Appeared in Russell Greiner (Ed.): Proceedings of the AAAI Fall Symposium Series on Relevance, pp. 101-104. New Orleans, Louisiana, November 4-6, 1994

How to Retrieve Relevant Information?

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

Abstract

Relevant information is required in all kinds of information-based systems. It is not a trivial task to locate the right piece of information since it may be difficult (or impossible) to specify query requests precisely and completely. Thus, a flexible retrieval algorithm is required, allowing for an imprecise query specification.

The document presents an approach to judging relevance of retrieved information based on a novel approach to similarity assessment. Contrary to other systems, we define relevance measures (context in similarity) at query time. This is necessary if since without a context in similarity one cannot guarantee that similar items will also be relevant.

1 Introduction

In some situations it may be difficult or impossible to locate information directly. In such cases, one may use similarity-based retrieval system to find relevant approximations to the required information. There are two types of similarity-based retrieval systems:

1. Similarity relations among items are predefined. According to our definition of similarity [JL94], this approach can be characterized as a limited similarity in retrieval since the context is fixed. Thus, similarity holds only in a predefined context which is the measure used in relevance assessment.
2. Similar items are located by defining a similarity relation during query time, allowing for changing context and criteria flexibly. This relation is defined as a similarity in retrieval. In general, similarity is a function with three parameters: a set of items (SI), a context (Ω), and an information base ($\Delta \supseteq SI$ (see Section 3). All retrieved items are relevant in the current context.

If the similarity relations are predefined then the system is not flexible. Such systems can be used to answer the question "are the items similar?" but cannot be used to answer the question "how they are similar?". Moreover, similarity between items in the information base may be changed in different situations because of context change [Jur94] and this system would not capture this change effectively.

The second approach is more favorable because of flexibility it provides and because it gives us additional information for relevance judgment. It is this additional information which allows us to answer the question "how are the items similar?".

Similarity theories developed so far (e.g., [Tve77, Hol85, Lea92, SOS92, Mic93]) do not support relevance assessment if the context is changed. In order to support flexible retrieval of relevant information, we include context in our definition of similarity.

2 The Role of Context

Similarity judgments are always made with respect to representations of entities, not with respect to the entities themselves [MO89]. It is known that similarity is context dependent [Mic93]. Systems with predefined similarity relations predefine context as well. However, if the formalism for similarity assessment does not capture context (e.g., [Tve77]) then one cannot model similarity changes flexibly.

Context in similarity allows for finding information approximation by attention focusing on relevant parts of knowledge.Thus, what is similar in a specified context is considered relevant. In other words, *context in similarity is a relevance measure.* Even though similarity is neither symmetric nor transitive in general, context allows for judging when similarity is transitive, symmetric and monotonic [Jur94].

In the example presented in Figure 1, the goal is to retrieve all items, similar in the specified context, from an information base. On the one hand, if the context is *area*, as large as the specified 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 changes with changed tasks or reasoning situations.

In formal notation, A represents a real world object

*Supported by a Research Assistantship from the Department of Computer Science. I also gratefully acknowledge receiving many helpful comments from John Mylopoulos, Janice Glasgow and David Lauzon.

