Computer Science and Information Systems 12(3):979–1008 DOI: 10.2298/CSIS141103036S

Towards Analogy-Based Reasoning in Semantic Network

Milos Stojkovic, Milan Trifunovic, Dragan Misic, and Miodrag Manic

Faculty of Mechanical Engineering in Nis, Aleksandra Medvedeva 14,
18000 Nis, Serbia
{milos.stojkovich, milantrifunovic, misicdr, miodrag.manich}@gmail.com

Abstract. In this paper an approach in realization of analogy-based reasoning in semantic networks is presented. New semantic model, called Active Semantic Model (ASM), was used. Core of the process is performed by ASM's association (semantic relation) plexus upgrading procedure based on recognition and determining similarity between association plexuses. Determining similarity between association plexuses is performed by recognition of topological analogy between association plexuses. ASM responds to unpredicted input by upgrading new association plexus modeled on remainder of the context whose subset is recognized as topologically analogous association plexus.

Keywords: Artificial Intelligence, Analogy-Based Reasoning, Active Semantic Model, Cognitive Data Processing, Semantic Features

1. Introduction

Example: Angelina, a preschool girl, enters the office of school's psychologist who assesses cognitive abilities of newcoming first class pupils. After welcoming Angelina, psychologist asks the girl to seat and look the paper laying on the table. Psychologist begins to explain the task to Angelina: Here you have two pictures in the first row, and your task is to choose the missing picture in the second row (Fig. 1).

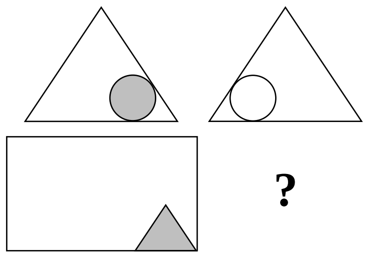

Fig. 1. Angelina's task

Angelina's mother, IT expert, also present in the office, intrigued by the task, tries to come up with computational procedure, method or approach that already exists, and which can be employed for solving Angelina's task. She also contemplates about efforts needed to "force" such system to bring proper conclusion without input pre-planning. First it's necessary to describe the pictures in a way so that system would be able to provide reasonable response. To ensure such a response, system should be prepared to interpret pictures and their spatial relations semantically, as well the task itself, i.e. to identify what it is expected to return. Finally, the system should be able to generate a kind of unique (previously never defined) and valuable response based on a very small portion of domain knowledge already modeled within it. Additionally, that system should not be custom-made for Raven's progressive matrices solving. While mother was thinking about the system she would need, Angelina chose the correct picture. "That's my girl!", she thought.

Semantic interpretation of data represents one of the biggest challenges faced by modern information technologies. In fact, this problem is closely related to the ability of computer applications to attach certain meaning to data which is being processed. The motive for solving this problem lies in ever increasing need to enable software applications to provide meaningful answers when it is not possible to predict the input, and consequently the code by which a meaningful response is programmed.

2. Related Work

2.1. Issues with Reasoning and Learning in DL Based Ontologies

The most exploited current model of knowledge representation in form of semantic network is ontology. Actually, ontology in IT and AI contexts usually refers to a kind of vocabulary of terms (concepts) and relations among the terms codified in a description logic (DL) formalism that should enable a computer application to interpret meaning of the terms from the vocabulary [1]. Aiming the ultimate objective to enable computer to draw meaningful response on unpredicted input, which is essentially related to its ability to interpret semantics of the input data, IT experts in last two decades were focusing mainly on development of DL-based ontologies that utilize first-principles reasoning for semantic interpretation. First-principles reasoning ensures logical deduction applying logical inference rules on axioms (mainly employing First-Order Logic – FOL) that are related to closed domain of richly axiomatized discourse [2]. In particular, almost all DL oriented reasoners are based on tableau-based decision algorithms or resolution-based decision procedures [3], [4], [5]. Besides the core set of logical rules, these semantic models usually also provide a production rules dialect [6] which allows creator of vocabulary to define domain specific rules of inference to ensure correct entailments for the case of not logically derivable semantic interpretation.

By insisting on richly axiomatized ontologies current semantic models are struggling to stay in domain of strongly structured knowledge which allows application of first-principles reasoning approach. The remarkable effort has been carried out in last two decades to generate a large set of different ontologies that demonstrate undoubted power of DL and FOL. However, the most common real situations where computer-autonomous semantic interpretation is highly required are related to data sets whose semantics is not consistently and precisely logically modeled. Besides, constructing richly axiomatized ontologies for real world knowledge intensive applications is a time consuming and difficult task [7], [8], which often results in incompleteness of ontology.

Due to lack of explicit specification of vocabulary terms intended meaning or insufficiently structured data semantics, it is hard for current semantic reasoners to provide relevant entailments [9], [10]. The same cause generates challenges related to ontology learning, assertions populating, ontology enrichment and evolution, as well as ontology matching and mapping [11], [12]. To extract ontological elements from an input and learn about new ontology elements or to enrich existing ontology from that input autonomously (by computer application), it is necessary for the set of input elements to be already formalized in a way that automated inference will yield the expected results [11], [13]. To match and align two different ontologies, even semantically very close, but created by different domain experts, first and inevitable step is identifying common concepts [14], [15], [16], [17], [18]. The key question is how to measure "commonality", i.e. similarity between differently conceptualized and hence differently described concepts; even more, how to measure similarity between concepts of semantically very distant vocabularies (completely disjointed) [19]. Existing research in the domain of ontology alignment has developed several semi-automatic approaches for measuring concept similarity [14], [17], like lexical similarity between concepts, structural similarity of concepts in ontologies (similarity of ancestors/descendant, depth and length of path in the tree) and similarity of concept instances/annotations. