# A Cognition-inspired Knowledge Representation Approach for Knowledge-based Interpretation Systems

Joel Luis Carbonera and Mara Abel
*Institute of Informatics, Federal University of Rio Grande do Sul, Porto Alegre, Brazil*

Keywords: Knowledge Representation, Reasoning, Ontology, Petroleum Geology.

Abstract: We propose a hybrid approach for knowledge representation that combines classic representations (such as rules and ontologies) with cognitively plausible representations, such as prototypes and exemplars. The resulting framework can be used for developing knowledge-based systems that combine knowledge-driven and data-driven techniques. We also present how this approach can be used for developing an application for interpretation of depositional processes in Petroleum Geology.

## 1 INTRODUCTION

Nowadays, knowledge-based systems can be viewed as a mature technology, which have been applied for supporting the decision making in a wide range of scenarios. In general, knowledge-based systems are characterized by a top-down approach, where the relevant knowledge is explicitly represented in a computer-processable way and the problem-solving process performed by the system is supported by this knowledge. Recently, *ontologies* have been applied for representing the domain knowledge in knowledge-based systems. Since an ontology *specifies in a formal and explicit way the shared conceptualization in a given domain* (Studer et al., 1998), its adoption allows the *knowledge reusing* and promotes the *semantic interoperability* among systems.

Knowledge-based systems can be applied when there is no data available for applying bottom-up approaches, such as those based on machine learning techniques. Besides that, in general, the results achieved by knowledge-based systems are reliable, since they are obtained by applying knowledge in which the practitioners of the domain rely on. However, for developing these systems it is necessary to elicit the relevant knowledge from reliable knowledge sources (domain experts, domain literature, etc). This is one of the weak points of adopting this approach, since the knowledge acquisition is a notoriously difficult task, involving costly and error-prone processes that, in general, depends on the availability of domain experts. Besides that, sometimes, the elicited knowledge is not enough to cover all the situations with which the application should deal.

In this position paper, we claim that these weaknesses of the knowledge-based systems can be mitigated by providing to the system the capability of extracting useful knowledge from the available data; that is, from previously solved instances available to the system. The resulting approach can be viewed as a combination of knowledge-driven (top-down) and data-driven (bottom-up) approaches. Hybrid approaches like this were already proposed in the literature (Li and Love, 1999). However, in this paper, we go a step further, by proposing a hybrid *cognition-inspired* knowledge representation approach for supporting this kind of system.

Within the cognitive sciences there is an ongoing debate concerning how the knowledge is represented in the human mind. According to (Murphy, 2002), there are three main theories in this debate. 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. In 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. These exemplars are entities that were previously experienced by the agent. In


Carbonera J. and Abel M..
A Cognition-inspired Knowledge Representation Approach for Knowledge-based Interpretation Systems.
DOI: 10.5220/0005467106440649
In *Proceedings of the 17th International Conference on Enterprise Information Systems* (ICEIS-2015), pages 644-649
ISBN: 978-989-758-096-3

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, such as (Fiorini et al., 2014), which apply these alternative theories in computer applications.

Our main proposal in this position paper is to combine *classic representations* (such as *ontologies* and *rules*), *prototypes* and *exemplars* in a *knowledge representation* framework for supporting knowledge-based systems. The resulting framework combines the strengths of the *knowledge-driven* (*top-down*) and *data-driven* (*bottom-up*) approaches, overcoming the weaknesses of both. In our framework, part of the knowledge is represented classically, using *ontologies* and *rules*, and part of the knowledge is represented as *prototypes* and *exemplars*, which can be extracted from the data that is processed by the system. Thus, our approach can provide the reliance of the expert knowledge when it can be applied, however, it also is able to provide solutions for cases that cannot be covered by this knowledge, but that can be estimated from the available data. In this approach, the resulting system can perform problem-solving processes by combining *rule-based reasoning* with *similarity-based reasoning*. The rule-based reasoning can be used first for reducing the search space and, if a suitable solution is not found, a similarity-based reasoning component can be triggered for finding the suitable solution, by comparing the instance that is being analyzed with the available prototypes and exemplars.

