user or by the system.

[8]We consider all contexts to be explicit, unless defined otherwise.

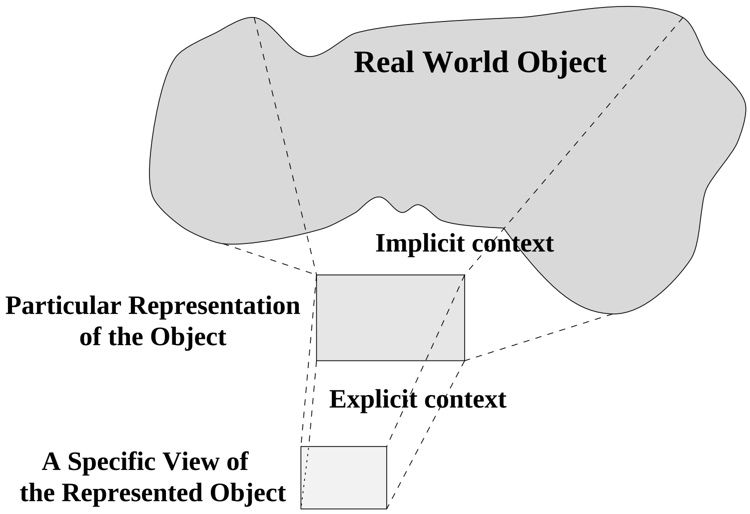

Figure 2. Implicit and explicit contexts.

Figure 5: *Implicit and explicit contexts.*

During the retrieval process, an agent (either the user or another part of the system) poses a query, where the initial context for retrieval (Ω_{init}) is specified. This context is called initial since it may be altered either by a system or the user if the retrieval is not successful.

It may be the case that during retrieval, too many or too few items are returned. In either case, the initial context needs to be adapted accordingly. Monotonicity of similarity allows for controlling the number of retrieved items by relaxing or constraining the context (see Section 4 and Section 3.3). This context-alternation process may become a chain reaction, producing successive contexts (Ω_i) and sets of items (SI_i). Because of monotonicity of similarity in context, successive contexts (Ω_i) are partially ordered. Partially ordered contexts have an effect on set of items (SI_i) - set of items are also partially ordered. Formally,

$$Assume:\ \Omega_i \subset \Omega_j, sim_r(var\_SI_i, \Omega_i, \Delta)\ and\ sim_r(var\_SI_j, \Omega_j, \Delta).$$

$$Then:\ var\_SI_i \supseteq var\_SI_j.$$

$$Assume:\ \Omega_i \supset \Omega_j, sim_r(var\_SI_i, \Omega_i, \Delta)\ and\ sim_r(var\_SI_j, \Omega_j, \Delta).$$

$$Then:\ var\_SI_i \subset SI_j.$$

After the agent is satisfied with the returned result, the question is how useful the result is. The retrieval process is depicted in Figure 6.

Considering Figure 6, a query is posed to satisfy the current goal which in turn helps to solve the current task. Because different strategies may lead to a successful retrieval, a particular goal can be achieved by asking different queries, which in turn can be described by different contexts (dotted arcs). For each goal there is only one query used, thus there is only one initial context. Without a loss of generality we will consider having only a single goal for description of the approach.

If the returned answer is not satisfactory (too many or too few items are returned) the initial

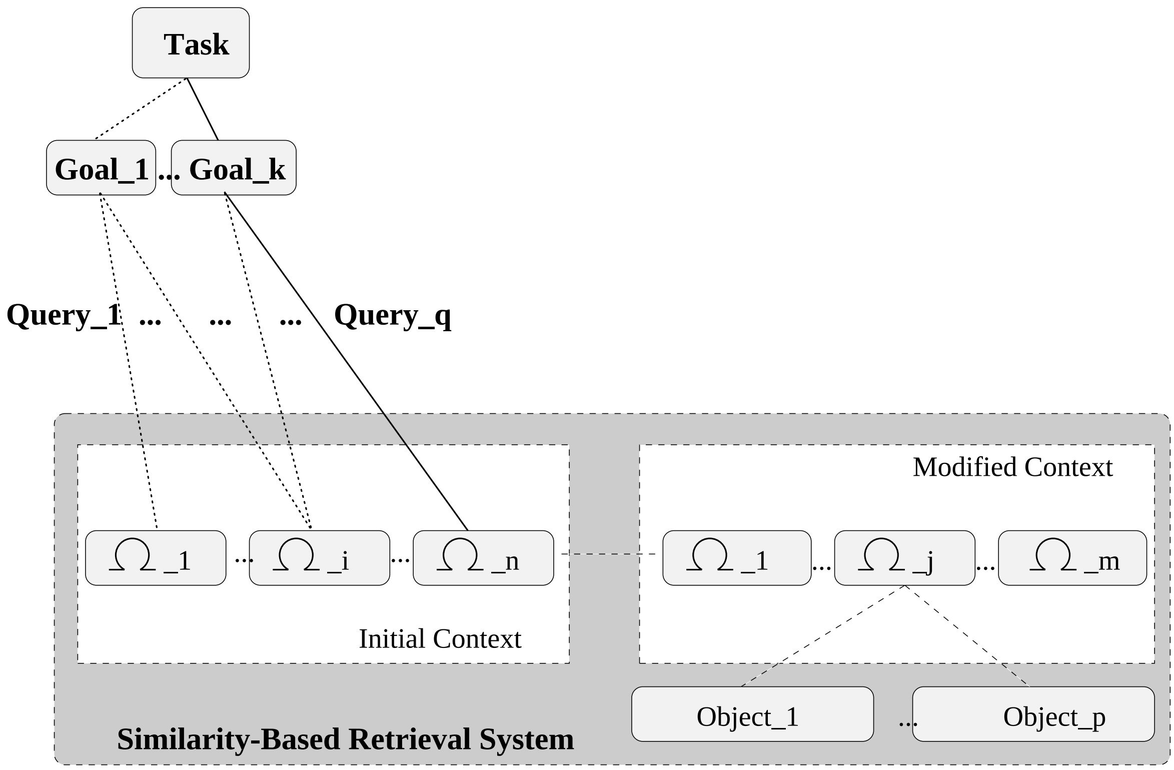

Figure 6: *Flexible approach to retrieval.*

context may be changed to obtain more suitable result. This allows for better control over the whole retrieval process, including easier specification and modification of a query.

