Lazy Induction of Descriptions for Relational Case-Based Learning

Eva Armengol and Enric Plaza

IIIA - Artificial Intelligence Research Institute,
CSIC - Spanish Council for Scientific Research,
Campus UAB, 08193 Bellaterra, Catalonia (Spain).
{eva, enric}@iiia.csic.es,

Abstract. Reasoning and learning from cases are based on the concept of similarity often estimated by a distance. This paper presents LID, a learning technique adequate for domains where cases are best represented by relations among entities. LID is able to 1) define a *similitude term*, a symbolic description of what is shared between a problem and precedent cases; and 2) assess the importance of the relations involved in a similitude term with respect to the purpose of correctly classifying the problem. The paper describes two application domains of relational case-based learning with LID: marine sponges identification and diabetes risk assessment.

1 Introduction

Reasoning and learning from cases is based on the concept of similarity. Often similarity is estimated by a distance (a metric) or a pseudo-metric. In addition to this, an assessment of which properties are "important" or "relevant" in the similarity is needed. This approach proceeds by a pairwise similarity comparison of a *problem* with every *precedent case* available in a case base; then one case (or k cases) with biggest (bigger) similarity is (are) selected. This process is called the *retrieval* phase in Case-based Reasoning (CBR), and also plays a pivotal role in lazy learning techniques like Instance-based Learning (IBL) and k-nearest neighbor. In classification tasks, the solution class of the *problem* is inferred from the solution class of the precedent case(s) selected.

However, distance-based approaches to case retrieval are mainly used for propositional cases, i.e. cases represented as attribute-value vectors. We are interested in this paper in learning tasks where cases are best represented in a scheme that uses relations among entities. We will call this setting *relational case-based learning*. One option to achieve case-based learning in a relational setting is to adapt the process of pairwise similarity comparison by defining a distance that works upon relational instances. This approach is taken in "relational IBL" [5] where cases are represented as collections of Horn clauses (see related work on §6).

The approach taken in this paper is different from pairwise similarity comparison based on metrics or pseudometrics. Basically, in our approach, similarity

L. De Raedt and P. Flach (Eds.): ECML 2001, LNAI 2167, pp. 13–24, 2001.

between two cases is understood as that which they "share". In addition, we need to be able to evaluate if what they share is what is important (or to which degree they share what is important). This paper presents a technique called LID for relational case-based learning. LID is based on two main notions: 1) similarity is constructed as a symbolic description of what is shared between precedent cases and a specific *problem* to be classified, and 2) there is some assessment function to help the system decide which relations among entities are "important" or "relevant" to be shared with the precedent cases.

The representation formalism used to represent cases, feature terms, is presented in §2. Then, §3 introduces the framework of relational case-based learning and the main building blocks that we will use in §4 to describe the LID method. In §5 two application domains of relational case-based learning with LID are described: diabetes risk assessment and marine sponges identification. The paper closes with the sections on related work and conclusions.

2 Representation of the Cases

LID handles cases represented as feature terms. *Feature terms* (also called feature structures or ψ-terms) are a generalization of first order terms [2,8]. The main difference is that first order terms parameters are identified by position, e.g. $f(x, y, g(x, y))$ can be formally described as a tree and a fixed tree-traversal order. The intuition behind a feature term is that it can be described as a labelled graph where arcs are labelled with feature symbols and nodes stand for sorted variables.

Given a signature $\Sigma = \langle \mathcal{S}, \mathcal{F}, \leq \rangle$ (where $\mathcal{S}$ is a set of sort symbols that includes $\bot$; $\mathcal{F}$ is a set of feature symbols; and $\leq$ is a decidable partial order on $\mathcal{S}$ such that $\bot$ is the least element) and a set ϑ of variables, we define *feature terms* as an expression of the form:

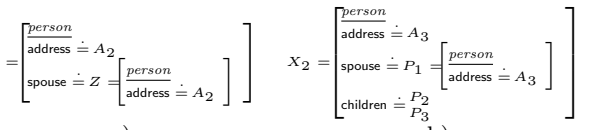
$$\psi ::= X : s[f_1 \doteq \Psi_1 \ldots f_n \doteq \Psi_n] \tag{1}$$

where X is a variable in ϑ called the *root* of the feature term, s is a sort in $\mathcal{S}$, $f_1 \ldots f_n$ are features in $\mathcal{F}$, $0 \leq n$, and each Ψ_i is a set of feature terms and variables. When $n = 0$ we are defining a variable without features. The set of variables occurring in ψ is noted as ϑ_ψ.