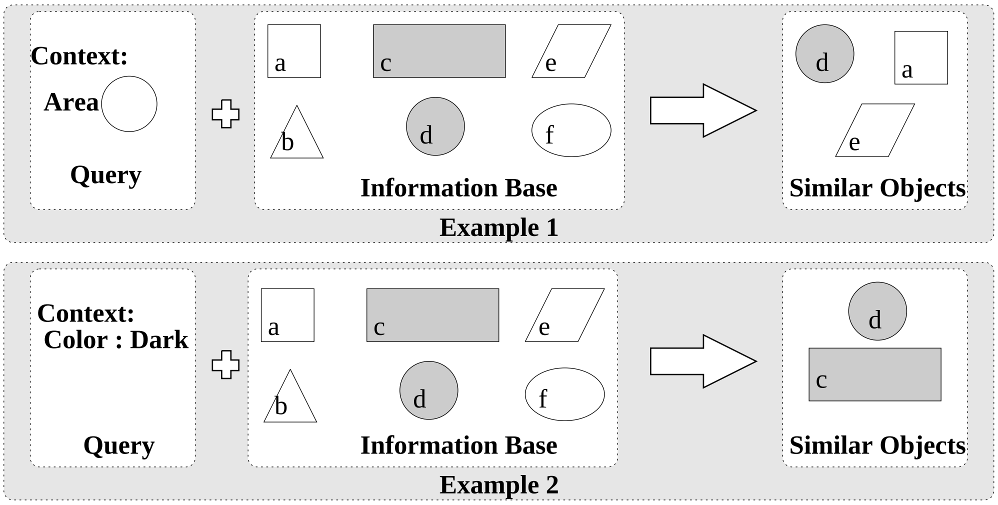

Figure 1: *An effect of changed context on similarity.*

in a particular representation scheme. A is represented as a finite set of attribute-value pairs, where a particular attribute can have more than one value:

$$A \equiv \{a_i : V_i\}_1^n \equiv \{A_i\}_1^n,$$

where $\{a_i : V_i\}$ is a set of attribute-value pairs, a_i is an attribute, and V_i is a set of values: $V_i \equiv \{v_j\}_1^m$.

Ω represents context. In general, context is defined as a finite set of attribute-value pairs. However, for our notion of similarity we need more flexibility. This can be achieved by grouping individual attribute-value pairs in a specific way. The enhanced context definition, which includes *category* for grouping attributes, is as follows:

$$\Omega \equiv \{category_j, \{a_i : V_i\}_{i=1}^k\}_{j=1}^l,$$

where: $V_i \equiv \{v_j\}_0^m$ or $V_i \equiv \{V_i' \ |!\{v_j\}_0^m\}$, v is the value which must match between the source and the target items ("allowed" value) and $!v$ is the value which should not match between the source and the target items ("prohibited" value). Note that a valid context may specify only attributes, leaving out attribute values. $\{V_i' \ |!\{v_i\}_0^m\}$ is used to represent situations where attribute can have values from a set V_i', except values $\{v_i\}_0^m$.

Thus, context allows us to specify relevance measures flexibly. Relevant items are located based on what is specified in context. The relevance of the retrieved information can be assessed even if the original query is automatically fine-tuned [Jur94].

3 Similarity Function

Using the presented approach, we define similarity as a function with three parameters: a set of items (SI), a context (Ω) and an information base ($\Delta \supseteq SI$). A set of items and a context may be either a constant or a variable (var_SI, var_Ω). This definition is flexible enough to be widely applicable in different tasks, e.g., comparison, retrieval, evaluation and analysis (see [Jur94]). The same similarity function is used in these tasks, with set of items and context being either variable or constant.

In this article we will discuss similarity in retrieval and evaluation (see Table 3, where **task** specifies activities in which a particular similarity may be used, **similarity** presents similarity definition and **function** defines a domain and a range of similarity operators).

Task	*Similarity*	*Function*
Retrieval	$sim_r(var\_SI, \Omega, \Delta)$	$\Omega \times \Delta \rightarrow SI$
Evaluation	$sim_e(SI, var\_\Omega, \Delta)$	$SI \subseteq \Delta \rightarrow \Omega$

Table 1: Similarity in retrieval and evaluation.

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. A possible use of similarity in retrieval is when one has to locate all relevant items in an information base and it is not feasible 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.

Similarity in evaluation (sim_e) evaluates the similarity between given items by finding possible contexts; hence, context is a variable. In the real world, however, the objective is to find the most restrictive context which satisfies user specified threshold (since there is an infinite number of other possibilities).

4 How Relevant Are the Similar Objects?

Similarity allows for retrieving approximate items from the information base. Context specifies the measure for judging closeness of these items. Moreover, context in similarity is also useful during the process of judging relevance of the returned answer to posed query. We will discuss two approaches to relevance assessment during the retrieval process.

Trivial Situation

In a trivial case, the system considers only context defined in the user's query. If the returned answer is not satisfactory, it is up to the user to make changes to the initial context and resubmit the query. Retrieved information is relevant to the posed query if user specified relevant context.