From our definition of similarity, all items in SI_i are equivalent in Ω_i. However, if the initial context did not lead to a successful retrieval it was modified either by the system or user. In such a case Ω_{init} is different from Ω_i. To measure the relevance of the result to the initial context, the system assess the similarity of the initial and returned contexts. This allows us to define two ends of the partial order of similarity, namely:

1. **Equivalence of items** - All items in SI_i are similar because they can be treated as equivalent in context Ω_i. In this case, the initial context and the final context are equivalent ($\Omega_{init} \equiv \Omega_i$). Because this similarity assessment is straightforward we may refer to equivalence of items as surface similarity.

2. **Similarity between contexts** - If the initial context was changed, we need to assess the relevance of the final context Ω_i to the initial context Ω_{init}. If contexts Ω_{init} and Ω_i are similar in context Ω_G then all items in SI_i would be equivalent in this context. Our confidence in usefulness of retrieved items depends on the relevance of the context Ω_G to the current goal, which is specified by the initial context Ω_{init}. Thus, Ω_G can be considered to be a semantical weight of similarity. Similarity in evaluation is used to find out the similarity between contexts: ($\Omega_{init} \sim_e \Omega_i) \Rightarrow \Omega_G$). Since similarity between contexts involves more computation, we may refer to it as a deep similarity.

4.2 Retrieval System

Based on the proposed theory of similarity we developed a prototype system $\mathcal{TA}\mathbf{3}$. The system aids the retrieval process by flexibly handling user queries. A query in similarity-based retrieval system uses similarity in retrieval and specifies what similarity measure should be used during retrieval. Formally, a query is defined as follows:

$$QUERY_r \ = \ [sim_r(var\_SI, \Omega, \Delta),\ Criteria],$$

where sim_r is a similarity in retrieval, var_SI represents a retrieved set of items, Ω is a context, Δ is a search space, and $Criteria$ specifies how the context should be handled and what criteria should be used for matching during an automatic context relaxation (more details are presented in Section 4.3).

The goal of similarity in retrieval is to find all items in an information base which are similar under current context. After retrieval is completed, var_SI will contain all similar items retrieved. A frame-like language is used to represent items and each is described by a set of attribute/value pairs.

The context Ω is the most important part of the similarity function. Context consists of a list of attribute declarations. Each attribute declaration specifies attribute/value pairs used for matching. A set of constraints can be applied to attributes to control the matching process. Moreover, attribute declarations can be grouped into *Categories*, allowing different constraints to be ascribed to different groups of attributes. Each item can have one or more categories. A priority is assigned to each category (0 is the highest priority) to allow for more reliable relaxation algorithm (see Section 4.3). The priority is used during a selection process, to help decide which attributes are the most important ones and should be treated first. This flexibility allows us to specify complex (but necessary) relations among attribute declarations more easily. In addition, value constraint and cardinality constraint are defined to structure attribute declarations, namely:

1. **Value constraints** specify constraints on the individual value for a particular attribute. This constraint can be used to relax constraint set by the current context. During the matching, the system compares items in an information base to the current context, trying to find an exact match. This may not be required all the time and value constraints are used as a mechanism to allow for this relaxation. value constrints can be of the following forms:

   - **Instance** - the attribute value in the source item and the target item must be the same. Source item is an item in information base. Target item in similarity in retrieval is context. However, in similarity in comparison and similarity in evaluation, target item is an item specified in the set of items SI (context is considered as a constraint during matching between source and target items).
   - **Range** - the attribute value in the source item must be a number in a range specified in a target. This allows for exact match in cases when source value is within an interval specified in context (or in the item from the set of items, when similarity in comparison and similarity in evaluation is considered).
   - **Domain** - the attribute value in the source item must be an instance of the attribute value in the target item. This constraint allows for even more relaxation of context by

considering its domain instead of its value (Instance) or interval (Range). As an example consider value for one attribute of a source item to be *5*. The system would consider exact match for this attribute if the context for this attribute is *integer* and the value constrain for the attribute is *Domain*.

2. **Cardinality constraints** specify the number of attributes required to match for a particular category. Category groups relevant attributes together and cardinality constraint allow to consider partial matches as exact matches if so desired. The following constraints are allowed:

   - **Exact** specifies that there must be a one-to-one match of the attributes in a category between the source and the target, where *source* is the actual item in the repository and *target* is the context (or an item from a set of items if similarity in comparison and similarity in evaluation is considered). It should be noted that $Exact = Some(n)$, *iff* $n = |A|$, i.e., *iff* n is equal to number of attributes of an item (considering a current context).
   - **Some(n)** determines that not all attributes in a category are required to match, but at least n attributes of the target must match the target.

3. **Schema constraints** specify the attribute match criteria for attributes. The following modes are allowed:

   - **AttributeMode** requires that both the attribute name and the corresponding value match.
   - **MapMode** allows attribute names to be mapped (put into equivalence) and only afterwards a set of attribute values is required to match.