In our project, our main focus of interest are knowledge-based systems for data interpretation, in the domain of *Petroleum Geology*. In this paper, we discuss how to use our approach for developing a knowledge-based system for *visual interpretation of depositional processes*. This task is a resource-consuming job, which relies intensively on the visual knowledge of geologists, and that is considered a crucial step in petroleum exploration.

## 2 SEDIMENTARY STRATIGRAPHY

Since we will discuss an application for the Sedimentary Stratigraphy field, in this Section, we will present an overview of the domain. Sedimentary Stratigraphy is a sub-field of Geology that studies the sedimentary terrains in the surface or subsurface of the Earth, in order to determine the geological history of their formation. The main objects of study and description are *Body of Rock*, *Well Core*, *Outcrop*, *Sedimentary Facies*, *Sedimentary Structures* and *Depositional Processes*. A body of rock can be a well core, which is a cylinder of rock extracted from the subsurface by means of drilling; or an outcrop, which is a body of rock exposed in the surface. A sedimentary facies is a region in a body of rock, visually distinguishable from adjacent regions. Each sedimentary facies is a direct product of a depositional process. A sedimentary structure is the external visual aspect of some internal spatial arrangement of the rock grains. Finally, depositional processes are events that involve the complex interaction of natural forces and sediments, and which are responsible for the formation of sedimentary rocks. Figure 1 presents an example of a well core, emphasizing two distinct sedimentary facies.

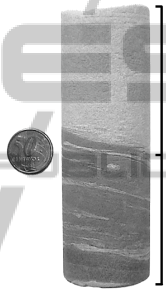

Figure 1: a well core, emphasizing two distinct sedimentary facies (adapted from (Lorenzatti et al., 2009)).

In the task of visual interpretation of depositional processes, the geologists visually inspect sedimentary facies (in well cores or outcrops) and interpret which was the corresponding depositional process that was responsible by the formation of this facies.

## 3 OUR APPROACH

In this section, we will present our approach and how it can be applied in a knowledge-based system for visual interpretation of depositional processes.

In our approach, we assume the availability of *static knowledge* about the domain, represented as a *domain ontology*. In our application, we adopted a domain ontology for Sedimentary Stratigraphy. Figure 2 represents an excerpt of our domain ontology, presenting the core domain entities that were explained in Section 2. Figure 3 represents an excerpt of the taxonomy of depositional processes.

This domain ontology is used in our application for providing a formal basis for describing facts about the domain and for articulating the *inferential knowledge* (rules), which are used for performing the interpretation. Our system takes as inputs descriptions

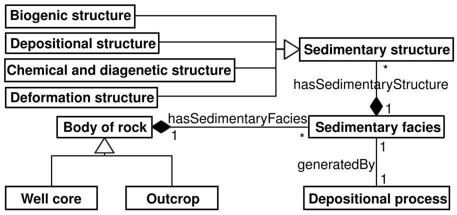


Figure 2: Representation of the core domain entities of the ontology. This excerpt omits the attributes of the entities and the other sub-types of sedimentary structure.

of *bodies of rock*, which are made according the ontology. When the user requires the execution of the interpretation process, the application analyses each sedimentary facies of the body of rock, interpreting the respective depositional process that have generated it.

Regarding the visual interpretation task performed by our system, following (Carbonera et al., 2015; Carbonera et al., 2013; Carbonera et al., 2011), we also consider that it is inferentially characterized by the application of *rules* in the form of

$$observation \Rightarrow interpretation \tag{1}$$

, where *observation* is a set of statements that describe the features that should be perceived in some *observable entities* for supporting the *interpretation*, and *interpretation* is a set of statements about *interpretable entities*. In our application, the observable entities are *sedimentary facies* and *sedimentary structures* and the interpretable entities are *depositional processes*. In this way, the *observation* comprises statements about features that should be perceived in a given instance $f$ of sedimentary facies for inferring the *interpretation*, which specifies what is the specific type of depositional process (see Figure 3) that was responsible by generating $f$. Following, it is presented an example of rule used in our application:

$$\forall f,s,p\ \mathsf{SedimentaryFacies}(f) \land \mathsf{hasSorting}(f, \mathit{PoorlySorted}) \land \mathsf{hasLithology}(f, \mathit{Sandstone}) \land \mathsf{TroughCrossStratification}(s) \land \mathsf{hasSedimentaryStructure}(f,s) \land \mathsf{DepositionalProcess}(p) \land \mathsf{generatedBy}(f,p) \rightarrow \mathsf{3dDuneMigration}(p) \tag{2}$$