General Situation

In a more complex case, the system is able to control the amount of retrieved information by using monotonicity of similarity. Monotonicity of similarity is only guaranteed if context is used. Then, the more restrictive context is used, the less items will satisfy it and vice versa. Thus, if the returned answer is not satisfactory - either too many or only a few items are retrieved - the system may alter the initial context accordingly [Jur94], to converge to a more suitable answer:

- if less items than required are retrieved - context should be relaxed, i.e., context becomes less specific; thus, more items would satisfy it;
- if too many items are retrieved - context should be constrained, i.e., context becomes more specific; thus, less items would satisfy it.

When these alternations are performed by the system, then the question "how relevant are the retrieved items?" is more difficult to answer. Considering this complex case in general, the system may produce a chain of contexts $\Omega_i, i = 1, ..., n$ from the initial context Ω_{init} (see Figure 2, where $\Omega_{init} = \Omega_1$). Using the similarity in retrieval, this chain of contexts will result in a corresponding chain of retrieved items (SI_i).

After an agent is satisfied with returned results, the important question to be answered is "how useful is the result?" (e.g., "what relevance to the original query does it have?").

Similarity-Based Retrieval

During the construction of an information base, a designer decides on a representation scheme; thus, the implicit context is embedded into the system which restricts the reasoning capabilities of the system. 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.

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 amount of retrieved items by relaxing or constraining the context. This context-alternation process may become a chain reaction, producing successive contexts (Ω_i) and sets of items (SI_i). After the agent is satisfied with the returned result, the question is how useful the result is. The retrieval process is depicted in Figure 2.

Considering Figure 2, 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 described by different contexts (dotted arcs). If the returned answer is not satisfactory (too many or too few items are returned) the initial 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.

A query in a flexible similarity-based retrieval system specifies what similarity measure should be used during the retrieval. Formally, a query is defined as follows:

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

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.

var_SI is a variable used to collect all retrieved items. A frame-like language is used to represent items and each is described by a set of attribute-value pairs.

The context Ω consists of a list of attribute declarations. Each attribute declaration specifies attribute-value pairs used for matching. Attribute declarations can be grouped into *Categories*, allowing different constraints being applied to different attributes. This flexibility allows us to specify complex (but necessary) relations among attribute declarations more easily.

In order to judge whether the final context (Ω_n) can be used to retrieve relevant information, we use monotonicity and transitivity of similarity in context [Jur94].

From our definition of similarity, all items in SI_i are equivalent in Ω_i, however, Ω_{init} may not be equivalent to Ω_i. We need to assess similarity of the initial and returned contexts; thus, the same similarity relation can be used to determine the similarity between the final (Ω_n) and initial contexts (Ω_{init}). Finding a context in which contexts Ω_n and Ω_{init} are similar gives us a measure to determine the relevance of retrieved items to the initial query. These allows us to define two levels of similarity.

Two Levels of Similarity

Previous discussion leads to the conclusion that there are two levels of similarity, namely:

1. **Equivalence of items** - All items in the set of items SI_i are similar because they can be treated as equivalent in context Ω_i. In other words, these items can be considered similar, because they match all attribute-value pairs specified by the current context. According to our definition, similar items are considered relevant to the query.
2. **Similarity between contexts** - In the general case, the system produces a chain of contexts. Even though all items in SI_i are similar in the context Ω_i, this context may be different from the initial context. Thus, we are interested in the similarity