The rationale behind value constraints is obvious. To explain the reason for supporting cardinality constraints, consider the following example. Assume having two schemas for item representation (only relevant part of it is shown):

Schema$_1$:	Name, HomePhone, BussinessPhone
Schema$_2$:	Name, Phone.

It is clear that two items beeing defined in the two respective schemas would not match, unless we use cardinality constraint to specify, that item in the Schema$_1$ will suffice to match one of the two phone numbers with the item in the Schema$_2$.

4.3 Context Relaxation

Using the fact that similarity in context is monotonic (see Section 3.3) context relaxation allows for retrieving more items from the information base. Monotonicity in similarity in context enables us to control the number of retrieved items by relaxing or constraining the context. On the one hand, if we constrain the context - the matching criteria become tighter - less items will satisfy it. On the other hand, if we relax the context - the matching criteria become less tight - more items will match (in the extreme, if we specify no constraints in context, all items from an information base would match). Considering the formal query definition, one of the following steps may be used for relaxation:

1. **Expanding the search space** either by expanding the set of items SI (for similarity in evaluation and comparison) or by expanding the information base Δ (for all similarities).

2. **Generalization of attribute constraints** is provided by relaxing the restriction on the number of relevant attributes of the source that must match with the source. Attribute constraint generalization is provided by one of the following ways:

   - **Cardinality constraint** is generalized by lowering the number of attributes required to match between the target and source. Priority is used to decide which part of a attribute declaration should be considered first. There are two types of generalization available:
     - *Exact* generalizes to $Some(n)$, where n specifies the minimum number of attributes required to match; $n \leq k-1$, where k is the number of attributes in the specific category;
     - $Some(n)$ generalizes to $Some(n-1)$, $k > n > 1$, where k is the number of attributes in the specific category. This lowers the minimum number of relevant attributes by at least one.
   - **Schema constraint** is generalized by relaxing the matching criteria from the $AttributeMode$, where both the attribute name and the corresponding values are required to match, to the $MapMode$, where only values are relevant for a specific mapping.
   - **Value constraint** is relaxed by using semantically neighboring values for the current ones. The value constraints generalization is provided as follows:
     - **Instance** allows for relaxation by allowing that the attribute value V of the source must be an immediate generalization of V, instead of requiring it to be the same value V. Generalization of value V is obtained from the generalization hierarchy. Note that user may specify a *Range* instead of an *Instance* value constraint as a form of non-automatic relaxation.
     - **Range** generalizes from the constraint that the attribute value of the source must be in the specific range to the constraint that the attribute value of the source must be *Domain* of the *Range*.
     - **Domain** allows for relaxation of the source value (V) by allowing the attribute value to be one of the immediate generalizations of *Domain*.

The $\mathcal{TA}$**3** system supports all presented forms of context relaxation When the automatic expansion of the search space is selected, the algorithm finds super-classes for each class of the search space, selectively merge them, leaving out everything what was already checked, and then start the retrieval again. Thus, similarity-based retrieval can change the metrics on which similarity assessment is based. Exploiting the monotonicity of similarity, constraining or relaxing a context is used to guide a retrieval process. While context relaxation could be done either by the user or by the system, automatic context constraining is not yet supported by our system.

As an example, consider Figure 7 for a fictive information base on people. Initial context specifies what attributes and what values are important. For more versatility we grouped attributes into three categories. Assume that all categories have the same priority and value constraints but they have different cardinality constraints. For easier explanation we do not show source items,

