Identification with iterative nearest neighbors using domain knowledge

David Grosser, Noël Conruyt, Henri Ralambondrainy

Abstract — A new iterative and interactive algorithm called CSN (Classification by Successive Neighborhood) to be used in a complex descriptive objects identification approach is presented. Complex objects are those designed by experts within a knowledge base to describe taxa (monography species) and also real organisms (collection specimens). The algorithm consists of neighborhoods computations from an incremental basis of characters using a dissimilarity function which takes into account structures and values of the objects. A discriminant power function is combined with domain knowledge on the features set at each iteration. It is shown that CSN consistently outperforms methods such as identification trees and simplifies interactive classification processes comparatively to search for K-Nearest-Neighbors method.

Index Terms — identification, Similarity, K-Nearest-Neighbors, Decision Trees, structured data, knowledge base, life science.

1 INTRODUCTION

In the frame of environmental sciences, for helping to preserve rich ecosystems from biodiversity loss, the acquisition and production of knowledge on biological specimens and taxa is an essential part of the work of systematologists [1]. Indeed, being able to describe, classify and identify a specimen from morphological characters is a first step for monitoring biodiversity, because it gives access to information relative to its species name (Biology, Geography, Ecology, Taxonomy, bibliography, photography). These tasks can be assisted in biodiversity informatics by databases for storing information and computer science decision support tools for description, classification and identification purpose with knowledge bases. In return, these complex representations deliver interesting models and processing problems to deal both with domain knowledge and specimen descriptions.

In many fields of real world applications, we can capture a given aspect of the descriptive domain knowledge by associating attributes of the problem.

The authors are with the Computer Science and Mathematics laboratory (LIM-IREMIA) of Reunion University – 97400 Saint-Denis, France. E-mail: grosser@univ-reunion.fr.
structure with objects linked by composition and/or specialization relationships. We can also structure the domain definition of nominal attributes by a hierarchy of values. These techniques enable the algorithms to take into account mutual dependencies between attributes and values and to compare case properties with more accuracy.

For instance, for the knowledge base on Corals of the Mascarene Archipelago (http://coraux.univ-reunion.fr/), the descriptions of specimens are often highly structured (composite objects, taxonomic attributes), highly noisy (erroneous or unknown data) and polymorphous (variable, i.e. simultaneous presence of states or imprecise data). To take into account this complexity, we need to define a descriptive model (or Ontology) that includes information about objects’ relationships, attribute types and other semantic aspects: scope of the values, meaning of special values (defaults, exceptions), observation cost of characters.

For engineering Systematics, we have developed a type of knowledge base system that supports both taxa and specimens descriptions. IKBS (Iterative Knowledge Base System) is a knowledge management system available on the Internet (http://ikbs.sourceforge.net) that helps to define descriptive models, describe instances of these models (see Fig. 1) and then identify new specimen descriptions with different identification methods: an Identification Tree based method (monothetic) and a K-Nearest Neighbors method (polythetic) that uses a dissimilarity function designed to deal with such complex objects representations [2].

Classification by Successive Neighborhood (CSN)

---

Fig. 1 – Part of a specimen description made with IKBS. Characters (attributes) are attached to objects (eventually missing or absent) that are organized with composition relationships.
2 CLASSIFICATION BY SUCCESSIVE NEIGHBORHOOD (CSN)

CSN is a new iterative and interactive method that uses a similarity measure and a discriminant character selection to identify complex objects [3]. Starting from a partially described unlabeled object, the method consists in selecting at each step an objects’ neighborhood in regards to a similarity measure. A set of candidate classes is computed from the neighbor set considering class frequencies. A list of discriminant characters is built from the neighborhood and the best is chosen among that list. The value is obtained interactively from users (or another data source). A new neighborhood is computed on the basis of the new partial description of the object. The process iterates until the candidate classes set is homogeneous.