This rule represents that a sedimentary facies $f$, which is *poorly sorted*, whose lithology is *sandstone* and that has a sedimentary structure $s$, which is a *Trough-cross Stratification* (a sub-type of sedimentary structure); was generated by a depositional process $p$, which is a *3D dune migration* (a sub-type of depositional process).

It is important to notice that, in our domain of interest, it is necessary to provide to the system the capability of providing interpretations in several levels of generalization/specificity. This is important because sometimes there is not enough information in the sedimentary facies description for supporting the interpretation of the specific type of depositional process that have generated the facies. We provide this capability to our application by including a taxonomy (with several levels of generalization) of depositional processes in the domain ontology and by including the necessary rules for interpreting each different type of depositional process specified by this taxonomy.

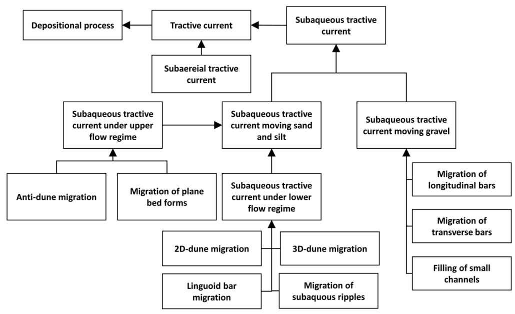


Figure 3: Representation of an excerpt of the taxonomy of depositional processes.

The *ontology*, the *rule base* and the *rule-based reasoning engine* constitute the *knowledge-driven* component of our system. In general, it can be very effective in reducing the set of alternative possible interpretations of a given facies. It can prune entire branches of alternatives in the taxonomy of depositional processes. It can even reach specific interpretations (some leaf of the taxonomy, such as *3D dune migration*), if the description of sedimentary facies is detailed enough. However, this is not always the case. It is common the absence of important information in facies descriptions, because some features are not identifiable by the geologist in the object, or because some identified features are omitted during the description. Besides that, the inferential knowledge can be incomplete, in a way that the set of rules cannot cover all the possible cases that arise in real scenarios. In these cases, the knowledge-driven component can only provide a more general interpretation (such as *Subaqueous tractive current*, for example).

For overcoming these weaknesses, we propose a *data-driven component* that acts in conjunction with the knowledge-driven component. This component relies on a knowledge representation approach that is inspired by the *prototype theory* and *exemplar theory* of knowledge representation in the human mind. Following this idea, we assume that for each possible relevant *interpretable entity ie*, the system has a *prototype* and a *set of exemplars* of the *observations* that can support the interpretation of *ie*. In this component, the visual interpretation is carried out through *similarity-based reasoning*, where the instances that should be interpreted are compared with the prototypes and exemplars for determining the suitable interpretation. In addition, since the system should *extract* the prototypes and exemplars from the available data (that can include instances that were previously solved by the system); it includes a component for extracting the prototypes and exemplars: the *knowledge extractor*.

Before presenting the details of the *data-driven component*, let us consider a formal characterization of some important notions.

