Extended ontologies: a cognitively inspired approach

Joel Luis Carbonera[1], **Mara Abel**[1]

[1]Instituto de Informática – Universidade Federal do Rio Grande do Sul (UFRGS)
Caixa Postal 15.064 – 91.501-970 – Porto Alegre – RS – Brazil

{jlcarbonera,marabel}@inf.ufrgs.br

Abstract. *Within the Knowledge representation community, in general, an ontology is considered as a formal specification of a shared conceptualization. In this sense, ontologies would be constituted of concepts and could be understood as an approach of representing knowledge. In general, ontologies represent concepts in a logical way, adopting the so-called classical theory of representation. Due to this, ontologies can support classification, based on necessary and sufficient conditions, and rule-based reasoning. In this work, we discuss a cognitively inspired approach for extending the knowledge representation capabilities of ontologies. We propose an extended notion of ontologies which incorporates other cognitively plausible representations, such as prototypes and exemplars. The extended ontology has the advantage of supporting similarity-based reasoning, besides the usual logical reasoning.*

1. Introduction

Nowadays, ontologies are widely adopted for *knowledge reusing* and for promoting the *semantic interoperability* among different systems (and humans). Within the knowledge representation community, in general, ontologies are considered as *formal and explicit specifications of a shared conceptualization in a given domain* [Studer et al. 1998]. It is important to notice that, according to this perspective, ontologies would be constituted of *concepts*. In this work, following other works in the field of Artificial Intelligence [Oltramari and Lebiere 2011, Carbonera et al. 2015], we adopt this *conceptualist* [Smith 2004] view about ontologies.

In general, ontologies represent concepts in a *logical* way, assuming the so-called *classical theory* of representation [Murphy 2002], where the concepts are represented by sets of features that are *shared by all the entities* that are abstracted by the concept. Due to this, ontologies are well suited for supporting classification based on *necessary and sufficient conditions* and for supporting *rule-based reasoning*. However, in general, ontologies cannot deal naturally with *typical* features of the concepts [Gärdenfors 2004]; that is, the features that are common to the entities abstracted by the concepts, but that are neither necessary nor sufficient. In this paper, we propose the notion of *extended ontology*, which incorporates other cognitively plausible representations, such as *prototypes* and *exemplars*, and that can support *similarity-based reasoning* (dealing with prototypical effects), besides the usual *rule-based reasoning*.

2. Theories of knowledge representation

Within the Cognitive Sciences there is an ongoing debate concerning how the knowledge is represented in the human mind. According to [Murphy 2002] in this debate there are three main theories. The *classical theory* assumes that each concept is represented by a *set of features* that are *shared* by *all* the entities that are abstracted by the concept. In this way, this set of features can be viewed as the *necessary and sufficient conditions* for a given entity to be considered an instance of a given concept. Thus, according to this theory, concepts are viewed as *rules* for classifying objects based on features. The *prototype theory*, on the other hand, states that concepts are represented through a *typical instance*, which has the typical features of the instances of the represented concept. Finally, the *exemplar theory* assumes that each concept is represented by a set of *exemplars* of it, which are explicitly represented in the memory. In theories based on prototypes or exemplars, the categorization of a given entity is performed according to its *similarity* with prototypes or exemplars; the instance is categorized by the category that has a prototype (or exemplar) that is more similar to it. There are some works that apply these alternative theories in computer applications [Fiorini et al. 2014].

3. Extended ontologies

As previously discussed, ontologies can be viewed as a paradigm of knowledge representation that adopts the *classical theory* of knowledge representation. In this sense, the classification of instances is performed by checking if they meet the necessary and sufficient conditions of the considered concepts. However, it is well known in the knowledge representation community that, for most of the common sense concepts, finding their necessary and sufficient conditions can be a challenging task [Gärdenfors 2004]. Besides that, according to evidences taken from the research within the Cognitive Sciences [Gärdenfors 2004], for most of the concepts, humans can perform similarity-based classifications, and can consider the typical features of the concepts during the classification process. In this work, we assume that a knowledge representation framework that preserves the flexibility of the human cognition can provide advantages for knowledge-based systems. For example, a system with this capability could classify some individual i as c (where c is some concept) if it is sufficiently similar to a given prototype of c, even when it does not present all the logically necessary features for being considered an instance of c.

In this work, we propose the notion of *extended ontology* ($\chi\mathcal{O}$), which incorporates the conventional features and capabilities of the *classical ontologies* with the possibility of representing typical features of the concepts and of supporting similarity-based reasoning. This proposal adopts some notions originally proposed in our previous works [Carbonera and Abel 2015a, Carbonera and Abel 2015b].