The iterative process to identify an unlabeled description called e is made of the following functions:

2.1 BUILDING NEIGHBORS SET

The neighbors of e at iteration m is the set \( V_m \) of objects inside of the sphere of radius \( \Delta_m \) centered at e:

\[
V_m = \{ o \in O \mid d(e, o) < \Delta_m \}, \quad d \text{ is a dissimilarity function.}
\]

The radius value is determined from the maximum distance, the max dissimilarity value between e and the set \( V_{m-1} \):

\[
\Delta_m = \max \{ d(e, o_i) \}, \quad o_i \in V_{m-1}.
\]

Then \( \{ \Delta_m \} \) is a decreasing sequence.

2.2 SELECTING DISCRIMINANT CHARACTERS

An ordered list of informative variables is computed at each step of the classification process from \( V \). The first element is exposed as a question to the user who can choose an alternative variable from the list. The list is built in function of three criteria:

- Discriminant power. Choice of different classical criteria computing the information gain used in machine learning such as Shannon entropy measure or Gini index. Straightforwardly, this type of criterion minimizes the number of questions.
- Selectable characters. The method considers only at each step characters that may be indicated for choice. Relative questions about presence or absence of components are also considered as selectable attributes.
- Using attributes weighting in the data model reflecting observation cost or other strategic knowledge about characters.

3 EXPERIMENTS

In the following experiments, we have extracted some descriptions from the Fungiidae Knowledge base on Corals of the Mascarene Archipelago that counts approximately 150 classes and 800 complex objects. We follow a double
objective. Firstly, we aim to illustrate the execution of the CSN algorithm in the IKBS software. Secondly, we want to compare the classification (identification) accuracy of the CSN method in regards to an identification tree (IT) based method and a simple K-Nearest-Neighbors (KNN). Both methods already exist in IKBS and use respectively the same discriminant character selection method and the same dissimilarity function.

3.1 Execution of CSN algorithm

The example in Tab. 1 illustrates the identification process of an unlabeled description $e$ initially empty. The *Fungiidae* Knowledge base is made of 63 cases with 94 characters and 15 species names. To simulate user interactions, the data source of the description $e$ to identify is set to $e_r$ corresponding to a complete and referenced description of a specimen pertaining to species *Fungia concinna* (case number 8 among 63). For the correct identification, $e_c$ class (species name) is compared with $e_r$ class at the end of the process. The criterion used for the character selection function is the classical Shannon’s information gain measure.

<table>
<thead>
<tr>
<th>Num</th>
<th>Attribute</th>
<th>Value</th>
<th>Neighbors set</th>
<th>Distance</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>form</td>
<td>colony</td>
<td></td>
<td>case</td>
</tr>
<tr>
<td>1</td>
<td></td>
<td></td>
<td>circular</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>51</td>
</tr>
<tr>
<td>7</td>
<td>size</td>
<td>skeleton</td>
<td></td>
<td>62</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10.3 cm</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>8</td>
<td>density</td>
<td>spines</td>
<td>[10 16]</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>9</td>
<td>profile</td>
<td>skeleton</td>
<td>flat</td>
<td>60</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>62</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
<tr>
<td>19</td>
<td>dev</td>
<td>coenosteum</td>
<td>sub equal</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>39</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>20</td>
<td>lobes</td>
<td></td>
<td>absent</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>39</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>60</td>
</tr>
<tr>
<td>21</td>
<td>form</td>
<td>spines</td>
<td>cylindrical</td>
<td>39</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>46</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>62</td>
</tr>
</tbody>
</table>

Tab. 1 – Example of identification process by successive iterations (Num) of $e$.

Tab. 1 shows a selected subset of iterations that conducted to a good detection of $e$. 21 iterations (and so 21 character values) were necessary. For each line, the selected character, the corresponding value and the information gain associated are showed. For each neighborhood, information about the 3 first objects in $V$ is shown: cases indexes (in the case base), attaching classes,
values for the selected character and the dissimilarity values. For convenience needs, the stopping criterion used is the exact matching with the class of the first case (in bold in the table).

The most interesting information to observe is the progression of \( V \). Variations of positions show how supplying information to \( e \) modifies distances and consequently the order of cases in \( V \). Thus, for instance, between iteration 8 and 9, the case 60 goes up to first position because the value of the character profile of the Skeleton (noted profile[Skeleton]) corresponds to the reference value, but not cases 62 and 46. At iteration 19 appears the case 39 in position 2, labeled with the “good class”. To finish, at iteration 21, the case 39 reached the first position in front of the case 46 and the process stop with a “good matching”.