- $DS = \{r_1, r_2, ..., r_n\}$ is a data set, which can be abstractly considered as a set of $n$ *records*. Each $r_i$ can be viewed as a 2-tuple $(observation, interpretation)$; where *interpretations* are articulated using concepts in the domain ontology that are generically called *Interpretable entities*, and *observations* are articulated using concepts in the domain ontology that are generically called *observable entities*.
- $O_{DS} = \{o_1, o_2, ..., o_n\}$ is the set of the *observations* of the records in *DS*, in a way that $o_i \in O_{DS}$ is the observation of the record $r_i \in DS$. There ir one *observation* for each *record*.
- $ro\colon DS \to O_{DS}$ is a function that maps a given record $r_i \in DS$ to its *observation* $o_i \in O_{DS}$.
- $IE = \{ie_1, ie_2, ...ie_m\}$ is a set of $m$ *interpretable entities*, where each $ie_i$ is a concept provided by the domain ontology. The elements of $IE$ are organized in a taxonomy, which is a *partially ordered set* $(IE, \leq)$, where $\leq$ is the subsumption relation held between elements of $IE$.
- $P = \{p_1, p_2, ..., p_m\}$ is a set of $m$ prototypes of observations, where each $p_i$ represents the typical observation whose interpretation is specified by the interpretable entity $ie_i$.
- $E = \{es_1, es_2, ..., es_m\}$ is a set whose elements are sets of exemplars, such that, each $es_i = \{e_1, e_2, ..., e_k\}$ is a set of $k$ exemplars of observations whose interpretation is specified by the interpretable entity $ie_i$.
- $prototype\colon IE \to P$ is a function that maps a given *interpretable entity* $ie_i \in IE$ to the prototype $p_i \in P$, which represents the typical observation hose interpretation is specified by $ie_i \in IE$.
- $exemplars\colon IE \to E$ is a function that maps a given *interpretable entity* $ie_i \in IE$ to the set $es_i$ of exemplars, such that $es_i \in E$, which represents the set of $k$ exemplars of observations whose interpretation is specified by the interpretable entity $ie_i$.
- $records\colon IE \to 2^{DS}$ is a function that maps a given *interpretable entity* $ie_i \in IE$ to the set $DS_{ie} \subseteq DS$ of records in *DS*, which represents the set of records in *DS* whose interpretation is specified by the interpretable entity $ie_i$.
- $obs\colon IE \to 2^{O_{DS}}$ is a function that maps a given *interpretable entity* $ie_i \in IE$ to a set $O_{ei} \subseteq O_{DS}$, which represents the set of observations related to the records specified by $records(ei)$, where $\forall o_i \in O_{ei}$, $\exists r_i \in records(ei)$, $o_i = ro(r_i)$.

The user data in the *system database* is stored as a set of instances of the domain ontology. However, for the purposes of the data-driven component, this database can be abstractly viewed as the dataset *DS*, where *observations* (including individual exemplars of observations in the set $E$, and typical observations in the set $P$) can be viewed as *vectors of features* of length $l$, called *observation vectors*. The vectorial representation is preferable for understanding how to perform similarity judgments. Figure 4 presents the process of representing instances of our domain ontology as vectors of features. Notice that, in our application, observations consist basically in descriptions of instances of sedimentary facies (our *observable entities*), and interpretations are basically types of depositional processes (our *interpretable entities*). Thus, in the resulting *observation vectors*, each attribute of Sedimentary Facies is represented as a position of the resulting vector of features. Besides that, the resulting vector has a special position for representing the type of the Sedimentary Structure of the Sedimentary Facies.

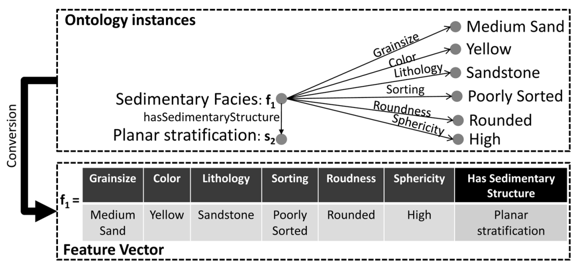


Figure 4: Representation of the process of converting the instance of Sedimentary facies (represented according the domain ontology) in a vector of features.

The extraction of *prototypes* of observations from data is performed by the *knowledge extractor* (in the knowledge-driven component). This component extracts a prototype of the observations whose interpretation is some *interpretable entity ie* ($obs(ie)$). It basically analyses the observations in $obs(ie)$ and determines the typical value of each one of the $l$ attributes of the observations. If the attribute is numeric, the typical value can be the average; if the attribute is nominal, the typical value can be the most frequent.

The extraction of a set of *exemplars* of observations from data is also performed by the *knowledge extractor*, in the knowledge-driven component. Considering a given $ie \in IE$, the exemplars in $exemplars(ie)$ will be used by the system as references for interpreting a new observation $o_{new}$, by performing judgments of the similarity between $o_{new}$ and each considered exemplar in $exemplars(ie)$. In our approach, we assume that it is not desirable to consider all records in $obs(ei)$ as exemplars for representing the observations in $obs(ei)$, since the computational cost of the interpretation process is proportional to the number of exemplars that are selected for representing the observations. For reducing the computational overhead of the interpretation process, in our approach we consider that the number of exemplars related to each interpretable entity $ei_i \in IE$ is defined as a percentage $ep$ (defined by the user) of the total number of observations in $obs(ei_i)$. This raises the problem of how to select which observations in $obs(ei_i)$ will be consider as the exemplars in $exemplars(ei_i)$. We select three main criteria that an observation $o_e \in obs(ei_i)$ should meet for being included in $exemplars(ei_i)$, considering an *interpretable entity ie*: (i) $o_e$ should have a high degree of dissimilarity with the prototype given by $prototype(ie)$; (ii) $o_e$ should have a high degree of similarity with a big number of observations in $obs(ei_i)$; and (iii) $o_e$ should have a high degree of dissimilarity with each exemplar already included in $exemplars(ei_i)$. This set of criteria was developed for ensuring that the set of exemplars in $exemplars(ei_i)$ will cover in a reasonable way the spectrum of variability of the observations in $obs(ei_i)$. That is, our goal is to preserve in $exemplars(ei_i)$ some uncommon observations, which are not well represented by $prototype(ie)$, but that represent the variability of the observations. In our approach, we apply these criteria, by including in $exemplars(ei_i)$ the $h$ first observations from $obs(ei_i)$ that maximize their *exemplariness index*. The exemplariness index is computed using the notion of *density* of a given observation (represented in the form of an observation vector). Regarding some *interpretable entity ie*, the density of some observation $o_i \in obs(ie)$, is computed by the function *density*, such that, $density(o_i) = -\frac{1}{|obs(ie)|}\sum_{j=1}^{|obs(ie)|} d(o_i, o_j)$; where $d$ is some dissimilarity (or distance) function (a function that measures the dissimilarity between two entities). Considering this, the set $exemplars(ei_i)$ of some *interpretable entity ie*, with $h$ exemplars, can be computed by the Algorithm 1.

Notice that the Algorithm 1 basically selects from $obs(ie)$, the observations that maximize the *exemplariness index*, which is the sum of: (i) distance (or dissimilarity) of the observation from the $prototype(ie)$; (ii) the *density* of the observation, considering the set $obs(ie)$; and the distance (or dissimilarity) of the observation from its nearest exemplar, already included in $exemplars_{ie}$.

**Algorithm 1:** extractExemplars.

**Input**: An *interpretable entity ie* and a number $h$ of exemplars
**Output**: A set $exemplars_{ie}$ of $h$ observations representing the exemplars of the observations whose interpretation is specified by *ie*.

**begin**
  $exemplars_{ie} \leftarrow \varnothing$;
  **for** $j \leftarrow 1$ **to** $h$ **do**
    $eIndex_{max} \leftarrow -\infty$;
    $obs_{max} \leftarrow null$;
    **foreach** $o_i \in obs(ie)$ **do**
      $density \leftarrow density(o_i)$;
      $dp \leftarrow d(o_i, prototype(ie))$;
      $med \leftarrow 0$;
      **if** *$exemplars_{ie}$ is not empty* **then**
        Compute the distance between $o_i$ and each exemplar $e$ already included in $exemplars_{ie}$ and assign to $med$ the distance of the nearest exemplar from $o_i$;
      `/* eIndex is the exemplariness index */`
      $eIndex = dp + density + med$;
      **if** $eIndex > eIndex_{max}$ **then**
        $eIndex_{max} \leftarrow eIndex$;
        $obs_{max} \leftarrow o_i$;
    $exemplars_{ie} \leftarrow exemplars_{ie} \cup obs_{max}$;
  **return** $exemplars_{ie}$;

Once the system has the hybrid knowledge representations (prototypes and exemplars), they can be applied by the *interpretation engine* for performing interpretations. This component takes as input a new observation and provide its corresponding interpretation. In our application, the observation is an ontology-based description of a sedimentary facies, and the interpretation is a specific type of depositional process that was responsible by the formation of the facies. Firstly, the interpretation engine applies rule-based reasoning (using classical knowledge representations in the form of a domain ontology and a rule base) for providing a first set of hypothesis. Notice that the rule-based reasoning can infer more than one interpretation for the same observation, depending on the rules in the rule base. If the rule-based reasoning provides interpretations that are not specific (the leaves of the taxonomy of interpretable entities in the domain ontology), the similarity-based reasoning can be used for determining more specific interpretations. The *visual interpretation engine* implements the Algorithm 2.

Notice that the Algorithm 2 uses the notion of *applicability*, which, intuitively measures the degree in that a given *interpretable entity ie* can be applied as an interpretation for a given observation *obs*. The applicability is computed by the Algorithm 3, using the

**Algorithm 2:** visualInterpretation.

**Input**: An observation *obs*.
**Output**: A set $int_{set}$ of *interpretable entities* representing the interpretations of *obs*.
**begin**
  $int_{set} \leftarrow \varnothing$;
  Perform the rule-based reasoning for interpreting *obs* (applying the rules in the rule base), and include the *interpretable entities* of the resulting interpretations in $int_{set}$;
  **if** *the interpretable entities in* $int_{set}$ *are not specific* **then**
    $hyp_{set} \leftarrow \varnothing$;
    **foreach** $ie \in int_{set}$ **do**
      Find the leaves in the taxonomy of interpretable entities, whose root is *ie*, and include them in $hyp_{set}$;
    $int_{set} \leftarrow \varnothing$;
    $MAX \leftarrow -\infty$;
    **foreach** $ie \in hyp_{set}$ **do**
      $app \leftarrow applicability(ie, obs)$;
      **if** $app > MAX$ **then**
        $MAX \leftarrow app$;
        $int_{set} \leftarrow \varnothing$;
        $int_{set} \leftarrow int_{set} \cup ie$;
      **else if** $app = MAX$ **then**
        $int_{set} \leftarrow int_{set} \cup ie$;
  **return** $int_{set}$;

**Algorithm 3:** applicability.

**Input**: An *interpretable entity ie* and an observation *obs*.
**Output**: A value $r \in \mathbb{R}$, which is the degree in that *ie* can be applied as an interpretation for *obs*.
**begin**
  $app \leftarrow 0$;
  $pSimilarity \leftarrow sim(obs, prototype(ie))$;
  $eSimilarity \leftarrow 0$;
  Calculate the similarity $sim(obs, ex_i)$ between *obs* and each $ex_i \in exemplars(ie)$, and assign to *eSimilarity* the similarity value of the most similar $ex_i$;
  $app \leftarrow pSimilarity + eSimilarity$;
  **return** *app*;

prototypes and exemplars of the observations in $obs(ie)$.

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 observations. Thus, *sim* has values that are inversely proportional to the values obtained by the function $d$. Here, we assume that $sim(o_i, o_j) = exp(-d(o_i, o_j))$.

## 4 CONCLUSIONS AND FUTURE WORKS

In this paper, we propose a cognition-inspired *knowledge representation* framework for supporting knowledge-based systems. The resulting framework can combine the strengths of the *knowledge-driven* (*top-down*) approaches, overcoming the weaknesses of both. In future works, we intend to provide a detailed account of the whole systems developed using this approach, and to present a validation of the approach in real cases. In addition, we plan to investigate how the system can update its representations (prototypes and exemplars), when new data is included in its database.

## ACKNOWLEDGEMENTS

We gratefully thank to PRH PB-217 program (supported by Petrobras), for the financial support to this work, and to Sandro Fiorini for comments and ideas.



## REFERENCES

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*, 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.

Li, H. and Love, P. E. (1999). Combining rule-based expert systems and artificial neural networks for mark-up estimation. *Construction Management & Economics*, 17(2):169–176.

Lorenzatti, A., Abel, M., Nunes, B. R., and Scherer, C. M. S. (2009). Ontology for imagistic domains: Combining textual and pictorial primitives. In *ER Workshops*.

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

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