Definition 1. An *extended ontology* ($\chi\mathcal{O}$) is a tuple

$$\chi\mathcal{O} = (\mathcal{C}, \leq, \mathcal{R}, \mathcal{A}, \hookrightarrow, \mathcal{D}, d, \mathcal{I}, v, ext, \mathcal{E}, ex, \mathcal{P}, prot) \tag{1}$$

, where:

- $\mathcal{C}$ is a set $\mathcal{C} = \{c_1, c_2, ..., c_n\}$ of n symbols that represents concepts (or classes), where each c_i is a symbolic representation of a given concept.
- $\leq$ is a *partial order* on $\mathcal{C}$, that is, $\leq$ is a binary relation $\leq \subseteq \mathcal{C} \times \mathcal{C}$, which is reflexive, transitive, and anti-symmetric. Thus, $\leq$ represents a relation of subsumption between two concepts.
- $\mathcal{R}$ is a set $\mathcal{R} = \{r_1, r_2, ..., r_m\}$ of m symbols that represents relations, where each r_i is a symbolic representation of a given relation.

- $\mathcal{A}$ is a set $\mathcal{A} = \{a_1, a_2, ..., a_l\}$ of l symbols that represents properties (or attributes or features), where each a_i is a symbolic representation of a given property.
- $\hookrightarrow$ is a binary relation that relates properties in $\mathcal{A}$ to concepts in $\mathcal{C}$, such that $\hookrightarrow \subseteq \mathcal{A} \times \mathcal{C}$. Thus $a_i \hookrightarrow c_j$ means that the attribute $a_i \in A$ is an attribute of the concept c_j, in the sense that a_i characterizes c_j.
- $\mathcal{D}$ is the set of every possible value of every attribute $a_i \in A$.
- $d\colon \mathcal{A} \rightarrow 2^{\mathcal{D}}$ is a function that maps a given attribute $a_i \in \mathcal{A}$ to a set $\mathcal{D}_{a_i} \subseteq \mathcal{D}$, which is its domain of values. Notice that $D = \bigcup_{i=1}^{l} d(a_l)$.
- $\mathcal{I}$ is a set $\mathcal{I} = \{i_1, i_2, ..., i_p\}$ of p symbols that represents individuals, where each i_j represents a given individual.
- $v\colon \mathcal{I} \times \mathcal{A} \rightarrow \mathcal{D}$ is a function that maps a given individual $i_j \in \mathcal{I}$ and a given attribute $a_i \in \mathcal{A}$ to the specific value $v \in \mathcal{D}$ that the attribute a_i assumes in i_j.
- $ext\colon \mathcal{C} \rightarrow 2^{\mathcal{I}}$ is a function that maps a given concept $c_i \in \mathcal{C}$ to a set $I_{c_i} \subseteq \mathcal{I}$, which is its extension (the set of individuals that it classifies).
- $\mathcal{E}$ is a set $\mathcal{E} = \{e_1, e_2, ..., e_n\}$ of n sets of individuals, where each $e_i \in \mathcal{E}$ represents the set of *exemplars* of a given concept c_i. Notice that $\mathcal{E} \subseteq 2^{\mathcal{I}}$.
- $ex\colon \mathcal{C} \rightarrow \mathcal{E}$ is a function that maps a given concept $c_i \in \mathcal{C}$ to its set of exemplars $e_i \in \mathcal{E}$.
- $\mathcal{P}$ is a set $\mathcal{P} = \{p_1, p_2, ..., p_n\}$ of n prototypes, where each $p_i \in \mathcal{P}$ represents the prototype of a given concept $c_i \in \mathcal{C}$.
- $prot\colon \mathcal{C} \rightarrow \mathcal{P}$ is a function that maps a given concept $c_i \in \mathcal{C}$ to its prototype $p_i \in \mathcal{P}$.

Besides that, for our purposes, the individuals (members of $\mathcal{I}$) are considered as $q - tuples$, representing the respective values of the q attributes that characterize each instance. Thus, each $i_j \in \mathcal{I} = (v(i_j, a_h), v(i_j, a_l), ..., v(i_j, a_p))$, where a_h, a_l and a_p are attributes of i_j.

In our proposal, the sets $\mathcal{E}$ and $\mathcal{P}$ can be *explicitly assigned* to the members of $\mathcal{C}$, or can be automatically determined from the set $\mathcal{I}$. As a basic strategy, a *prototype* $p_i \in \mathcal{P}$ of a given concept $c_i \in \mathcal{C}$, such that $prot(c_i) = p_i$ can be extracted by analyzing the individuals in $ext(c_i)$ and by determining the *typical value* of each attribute of the individuals. If the attribute is numeric, the typical value can be the *average*; if the attribute is *categorical* (or nominal or symbolic), the typical value can be the *most frequent* (the *mode*).

