Dual PECCS: a cognitive system for conceptual representation and categorization

(Article begins on next page)

17 August 2022

Dual PECCS: A Cognitive System for Conceptual Representation and Categorization

Antonio Lieto[a,b]* Daniele P. Radicioni[a]* and Valentina Rho[a]*

[a]*University of Turin, Dipartimento di Informatica, 10149 Corso Svizzera 185, Turin, Italy;*
[b]*ICAR-CNR, 90128 Viale delle Scienze 12 Ed.11, Palermo, Italy*

In this article we present an advanced version of Dual-PECCS, a cognitively-inspired knowledge representation and reasoning system aimed at extending the capabilities of artificial systems in conceptual categorization tasks. It combines different sorts of common-sense categorization (*prototypical* and *exemplars-based* categorization) with standard monotonic categorization procedures. These different types of inferential procedures are reconciled according to the tenets coming from the dual process theory of reasoning. On the other hand, from a representational perspective, the system relies on the hypothesis of conceptual structures represented as *heterogeneous proxytypes*. Dual-PECCS has been experimentally assessed in a task of conceptual categorization where a target concept illustrated by a simple common-sense linguistic description had to be identified by resorting to a mix of categorization strategies, and its output has been compared to human responses. The obtained results suggest that our approach can be beneficial to improve the representational and reasoning conceptual capabilities of standard cognitive artificial systems, and –in addition– that it may be plausibly applied to different general computational models of cognition. The current version of the system, in fact, extends our previous work, in that Dual-PECCS is now integrated and tested into two cognitive architectures, ACT-R and CLARION, implementing different assumptions on the underlying invariant structures governing human cognition. Such integration allowed us to extend our previous evaluation.

Keywords: Knowledge Representation, Categorization, Conceptual Spaces, Cognitive Architectures, Heterogeneous Proxytypes, ACT-R, Prototypes, Exemplars, Common-sense Reasoning, CLARION.

1. Introduction

In this work we present the extended version of an integrated knowledge representation system aimed at performing conceptual categorization tasks. It is named Dual-PECCS (after Prototypes and Exemplars-based Conceptual Categorization System), since it relies on two different sorts of cognitively-inspired common-sense categorization: *prototypical* and *exemplars-based* categorization. In addition, it is grounded on the theoretical tenets coming from the dual process theory of mind, and on the hypothesis of "heterogeneous proxytypes" developed in the area of the biologically inspired cognitive architectures (BICA).

The system aims at providing a unified framework for the conceptual categorization simulating some of the common-sense heuristic strategies exploited by humans in categorization tasks. More specifically, it integrates strategies based on prototypes and exemplars-based reasoning, as suggested by the psychological results coming form the area of experimental Cognitive Science. The current version of DUAL-PECCS has been integrated and tested in the ACT-R and CLARION cognitive architectures to investigate its compatibility with the models of cognition therein implemented (Anderson et al., 2004; Langley et al., 2009; Sun, 2006).[1]

While existing systems and architectures allow to selectively perform either prototype or exemplar-based categorization rather than autonomously adapting their strategy to the input being categorized (Anderson and Betz, 2001), conversely, DUAL-PECCS addresses this issue. In addition to the deployment of such common-sense categorization strategies, DUAL-PECCS also provides an integration of such types of non-monotonic reasoning with the classical categorization based on standard, deductive, processes. The flow and the interaction of such diverse reasoning mechanisms have been devised based on the tenets coming from the dual process theory of reasoning, and compared to the answers provided by human subjects. In this respect, the main goal of the current system is proposing a cognitively-inspired framework for conceptual representation and categorization. Nonetheless, some preliminary results (see (Lieto et al., 2015a)) show that the resulting system also provides advances in performing common-sense categorization of linguistic descriptions compared with state-of-the-art question-answering systems (including Bing, Google and Wolfram-Alpha). However, a deeper cross-systems comparison is out of the scope of our present contribution, and is ongoing work.

This work is organized as follows: in Section 2 we introduce the theoretical bases inspiring our system, coming from the research in Cognitive Science. In Section 3 we start by outlining the overall architecture and describe the heterogeneous Knowledge Base adopted (Section 3.1), we then illustrate the pipeline going all throughout from the input textual description to the category in output (Section 3.2) and provide the detailed algorithms devised to implement the categorization task (Section 3.3). In Section 4 we show how the hybrid system for the conceptual categorization was integrated into ACT-R and CLARION, and in Section 5 we describe the evaluation of the resulting system, by introducing the experimental setting adopted and discussing the obtained results. Finally, we elaborate on future work.

2.   Prototypes, Exemplars and Proxytypes

In Cognitive Science different theories about the nature of concepts have been proposed. According to the traditional view, known as "classical" or Aristotelian theory, concepts can be simply defined in terms of sets of necessary and sufficient conditions. Such theory was dominant until the mid '70s of the last Century, when Rosch's experimental results demonstrated the inadequacy of such a theory for ordinary –or *common-sense*– concepts (Rosch, 1975). Rosch's results seemed to suggest, on the other hand, that ordinary concepts are characterized and organized in our mind in terms of *prototypes*. Since then, different theories of concepts have been proposed to explain different representational and reasoning aspects concerning the problem of typicality: we focus here on the proto-

---

[1]The previous version of the system was integrated only into ACT-R (Lieto et al., 2015c). Thanks to this additional integration, the current evaluation has been performed on both cognitive architectures and tested on a larger dataset.

type theory and on the exemplars theory.[2] According to the *prototype* view, knowledge about categories is stored in terms of prototypes, i.e., in terms of some representation of the "best" instance of the category. In this view, the concept *bird* should coincide with a representation of a typical bird (e.g., a robin). In the simpler versions of this approach, prototypes are represented as (possibly weighted) lists of typical features. According to the *exemplar* view, a given category is mentally represented as set of specific exemplars explicitly stored in memory: the mental representation of the concept *bird* is a set containing the representation of (some of) the birds we encountered during our past experience.

Although these approaches have been largely considered as competing ones (since they propose different models and predictions about how we organize and reason on conceptual information), they turned out to be not mutually exclusive (Malt, 1989). Rather, they seem to succeed in explaining different classes of cognitive *phenomena*, such as the fact that human subjects use different representations to categorize concepts: some use exemplars, a few rely on prototypes, and often both exemplars and prototypes are employed (Smith and Minda, 1998). This distinction also has neural plausibility, as witnessed by empirical research (Squire and Knowlton, 1995). Such experimental evidences led to the development of the so called "heterogeneous hypothesis" about the nature of concepts: this approach assumes that concepts do not constitute a unitary phenomenon, and hypothesizes that different types of conceptual representations may co-exist: prototypes, exemplars, classical representations, and so on (Machery, 2009). All such representations, in this view, constitute different *bodies of knowledge* and contain different types of information associated to the the same conceptual entity. Furthermore, each body of conceptual knowledge is featured by specific processes in which such representations are involved (e.g., in cognitive tasks like recognition, learning, categorization, *etc.*). In particular prototypes and exemplars-based representations are associated with the possibility of dealing with non-monotonic strategies of reasoning and categorization, while the classical representations (i.e. that ones based on necessary and/or sufficient conditions) are associated with standard deductive mechanism of reasoning.[3]

In recent years an alternative theory of concepts has been proposed: the *proxytype theory*. It postulates a biological localization and interaction between different brain areas for dealing with conceptual structures, that have a direct counterpart in the distinction between *long term* and *working memory* (Prinz, 2002). Such characterization is partic-

---

[2]In Cognitive Science there is another well known typicality-based theory of concepts, called the theory-theory. Such approach adopts some form of holistic point of view about concepts. According to some versions of the theory-theories, concepts are analogous to theoretical terms in a scientific theory. For example, the concept *cat* is individuated by the role it plays in our mental theory of zoology. In other versions of the approach, concepts themselves are identified with micro-theories of some sort. For example, the concept *cat* should be identified with a mentally represented microtheory about cats. In the current contribution we do not take into consideration the theory-theory approach, since it is, to a certain extent, more vaguely defined when compared to both prototypes and exemplar based approaches. As a consequence, at present its computational treatment seems to be more problematic. Full-detailed reviews can be found in (Machery, 2009; Murphy, 2002).

[3]Let us assume that we have to categorize a stimulus with the following features: "it has fur, woofs and wags its tail". In this case, the result of a *prototype-based categorization* would be *dog*, since these cues are associated to the prototype of *dog*. Prototype-based reasoning, however, is not the only type of reasoning based on typicality. In fact, if an exemplar corresponding to the stimulus being categorized is available, too, it is acknowledged that humans use to classify it by evaluating its similarity w.r.t. the exemplar, rather than w.r.t. the prototype associated to the underlying concepts (Frixione and Lieto, 2013). For example, a penguin is rather dissimilar from the prototype of *bird*. However, if we already know an exemplar of penguin, and if we know that it is an instance of *bird*, it is easier to classify a new penguin as a *bird* w.r.t. a categorization process based on the similarity with the prototype of that category. This type of common-sense categorization is known in literature as *exemplars-based categorization*. Finally, an example of standard deductive categorization is the categorization as *triangle* of a stimulus described by the following features: "it is a polygon, it has three corners and three sides".