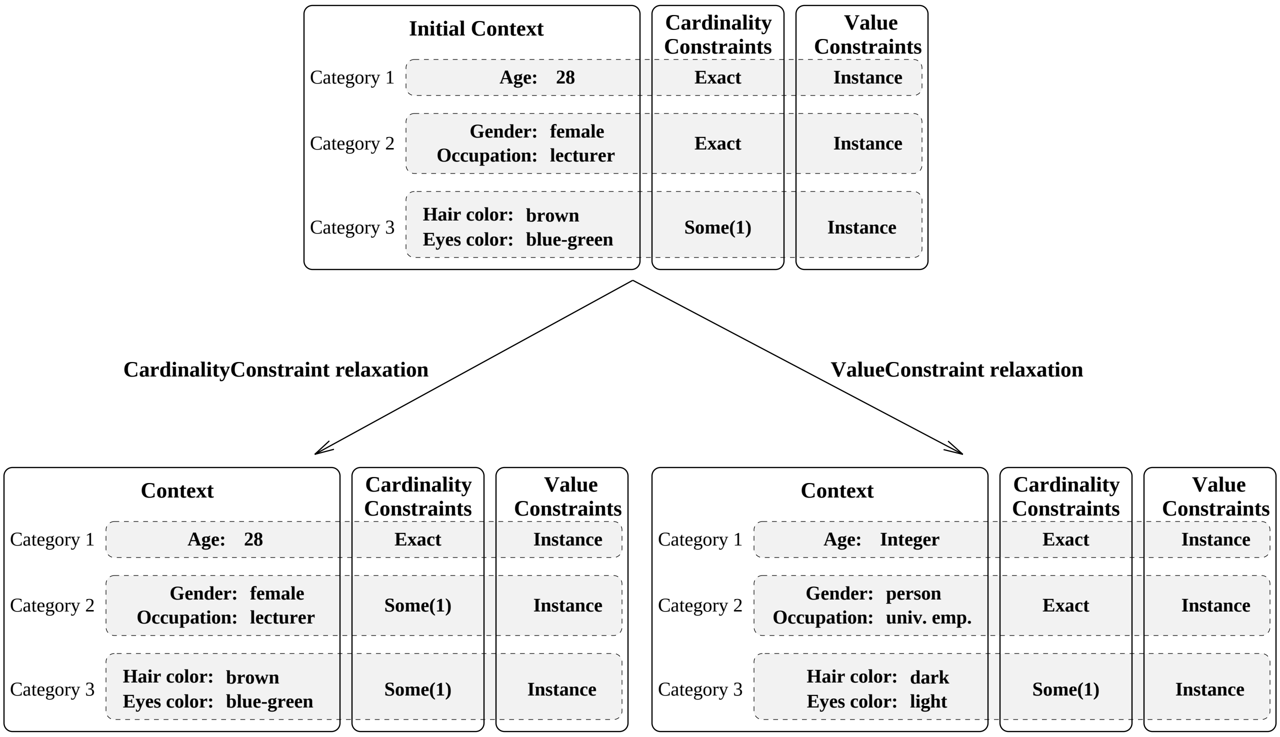

Figure 7: *Context relaxation by:* **a.** *relaxing cardinality constraints,* **b.** *relaxing value constraints.*

instead, relaxation is presented on a context. Schema constraints are not shown in the figure for the simplicity reason. It is assumed, however, that schema constraints are of AttributeMode type. The meaning is the same and the retrieval result would be equivalent as well.

In the first scenario, we relax cardinality constraints. It is not possible to relax category 1 and 3, but it is possible to relax category 2. $Some(1)$ means that at least one of the attribute/value pairs need to match for the whole category to match.

In the second scenario, value constraints are relaxed. Since all categories required *Instance* matches, using a generalization hierarchy we can update context values. Thus, *Age* would be changed to *Integer*, as an immediate generalization of 28. There is also an option not to rely on automatic relaxation. This allows for modifying context more precisely, since user could have information which is not available to the system. For example, user can relax context by changing value constraints from *Instance* (28) to *Range* $(25 - 30)$, which is not possible to do automatically. Similar generalizations are performed on other attributes.

4.4 The Similarity Query Language - SimQL

The main advantage of our system is its flexibility. This is accomplished by defining context and matching constraints within the query. Thus, the user can easily define different views on the repository, i.e., making the repository multifunctional. Moreover, the system records all similarity requests and allows for browsing through earlier contexts and criteria, having the retrieved set of items available. The query language syntax (restricted to similarity in retrieval) is presented in Table 2.

```
<Query>                     ::=  sim_r, Δ, var_SI, < Ω >, <Criteria>
< SI >                      ::=  {attribute : value}
< Ω >                       ::=  {<Category>}
<Category>                  ::=  <Priority>, <CardinalityConstraint>,
                                 {[attribute, <ValueConstraint>] |
                                 [(attribute : {value}), <ValueConstraint>] |
                                 [(attribute : {value}/!{value}), <ValueConstraint>]}
<Priority>                  ::=  Number
<CardinalityConstraint>     ::=  Exact | Some_n
<SchemaConstraint>          ::=  AttributeMode | MapMode
<ValueConstraint>           ::=  Instance | Domain | Range
<Criteria>                  ::=  {[<CardinalityConstraint> |
                                   Relax-CardinalityConstraint [, < num-of-iterations >]],
                                 {[(attribute, <ValueConstraint>) |
                                   (attribute, Relax-ValueConstraint [, <num-of-iterations>])]}}
```

Table 2: The Similarity Query Language - SimQL.

In SimQL, *Criteria* specifies how individual categories are to be treated. A cardinality constraint may be specified directly or the relaxation of it may be left to the algorithm (by specifying *Relax-CardinalityConstraint* or *Relax-ValueConstraint*). However, there is a possibility to control the number of iterations by specifying *num-of-iterations*. Note that if no number follows the specification for automatic relaxation then the algorithm will explore all possibilities available.

Priority of the category plays an important role during the relaxation. This number specifies the order in which categories are affected by the criteria. Thus, the criteria specified for category with the highest priority will be considered first. This allows for ordering the context changes.

In Table 3, SimQL is used to specify a query for retrieval, where the search space consists of all objects presented in Figure 2 and two categories are specified. Extra attributes are used to introduce some complexity to items which allows us to point out advantages of categories. The first category groups attributes *color* and *shape* with values *black* and *rectangle* respectively. The category has a priority 0, exact match is required, and an attribute mode is to be considered. The second category is defined similarly.

Criteria specifies that during an automatic relaxation, value constraint of an attribute *color* may be changed allowing the attribute value in the source to be an instance of at least one of the attribute values in the target. Moreover, the cardinality constraint for the category *manufacturer* may be relaxed in three iterative steps (and not more). Because category *surface* is defined to be the most important (it has a priority 0) criteria dealing with it will be applied first. The next category is used only when this relaxation is not successful. The relaxation process terminates when sufficient number of items is retrieved or when there is no option for automatic relaxation left.

```
Similarity
  sim_r
Search Space
  all objects presented in Figure 1.
Result
  var_SI

Context
  Category Surface, 0, Exact
    color : black, Instance
    shape : rectangle, Instance

  Category Manufacturer, 1, Some_2
    manufacturer : XCo, Instance
    price : 2much, Instance

Criteria
  Category Surface
    color, Domain
  Category Manufacturer, Relax_CardinalityConstraint, 3
```

Table 3: An example of a query in SimQL.

5 Discussion

The research on similarity assessment described in this paper has been conducted on both a theoretical and an application levels. Following, we discuss several representative applications of similarity-based retrieval in different domains and present some concluding remarks.