Sorts have an informational order relation ($\leq$) among them, where $s \leq s'$ means that s has less information than s' Ñor equivalently that s is more general than s'. The minimal element ($\bot$) is called *any* and it represents the minimum information. When a feature has unknown value it is represented as having the value *any*. All other sorts are more specific than *any*.

A *path* $\pi(X, f_i)$ is defined as a sequence of features going from the variable X to the feature f_i. When two paths $\pi(X, f_i)$ and $\pi(Y, f_j)$ point to the same value we say that there is a *path equality*.

The function *root(ψ)* returns the sort of the root of ψ. We note F_ψ the set of features $\{f_1 \ldots f_n\}$ of the root of ψ.



a) b)

Fig. 1. Examples of feature terms.

The *depth* of a feature f in a feature term ψ with root X is the number of features that compose the path from the root X to f, including f, with no repeated nodes.

Given a particular maximum feature depth k, a *leaf feature* of a feature term is a feature f_i such that either 1) the depth of f_i is k or 2) the depth of f_i is less than k and the value of f_i is a term without features.

The semantic interpretation of feature terms brings an ordering relation among feature terms that we call *subsumption*. Intuitively, a feature term ψ subsumes another feature term ψ' ($\psi \sqsubseteq \psi'$) when all information in ψ is also contained in ψ'. In other words, a feature term ψ subsumes another feature term ψ' when the following conditions are satisfied: 1) the sort of $root(\psi')$ is either the same or a subsort of $root(\psi)$, 2) if F_ψ is the set of features of ψ and $F_{\psi'}$ is the set of features of ψ' then $F_\psi \subseteq F_{\psi'}$ and 3) the values of the features in F_ψ and $F_{\psi'}$ satisfy the two conditions above.

For instance, the feature term X_1 in Figure 1a represents a person that is married with a person Z and both live at the address A_2 (i.e. there is a path equality since X_1.address = X_1.spouse.address). Because all the information in X_1 is also present in X_2 (Figure 1b), X_1 subsumes X_2 ($X_1 \sqsubseteq X_2$). However X_2 does not subsume X_1 —since X_1 has not the feature *children*.

A more detailed explanation about the feature terms and the subsumption relation can be found in [3]. In this reference there is also a detailed explanation of how feature terms can be translated to clause and graph representations.

3 Relational Case-Based Learning

There are three aspects that we need to define in order to perform CBR on relational cases: 1) to define a *case* from a constellation of relations, 2) to define a way to assess similarity between cases, and 3) to establish a degree of importance for the relations involved in the similarity.

A *case base* contains a constellation of relations between objects. The first step is to determine which of these relations constitute a case. A *case* (in feature terms) is specified from a relational case base using two parameters: a *root sort* and a *depth*. Assuming a *case base* expressed as a collection of feature terms, a case is a feature term whose root node is subsumed by the *root sort* and whose depth is at most *depth*. Examples of case specification are *case*[root-sort ≐

patient, depth $\doteq 5$] in the diabetes domain and *case*[root-sort $\doteq$ *sponge*, depth $\doteq$ 4] in the marine sponges domain (see §5).

The estimation of the similitude between cases is one of the key issues of the lazy learning algorithms. Those techniques (such as IBL [1] and k-nearest neighbor) that use cases represented as attribute-value vectors define the *similitude* of two cases by means of a distance measure. The LID method (§4) uses a symbolic estimation of the similitude between cases. The intuition of a symbolic similitude is that of a description containing the features shared by the two cases. In feature terms this intuition is formalized using the subsumption.