3

ularly interesting for the explanation of phenomena such as, for example, the activation (and the retrieval) of conceptual information. In this setting, concepts are seen as *prox-ytypes*.

Definition 1 (Proxytypes)**:** *A* proxytype *is any element of a complex representational network* stored in long-term memory *corresponding to a particular* category *that can be tokenized in working memory to 'go proxy' for that category* (Prinz, 2002).

In other terms, the proxytype theory, inspired by the work of Barsalou (Barsalou, 1999), considers concepts as *temporary constructs* of a given category, activated (tokenized) in working memory as a result of conceptual processing activities, such as concept identification, recognition and retrieval.

In its original formulation, however, proxytypes are depicted as monolithic conceptual structures, primarily intended as prototypes (Prinz, 2002). A revised view of this approach has been recently proposed in the area of BICA, hypothesizing the availability of a wider range of representation types than just prototypes (Lieto, 2014). They correspond to the kinds of representations hypothesized by the above mentioned heterogeneous approach to concepts. In this sense, proxytypes are assumed to be heterogeneous in nature (i.e., they are composed by heterogeneous networks of conceptual representations and not only by a monolithic one).

Definition 2 (Heterogeneous Proxytypes)**:** *Heterogeneous representations (such as* prototypes, exemplars, etc.*) for each conceptual category are* stored in long-term memory. *They can be activated and accessed by resorting to different categorization strategies. In this view, each representation has its associated accessing procedures.*

In the design of our system we followed the approach based on heterogeneous proxytypes[4] for both the representational level (that is, we devised a hybrid knowledge base composed of heterogeneous representations, each endowed with specific reasoning mechanisms) and for the 'proxyfication' (that is, the set of procedures implementing the tokenization of the different representations in working memory).

3.   Dual Process Architecture for Conceptual Representation and Processing

As earlier mentioned, the DUAL-PECCS relies –at the representational level– on the heterogeneous proxytypes approach, and it is also inspired by the dual process theory of reasoning and rationality. More in detail, DUAL-PECCS is equipped with a hybrid knowledge base composed of heterogeneous representations of the same conceptual entities: that is, the hybrid knowledge base includes prototypes, exemplars and classical representations for the same concept. Such different bodies of knowledge act like *semantic pointers* towards the same conceptual entity (Blouw et al., 2015; Eliasmith et al., 2012; Thagard, 2012).[5]

---

[4]The characterization in terms of "heterogeneous proxytypes" also allows us to deal with the problem of the "contextual activation" of a given information based on the external stimulus being considered. The general idea is that, when we categorize a percept, we do not activate the whole network of knowledge related to its assigned category. Conversely, we only activate the knowledge that is "contextually relevant" in its respect. In other terms, we only proxyfy the type of representation that minimizes the distance w.r.t. the percept (see (Lieto, 2014) for further details).

[5]It is worth-noting, however, that the similarity with the semantic pointer perspective is limited to this point. In fact, as pointed out in (Lieto, 2014), while (Blouw et al., 2015; Eliasmith et al., 2012; Thagard, 2012) propose a

Both prototypes and exemplars are represented by adopting the conceptual spaces framework (see Section 3.1), while classical information is represented through standard symbolic formalisms (i.e., by means of a formal ontology).

From a reasoning perspective, instead, the retrieval of such representations is driven by different process types. In particular, prototype and exemplar-based retrieval is based on a fast and approximate kind of categorization, and benefits from defeasible, common-sense information associated to concepts. On the other hand, the retrieval of classical representation of concepts is featured by explicit rule following, and makes no use of defeasible, common-sense information. These two differing categorization strategies have been widely studied in psychology of reasoning in the frame of the dual process theory, that postulates the co-existence of two different types of cognitive systems (Evans and Frankish, 2009; Kahneman, 2011). The systems of the first type (*type 1*) are phyloge-netically older, unconscious, automatic, associative, parallel and fast. The systems of the second type (*type 2*) are more recent, conscious, sequential and slow, and featured by explicit rule following. We assume that both systems can be composed in their turn by many sub-systems and processes. According to the hypotheses in (Frixione and Lieto, 2012, 2014), the conceptual representation of our system includes two main sorts of components, based on these two sorts of processes. *Type 1* processes have been designed to deal with prototypes- and exemplar-based retrieval, while *Type 2* processes have been designed to deal with deductive inference.

The two sorts of system processes interact (Algorithm 1), since Type 1 processes are executed first and their results are then refined by Type 2 processes. In the implemented system the typical representational and reasoning functions are assigned to the System 1 (hereafter $\mathcal{S}1$), which executes processes of Type 1, and is associated to the conceptual spaces framework (Gärdenfors, 2000, 2014). On the other hand, the classical representational and reasoning functions are assigned to the System 2 (hereafter $\mathcal{S}2$) to execute processes of Type 2, and are associated to a standard Description Logics based ontological representation.

Figure 1 shows the heterogeneous representation for the concept *tiger*, with prototypical and exemplar-based representations semantically pointing to the same conceptual entity. In this example, the exemplar and prototype-based representations make use of non classical (or typical) information. Namely, the prototypical representation grasps information such as that tigers are wild animals, their fur has yellow and black stripes, etc.; the exemplar-based representations grasp information on individuals. For example, in Fig. 1 is represented an individual of *white-tiger*, which is a particular type of *tiger* with white fur.[6] Both sorts of representations activate Type 1 processes. On the other hand, the classical body of knowledge is filled with necessary and sufficient information to characterize the concept (representing, for example, the taxonomic information that a tiger is a mammal and a carnivore), and activates Type 2 processes. For the sake of readability the information in Figure 1 is visualized with a uniform format, even though the different representations are actually encoded in different formalisms.

In the following we introduce the two representational and reasoning frameworks adopted in our system, by focusing *i)* on how typicality information (including both prototypes and exemplars) and their corresponding non-monotonic reasoning procedures

---

heterogeneity at the level of the information channel (i.e., they consider, as "conceptual components", the different information channels through which the content of the information is provided) in our framework the focus is on the heterogeneity regarding the content of the represented information. In our opinion our axis of representation provides an additional level of generality, since the content distinction is cross-channel.

[6]This implies that subclasses can be possibly represented by several exemplars.
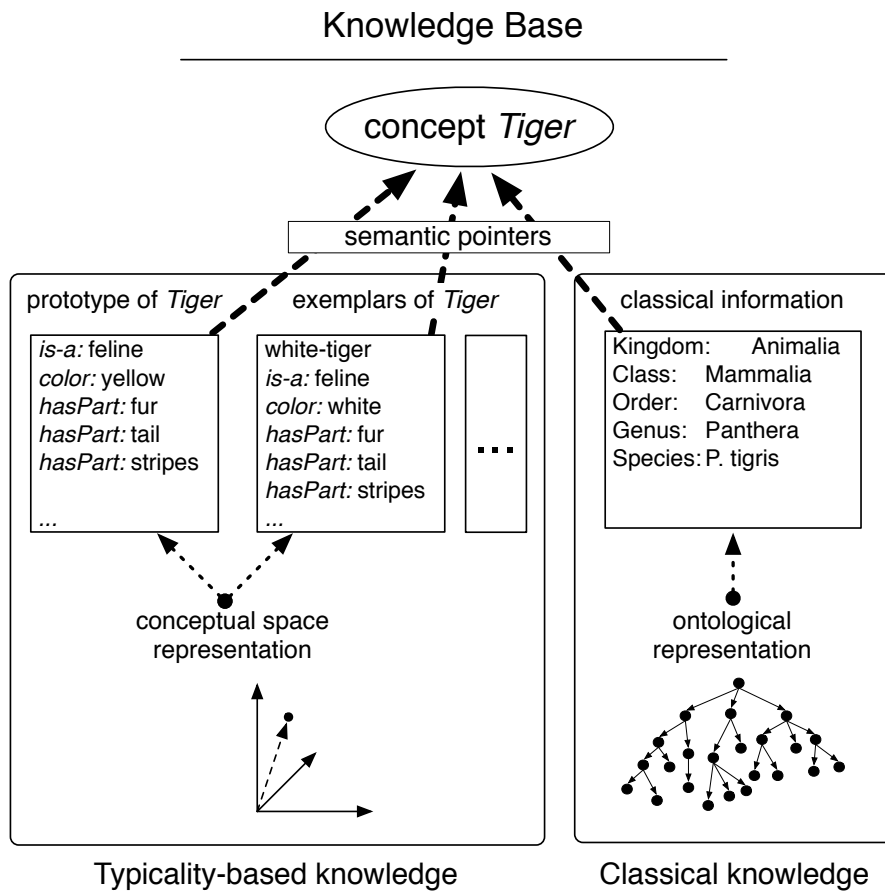
Figure 1.: Heterogeneous representation of the *tiger* concept in the hybrid knowledge base.

can be encoded through conceptual spaces; and *ii)* on how classical information can be naturally encoded in terms of formal ontologies.