Watanabe (Watanabe, 1985) presented similarity assessment in the area of pattern recognition. Even though he realized the importance of context in similarity, his formalism was limited only to a distance measure of similarity (this approach is used extensively in many other systems). In our framework, the distance measure can be viewed as a limited version of similarity in retrieval where similarity relations are predefined and thus the system is single context. Moreover, a numerical value is assigned to all similarity relations among items. Thus, one could say that the apple and pear are similar to the degree of 0.7. Even though this is sufficiently limited - there is no semantic information available to explain "how the items are similar?", there is another limitation; namely, systems based on distance measures do not include context in similarity assessment which makes apple and pear being equally similar in different situations.

In (Hammond, Converse and Marks, 1990) and other works on case-based reasoning, relevant (similar) items are retrieved based on indexing. Even though this approach works in several specific domains, the problem is with its inflexibility. The designer decides what is important for retrieval in advance and during the actual retrieval, many system do not allow to change this preference. Even if the system does allow this change, they are usually limited to a simple selection of important attributes. There are also approaches which use machine learning techniques to index the cases.

This may help but it will not solve the flexibility problem completely. In case-based planning systems, e.g., CHEF (Hammond, 1990) or TRUCKER (Hammond, Converse and Marks, 1988), cases are indexed (and later accessed) based on goals they achieve and problems they avoid. Even though this may seem to be sufficient, there are situations where this need not to be the case, e.g., if one wants to retrieve cases based on different criteria (e.g., resources used, recency, frequency, etc.). In some areas it may be possible to predict all reasonable ways to retrieve relevant information. However, in the design domain (e.g., software reuse) it is hardly possible. This suggests that the system should be capable of flexibly changing the context and thus flexibly changing the rules under which relevant items are retrieved.

In the HYPO-based tutoring system (Ashley and Aleven, 1991) the relevance concepts are explicitly represented. The system is used in teaching law students to argue with cases. In order to succeed in this task, the system needs to explain the concepts of relevance. Relevant similarities and differences among cases are represented as logical expressions - called factors - using a knowledge representation language Loom (MacGregor, 1988). Factors are used during retrieval since they make cases stronger or weaker for plaintiff. The relevance concepts are used to interpret student responses, to generate feedback and to explain the reasoning process.

In the Clavier system (Hennessy and Hinkle, 1992) it was noted that the context information is critical during problem solving. However, the case representation in Clavier was not sufficient to provide reliable suggestions and the system left the final decision to a human operator. Better context handling could improve the system's reliability.

The CAReT system (Lee and Harandi, 1993) is similar to our work in the sense that it supports flexible retrieval of software components. CEReT is an analogy-based retrieval system applied to software design reuse. It also supports incomplete query specification as well as similar and exact matching of components. Its knowledge base consists of the background knowledge and the design library. The system uses $is-a$ and $part-of$ links to navigate through the library. The retrieval mechanism uses a three-phase procedure - locate-evaluate-select. Flexible retrieval is supported by allowing to change the level at which attribute matches occur. The system uses two levels of matching - object-based and attribute-based. The first level compares object classes, object type and attribute properties and their relationships to other object types to support retrieval. The second level compares attribute data types and properties. Based on similarity score obtained from matching object attributes, the similarity between objects is computed.

An expert diagnostic system FIXIT (Hart and Graham, 1994) is related to our work by its consideration of context. The system has two parts - a diagnostic system and a retrieval tool. After the diagnosis is made the system automatically retrieves all necessary maintenance manuals. This is possible through automatic context generation by a diagnostic expert system. FIXIT uses semantically-based indexing structure for the legacy full-text database of maintenance manuals to search for relevant information. In a diagnostic domain one can assume that there is only a certain set of manuals and thus only a limited number of possible contexts. However, for the software repository, a finite number of contexts would be a limiting factor, unless the tool would allow a user to refine a chosen context.

AIRS (AI-based Reuse System) (Ostertag et al., 1992) is an AI-based library system for software reuse. Even though this system was originally designed to reuse Ada packages it has evolved to a general tool for reuse. AIRS is a hybrid system that integrates faceted indexing and a hierarchical frame system. The faceted aspect describes the items in terms of a set of feature-term pairs which helps in creating a semantic structure. A hierarchical frame system is used to maintain groupings of items and relations among features. The system is based loosely on techniques developed for case-based reasoning systems. Despite many similarities, the main distinction between AIRS and case-based reasoning systems is the numerical nature and efficient computation of the heuristic estimations during similarity assessment. Similarity between two components is based on a distance measure of their respective descriptions. The distance measure is based either on the closeness or the subsumptions relation. The closeness relation is intended to capture the fact that new components can be constructed by modifying existing constructs. The subsumption relation is defined among the components of the system and is intended to capture the idea that certain components can be built by composing several other components. Based on this measure, two components are considered similar if the overall difference between their corresponding terms is small. The main restriction of the system is its inflexibility. Because the components must be described in terms of all features, adding new features would involve modifying all components which are already stored in the library.

Most of the current case-based reasoning systems do not represent context explicitly. Similarity assessment in these systems is usually based on the equivalence of cases in a predefined context. Moreover, the similarity is measured numerically. Thus, these systems cannot specify the relevance of the returned answer to the current task in a semantic way. Thus, the systems can be used only as a single information bases.

This paper proposes a new approach to similarity assessment. Context-based similarity is used for flexible retrieving of similar items and for specifying their relevance to the current goal. Context also allows for schema mappings and for multifunctional information bases. Moreover, context-based similarity allows for stating the circumstances where the relevance and irrelevance statements are reflexive, symmetric, transitive and monotonic.

Future work on context-based similarity is to provide experimental evidence on its soundness from different applications. Preliminary results from the medical domain (prediction and diagnosis) are encouraging.