We say that a term s is a *similitude term* of two cases c_1 and c_2 if and only if $s \sqsubseteq c_1$ and $s \sqsubseteq c_2$ i.e. the similitude term of two cases subsumes both cases. In this framework, the task of similarity assessment is a search process over the space of similarity descriptions determined by the subsumption relation.

The next subsection explains a technique to assess the importance of a feature using the cases present in the case base. This technique is a heuristic measure based on the López de Mántaras (RLM) distance. Then, §4 explains how LID incrementally builds a similitude term based on the RLM heuristic.

3.1 Relevance of Attributes

Given a new example to be classified, the goal is to determine those features that are most relevant for the task. The relevance of a feature is heuristically determined using the RLM distance [11] that assesses how similar are two partitions (in the sense that the lesser the distance the more similar they are). Each feature $f_i \in \mathcal{F}$ induces a partition P_i of the case-base, namely a partition whose sets are formed by those cases that have the same value for feature f_i. The *correct partition* is a partition $P_c = \{C_1 \ldots C_m\}$ where all the cases contained into a set C_i belong to the same solution class. For each partition P_i induced by a feature f_i, LID computes the RLM distance to the correct partition P_c. The proximity to P_c of a partition P_i estimates the relevance of feature f_i.

Let P_i and P_j the partitions induced by features f_i and f_j respectively. We say that the feature f_i is *more discriminatory than* the feature f_j iff $RLM(P_i, P_c) < RLM(P_j, P_c)$, i.e. when the partition induced by f_i is closer to the correct partition P_c than the partition induced by f_j. Intuitively, the most discriminatory feature classifies the cases in a more similar way to the correct classification. LID uses the *more discriminatory than* relationship to estimate the features that are more relevant for the purpose of classifying a current problem.

4 Lazy Induction of Descriptions

In this section we introduce a new method called Lazy Induction of Descriptions (LID). The goal of LID is to classify a problem as belonging to one of the solution classes. The main idea of LID is to determine which are the more relevant features of the problem and to search in the case base for cases sharing these relevant

```
Function LID (S_D, p, D, C)
    if stopping-condition(S_D)
        then return class(S_D)
        else f_d := Select-leaf (p, S_D, C)
             D' := Add-path(π(root(p), f_d), D)
             S_D' := Discriminatory-set (D', S_D)
             LID (S_D', p, D', C)
    end-if
end-function
```

Fig. 2. The LID algorithm. D is the similitude term, S_D is the discriminatory set of D, C is the set of solution classes, $class(S_D)$ is the class $C_i \in C$ to which all elements in S_D belong.

features. The problem is classified when LID finds a set of relevant features shared by a subset of cases belonging all of them to the same solution class. Then, the problem is classified into that solution class.

Given a case base B containing cases classified into one of the solution classes $C = \{C_1 \dots C_m\}$ and a problem p, the goal of LID is to classify p as belonging to one of the solution classes. The problem and the cases in the case base are represented as feature terms (see §2). We call *discriminatory set* the set $S_D = \{b \in B | D \sqsubseteq b\}$ that contains the cases of B subsumed by the similitude term D.

The main steps of the LID algorithm are shown in Figure 2. In the first call $LID(S_D, p, D, C)$ parameter S_D is initialized to B (the whole case base) and parameter D can be initialized to *any* or to a value $D = D^0$ (where $D^0 \neq any$) based on domain knowledge we may have (see an example in §5.1).

The specialization of a similitude term D is achieved by adding features to it. In principle, any of the features used to describe the cases could be a good candidate. Nevertheless, LID uses two biases to obtain the set F_l of features candidate to specialize D. First, of all possible features in $\mathcal{F}$, LID will consider only those features present in the problem p to be classified. As a consequence, any feature that is not present in p will not be considered as candidate to specialize D. The second bias is to consider as candidates for specializing D only those features that are leaf features of p (see §2). This bias is similar to that of the *relational pathfinding* method [16] in that it favours the selection of relations chained together in the examples.

The next step of LID is the selection of a leaf feature $f_d \in F_l$ to specialize the similitude term D. Selecting the most discriminatory leaf feature in the set F_l is heuristically done using the RLM distance of §3.1 over the features in F_l. Let us call f_d the most discriminatory feature in F_l.

The feature f_d is the leaf feature of path $\pi(root(p), f_d)$ in problem p. The specialization step of LID defines a new similitude term D' by adding to the current similitude term D the sequence of features specified by $\pi(root(p), f_d)$. After this addition D' has a new path $\pi(root(D'), f_d)$ with all the features in the

path taking the same value that they take in p. After adding the path π to D, the new similitude term $D' = D + \pi$ subsumes a subset of cases in S_D, namely the discriminatory set $S_{D'}$ (the subset of cases subsumed by D').

Next, LID is recursively called with the discriminatory set $S_{D'}$ and the similitude term D'. The recursive call of LID has $S_{D'}$ as first parameter (instead of S_D) because the cases that are not subsumed by D' will not be subsumed by any further specialization. The process of specialization reduces the discriminatory set $S_D^n \subseteq S_D^{n-1} \subseteq \ldots \subseteq S_D^0$ at each step.

The stopping condition of LID, given the current similitude term D, is that all the cases in its discriminatory set S_D belong to only one solution class $C_k \in C$. LID gives D as an explanation of classifying p in C_k and S_D as the cases justifying that result. The similitude term D can be viewed as a *partial* description of C_k because it contains a subset of features that are discriminant enough to classify a case as belonging to C_k. Notice that D is not the most general generalization of C_k since in general D does not subsume all the cases belonging to C_k but only a subset of them (those sharing the features of D with the new problem). The similitude term D depends on the new problem, for this reason there are several partial descriptions (i.e. similitude terms) for the same class.

5 Experiments

In this section we describe two applications developed usimg LID: diabetes risk assessment (5.1) and identification of marine sponges (5.2).

5.1 Complications Risk Assessment in Diabetes

Diabetes mellitus is one of the most frequent human chronic diseases. There are two major types of diabetes: diabetes type I (or insulin-dependent) usually found in people younger than 40 years, and diabetes type II (or non insulin-dependent) often developed in people over this age. Both forms of diabetes produce the same short-term symptoms (i.e. increase of thirst, and high blood glucose values) and long-term complications (i.e. blindness, renal failure, gangrene and amputation, coronary heart disease and stroke).

The main concern in the management of the diabetes is reducing the individual risks of patients in developing new long-term complications and reducing risk of progression in the complications already present. In fact, the expected risks are different whether the patient has diabetes type I than he has diabetes type II. Moreover, the risk is also different whether the patient has no complications (*development risk*) or he has developed some complication (*progression risk*). We have developed a case base of 370 patients in collaboration with an expert.

The goal of LID in this domain is to assess the individual risks of complications for diabetic patients. Specifically, LID has to classify a patient in one of the following risk classes: *low risk*, *moderate risk*, *high risk*, and *very high risk*. We will focus on three macro-vascular complications: infarct, stroke and amputations. Each of these tasks requires LID to independently classify a problem using

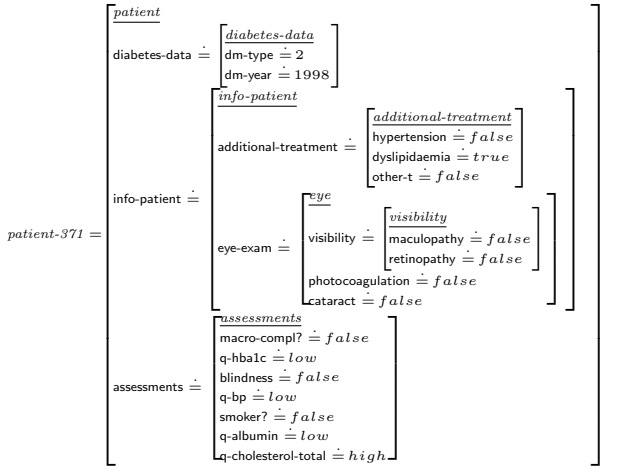


Fig. 3. Partial description of a diabetic patient. Here only 15 leaves are shown out of a total of more than 50 leaves present in the complete description of the patient.

the patients available in the case base. In order to illustrate the performance of LID in the diabetes domain we will focus on the assessment of the infarct risk. For this task the correct partition P_C is formed by the sets of patients that have the same risk class for the infarct complication.