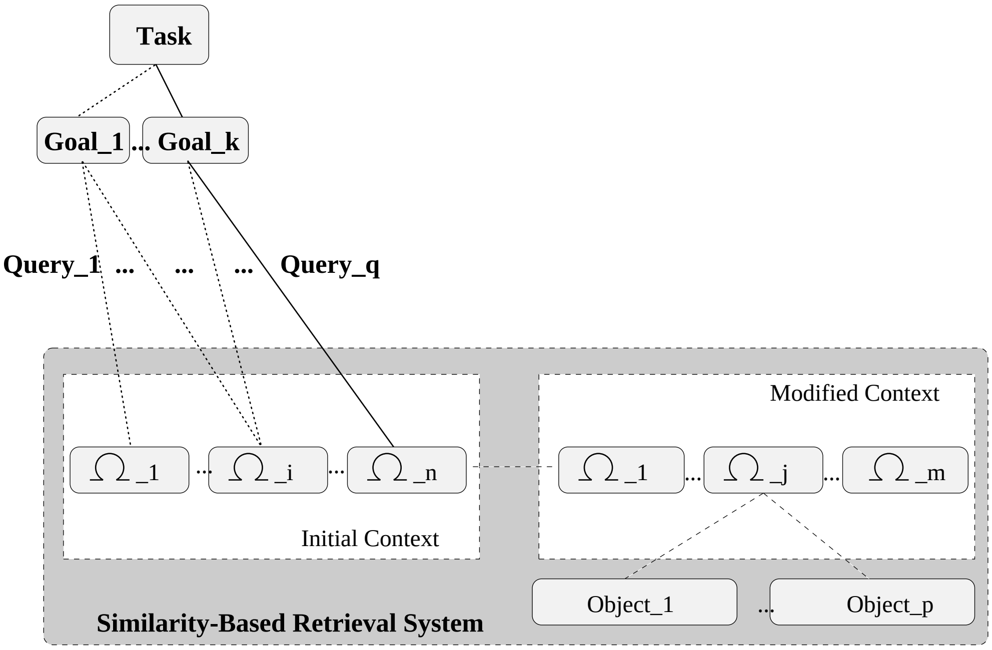

Figure 2: Flexible approach to similarity-based retrieval.

of all items in SI_i in the context Ω_{init}. This requires that Ω_i is compared to Ω_{init}. In other words, our confidence in usefulness of SO_i depends on the relevance of Ω_i to the current goal (represented by Ω_{init}). Using similarity in evaluation, the system finds the context in which Ω_i and Ω_{init} are similar: $(\Omega_{init} \sim_e \Omega_i) \Rightarrow \Omega_G$. If Ω_i is similar to Ω_{init} in Ω_G, then all SI_i objects would be equivalent in Ω_G. Thus, Ω_G can be considered a semantical similarity measure. Moreover, items similar under Ω_G are relevant to the original query.

5 Conclusion

There are many articles dealing with similarity assessment. However, most of them use only a limited notion of similarity (predefined similarity relations among items in an information base, no use of explicit context during similarity assessment, etc.). Moreover, features of similarity are usually ignored and are not studied accordingly to answer the questions "why" and "when" similarity is reflexive, symmetric, transitive and monotonic. Most of the systems built on these theories are based only on equivalence of items in a predefined context and they cannot specify the relevance of the returned answer to the current task in a semantic way. Even though these systems can be proved to be useful in a certain task they are not flexible enough to support the possible context changes.

This paper discusses a new approach to similarity assessment used in a flexible retrieval of relevant information. This approach allows for finding a semantic ground for measuring the relevance of retrieved items to the query and for stating the circumstances where the relevance statements are reflexive, symmetric, transitive and monotonic.

References

[Hol85] Keith J. Holyoak. The pragmatics of analogical transfer. In G. Bower, editor, *The Psychology of Learning and Motivation*, New York, NY, 1985. Academic Press.

[JL94] Igor Jurišica and David Lauzon. Similarity assessment; A unified approach. Technical report, Department of Computer Science, University of Toronto, Toronto, Canada, 1994. In preparation.

[Jur94] Igor Jurišica. Context-based similarity applied to retrieval of relevant cases. Technical Report DKBS-TR-94-5, University of Toronto, Department of Computer Science, Toronto, Ontario, 1994.

[Lea92] David B. Leake. 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, 1992.

[Mic93] Ryszard S. Michalski. Inferential theory of learning as a conceptual basis for multistrategy learning. *Machine Learning*, 11(2):3–151, 1993.

[MO89] Douglas Medin and Andrew Ortony. Comments on Part I: Psychological essentialism. In *Similarity and Analogical Reasoning*, pages 179–195, New York, NY, 1989. Cambridge University Press.

[SOS92] Hiroaki Suzuki, Hitoshi Ohnishi, and Kazuo Shigemasu. Goal-directed processes in similarity judgement. In *Proc. of the 14^{th} Annual Conference of the Cognitive Science Society*, pages 343–348, Bloomington, IN, 1992.

[Tve77] Amos Tversky. Features of similarity. *Psychological Review*, 84(4):327–352, 1977.