References

Acker, L. and Porter, B. (1992). Extracting viewpoints from multifunctional knowledge bases. Technical report, Department of Computer Science, University of Texas, Austin, TX.

Arikawa, S., Miyano, S., Shinohara, A., Kuhara, S., Mukouchi, Y., and Shinohara, T. (1993). A machine discovery from amino acid sequences by decision trees over regular patterns. *New Generation Computing*, 11(3-4):361–375.

Ashley, K. D. (1989). Assessing similarities among cases. In *Proc. of the 2 nd Workshop on Case-Based Reasoning*, pages 72–76, Pensacola Beach, FL.

Ashley, K. D. and Aleven, V. (1991). A computational approach to explaining case-based concepts of relevance in a tutorial context. In *Proc. of the 4 th Workshop on Case-Based Reasoning*, pages 257–268, Washington, D.C.

Bain, A. (1855). *Senses of Intellect.* J.W. Parker, London, UK.

Bareiss, E. and King, J. A. (1989). Similarity assessment in case-based reasoning. In *Proc. of the 2 nd Workshop on Case-Based Reasoning*, pages 67–71, Pensacola Beach, FL.

Barsalou, L. W. (1989). Intraconcept similarity and its implications for interconcept similarity. In *Similarity and Analogical Reasoning*, pages 76–121, New York, NY. Cambridge University Press.

Brewer, W. F. (1989). Comments on Part III: The activation and acquisition of knowledge. In *Similarity and Analogical Reasoning*, pages 532–545, New York, NY. Cambridge University Press.

Brown, A. L. (1989). Analogical learning and transfer: What develops? In *Similarity and Analogical Reasoning*, pages 369–412, New York, NY. Cambridge University Press.

Brown, M. (1992). Importing similarity measures into case retrieval using analogue marker passing. In *Proc. of the 10 th ECAI*, pages 590–592, Vienna, Austria.

Carnap, R. (1967). *The logical structure of the world. Pseudoproblems in philosophy.* Routledge and Kegan Paul, London, UK.

Cheng, Y. (1991). Context-dependent similarity. *Uncertainty in Artificial Intelligence 6*, 12:41–47.

Chi, M., Feltovich, P., and Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Coognitive Science*, 5:121–152.

Collins, A. and Burstein, M. (1989). Afterword: Comments on Parts I, II, and III: A framework for a theory of comparison and mapping. In *Similarity and Analogical Reasoning*, pages 546–566, New York, NY. Cambridge University Press.

Collins, A. M. and Michalski, R. S. (1989). The logic of plausible-reasoning: A core theory. *Cognitive Science*, 13:1–49.

Cook, D. J. and Holder, L. B. (1994). Substructure discovery using minimum description length and background knowledge. *Journal of Artificial Intelligence Research*, 1.

Davies, G. M. and Thomson, D. M., editors (1988). *Memory in Context: Context in Memory.* Ellis Horwood, Chichester, UK.

Féret, M. P. and Glasgow, J. I. (1993). Hybrid case-based reasoning for the diagnosis of complex devices. In *Proc. of AAAI-93*, Washington, DC.

Fertig, S. and Gelertner, D. H. (1991). FGP: A virtual machine for acquiring knowledge from cases. In *Proc. of the 12 th IJCAI*, pages 796–802, Sidney, Australia.

Frawley, J. and Piatetsky-Shapiro, G. (1991). *Knowledge Discovery in Databases.* AAAI Press/MIT Press.

Fullerton, G. (1890). *On sameness and Identity.* University of Pensylvania Press, Philadelphia, PA.

Gentner, D. (1983). Structure-mapping: A theoretical framework for analogy. *Cognitive Science*, 7(2):155–170.

Gentner, D. (1989). The mechanisms of analogical learning. In Vosniadou, S. and Ortony, A., editors, *Similarity and Analogical Reasoning*, pages 199–241, New York, NY. Cambridge University Press.

Gentner, D. and Bowdle, B. F. (1994). The coherence imbalance hypothesis: A functional approach to asymmetry in comparison. In *Proc. of the 16 th Annual Conference of the Cognitive Science Society*, pages 351–356, Atlanta, GA.

Gentner, D. and Forbus, K. D. (1991). MAC/FAC: A model of similarity-based retrieval. In *Proc. of the 13 th Annual Conference of the Cognitive Science Society*, pages 504–509, Chicago, IL.

Gick, M. and Holyoak, K. J. (1980). Analogical problem solving. *Cognitive Psychology*, 12:306–355.

Golding, A. R. and Rosenbloom, P. S. (1989). Combining analytical and similarity-based CBR. In *Proc. of the 2nd Workshop on Case-Based Reasoning*, pages 259–263, Pensacola Beach, FL.

Goodman, N. (1951). *The Structure of Appearance.* Harvard University Press, Cambridge, MA.

Gregson, R. A. (1975). *Psychometrics of Symilarity.* Academic Press, New York, NY.

Greiner, R. (1994). *AAAI Fall Symposium Series on Relevance.* AAAI Press, Menlo Park, CA.

Hamilton, H. J. (1990). Datex: A framework for machine discovery of regularity in data. In Patel-Schneider, P. F., editor, *CSCSI-90: Proc. of the Eighth Biennial Conference of the Canadian Society for Computational Studies of Intelligence*, pages 196–203. Kaufmann, Palo Alto, CA.

Hammond, K. J. (1990). Case-based planning: A framework for planning from experience. *Cognitive Science*, 14(3):385–443.

Hammond, K. J., Converse, T., and Marks, M. (1988). Learning from opportunities: Storing and re-using execution-time optimization. In *Proc. of AAAI-88*, pages 537–540, St. Paul, MN.