Let us suppose that LID is used to assess the risk of infarct for the *patient-371* described in Figure 3. The search of LID can be constrained using domain knowledge. In this domain, we are only interested in considering patients in the case base that share with our current patient the same type of diabetes and the fact of whether or not the patient has macro-complications. Specifically, LID is initialized with D^0 (Figure 4a) having features *dm-type* with value 2 and *macro-compl?* with value *false* as in *patient-371*. The discriminatory set S_D^0 contains 144 cases having different infarct risk. Therefore the similitude term D^0 has to be specialized. The first step is to select the most discriminatory leaf feature using the RLM distance where the correct partition P_C is the classification of the cases according to their infarct risk degree.

LID finds the leaf feature *q-albumin* as the most discriminatory and builds the similitude term D^1 by adding to D^0 the path π(*patient*, *q-albumin)* with *q-albumin* taking value *low* as in the *patient-371* (see Figure 4b). The discriminatory set S_D^1 contains 99 cases with different infarct risk. Therefore the similitude term D^1 has to be specialized by adding a new leaf feature. LID finds now the leaf feature *maculopathy* as the most discriminatory. The similitude term D^2 is obtained by adding the path π(*patient*, *maculopathy)* to D^1 with value *false* as in *patient-371*.

The discriminatory set S_D^2 contains 66 cases with different infarct risk therefore D^2 has to be specialized. Now LID finds *other-t* as the most discriminant leaf feature. The similitude term D^3 subsumes 57 cases with different infarct risk. This means that D^3 has to be specialized. The next more discriminant

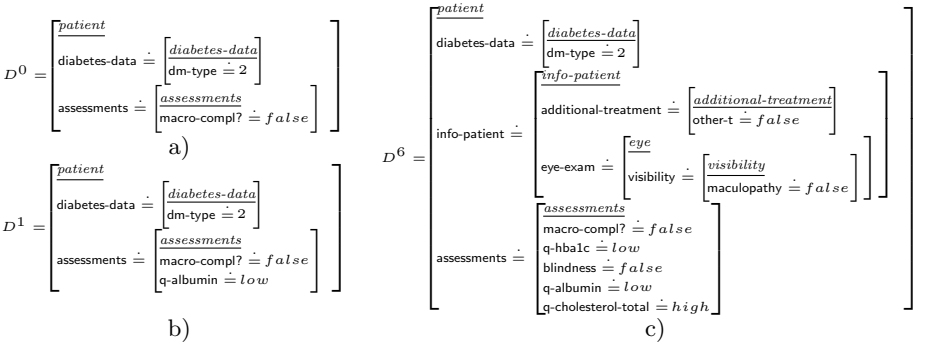


Fig. 4. Three similitude terms constructed for assessing the infarct risk of *Patient-371*.

leaf feature is *q-cholesterol-total*. The similitude term D^4 including the path π(*patient*, *q-cholesterol-total)* with value *high* subsumes 16 cases that also have different infarct risk. The similitude term D^5 is obtained by adding to D^4 the path π(*patient*, *blindness)* with value *false*. The discriminatory set S_D^5 still contains cases having either *high* or *moderate* infarct risk. Finally, the similitude term D^6 (Figure 4c) is obtained by adding to D^5 the path π(*patient*, *q-hba1c)* with value *low*. The discriminatory set S_D^6 contains two cases with *moderate* risk of infarct. Therefore LID concludes that *patient-371* has a *moderate risk* of infarct. LID explains this classification with the similitude term D^6 of Figure 4c and justifies this result with the 2 cases of S_D^6.

LID has been evaluated in the following way. An expert diabetologist constructed a gold standard consisting of a risk pattern for macro-complications for the 370 cases of the base. This gold standard gives a unique "correct" risk value for each complication and considers all other risk estimations "incorrect". In fact, this assumption is too strong because often the expert assesses a range of risks (e.g. *very high* or *high*).