3.1. $\mathcal{S}$1-$\mathcal{S}$2: Conceptual Spaces and Ontologies

Conceptual spaces (CS) are a representational framework where knowledge is represented as a set of *quality dimensions*, and where a geometrical structure is associated to each quality dimension.[7] In this setting, concepts correspond to convex regions, and regions with different geometrical properties correspond to different sorts of concepts (Gärdenfors, 2000). In addition, concepts are characterized in terms of *domains*; a domain is "a set of integral dimensions that are separable from all other dimensions" (Gärdenfors, 2014). Typical domain examples are color, size, shape, texture. In turn, domain information can be specified along some dimensions: e.g., in the case of the *color* domain, relevant dimensions are hue, chromaticity, and brightness.

   Prototypes are geometrically interpreted in conceptual spaces, in that they correspond to the geometrical centre of a convex region. This can be thought of as a centroid, that is the mean position of all the points in all dimensions. This representation also allows

---

[7]This framework has been adopted in different applicative contexts; for a recent overview, please refer to (Zenker and Gärdenfors, 2015).

us, given a convex region, to associate each point to a certain centrality degree, that can be interpreted as a measure of its typicality. Instances can be represented as points in a multidimensional space, and their similarity can be computed as the intervening distance between each two points, based on some suitable metrics.[8] This framework has been employed also to represent exemplars, that are modelled as points in the multidimensional space. In general, conceptual spaces can be used to compute the proximity between any two entities, and between entities and prototypes. In order to compute the distance between two points p_1, p_2 we use Euclidean metrics to calculate within-domain distance, while for dimensions from different domains we use the Manhattan distance metrics, as suggested in (Adams and Raubal, 2009; Gärdenfors, 2000). The weighted Euclidean distance $dist_E$ is computed as follows

$$dist_E(p_1, p_2) = \sqrt{\sum_{i=1}^{n} w_i(p_{1,i} - p_{2,i})^2} \ , \tag{1}$$

where i varies over the n domain dimensions and w_i are dimension weights.

In our implementation of conceptual spaces we represent points as vectors (with as many dimensions as required by the considered domain), whose components correspond to the point coordinates, so that a natural metrics to compute the similarity between them is *cosine similarity*. In this perspective two vectors with same orientation have a cosine similarity 1, while two orthogonal vectors have cosine similarity 0. The normalized version of cosine similarity ($\hat{cs}$), also accounting for the above weights w_i is computed as

$$\hat{cs}(p_1, p_2) = \frac{\sum_{i=1}^{n} w_i(p_{1,i} \times p_{2,i})}{\sqrt{\sum_{i=1}^{n} w_i(p_{1,i})^2} \times \sqrt{\sum_{i=1}^{n} w_i(p_{2,i})^2}}. \tag{2}$$

In the metric space we have defined, the distance between an individual and prototypes is computed with the Manhattan distance metrics. The distance between two concepts can be computed as the distance between two regions: namely, we can compute the distance between their prototypes, or the minimal distance between their *individuals*[9], or we can apply more sophisticated algorithms. Further details about technical issues can be found in (Ghignone et al., 2013).

Optionally, a context k can be defined as a set of weights, to grade the relative relevance of the considered dimensions –thus resulting in the following formulas: $dist_E(p_1, p_2, k)$ and $\hat{cs}(p_1, p_2, k)$ from Eq. 1 and 2, respectively–, and to adapt the computation to a variety of settings, such as, e.g., default values *vs.* known values, explicitly asserted values *vs.* computed values and/or inherited, *etc.*. Although it is widely accepted that context plays a major role in considering similarity issues (both in human judgements and in computational models), presently our similarity metrics does not involve contextually

---

[8]Our prototype-based representations and the corresponding categorization strategy adopt the same rationale as proposed by Edelman in his "Chorus of Prototypes" hypothesis where the comparison of a stimulus with a small number of prototypes is obtained via the calculation of proximity scores (e.g., in terms of Euclidean distance) between the feature vector of the perceived input and the *synthetic* vector of the prototypical representations (Edelman, 1995). Beyond what has been proposed by Edelman, however, our conceptual representation is enriched with the exemplars-based representations (and its corresponding reasoning strategy), as explained in the following pages. The integration (and harmonization) of both prototypes and exemplars-based representations and reasoning procedures represents one of the main advancements of our proposal.

[9]Individuals can be thought of as *exemplars*, that is elements that belong to the given concept by sharing properties and their related sets of values.

weighted dimensions, nor tunable metrics, such as those devised in (Aisbett and Gibbon, 1994). These aspects represent an interesting enhancement that we defer to future work. However, it is worth noting that linguistic context, also known as *discourse context* – which is the focus of the present work– is implicitly accounted for by DUAL-PECCS categorization strategy: in our setting we consider as contextually relevant elements only those explicitly expressed in the input description. The explicit elements of the discourse structure are therefore at the base of our notion of context: this implies that we compute similarity (between the *stimulus* and the representations in the hybrid knowledge base, see Figure 1) by considering only dimensions whose values are explicitly expressed in the linguistic descriptions.

The internal representation format for handling conceptual spaces has been introduced in (Lieto et al., 2014); we briefly recall it for the sake of self-containedness. Although the format has been developed by attempting to keep it as general as possible (so to extend its usage to further domains), the current implementation has been devised based on specific representational needs described in Section 5. The basic representational structure processed by the system is named genericDescription; it is actually a super-domain that hosts information about both physical and non physical features, arranged into nine further domains, such as size, shape, color, location, feeding, locomotion, hasPart, partOf, manRelationship.

To give an example of the conceptual spaces representation, let us consider the size domain: the size of entities is expressed through the three Euclidean dimensions; the shape allows expressing that an object has circular, square, spherical, cubic, etc., shape. The color space maps object's features onto the three dimensional $L^\star a^\star b^\star$ color space, in the same spirit as drawn by Gärdenfors in (Gärdenfors, 2000): in particular, in $L^\star a^\star b^\star$ terms $L^\star$ ($0 \leq L \leq 100$) is the correlate of lightness, $a^\star$ ($-128 \leq a \leq 127$) is the chromaticity axis ranging from green to red, and $b^\star$ ($-128 \leq b \leq 127$) is the chromaticity axis ranging from blue to yellow. The location domain indicates the place or the environment where the object being modeled can be typically found. It actually results from the combination of five dimensions, and namely: humidity, indicated as a percentage; temperature, ranging in $[-40°, 50°]$; altitude, ranging in $[-11000, 8848]$; vegetation, ranging in $[0, 100]$; time. In turn, time contains a partitioning of the hours of the day into sunrise (4–6 AM), morning (6–12 AM), afternoon (12–5 PM), evening (5–10 PM) and night (10 PM–4 AM). The locomotion domain combines two dimensions: the former dimension is used to account for the type of movement (1: swim, 2: dig, 3: crawl, 4: walk, 5: run, 6: roll, 7: jump, 8: fly), and the latter one is used to account for the speed, expressed in km/h (Bejan and Marden, 2006). The hasPart and partOf domains are used to complement the analogous ontological properties: in particular, we collected information about the following dimensions: name, number, and partSize, partColor that are intended to specialize the above illustrated spaces.

A simplified example of the *lion* prototype information is reported below.

```
<object name="lion">
  <genericPhysicalDescription>
    <size name="lion_size">
      <x>70</x>
      <y>120</y>
      <z>200</z>
    </size>
    <color name="beige">
      <l>63</l>
      <a>13</a>
      <b>32</b>
    </color>
```

```
        <location name="savanna">
          <humidity>50</humidity>
          <temperature>40</temperature>
          <altitude>100</altitude>
          <vegetation>50</vegetation>
        </location>
        <locomotion name="walk">
          <movement>4</movement>
          <speed>10</speed>
        </locomotion>
        <locomotion name="run">
          <movement>5</movement>
          <speed>40</speed>
        </locomotion>
      </genericPhysicalDescription>
</object>
```

On the other hand, the representation of the classical information regarding a given concept is demanded to classical ontological formalizations. Although relevant efforts have been made in the Description Logics community to enhance the ontological representation and reasoning capacities with the design of formalisms endowing types of typical representation and non-monotonic inference (Giordano et al., 2013), nonetheless the problem of representing and reasoning on typicality remains computationally expensive and practically intractable, and therefore of limited interest for concrete applications (Frixione and Lieto, 2012).[10] Formal ontologies, then, provide the characterization of concepts in terms of necessary and sufficient conditions (if these conditions exists: as mentioned, most common-sense concepts cannot be characterized in these terms). In our implementation, the ontological representation adopted by the $\mathcal{S}2$ component is grounded on the OpenCyc ontology, one of the widest ontological resources currently available containing more than $230,000$ concepts.

3.1.1.   On the Entity Mapping in Heterogeneous Resources

A relevant issue we face is aligning knowledge resources based on different sorts of representational formalisms. Although many efforts have been recently invested in the ontology mapping task (Euzenat et al., 2007; Shvaiko and Euzenat, 2013), in the present setting a different effort is required by knowledge bases that are as diverse as conceptual spaces and ontologies.

Under an architectural perspective, $\mathcal{S}1$ and $\mathcal{S}2$ rely on knowledge bases encoded in different ways, which need to be connected and mapped onto a shared and uniform representation of meaning in order to allow the ACT-R and CLARION layers to operate them. A rather common approach is to provide entities with sense identifiers, such as WordNet or BabelNet identifiers. WordNet (WN) is a lexical database for the English language (Miller, 1995). Different from common dictionaries –organizing terms alphabetically, but possibly scattering senses– it relies on the idea of grouping terms into synonyms sets (called *synsets*), that are equipped with short definitions and usage examples. Such sets are represented as nodes of a large semantic network, where the intervening edges represent a number of semantic relations among synset elements (such as hyponymy, hypernymy, antonymy, meronymy, holonymy). BabelNet is a wide-coverage multilingual semantic network resulting from the integration of lexicographic

---

[10]In this respect, then, our approach is alternative w.r.t. these ones, since its underlying cognitive assumptions allowed us to practically implement a system, thus avoiding intractability problems which are inherent in the above mentioned logic-oriented extensions.

	TOTAL	MAPPED	% SUCCESS
Wikipedia links	$19,876$	$7,698$	38.7
WordNet 2.1 links	$8,779$	$5,197$	59.2
total num of classes	$27,823$	$12,723$	45.7

Table 1.: Summary of the automatic mapping of OpenCyc classes onto WordNet 3.0 synset IDs.

and encyclopedic knowledge from WordNet and Wikipedia (Navigli and Ponzetto, 2010); it extends the constructive rationale of WN –based on sets of synonyms– through the structure of Wikipedia composed of redirect pages, disambiguation pages, internal links, inter-language links, and categorical information. More on the algorithm used to build BabelNet can be found in (Navigli and Ponzetto, 2012).

Entities represented in the conceptual spaces domains, processed by $\mathcal{S}1$, are provided with WordNet synset IDs; on the other side, classes in $\mathcal{S}2$ are featured by Cyc IDs. The linking between such identifiers is presently managed through a mapping table. It turns out, then, that the mapping process is a crucial one in order to practically handle the considered resources; a more detailed account on such anchoring aspects is provided in Section 3.1.2. Our mapping proceeded by considering that about 20% classes in OpenCyc contain references to external knowledge bases (KBs) such as WordNet 2.1 and Wikipedia. The automatically extracted mappings have been computed by following the references in OpenCyc towards BabelNet and WordNet, which produced as output about $13,000$ new associations. Namely, linking information in OpenCyc can be arranged in two classes:

- Wikipedia Links. A subset of classes in OpenCyc are provided with references to Wikipedia pages. Such references can be directly mapped onto a BabelSynset (in turn providing a pointer to the WordNet synset ID); each link from a BabelNet ID to a WordNet ID has been added to our mapping table. The steps to perform this kind of mapping were: [OpenCyc Class] $\rightarrow$ [BabelSynset ID] $\rightarrow$ [WordNet 3.0 ID].
- WordNet 2.1 Links. A subset of classes in OpenCyc contain a link to the online WordNet 2.1 APIs, whose IDs can be directly converted into WN 3.0 IDs through the SenseMap library.[11] The synset *name* has then been converted into a numeric ID through the online WordNet 3.0 APIs, and added to the mapping table. The steps to perform this kind of mapping were: [OpenCyc Class] $\rightarrow$ [WordNet 2.1 SynsetName] $\rightarrow$ [WordNet 3.0 SynsetName] $\rightarrow$ [WordNet 3.0 ID].

Table 1 illustrates some figures about the mapping of Wikipedia links and WordNet links, by showing how many links were found in OpenCyc, and how many of them could be mapped onto WN synset IDs. We observed that *i)* sometimes Cyc classes are linked to several WN synsets; and *ii)* in some cases such links are not precise, mostly due to the differing granularity of the information contained in the KB involved in the mapping process.

At the current stage of development, although the entities described through the conceptual spaces used by $\mathcal{S}1$ module require being substantially enriched, the mapping procedure allows handling some $13K$ descriptions.

---

[11]https://goo.gl/lRZYnW.

3.1.2.   On Mapping Conceptual Spaces and Ontological Representations

As above mentioned, an important aspect related to the proposed artificial architecture for conceptual systems regards the way in which the different kinds of representations (i.e., the symbolic structures of the ontological representation and the corresponding underlying conceptual space representations) are mapped one to another. We dealt with this problem by anchoring both the representations at the level of the general concept enveloping them. The heterogeneous proxytypes approach, in fact, requires the existence of co-referring representational structures where the different bodies of knowledge are assumed to semantically point to the main reference conceptual container (see Figure 1). In our approach such a container has been automatically provided with a WordNet synset identifier. In addition, also the corresponding pointing representations (conveying the different types of conceptual knowledge) have been equipped with the same WordNet synset ID.

The anchoring mechanism between the representations follows two different ways. The first one corresponds to the mapping between the general concept and its related ontological component ($\mathcal{S}2$ mapping). This mapping is already provided in the OpenCyc ontology, which is sometimes equipped with the information regarding the corresponding WordNet synset ID. On the other hand, the anchoring between the conceptual space representations and the corresponding general concept ($\mathcal{S}1$ mapping) is obtained as follows: the linguistic elements extracted through an Information Extraction step (please refer to Section 3.2) are automatically equipped with the corresponding WordNet synset ID, if any. After this process, Dual-PECCS inspects the Conceptual Spaces Knowledge Base searching for the elements corresponding to the extracted linguistic entities. This process is presently based on a simple string-matching between the names of the extracted linguistic entities and the names of the concepts available in the Conceptual Spaces. Once this mapping is provided, the WordNet synset ID assigned to the extracted linguistic entities is transferred to the corresponding representation in the Conceptual Spaces. Finally, the ID obtained in Conceptual Spaces is linked to the WN synset ID already associated to the general concept.

Interestingly, this approach could be easily extended to visual categorization systems by adding, to the whole networks of heterogeneous representations, the ImageNet identifiers (Deng et al., 2009). In the context of artificial vision and robotic applications there are proposals similar to our approach. In particular, the works in (Chella et al., 1997, 2003) also provides a mapping between different levels of representations, including conceptual spaces and symbolic representations. In this case, anchoring is performed for static and dynamic visual scenes. In particular, in the case of dynamic scenes it is based on an anchoring function defined from time t to couple assertions (formulas for the Situation Calculus) and Conceptual Space structures by using look-up tables. The main similarity with their approach by (Chella et al., 1997) is that also our model is primarily focused on the representational issue of the anchoring problem, and does not consider the procedural one (on these aspects, please refer to (Coradeschi and Saffiotti, 2000)). Conversely, the main difference w.r.t. their approach is that we adopt externally developed and well known resources for providing the linguistic anchoring (such as Wordnet). This allows our system a good level of compatibility and interoperability with other language technologies adopting the same resources. In the next sections we present the details regarding the categorization strategies adopted in the Dual-PECCS.

Data: Linguistic description d
Result: A class assignment, as computed by $\mathcal{S}1$ and $\mathcal{S}2$

1 $trialCounter \leftarrow 0$;
2 $closed^{\mathcal{S}1} = \{\emptyset\}$
3 **while** $trialCounter < maxTrials$ **do**
    // conceptual spaces output
4 $\mathsf{c} \leftarrow \mathcal{S}1(d,\ closed^{\mathcal{S}1})$;
5   **if** $trialCounter == 0$ **then** $\mathsf{c}^* \leftarrow \mathsf{c}$;
    // ontology based consistency check
6 $\mathsf{cc} \leftarrow \mathcal{S}2(d, conceptPointedBy(\mathsf{c}))$;
7   **if** cc *equals(conceptPointedBy(*c*))* **then**
8 | **return** $\langle \mathsf{c}^*, \mathsf{cc} \rangle$;
9   **else**
10 | $closed^{\mathcal{S}1}$ *add(conceptPointedBy(*c*))*
11   **end**
12 $++trialCounter$;
13 **end**
14 $\mathsf{cc} \leftarrow \mathcal{S}2(\langle d, \mathsf{Thing} \rangle)$;
15 **return** $\langle \mathsf{c}^*, \mathsf{cc} \rangle$;

Algorithm 1: The $\mathcal{S}1$-$\mathcal{S}2$ categorization process.

3.2. *Categorization Pipeline of the* Dual-PECCS

The whole categorization pipeline implemented by DUAL-PECCS works as follows. The input to the system is a simple linguistic description, like 'The animal that eats bananas', and the expected output is a given category evoked by the description (e.g., the category *monkey* in this case). After an Information Extraction (IE) step, the input information is encoded into an internal format devised to store conceptual spaces information, which is then used as input in the categorization task by adopting the strategies that will be described below.

A shallow IE approach has been devised, where the morphological information computed from input sentences has been used to devise a simple finite-state automaton describing the input sentences' structure (more on the input descriptions in Section 5). This approach would not scale to handle more complex sentences; its limitations are due to using morphological information, and in general are inherent in finite-state machines (which prevents us from dealing with parenthetical clauses, like relative clauses). We defer to future work the adoption of richer language models. Despite these limitations, however, it allowed us to complete the automatization of the software pipeline going all throughout from the simple linguistic input description used for the evaluation (that will be described later) to its final conceptual categorization.