Hammond, K. J., Converse, T., and Marks, M. (1990). Towards a theory of agency. In Sycara, K. P., editor, *Proc. of the Workshop on Innovative Approaches to Planning, Scheduling and Control*, pages 354–365, San Diego, CA.

Hart, P. E. and Graham, J. (1994). Query-free information retrieval. In *Proc. of the 2nd International Conference on Cooperative Information Systems*, pages 36–46, Toronto, ON.

Hennessy, D. and Hinkle, D. (1992). Applying case-based reasoning to autoclave loading. *IEEE Expert*, 7:21–26.

Holyoak, K. J. (1985). The pragmatics of analogical transfer. In Bower, G., editor, *The Psychology of Learning and Motivation*, New York, NY. Academic Press.

Holyoak, K. J. and Thagard, P. R. (1989). A computational model of analogical problem solving. In *Similarity and Analogical Reasoning*, pages 242–265, New York, NY. Cambridge University Press.

Hume, D. (1947). An enquiry concerning human understanding. In *The World's Great Thinkers. Man and Spirit: The Speculative Philosophers*, pages 341–423, New York. Random House.

Jagadish, H. (1991). A retrieval technique for similar shapes. In *Proc. of the 10th ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, pages 208–217, Denver, CO.

James, W. (1890). *The Principles of Psychology.* Holt, New York, NY.

Jurišica, I. and Lauzon, D. (1994). Similarity assessment; A unified approach. Technical report, Department of Computer Science, University of Toronto, Toronto, Canada. In preparation.

Kashyap, V. and Sheth, A. (1993). Semantics-based information brokering. Technical Report DCS-TR-307, Department of Computer Science, Rutgers University.

Keane, M. T., Ledgeway, T., and Duff, S. (1991). Constraint on analogical mapping: The effects of similarity & order. In *Proc. of the 13th Annual Conference of the Cognitive Science Society*, pages 275–280, Chicago, IL.

Kolodner, J. L. (1993). *Case-Based Reasoning.* Morgan Kaufmann, San Mateo, CA.

Külpe, O. (1895). *Outlines of Psychology.* Swan Sonnenschein, London, UK.

Langley, P., Simon, H. A., and Bradshaw, G. L. (1990). Heuristics for empirical discovery. In Shavlik, J. W. and Dietterich, T. G., editors, *Readings in Machine Learning*, pages 356–372. Kaufmann, San Mateo, CA.

Lauzon, D. and Rose, T. (1994). Task-oriented and similarity-based retrieval. In *Conference on Knowledge-Based Software Engineering*.

Leake, D. B. (1991). ACCEPTER: A program for dynamic similarity assesment in case-based explanation. In *Proc. of the 4^{th} Workshop on Case-Based Reasoning*, pages 51–62, Washington, D.C.

Leake, D. B. (1992). Constructive similarity assessment: Using stored cases to define new situations. In *Proc. of the 14^{th} Annual Conference of the Cognitive Science Society*, pages 313–318, Bloomington, IN.

Lee, H.-Y. and Harandi, M. T. (1993). An analogy-based retrieval mechanism for software design reuse. In *Proc. of the Eight Knowledge-Based Software Engineering Conference*, pages 152–159, Chicago, IL.

Light, P. and Butterworth, G., editors (1993). *Context and Cognition: Ways of Learning and Knowing*. L. Erlbaum Associates, Hillsdale, NJ.

Lockhart, R. S. (1988). Conceptual specificity in thinking and remembering. In Davies, G. M. and Thomson, D. M., editors, *Memory in Context: Context in Memory*, Chichester, UK. Ellis Horwood.

MacGregor, R. M. (1988). A deductive pattern matcher. In *Proc. of AAAI-88*, pages 403–408, St. Paul, MN.

Markham, A. B. and Gentner, D. (1991). Commonalities, differences and the alignment of conceptual frames during similarity judgements. In *Proc. of the 13^{th} Annual Conference of the Cognitive Science Society*, pages 287–292, Chicago, IL.

Matheus, C. J., Chan, P. K., and Piatetsky-Shapiro, G. (1993). Systems for knowledge discovery in databases. *IEEE Transactions on Knowledge and Data Engineering*, 5(6):903–913.

Medin, D. and Ortony, A. (1989). Comments on Part I: Psychological essentialism. In *Similarity and Analogical Reasoning*, pages 179–195, New York, NY. Cambridge University Press.

Michalski, R. S. (1993). Inferential theory of learning as a conceptual basis for multistrategy learning. *Machine Learning*, 11(2):3–151.

Milosavljevic, A. and Jurka, J. (1993). Discovery by minimal length encoding - a case study in molecular evolution (vol 12, pg 75, 1993). *Machine Learning*, 13(1):153.

Mylopoulos, J., Borgida, A., Jarke, M., and Koubarakis, M. (1990). Telos: Representing knowledge about information systems. *ACM Transactions on Information Systems*, 8(4):325–362.

Nosofsky, R. M. (1990a). Exemplars, prototypes, and similarity rules. Technical Report 17, Indiana University, Department of Psychology, Bloomington, IN.

Nosofsky, R. M. (1990b). Stimulus bias, assymetric similarity, and classification. Technical Report 14, Indiana University, Department of Psychology, Bloomington, IN.

Ohnishi, H., Suzuki, H., and Shigemasu, K. (1994). Similarity by feature creation: Reexamination of the asymmetry of similarity. In *Proc. of the 16^{th} Annual Conference of the Cognitive Science Society*, pages 687–692, Atlanta, GA.

Ostertag, E., Hendler, J., Díaz, R. P., and Braun, C. (1992). Computing similarity in a reuse library system: An AI-based approach. *ACM Transactions on Software Engineering and Methodology*, 3(1):205–228.

Pankakoski, J., Eloranta, E., Luhtala, M., and Nikkola, J. (1991). Applying case-based reasoning to manufacturing systems design. *Computer Integrated Manufacturing Systems*, 4(4):212–220.

Parsaye, K., Chignell, M., Khoshafian, S., and Wong, H. (1991). Intelligent data base and automatic discovery. *Neural and Intelligent Systems Integration*, pages 615–628.

Pavlov, I. (1927). *Conditioned Reflexes.* Oxford University Press, London, UK.

Porter, B. W. (1989). Similarity assessment: Computation *vs.* representation. In *Proc. of the 2nd Workshop on Case-Based Reasoning*, pages 82–84, Pensacola Beach, FL.

Rips, L. J. (1989). Similarity, typicality, and categorization. In *Similarity and Analogical Reasoning*, pages 21–59, New York, NY. Cambridge University Press.

Rissland, E. L., Daniels, J. J., Rubinstein, Z. B., and Skalak, D. B. (1993). Case-based diagnostic analysis in a blackboard architecture. In *Proc. of AAAI-93*, Washington, DC.

Ross, B. H. (1989). Remindings in learning and instruction. In *Similarity and Analogical Reasoning*, pages 438–469, New York, NY. Cambridge University Press.

Rubin, S. H. (1992). Case-based learning - A new paradigm for automated knowledge acquisition. *ISA Transactions*, 31(2):181–209.

Russell, S. (1986). Quantitative analysis of analogy by similarity. In *Proc. of AAAI-86*, pages 284–288, Philadelphia, PA.

Russell, S. (1989). *The Use of Knowledge in Analogy and Induction.* Morgan Kaufmann Publishersm Inc., San Mateo, CA.

Skalak, D. B. (1992). Representing cases as knowledge sources that apply local similarity metrics. In *Proc. of the 14th Annual Conference of the Cognitive Science Society*, pages 325–330, Bloomington, IN.

Skalak, D. B. (1993). Using genetic algorithm to learn prototypes for case retrieval and classification. In *Proc. of the AAAI-93 Workshop on Case-Based Reasoning*, Washington, DC.

Smith, L. B. (1989). From global similarities to kinds of similarities: The construction of dimensions in development. In *Similarity and Analogical Reasoning*, pages 146–178, New York, NY. Cambridge University Press.

Spiro, R. J., Feltovich, P. J., Coulson, R. L., and Anderson, D. K. (1989). Multiple analogies for complex concepts: Antidotes for analogy-induced misconception in advanced knowledge acquisition. In *Similarity and Analogical Reasoning*, pages 498–531, New York, NY. Cambridge University Press.

Stanfill, C. and Waltz, D. L. (1986). Toward memory-based reasoning. *Communications of the ACM*, 29(12):1213–1228.

Subramanian, D. and Genesereth, M. R. (1987). The relevance of irrelevance. In *Proc. of the 10th IJCAI*, pages 416–422, Milan, Italy.

Sun, R. (1995). Robust reasoning: Integrating rule-based and similarity-based reasoning. *Journal of Artificial Intelligence.* To appear.

Suzuki, H., Ohnishi, H., and Shigemasu, K. (1992). Goal-directed processes in similarity judgement. In *Proc. of the 14th Annual Conference of the Cognitive Science Society*, pages 343–348, Bloomington, IN.

Tanaka, T., Hattori, M., and Sueda, N. (1992). Use of multiple cases in case-based design. In *Proc. of the 8th IEEE Computer Society Conference on Artificial Intelligence Applications*, pages 233–239, Monterey, CA.

Thagard, P. R. and Holyoak, K. J. (1989). How to compute semantic similarity. In *Proc. of the 2^{nd} Workshop on Case-Based Reasoning*, pages 85–86, Pensacola Beach, FL.

Tversky, A. (1977). Features of similarity. *Psychological Review*, 84(4):327–352.

Tversky, A. and Gati, I. (1982). Similarity, separability, and the triangle inequality. *Psychological Review*, 89:123–154.

Veloso, M. M. and Carbonell, J. G. (1993). Derivational analogy in PRODIGY: Automating case acquisition, storage,and utilization. *Machine Learning*, 10(3):249–278.

Vosniadou, S. (1989). Analogical reasoning as a mechanism in knowledge acquisition: A developmental perspective. In *Similarity and Analogical Reasoning*, pages 413–437, New York, NY. Cambridge University Press.

Vosniadou, S. and Ortony, A. (1989). *Similarity and Analogical Reasoning.* Cambridge University Press, New York, NY.

Wallach, M. A. (1958). On psychological similarity. *Psychological Review*, 65(2):103–116.

Waltz, D. (1989). Is indexing used for retrieval? In *Proc. of the 2^{nd} Workshop on Case-Based Reasoning*, pages 41–44, Pensacola Beach, FL.

Watanabe, S. (1985). *Pattern Recognition. Human and Mechanical.* John Wiley, New York.

Wharton, C. M., Holyoak, K. J., Downing, P. E., Lange, T. E., and Wickens, T. D. (1992). The story with reminding: Memory retrieval is influenced by analogical similarity. In *Proc. of the 14^{th} Annual Conference of the Cognitive Science Society*, pages 588–593, Bloomington, IN.

Whitaker, L. A., Wiggins, S., and Klein, G. A. (1989). Using qualitative or multi-attribute similarity to retrieve useful cases from a case base. In *Proc. of the 2^{nd} Workshop on Case-Based Reasoning*, pages 345–347, Pensacola Beach, FL.