The experimental evaluation has been performed with this definition of correctness for the tasks of assessing the risk of stroke, infarct and amputation. For each task, we have built 15 test sets with the 370 patients case base, where each test set has 300 cases randomly chosen as training set. The results of LID upon the remaining 70 cases for each test set where compared with the gold standard and averaged for each task. The accuracy of LID is the following: 100% correct in assessing the stroke risk, 90% correct in assessing amputation risk, and 72.45% correct in assessing the infarct risk. In fact, the incorrect assessment of the infarct risk fail only by one degree (e.g. high risk vs. very-high risk) in 81.69% of the cases. We are currently analyzing those cases with the support of the expert since often he assesses a range of risks that includes the answer of LID.

In addition to estimate the accuracy of LID we have also analyzed the justifications (similitude terms) in order to determine whether the risk has been obtained based on correct assumptions or, conversely, whether it has been obtained from assumptions that the expert considers irrelevant for estimating a risk. Let

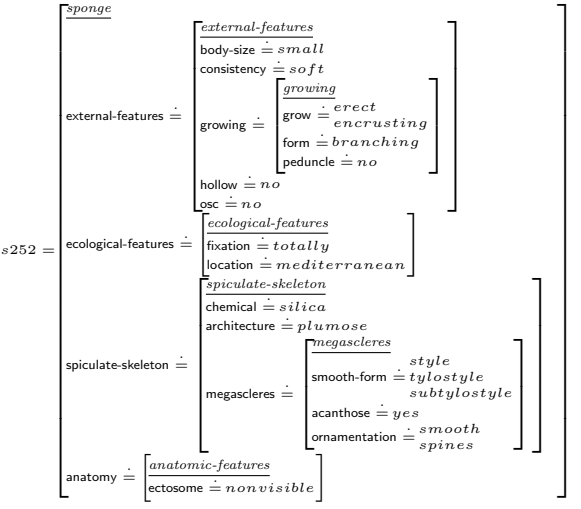


Fig. 5. Partial representation of a sponge using feature terms.

us consider assessing the stroke risk where LID always obtains a similitude term containing only one feature: the blood pressure. The expert confirms that the blood pressure is the determining factor for the risk of stroke. When assessing the risk of amputation, the justification contains the feature *polyneuropathy* that is one of the most important factors according to the expert criteria. Concerning the assessment of the infarct risk, the LID explanations also use among others features such as diabetes duration, cholesterol or haemoglobin that the expert considers as determinant factors for assessing the infarct risk.

Notice that LID explanations are not generalizations describing a class, they are symbolic descriptions of the (important) aspects shared between the problem and similar cases.

5.2 Identification of Marine Sponges

Marine sponges are relatively little studied and most of the existing species are not yet fully described. The main problems in sponge identification are due to the morphological plasticity of the species, to the incomplete knowledge of many of their biological and cytological features and to the frequent description of new taxa. Moreover, there is no agreement around the species delimitation since it is not clear how the different taxa can be characterized. The application of LID to this domain allows the classification of new specimens based on their similarity to specimens clearly classified in some taxa. This similarity, in turn, is based on the relevant features of the specimen that has to be classified.

With the support of two experts, we have developed a case base with the descriptions of 280 sponges belonging to the orders: $C = \{$*astrophorida, hadromerida, axinellida*$\}$. LID has been used to identify the order of new specimens.

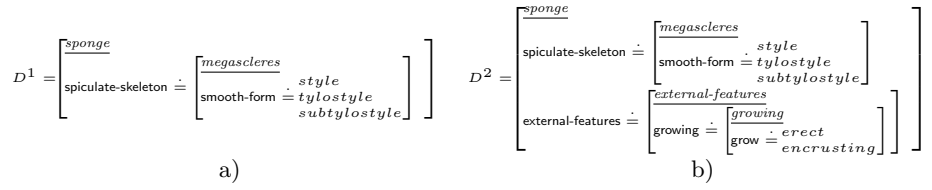


Fig. 6. Sequence of similitude terms constructed by LID for classifying sponge *s252*.