3.3. *Dual Process Categorization*

The overall control strategy implemented by DUAL-PECCS governs the flow of interaction between the $\mathcal{S}1$ and $\mathcal{S}2$ systems (it is thus referred to as $\mathcal{S}1$-$\mathcal{S}2$ categorization). Its underlying rationale is to assess the approximate categorization results obtained by Type 1 processes in $\mathcal{S}1$ with the ontological information and the deliberative processes of reasoning implemented by $\mathcal{S}2$. The $\mathcal{S}1$-$\mathcal{S}2$ categorization process can be summarized as follows (Algorithm 1). The system takes in input a textual description d and produces in

Data: Linguistic description: d; list of inconsistent concepts: $closed^{\mathcal{S}1}$.
Result: A typicality based representation of a category.

1 $\mathcal{S}1_{\mathrm{EX}} \leftarrow categorizeExemplars(d)$;
2 **if** $firstOf(\mathcal{S}1_{\mathrm{EX}}, closed^{\mathcal{S}1}).distance(d) < similarityThreshold$ **then**
3 $\quad\mid\quad$ **return** $firstOf(\mathcal{S}1_{\mathrm{EX}}, closed^{\mathcal{S}1})$;
4 **else**
5 $\quad\mid\quad$ $\mathcal{S}1_{\mathrm{PR}} \leftarrow categorizePrototypes(d)$;
$\quad\quad\quad$ *// in case of equal distance prefer exemplars*
6 $\quad\mid\quad$ $typicalityCategorization \leftarrow sortResults(\mathcal{S}1_{\mathrm{EX}}, \mathcal{S}1_{\mathrm{PR}})$;
7 $\quad\mid\quad$ **return** $firstOf(typicalityCategorization, closed^{\mathcal{S}1})$;
8 **end**

Algorithm 2: $\mathcal{S}1$ categorization with prototypes and exemplars implementing the instruction in Algorithm 1: line 4.

output a pair $\langle \mathsf{c}^*, \mathsf{cc} \rangle$, the output of $\mathcal{S}1$ and $\mathcal{S}2$, respectively. If the categorization result provided by $\mathcal{S}1$ (based on the similarity calculation between the input and $\mathcal{S}1$ representations) is consistent with the ontology, then the categorization succeeded and the category provided by $\mathcal{S}2$ (cc) is returned along with c^*, the top scoring class returned by $\mathcal{S}1$. Otherwise the system evaluates a fixed amount ($maxTrials$) of $\mathcal{S}1$ candidates, meantime keeping track of the inconsistent elements that are added to the $closed^{\mathcal{S}1}$ list. In case all the $\mathcal{S}1$ candidates are inconsistent w.r.t. the ontology in $\mathcal{S}2$, the output of $\mathcal{S}2$, computed independently of $\mathcal{S}1$, is provided along with c^*. The control strategy implements a tradeoff between ontological inference and the output of $\mathcal{S}1$, which is more informative but also formally less reliable.[12]

The $\mathcal{S}1$ categorization algorithm implements the instruction in Algorithm 1, line 4 by determining which kind of $\mathcal{S}1$ output must be selected and then checked against the deliberative $\mathcal{S}2$ module (Algorithm 2). In particular, the algorithm is designed to activate either the prototypical-based or the exemplar-based representation, according to the actual input description. The implemented procedure works as follows: when the input *stimulus* –in our case a simple linguistic description– is similar enough to an exemplar representation (a threshold has been fixed to these ends), the corresponding exemplar of a given category is retrieved. Otherwise, the prototypical representations are also scanned and the representation (prototype or exemplar) that is closest to the input is returned. By following a preference that has been experimentally observed in human cognition (Medin and Schaffer, 1978), this algorithm favors the results of the exemplars-based categorization if the knowledge-base stores any exemplars similar to the input being categorized.

4.   Integrating Dual-PECCS into ACT-R and CLARION

The proposed system has been integrated into two of the most widely known cognitive architectures: ACT-R (Anderson et al., 2004) and CLARION (Sun, 2006). The underlying rationale behind such integration efforts is to investigate whether our approach is

---

[12]For example, given an input description such as "The very big fish that eats plankton", the $\mathcal{S}1$ component returns an approximate output that is *whale*, which is then discarded by the $\mathcal{S}2$ ontological reasoning process since whales are mammals and not fishes. The second item returned by $\mathcal{S}1$ is *whale-shark*, which is successfully checked against the ontological knowledge base. DUAL-PECCS then returns both *whale* (the top-scoring class according to $\mathcal{S}1$) and *whale-shark* (the second result of $\mathcal{S}1$ also consistent with the ontology).

compatible with architectures implementing different cognitive theories of mind; in this case, it can be considered a candidate general framework for representing and reasoning on conceptual information, and eventually tested with even further architectures.

One main difference between the two architectures is that CLARION natively assumes the perspective of the dual process theory; ACT-R, on the other hand, is not natively dual process based. Therefore, in the latter architecture, the dual mechanisms of reasoning need to be explicitly designed and implemented instantiated within an already existing general framework. In particular, in ACT-R cognitive mechanisms emerge from the interaction of two types of knowledge: the *declarative* knowledge, that encodes explicit facts that the system knows, and the *procedural* knowledge, that encodes rules for processing declarative knowledge. The declarative module is used to store and retrieve pieces of information called *chunks*, that are featured by a *type* and a set of attribute-value pairs, similar to frame slots. Finally, the central production system connects these modules by using a set of IF-THEN production rules.

Differently, in CLARION, cognitive processes are mainly subject to the activity of two sub-systems, called Action Centered Sub-system (ACS) and the Non-Action Centered Sub-system (NACS). Both sub-systems store information using a two-layered architecture, i.e., they both include an *explicit* and an *implicit* level of representation. The working memory, acting as temporary storage for decision making, is a part of the ACS, which also maintains the active behavior strategies. To hold general knowledge, the NACS provides a semantic memory consisting of both a rule-based layer that encodes explicit, symbolic knowledge, and of an underlying distributed layer with implicit, sub-symbolic representations. A rule in CLARION connects a condition, encoded as a chunk, with an action, encoded as another chunk. For both architectures we mainly focused on the Declarative Memory and Working Memory, and on the corresponding retrieval mechanisms (Figure 2).

Besides, the dual process strategies of concept categorization have been integrated into the ACT-R and CLARION processes and connected to the retrieval request executed in the Working Memory. More in detail: in the *Extended* Declarative Memory (equivalent to its counterpart, NACS, in CLARION) every concept is represented as an empty chunk (that is, a chunk having no associated information, except for its *WordNet synset ID* and a human readable *name*), referred to by the external bodies of knowledge (prototypes and exemplars) acting like semantic pointers. The novel dual process-based categorization mechanism triggers both the $\mathcal{S}1$ categorization and the $\mathcal{S}1$-$\mathcal{S}2$ categorization procedures. In this setting, when the categorization result of $\mathcal{S}1$ is returned, the representation activated in the Extended Declarative Long-Term Memory is *proxyfied* (i.e., recalled to the working memory, see Algorithm 1, line 4) in order to perform the $\mathcal{S}2$ consistency check (Algorithm 1, line 6), in the dual process perspective.

As regards as the ACT-R implementation, we have integrated our hybrid knowledge base directly into the declarative memory, differently from other approaches that have extended the knowledge capabilities of ACT-R based on the introduction of a new, *ad-hoc*, external module of declarative memory (Oltramari and Lebiere, 2012; Salvucci, 2014). We designed a novel retrieval request implementing the $\mathcal{S}1$-$\mathcal{S}2$ categorization mechanism by extending the repertoire of the retrieval buffer through a new action (symbolized by the operator $). Such action allows a direct access to the heterogeneous information represented by the $\mathcal{S}1$-$\mathcal{S}2$ external bodies of knowledge.[13] We designed two types of $