Considering a given $c_i \in \mathcal{C}$, the set of its exemplars, $ex(c_i)$, should be selected in a way that, collectively, its members provide a good sample of the variability of the individuals in $ext(c_i)$. Also, it is important to consider that the exemplars of a concept can be used for supporting the classification of a given individual i and that, for performing this process, it can be necessary to compare i with every exemplar of every concept of the ontology. Thus, it is not desirable to consider all records in $ext(c_i)$ as exemplars for representing c_i, since the computational cost of the classification process is proportional to the number of exemplars that are selected for representing the concepts. Due to this, in our approach we consider that the number of exemplars related to each concept $c_i \in \mathcal{C}$ is defined as a percentage ep (defined by the user) of $|ext(c_i)|$ (where $|S|$ is the cardinality of the set S). This raises the problem of how to select which individuals in $ext(c_i)$ will be consider as the exemplars in $e(c_i)$. We select three main criteria that an individual $i_j \in ext(c_i)$ should meet for being included in $ex(c_i)$: (i) i_j should have a high degree of dissimilarity with the prototype given by $prot(c_i)$; (ii) i_j should have a high degree of similarity with a big number of observations in $ext(c_i)$; and (iii) i_j should have a high degree of dissimilarity with each exemplar already included in $ex(c_i)$. This set of criteria was developed for ensuring that the set of exemplars in $ex(c_i)$ will cover in a reasonable way the spectrum of variability of the individuals in $ext(c_i)$. That is, our goal is to preserve in $ex(c_i)$ some uncommon individuals, which can be not well represented by $prot(c_i)$, but that represent the variability of the individuals. In our approach, we apply these criteria, by including in $ex(c_i)$ the k first individuals from $ext(c_i)$ that maximize their *exemplariness index*. The exemplariness index is computed using the notion of *density* of a given individual. Regarding some concept $c_i \in \mathcal{C}$, the density of some individual $i_j \in ext(c_i)$, is computed by the function $density\colon \mathcal{I} \times \mathcal{C} \to \mathbb{R}$, such that,

$$density(i_j, c_i) = -\frac{1}{|ext(c_i)|} \sum_{p=1}^{|ext(c_i)|} d(i_p, i_j) \tag{2}$$

, where d is some dissimilarity (or distance) function (a function that measures the dissimilarity between to entities). Considering this, the set $ex(c_i)$ of some concept c_i, with k exemplars, can be computed by the Algorithm 1.

Algorithm 1: extractExemplars

```
Input: A concept c and a number h of exemplars
Output: A set exemplars of h instances representing the exemplars of the concept c.
begin
    exemplars ← ∅;
    for j ← 1 to h do
        eIndex_max ← −∞;
        i_max ← null;
        foreach individual ∈ ext(c) do
            density ← density(individual, c);
            dp ← d(individual, prot(c));
            med ← 0;
            if exemplars is not empty then
                Compute the distance between individual and each exemplar already included in exemplars and assign
                to med the distance of the nearest exemplar from individual;
            /* eIndex is the exemplariness index */
            eIndex = dp + density + med;
            if eIndex > eIndex_max then
                eIndex_max ← eIndex;
                i_max ← individual;
        exemplars ← exemplars ∪ {individual};
    return exemplars;
```

Notice that Algorithm 1 basically selects from $ext(c)$, the individuals that maximize the *exemplariness index*, which is the sum of: (i) distance (or dissimilarity) of the individual from the $prot(c)$; (ii) the *density* of the individual, considering the set $ext(c)$; and the distance (or dissimilarity) of the individual from its nearest exemplar, already included in $exemplars$.

Once a given extended ontology has its concepts, prototypes and exemplars, they can be used by a *hybrid classification engine* for classifying individuals. This component takes as input an individual and provides its corresponding classifications (a set of concepts $classifications \subseteq \mathcal{C}$). Firstly, the classification engine applies a conventional logical reasoning procedure (using the classical part of the extended ontology) for providing a first set of classification hypothesis. Notice that this reasoning process can infer more than one classification for the same individual. If this process provides, as classifications, concepts that are not specific (if they are not leaves of the taxonomy), the similarity-based reasoning can be used for determining more specific interpretations. The *hybrid classification engine* implements the Algorithm 2.

Algorithm 2: hybridClassification

Input: An individual i.
Output: A set $classification_{set}$ of concepts representing the classifications of i.
begin
  $classification_{set} \leftarrow \varnothing$;
  Perform the logical reasoning for interpreting i, and include the concepts of the resulting classification in $classification_{set}$;
  if *the concepts in* $classification_{set}$ *are not specific* **then**
    $hyp_{set} \leftarrow \varnothing$;
    foreach $c \in classification_{set}$ **do**
      Find the leaves in the taxonomy, whose root is c, and include them in hyp_{set};
    $classification_{set} \leftarrow \varnothing$;
    $MAX \leftarrow -\infty$;
    foreach $c \in hyp_{set}$ **do**
      $app \leftarrow applicability(c, i)$;
      if $app > MAX$ **then**
        $MAX \leftarrow app$;
        $classification_{set} \leftarrow \{c\}$;
      else if $app = MAX$ **then**
        $classification_{set} \leftarrow classification_{set} \cup \{c\}$;
  return $classification_{set}$;

Notice that the Algorithm 2 uses the notion of *applicability*, which, intuitively measures the degree in that a given concept c can be applied as an interpretation for a given observation $individual$. The applicability is computed by the Algorithm 3, using the prototypes and exemplars of the concepts.