Let us consider how LID identifies the specimen *s252* of Figure 5 given a case base B. LID begins with the similitude term $D^0 = any$, the discriminatory set $S_D^0 = B$ and $p = s252$. Since the cases in S_D^0 belong to several orders, the similitude term D^0 has to be specialized. The first step of the specialization is to select the most discriminatory leaf of *s252*. Using the RLM distance LID finds that the most discriminatory feature is the leaf *smooth-form*. Then LID specializes D^0 to the new similarity term D^1 by adding the path $\pi(sponge,$ *smooth-form)* taking as value the set $\{style, tylostyle, subtylostyle\}$ (Fig. 6a).

The discriminatory set S_D^1 contains 25 cases subsumed by the similitude term D^1. Since these cases belong to several orders D^1 has to be specialized. Now LID finds that the most relevant leaf feature is *grow*. Then D^1 is specialized to D^2 adding the path $\pi(sponge, grow)$ with value the set {*erect, encrusting*} (see Figure 6b). The discriminatory set S_D^2 contains 10 cases belonging to the *axinellida* order. Therefore LID classifies the sponge *s252* in the *axinellida* order. The explanation of this classification is the similitude term D^2 of Figure 6b stating that the sponge *s252* is *axinellida* because 1) has megascleres of form *style, tylostyle* and *subtylostyle*, and 2) grows *erect* and *encrusting*.

We evaluated LID in the marine sponges domain with the goal of identifying the order of specimens. We performed six ten-fold crossvalidation runs on the case base containing 280 descriptions of marine sponges. The average accuracy of LID is 89.998% with a standard deviation of 5.827.

6 Related Work

There are two main lines of research closely related to our approach on similarity assessment in relational representations. On the one hand, there is research about similarity assessment of structured and complex cases in CBR; and, on the other hand, there is research on relational IBL.

Although a lot of work in similarity assessment of cases in CBR is focused on weighted distances among attribute value vectors there is also active research in establishing similarity estimates among structured representation of cases [6, 7,15]. Some approaches share the idea of building a "structural similarity" (as our "similitude term" approach) but they use techniques subtree-isomorphism or subgraph isomorphism detection for building the description of this "structural similarity" [7]. In [6] the "structural similarity" is used to guide adaptation phase of CBR [6] and not for the retrieval phase. Another way to construct a similitude

term is using antiunification (computing the most specific generalization of two cases) [15], and later having a measure to assess which similitude term is better (e.g. using an entropy measure [14]) to select the best precedent in the case-base.

A related approach to feature terms as case representation formalism is that of using Description Logics for this purpose; antiunification can also be used but some assessment measure is also needed, as shown in [12] where a probabilistic interpretation of the "least common subsumer" is used. LID does not use antiunification to build the similitude term; instead it constructs the similitude term guided by an example-driven heuristic to assess the relevance of the features to be added to the similitude term. Finally, some research work on case retrieval use inductive techniques involving the construction of decision trees (as in [4]) and only for cases that are attribute-value vectors.

Relevant work is being carried on for transferring IBL techniques to relational learning, mainly in the ILP framework of a Horn clause representation of cases and background knowledge. An approach in ILP using a concept closed to similitude terms [15,14] is that of Progol's Analogical Prediction [13]. Progol's AP is a lazy induction technique capable of binary classification. For each problem, AP tries to build an hypothesis using the Progol engine. If this hypothesis is found the problem is classified as *true*, otherwise the problem is classified as *false*. LID builds a similitude term ("hypothesis") but can deal with multiple classes. The RIBL system [9,10] first generates cases out of Horn clauses in a knowledge base, then calculates the distance among cases by estimating the relevance of predicates and attributes. RIBL 2.0 [10] uses an "edit distance" to estimate the distance between cases (that can contain lists and terms). Instead of the notion of "distance" among cases LID uses a distance in the heuristic assessment of the importance of features to be included in the similitude term. Moreover, our assessment is not based on a pairwise comparison of "problem vs. case" similarity but takes into account all cases that share a particular "structural similarity" embodied by those cases subsumed by the similitude term that LID builds.