---

[13]The standard symbols already existing in ACT-R for the retrieval buffer are: '+' for standard chunk requests to the declarative memory module, '=' for chunk modification and testing, and '?' for buffer and module status checking. The standard ACT-R retrieval is featured by basically testing the equality of slot values (with the
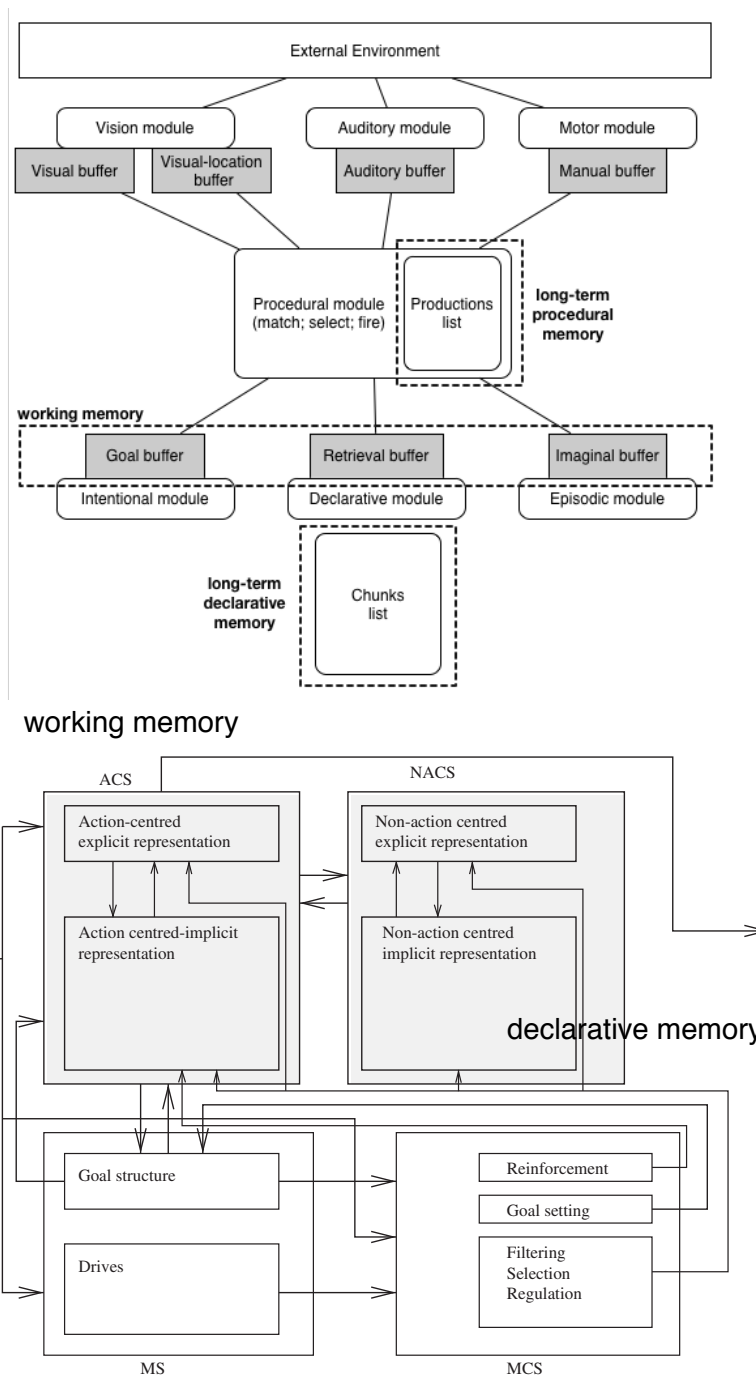
Figure 2.: ACT-R Architecture with the used memory slots in dotted frames (top), adapted from (Anderson et al., 2004); CLARION cognitive architecture with the working memory and the declarative memory emphasized through shaded frames (bottom), adapted from (Sun, 2007).

requests that are executed according to the specific type of request received from the

---

exception of some slots –that need to be explicitly indicated by the programmer– for which it is possible to exploit values similarity); conversely, our approach implements a similarity-based retrieval over all the considered dimensions, which is aimed at providing as result the element minimizing the distance from the input query.

retrieval buffer: the *approximate categorization request* and the *consistency request*. The *approximate categorization request* is activated when the retrieval request is generic: i.e., the request chunk lacks of the concept type (the *concept_id* slot set to `nil`), that in this kind of request needs to be retrieved. This task is similar to the *open request* that is possible to execute in ACT-R (Thomson et al., 2014), like in the following example.

```
$retrieval>
    concept_id nil
    family_1_name felin

    haspart_1_name fur
    haspart_1_color_1_name yellow
    haspart_1_color_1_l 80.0
    haspart_1_color_1_a -20.0
    haspart_1_color_1_b 94.0

    haspart_2_name stripe
    haspart_2_color_1_name black
    haspart_2_color_1_l 0.0
    haspart_2_color_1_a 0.0
    haspart_2_color_1_b 0.0
    [...]
```

This kind of request triggers the $\mathcal{S}1$ retrieval system, and its output is a chunk-like translation of the exemplar or the prototype resulting from the execution of the $\mathcal{S}1$ retrieval on the typicality-based knowledge. We used the *conceptual finsts*, by building on the notion of *declarative finsts* (Pylyshyn, 1989) delivered in ACT-R, to keep track of the representations that have been recently retrieved by the system $\mathcal{S}1$. In our implementation, conceptual finsts allow $\mathcal{S}1$ to exclude the elements already inspected and found inconsistent, by adding them to the $closed^{\mathcal{S}1}$ list. On the other hand, if the \$ request specifies the concept to which it refers (i.e., the \$ request chunk contains a filler for the *concept_id* slot), then we are dealing with a *consistency check* request, to be sent to the $\mathcal{S}2$ system: in this case, we convert the request and redirect it to the $\mathcal{S}2$ system that checks whether the features of the chunk are compatible with the proposed classification. The output of this request is a chunk where the slot *concept_id* is filled with the conceptual representation resulting from $\mathcal{S}2$.

The integration at the representational and reasoning level in CLARION followed the same rationale indicated in ACT-R, but has been adapted to the specific requirements of the architecture. In particular, we adopted both implicit and explicit representational layers provided by the NACS in order to create a direct mapping with our hybrid architecture: $\mathcal{S}1$ (and its typicality-based information represented with conceptual spaces) has been mapped onto the implicit layer, while $\mathcal{S}2$ (the classical, ontology-based representation), has been mapped onto the explicit one. The mapping between the sub-symbolic module of CLARION and the dimension-based representations of the conceptual spaces has been favored since such architecture also synthesizes the implicit information in terms of dimensions-values slots. The dual process based categorization mechanisms have been implemented based on the following procedure: every request is encoded in working memory as a particular type of instance (instance chunk). The dimensions and values of every instance chunk are filled through an update of the implicit module with the information extracted from the external stimulus (in the present case a linguistic description). After building the chunk request, a retrieval request is executed on the $\mathcal{S}1$ knowledge base with the aim at retrieving an exemplar or a prototype-based representation. Such result is stored in working memory, and checked, as previously illustrated, with the knowledge of the external $\mathcal{S}2$ knowledge base (the Cyc ontology in our case). More specifically, the

categorization process starts in CLARION when an instance chunk is built and added to the working memory; such process is executed in the ACS module, and it is arranged as a series of *rounds*, each producing a query to the implicit $\mathcal{S}1$ component and to the explicit $\mathcal{S}2$ module. The ACS module initializes the input layer of the $\mathcal{S}1$ module, based on the instance chunk being considered. This initialization requires to handle external stimuli (the *world*) along with internal information, at disposal of CLARION agents. Let us start from the *sensory input* space: this space represents the agent's percepts, and is encoded as a set of pairs $\langle dimension, value \rangle$. Populating the sensory input space (and therefore building the instance chunk for the request to be sent to the NACS declarative memory) involves adding the appropriate set of $\langle dimension, value \rangle$ pairs (when such information, extracted from the input *stimuli*, is available). Filling a value of a dimension in CLARION is based on the sub-symbolic activation of that dimension when the external input is processed.

In our implementation the available dimensions that a chunk can assume is based on the set of dimensions defined in (Lieto et al., 2015b) for the encoding of conceptual spaces (therefore the internal information that CLARION can process is fixed). It is worth nothing that the activated chunk can lack of some information (i.e., a *dimension* not filled with its corresponding *value*), since by definition percepts include noisy or partially missing information.

When the instance chunk is formed a request to the implicit $\mathcal{S}1$ component is called. The output will be either a chunk representing a prototype or an exemplar representation, corresponding to the prototype/exemplar returned by $\mathcal{S}1$. Such intermediate result is then added to the working memory for the consistency check, which is performed by the explicit module. In turn, the explicit module tests the consistency of the instance chunk recently added to the Working Memory: if the instance chunk is consistent with the ontological knowledge base, then the explicit module returns the concept chunk corresponding to the output of $\mathcal{S}1$, thus implementing the notion of *proxyfication*; otherwise, a special chunk is returned to indicate that an inconsistency was detected. In this case the explicit module evaluates the next result returned by the implicit module, and the whole $\mathcal{S}1$-$\mathcal{S}2$ process is iterated until a representation consistent with the KB is found (as illustrated in Algorithm 1: line 3), or by exiting with a failure in case the maximal number of rounds is reached (Algorithm 1: line 14).

In the next sections we present and discuss some results obtained by the system.

5.   Evaluation

A dataset composed of 112 descriptions (corresponding to very simple riddles), was collected and given in input to the implemented system: namely, we selected 56 descriptions for which an exemplar-based representation was expected to be retrieved, and 56 descriptions for which a prototype-based representation was expected to be retrieved. This experimentation extends a previous one, presented in (Lieto et al., 2015c). These stimuli have been built by a multidisciplinary team composed of neuropsychologists, linguists and philosophers in the frame of a project aimed at investigating the brain activation of visual areas in tasks of lexical processing even for very abstract concepts. An example of such descriptions is "The big carnivore with yellow fur and black stripes", where the expected category to be retrieved was *tiger*, and in particular its representation corresponding to the "prototype of *tiger*"; conversely, a description such as "The big carnivore with *white*

fur and black stripes" was expected to lead as answer to "exemplar of *white_tiger*".[14]

The *expected* categorical targets represent a gold standard, since they correspond to the results provided by human subjects in a twofold psychological experimentation. In the first experiment, 30 healthy volunteers (16 females and 14 males) were recruited for the experiment from the Computer Science and Psychology Departments of the University of Turin. Participants were all naïve to the experimental procedure and to the aims of the study. Participants were asked to perform an inferential task "Naming from definition": the subjects were presented the stimuli and asked to overtly name, as accurately and as fast as possible, the target concept corresponding to the definition, using a microphone connected to a response box. The stimuli were presented through the E-Prime software, which was also used to record data on accuracy and reaction times; a richer account of this experimentation is provided in (Radicioni et al., 2015). An additional experimentation has been run on a dataset also including exemplar-based representations along with prototype-based representations. In this case, 10 subjects (4 females and 6 males) were recruited among PhD students and Postdoc fellows from the Computer Science Department of the University of Turin. As for the previous experiment, participants were all naïve to the experimental procedure and to the aims of the study. The experimental design was borrowed from the third experiment described in (Malt, 1989): subjects underwent a training phase where exemplar information was also explicitly taught. For example, for the concept *tiger* they were presented an exemplar of *siberian_tiger*, an exemplar of *malaysian_tiger* and a prototypical *tiger*. After the training phase they were asked to perform the same inferential task "Naming from definition", by naming not only the target concept corresponding to the definition, but also the specific representation (either exemplar or prototype) activated by the linguistic definition.