3.2 Classification accuracy

In this second experiment, we evaluate relative performances of CSN comparatively to IT and to KNN methods already implemented in IKBS. The first IT method [4] is an extension of the supervised classification algorithm C4.5 [5] adapted to the use of a structured descriptive model.

The second KNN method uses the same dissimilarity measure as CSN for its neighbors set computation. The validation method used is a “leave-one-out” cross validation process [6] that consists to classify each case of the base using the others as training set. This method is applied for the three algorithms with similar conditions. For convenience needs, K is set to 1 in the KNN method.

Tab. 2 gives results of the validation process on six family knowledge bases. For each base, we show the number of cases and characters, and for each method, identification accuracy (score).

<table>
<thead>
<tr>
<th>Bases</th>
<th>nb cls</th>
<th>nb case</th>
<th>nb attr</th>
<th>Tree ident score</th>
<th>rate</th>
<th>K-neighbors score</th>
<th>rate</th>
<th>CSN score</th>
<th>rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faviinae</td>
<td>36</td>
<td>92</td>
<td>146</td>
<td>65</td>
<td>70.65%</td>
<td>85</td>
<td>92.39%</td>
<td>84</td>
<td>91.30%</td>
</tr>
<tr>
<td>Montastreinae</td>
<td>15</td>
<td>24</td>
<td>118</td>
<td>17</td>
<td>70.83%</td>
<td>22</td>
<td>91.66%</td>
<td>19</td>
<td>79.16%</td>
</tr>
<tr>
<td>Fungiidae</td>
<td>15</td>
<td>63</td>
<td>94</td>
<td>47</td>
<td>74.60%</td>
<td>58</td>
<td>92.06%</td>
<td>55</td>
<td>87.30%</td>
</tr>
<tr>
<td>Mussidae</td>
<td>15</td>
<td>56</td>
<td>28</td>
<td>49</td>
<td>87.50%</td>
<td>54</td>
<td>96.42%</td>
<td>51</td>
<td>91.07%</td>
</tr>
<tr>
<td>Poritidae</td>
<td>28</td>
<td>28</td>
<td>87</td>
<td>22</td>
<td>78.57%</td>
<td>24</td>
<td>85.71%</td>
<td>19</td>
<td>67.85%</td>
</tr>
<tr>
<td>Siderastreidae</td>
<td>14</td>
<td>60</td>
<td>99</td>
<td>49</td>
<td>81.67%</td>
<td>56</td>
<td>93.33%</td>
<td>57</td>
<td>95.00%</td>
</tr>
</tbody>
</table>

Tab. 2 – Comparison of IT, KNN and CSN on classification accuracy of 6 knowledge bases.

It demonstrates that IT method presents a low accuracy rate comparatively to CSN and KNN. Identification errors are frequent, from 12.5% for Mussidae family to 29.35% for Faviinae. The best method is KNN that may be intuitively explained by the fact that it uses the overall information of the objects: the reference case is fully described. CSN offers an intermediate score, near KNN and often outperforms IT. We may observe for instance results of the Faviinae base that show a difference among 20% of good identification.
4 Conclusion

To identify a biological object and to associate a class to it, experts usually proceed with two phases. The synthetic phase, by global observation of the most visible characters reduces the field of investigation. The analytical phase, by precise observation of discriminating attributes refines research until obtaining the result. Even if the k-nearest-neighbors approach gives a good classification rate, it is difficult to use in real conditions without background knowledge of the domain. In fact, it is very useful to dispose of an interactive process to design features selection such in decision tree approaches.

The classification by successive neighborhood (CSN) method that we proposed deals with structured and partial objects descriptions. It presents the interest to correspond to the reasoning followed by biologists. Starting from a partial description generally containing the most visible or easy to observe and describe features, the method suggests relevant information necessary to supplement to determine the most probable class.

We expect that the CSN method is generic and applicable on any fields where structured or semi structured data are considered, such as XML data format or RDF and OWL graph structures. It's enough to lay out a similarity index and a discriminant power function adapted to the considered data.

References