7 Conclusions

We have developed a technique for case-based learning where cases are best represented as collection of relations among entities. The LID approach is based on a similarity assessment used by a heuristic search in a space of symbolic descriptions of similitude. Moreover, the symbolic description of similitude provides an explanation of the grounds on which a precedent case is selected from the case base as most relevant (or "similar") to the current problem. The symbolic similitude that classifies a problem subsumes a subset of the elements in a class, and as such it is just a partial description of that class. Indeed, this is the main difference between a lazy learning approach like LID and an eager approach as that of induction (see the INDIE inductive method in [3]).

As for future work, we intend to explore a variation of LID that adaptively chooses a middle ground between the extreme points of lazy and eager approaches. Our assumption is that it is unlikely that a lazy (or eager) approach

is always the best suited for all application domains. Thus, our aim will be to investigate in which situations it is useful to store (memorize) the partial class descriptions provided by LID an use them to solve new problems and in which situations is better to keep a purely lazy approach.

Acknowledgements. This work has been supported by Projects SMASH (CICYT TIC96-1038-C04-01) and IBROW (IST-1999-19005). The authors thank Dr. Marta Domingo and physician Albert Palaudàries for their assistance in developing the marine sponges and diabetes applications.

References

[1] D. Aha, editor. *Lazy Learning*. Kluwer Academic Publishers, 1997.
[2] Hassan Aït-Kaci and Andreas Podelski. Towards a meaning of LIFE. *J. Logic Programming*, 16:195–234, 1993.
[3] E. Armengol and E. Plaza. Bottom-up induction of feature terms. *Machine Learning*, 41(1):259–294, 2000.
[4] E. Auriol, S. Wess, M. Manago, K.-D. Althoff, and R. Traphöner. Inreca: A seamless integrated system based on inductive inference and case-based reasoning. In *CBR Reseacrh and Development*, number 1010 in Lecture Notes in Artificial Intelligence, pages 371–380. Springer-Verlag, 1995.
[5] U. Bohnebeck, T. Horváth, and S. Wrobel. Term comparisons in first-order similarity measures. In D. Page, editor, *Proc. of the 8th International Workshop on ILP*, volume 1446 of *LNAI*, pages 65–79. Springer Verlag, 1998.
[6] K Börner. Structural similarity as a guidance in case-based design. In *Topics in Case-Based Reasoning: EWCBR'94*, pages 197–208, 1994.
[7] H Bunke and B T Messmer. Similarity measures for structured representations. In *Topics in Case-Based Reasoning: EWCBR'94*, pages 106–118, 1994.
[8] B. Carpenter. *The Logic of typed Feature Structures*. Tracts in theoretical Computer Science. Cambridge University Press, Cambridge, UK, 1992.
[9] W. Emde and D. Wettschereck. Relational instance based learning. In Lorenza Saitta, editor, *Machine Learning - Proceedings 13th ICML*, pages 122 – 130. Morgan Kaufmann Publishers, 1996.
[10] Tamas Horvath, Stefan Wrobel, and Uta Bohnebeck. Relational instance-based learning with lists and terms. *Machine Learning*, 43(1):53–80, 2001.
[11] Ramon López de Mántaras. A distance-based attribute selection measure for decision tree induction. *Machine Learning*, 6:81–92, 1991.
[12] T. Mantay and R. Moller. Content-based information retrieval by computing least common subsumers in a probabilistic description logic. In *Proceedings of the ECAI Workshop Intelligent Information Integration*, 1998.
[13] Stephen Muggleton and Michael Bain. Analogical prediction. In *Proc. ILP*, 1999.
[14] E. Plaza, R. López de Mántaras, and E. Armengol. On the importance of similitude: An entropy-based assessment. In I. Smith and B. Saltings, editors, *Advances in Case-based reasoning*, number 1168 in Lecture Notes in Artificial Intelligence, pages 324–338. Springer-Verlag, 1996.
[15] Enric Plaza. Cases as terms: A feature term approach to the structured representation of cases. In M. Veloso and A. Aamodt, editors, *Case-Based Reasoning, ICCBR-95*, number 1010 in Lecture Notes in Artificial Intelligence, pages 265–276. Springer-Verlag, 1995.
[16] B. L. Richards and R. J. Mooney. Learning relations by pathfinding. In *proceedings of AAAI-92*, pages 50–55, 1992.