In both experimentations the recorded answers were consistent with our assumptions and more in general with the literature concerning prototypes and exemplar-based retrieval, which allows us to consider the target concepts like the *expected categories*, and the target representations as *expected proxyfied representations*.

To assess the DUAL-PECCS system we have considered a twofold experimental setting. In the first case we tested the whole pipeline, where the salient information is extracted by starting from the linguistic description, the corresponding representation is retrieved, proxyfied and loaded in working memory according to the dual process approach. The information extraction of the linguistic input is not implemented in ACT-R, but it relies on the CoreNLP Stanford Parser (Manning et al., 2014), which is used to convert the textual description into a chunk request. This measure is intended to assess the robustness of the overall approach, from the input parsing to the final categorization. In the second case we tested the heterogeneous proxytypes approach by directly starting with a manual encoding of the linguistic stimulus in terms of chunk request: this measure is intended to assess the accuracy in the categorization task of the hybrid system, featured by dual process approach, heterogeneous representation and reasoning, proxyfication, integration in the ACT-R architecture, but no information extraction.

In both cases (all linguistic pipeline *vs.* 'clean' input) we recorded two distinct metrics:
i) Concept-categorization accuracy (CC-ACC metrics) This metrics was designed to evaluate the final categorization, that is the accuracy in retrieving the expected concept (in this case, the wrong proxyfication did not count as error).
ii) Proxyfication accuracy (P-ACC metrics) This metrics was designed to evaluate whether

---

[14] A prototype of the system is available at the URL `https://goo.gl/cyThwa` together with the example files containing the full list of descriptions employed in the experimentation. A stable version of the system is available at `http://www.dualpeccs.di.unito.it/`

Table 2.: The accuracy results (Table 2-a) and the analysis of the proxyfication errors (Table 2-b).

a. Accuracy rates obtained for the conceptual categorization accuracy (CC-ACC) and proxyfication accuracy (P-ACC) metrics.

test	CC-ACC	P-ACC
with no IE	89.3% (100/112)	79.0% (79/100)
with IE	77.7% (87/112)	71.3% (62/87)

b. Analysis of the errors in the proxyfication (P-ACC metrics).

test	Proxyfication error		
	Ex-Proto	Proto-Ex	Ex-Ex
with no IE	21.0% (21/100)	0.0% (0/100)	0.0% (0/100)
with IE	28.8% (26/87)	0.0% (0/87)	5.8% (5/87)

given in input a description evoking a given concept, the expected proxyfied representation was retrieved. In this case the confusion between prototype and exemplar (or between exemplars) is scored as error even if the expected category is returned.

5.1. *Results and Discussion*

The system obtained an accuracy of 89.3% in conceptual retrieval (that reduces to 77.7% when performing the IE step), and 79% in the proxyfication task (71.3% in the setting with the IE). These figures are reported in Table 2.

The results in the conceptual categorization are in line with those previously reported in (Lieto et al., 2015a), although the dataset was presently more diverse by including exemplars, as well. The results of the whole pipeline (Table 2-a, second row) provide a baseline for future implementations of the IE step; we observe that, although producing 11.6% error either in the POS tagging or in the IE step proper, this result is also in line with those reported in (Lieto et al., 2015b). This fact shows that the approach –devised to match the simple linguistic structures in the considered stimuli, and which was expected not to generalize to handle further linguistic descriptions– maintains its performance when dealing with a broader dataset. The IE step significantly affects also the P-ACC metrics: that is, if we restrict to considering cases where the concept was categorized as expected, the proxyfication step is performed correctly in 79.0% of descriptions with 'clean' input, and only in 71.3% of cases with the Information Extraction step.

Table 2-b reports the detailed errors committed in the proxyfication phase; here we distinguish three cases. Provided that proxyfication errors occur only when the concept has been correctly categorized, three kinds of proxyfication errors were recorded: an exemplar returned in place of an expected prototype (column Ex-Proto); a prototype returned in place of an expected exemplar (column Proto-Ex), or by retrieving a wrong exemplar (e.g., an individual of *siberian_tiger* in place of an individual of *malaysian_tiger*, column Ex-Ex). Notably, the vast majority of errors are due to confusion between exemplars and prototypes: in particular, in the 21% of the considered stimuli an exemplar-proxyfied representation has been returned by the system in spite of the expected prototype. This sort of error raises to 28.8% in the implementation including the IE task. This error was caused

by the fact that in case exemplar based representations in the KB are equipped with the typical information matching the linguistic description being categorized, then such representations are always favored w.r.t. their prototypical counterpart (see Section 3.1, Algorithm 2). However this fact, although in line with psychological experimental evidence where exemplars –if available– are mostly returned as first choice, is counterintuitive for very general descriptions situated at a high level of abstraction. For example, given the input description "The animal that eats bananas", the Dual-PECCS retrieves the representation of the exemplar associated to *roloway_monkey*, while the expected output, based on the answers provided by human subjects, is a generic prototype of *monkey* since the provided description is quite general. This particular class of errors deserves additional clarification in future works: albeit emerged experimenting the computational model, it also suggests that additional analysis is needed in the theoretical debate about exemplars and prototypes. Our results seem to demand deepening the heuristics that steer our categorical choices when the stimulus at hand spans different levels of abstraction. Therefore also from an epistemological perspective this result is interesting, since it shows how cognitively-inspired computational models of cognition taking into account a *structuralist* perspective, can fruitfully provide insights to the theoretical counterparts that they implement, in a continuous cycle of interaction between theoretical and experimental settings (see e.g. (Lieto and Radicioni, 2016) for the distinction between *structuralist* and *functionalist* artificial models of cognition).

The rest of errors are less relevant, except for the type Ex-Ex, where we observe a 5.8% error rate, which is mostly due to the noisy features extracted from the linguistic descriptions.

6.   Conclusions

This article has illustrated two main advancements, in that Dual-PECCS provides the heterogeneous proxyfication approach –governed by the dual process theory– with a working implementation; the resulting system is able to *autonomously* perform prototypical and exemplar-based categorization. Additionally, it has been integrated into ACT-R and CLARION, thus showing a good level of compatibility with two general cognitive systems making different theoretical assumptions about the architecture of human cognition. Although there is room for both refining the theory and tuning the implemented system, the obtained experimental results are encouraging. Our proposal, in addition, seems to suggest, in perspective, a suitable way to deal with both the size and the heterogeneity problems affecting the knowledge level in cognitive architectures (Lieto, 2015).

As mentioned, an aspect that emerged from the experimentation deserves further investigation: namely, it regards the phenomenon of the proxyfication errors[15] of exemplar-based representations, even for stimuli representing quite general typical descriptions. Such descriptions, in fact, determine in humans the activation of prototypes.

As a future work, we plan to extend the current integrated $S1$-$S2$ knowledge base (in particular the $S1$ component) by recurring to both semi-automatic enrichment of conceptual spaces through the use of existing resources such as ConceptNet (Liu and Singh, 2004), as well as to manual annotation by exploiting also crowdsourcing and gamification systems.

---

[15]We remark that we considered as errors results differing from the answers provided by the human subjects interviewed.

Additionally, as a mid term goal, we plan to integrate the proposed representational and reasoning framework into further general cognitive architectures (e.g., SOAR, LIDA, OpenCog etc.). Such set of integrations, should it prove to be feasible, would show whether the proposed representational and reasoning system may be used as a reference framework for different cognitive architectures. Moreover, it will allow us to simulate brain disorders related to the activation and retrieval of conceptual information by directly building on processes and patterns proper to each architecture. Some disorders can be thought of as involving the access to conceptual structures that may inhibit, in different ways, the "proxyfication" process of the different representational elements of our hybrid framework. A particularly interesting disorder, to our present concerns, regards the impairment known as Semantic Dementia, selectively involving the capability of the "proxyfication" of the $\mathcal{S}1$ representations and not the subsequent $\mathcal{S}2$ access and control check.

Finally, further representational levels and sources can be foreseen that fit the needs of cognitive architectures, e.g., dealing with sensory-motor representations, visual representations and imagery, *etc.*. Relatedly, it should be possible to extend the proposed framework to integrate such additional levels into the DUAL-PECCS: for example, *affordances* can be thought of as prototypical sensory-motor representations associated to a given concept. We plan, as a long-term goal, to provide a comprehensive account of such aspects in our system.