Algorithm 3: applicability

Input: A concept c and an instance i.
Output: A value $r \in \mathbb{R}$, which is the degree in that c can be applied as a classification for i.
begin
  $app \leftarrow 0$;
  $pSimilarity \leftarrow sim(i, prot(c))$;
  $eSimilarity \leftarrow 0$;
  Calculate the similarity $sim(i, ex_i)$ between i and each $ex_i \in e(c)$, and assign to $eSimilarity$ the similarity value of the most similar ex_i;
  $app \leftarrow pSimilarity + eSimilarity$;
  return app;

Notice that the Algorithm 3 uses the function sim for measuring the similarity. Intuitively, the similarity is the inverse of the dissimilarity (or distance) between two individuals. Thus, sim has values that are inversely proportional to the values obtained by the function d. Here, we assume that $sim(i_j, i_l) = exp(-d(i_j, i_l))$.

4. Conclusions and future works

In this paper, we propose the notion of *extended ontology*, which integrates the common features and capabilities of conventional ontologies (based on the classical paradigm of knowledge representation) with the capability of dealing with typical features in similarity-based reasoning processes. The extended ontologies can provide more flexibility in classification processes, in the cases that do not have enough information for being classified according to necessary and sufficient conditions.

In future works, we intend to investigate approaches of *instance selection* [Olvera-López et al. 2010] for enhancing our approach for selecting exemplars. Also, we intend to apply the notion of extended ontologies (as well as the algorithms proposed here) for improving the results obtained in [Carbonera et al. 2011, Carbonera et al. 2013, Carbonera et al. 2015] for the task of visual interpretation of depositional processes, in the domain of Sedimentary Stratigraphy. We are also investigating how this approach can be applied for solving other problems, such as ontology alignment. We hypothesize that it is possible to take advantage of the information represented in the form of prototypes and exemplars, as additional sources of evidences in the process of ontology alignment.

References

Carbonera, J. L. and Abel, M. (2015a). A cognition-inspired knowledge representation approach for knowledge-based interpretation systems. In *Proceedings of 17th ICEIS*, pages 644–649.

Carbonera, J. L. and Abel, M. (2015b). A cognitively inspired approach for knowledge representation and reasoning in knowledge-based systems. In *Proceedings of 24th International Joint Conference on Artificial Intelligence (IJCAI 2015)*, pages 4349–4350.

Carbonera, J. L., Abel, M., and Scherer, C. M. (2015). Visual interpretation of events in petroleum exploration: An approach supported by well-founded ontologies. *Expert Systems with Applications*, 42:2749–2763.

Carbonera, J. L., Abel, M., Scherer, C. M., and Bernardes, A. K. (2013). Visual interpretation of events in petroleum geology. In *Proceedings of ICTAI 2013*.

Carbonera, J. L., Abel, M., Scherer, C. M. S., and Bernardes, A. K. (2011). Reasoning over visual knowledge. In Vieira, R., Guizzardi, G., and Fiorini, S. R., editors, *Proceedings of Joint IV Seminar on Ontology Research in Brazil and VI International Workshop on Metamodels, Ontologies and Semantic Technologies*, volume 776.

Fiorini, S. R., Abel, M., and Carbonera, J. L. (2014). Representation of part-whole similarity in geology. *Earth Science Informatics*, Special Issue on Semantic e-Sciences.

Gärdenfors, P. (2004). *Conceptual spaces: The geometry of thought*. The MIT Press.

Murphy, G. L. (2002). *The big book of concepts*. MIT press.

Oltramari, A. and Lebiere, C. (2011). Extending cognitive architectures with semantic resources. In *Artificial General Intelligence*, pages 222–231. Springer.

Olvera-López, J. A., Carrasco-Ochoa, J. A., Martínez-Trinidad, J. F., and Kittler, J. (2010). A review of instance selection methods. *Artificial Intelligence Review*, 34(2):133–143.

Smith, B. (2004). Beyond concepts: ontology as reality representation. In *Proceedings of the third international conference on formal ontology in information systems (FOIS 2004)*, pages 73–84.

Studer, R., Benjamins, V. R., and Fensel, D. (1998). Knowledge engineering: Principles and methods. *Data and Knowledge Engineering*, 25(1-2):161–197.