Acknowledgements

Funding

References

Adams, B. and Raubal, M. (2009). A metric conceptual space algebra. In Hornsby, K. et al., editors, *COSIT*, volume 5756 of *LNCS*, pages 51–68. Springer.

Aisbett, J. and Gibbon, G. (1994). A tunable distance measure for coloured solid models. *Artificial Intelligence*, 65(1):143–164.

Anderson, J. R. and Betz, J. (2001). A hybrid model of categorization. *Psychonomic Bulletin & Review*, 8(4):629–647.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., and Qin, Y. (2004). An integrated theory of the mind. *Psychological review*, 111(4):1036.

Barsalou, L. W. (1999). Perceptual symbol systems. *Behavioral and Brain Sciences*, 22(04):577–660.

Bejan, A. and Marden, J. H. (2006). Constructing animal locomotion from new thermodynamics theory. *American Scientist*, 94(4):342.

Blouw, P., Solodkin, E., Thagard, P., and Eliasmith, C. (2015). Concepts as semantic pointers: a framework and computational model. *Cognitive Science.*

Chella, A., Frixione, M., and Gaglio, S. (1997). A cognitive architecture for artificial vision. *Artificial Intelligence*, 89(1):73–111.

Chella, A., Frixione, M., and Gaglio, S. (2003). Anchoring symbols to conceptual spaces: the case of dynamic scenarios. *Robotics and Autonomous Systems*, 43(2):175–188.

Coradeschi, S. and Saffiotti, A. (2000). Anchoring symbols to sensor data: Preliminary report. In Kautz, H. A. and Porter, B. W., editors, *AAAI/IAAI*, pages 129–135. AAAI Press / The MIT Press.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 248–255. IEEE.

Edelman, S. (1995). Representation, similarity, and the chorus of prototypes. *Minds and Machines*, 5(1):45–68.

Eliasmith, C., Stewart, T. C., Choo, X., Bekolay, T., DeWolf, T., Tang, Y., and Rasmussen, D. (2012). A large-scale model of the functioning brain. *Science*, 338(6111):1202–1205.

Euzenat, J., Shvaiko, P., et al. (2007). *Ontology matching*, volume 333. Springer.

Evans, J. S. B. and Frankish, K. E. (2009). *In two minds: Dual processes and beyond.* Oxford University Press.

Frixione, M. and Lieto, A. (2012). Representing concepts in formal ontologies: Compositionality vs. typicality effects. *Logic and Logical Philosophy*, 21(4):391–414.

Frixione, M. and Lieto, A. (2013). Representing non classical concepts in formal ontologies: Prototypes and exemplars. In *New Challenges in Distributed Information Filtering and Retrieval*, pages 171–182. Springer.

Frixione, M. and Lieto, A. (2014). Towards an Extended Model of Conceptual Representations in Formal Ontologies: A Typicality-Based Proposal. *Journal of Universal Computer Science*, 20(3):257–276.

Gärdenfors, P. (2000). *Conceptual spaces: The geometry of thought.* MIT press.

Gärdenfors, P. (2014). *The Geometry of Meaning: Semantics Based on Conceptual Spaces.* MIT Press.

Ghignone, L., Lieto, A., and Radicioni, D. P. (2013). Typicality-Based Inference by Plugging Conceptual Spaces Into Ontologies. In Lieto, A. and Cruciani, M., editors, *Procs. of the International Workshop on Artificial Intelligence and Cognition*, pages 68–79. CEUR.

Giordano, L., Gliozzi, V., Olivetti, N., and Pozzato, G. L. (2013). A non-monotonic description logic for reasoning about typicality. *Artificial Intelligence*, 195:165–202.

Kahneman, D. (2011). *Thinking, fast and slow.* Macmillan.

Langley, P., Laird, J. E., and Rogers, S. (2009). Cognitive architectures: Research issues and challenges. *Cognitive Systems Research*, 10(2):141–160.

Lieto, A. (2014). A computational framework for concept representation in cognitive systems and architectures: Concepts as heterogeneous proxytypes. *Procedia Computer Science*, 41:6–14.

Lieto, A. (2015). Some epistemological problems with the knowledge level in cognitive architectures. In *Proceedings of AISC 2015, 12th Italian Conference on Cognitive Science.*

Lieto, A., Minieri, A., Piana, A., and Radicioni, D. P. (2014). A Dual Process Architecture for Ontology-Based Systems. In Filipe, J., Dietz, J., and Aveiro, D., editors, *Proceedings of KEOD 2014, 6th International Conference on Knowledge Engineering and Ontology Development*, pages 48–55. SCITEPRESS – Science and Technology Publications.

Lieto, A., Minieri, A., Piana, A., and Radicioni, D. P. (2015a). A knowledge-based system for prototypical reasoning. *Connection Science*, 27(2):137–152.

Lieto, A. and Radicioni, D. P. (2016). From human to artificial cognition and back: New perspectives on cognitively inspired ai systems. *Cognitive Systems Research*, 39:1–3.

Lieto, A., Radicioni, D. P., Frixione, M., and Rho, V. (2015b). Extending ontological categorization through a dual process conceptual architecture. In Fred, A., Dietz, J. L. G., Aveiro, D., Liu, K., and Filipe, J., editors, *Knowledge Discovery, Knowledge Engineering and Knowledge Management*, volume 553 of *Communications in Computer and Information Science*, pages 313–328. Springer International Publishing.

Lieto, A., Radicioni, D. P., and Rho, V. (2015c). A common-sense conceptual categorization system integrating heterogeneous proxytypes and the dual process of reasoning. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 875–881. AAAI Press.

Liu, H. and Singh, P. (2004). Conceptnet — a practical commonsense reasoning tool-kit. *BT technology journal*, 22(4):211–226.

Machery, E. (2009). *Doing without concepts*. OUP.

Malt, B. C. (1989). An on-line investigation of prototype and exemplar strategies in classification. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 15(4):539.

Manning, C. D., Surdeanu, M., Bauer, J., Finkel, J., Bethard, S. J., and McClosky, D. (2014). The Stanford CoreNLP natural language processing toolkit. In *Proceedings of 52nd Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 55–60.

Medin, D. L. and Schaffer, M. M. (1978). Context theory of classification learning. *Psychological review*, 85(3):207.

Miller, G. A. (1995). Wordnet: a lexical database for english. *Communications of the ACM*, 38(11):39–41.

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

Navigli, R. and Ponzetto, S. P. (2010). Babelnet: Building a very large multilingual semantic network. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 216–225. Association for Computational Linguistics.

Navigli, R. and Ponzetto, S. P. (2012). Babelnet: The automatic construction, evaluation and application of a wide-coverage multilingual semantic network. *Artificial Intelligence*, 193:217–250.

Oltramari, A. and Lebiere, C. (2012). Pursuing artificial general intelligence by leveraging the knowledge capabilities of act-r. In *Artificial General Intelligence*, pages 199–208. Springer.

Prinz, J. J. (2002). *Furnishing the mind: Concepts and their perceptual basis.* MIT press.

Pylyshyn, Z. (1989). The role of location indexes in spatial perception: A sketch of the finst spatial-index model. *Cognition*, 32(1):65–97.

Radicioni, D. P., Garbarini, F., Calzavarini, F., Biggio, M., Lieto, A., Sacco, K., and Marconi, D. (2015). On Mental Imagery in Lexical Processing: Computational Modeling of the Visual Load Associated to Concepts. In Airenti, G. and Cruciani, M., editors, *Proceedings of the EuroAsianPacific Joint Conference on Cognitive Science (EAP-COGSCI 2015)*, pages 181–186.

Rosch, E. (1975). Cognitive representations of semantic categories. *J. Exp. Psychol. Gen.*, 104(3):192–233.

Salvucci, D. D. (2014). Endowing a cognitive architecture with world knowledge. In *Procs. of the 36th Annual Meeting of the Cognitive Science Society*, pages 1353–1358.

Shvaiko, P. and Euzenat, J. (2013). Ontology matching: state of the art and future challenges. *Knowledge and Data Engineering, IEEE Transactions on*, 25(1):158–176.

Smith, J. D. and Minda, J. P. (1998). Prototypes in the mist: The early epochs of category learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 24(6):1411.

Squire, L. R. and Knowlton, B. J. (1995). Learning about categories in the absence of memory. *Proceedings of the National Academy of Sciences*, 92(26):12470–12474.

Sun, R. (2006). The clarion cognitive architecture: Extending cognitive modeling to social simulation. *Cognition and multi-agent interaction*, pages 79–99.

Sun, R. (2007). The importance of cognitive architectures: An analysis based on clarion. *Journal of Experimental & Theoretical Artificial Intelligence*, 19(2):159–193.

Thagard, P. (2012). *The cognitive science of science: Explanation, discovery, and conceptual change.* Mit Press.

Thomson, R., Lebiere, C., Anderson, J. R., and Staszewski, J. (2014). A general instance-based learning framework for studying intuitive decision-making in a cognitive architecture. *Journal of Applied Research in Memory and Cognition.*

Zenker, F. and Gärdenfors, P. (2015). *Applications of Conceptual Spaces: The Case for Geometric Knowledge Representation.* Cham: Springer International Publishing.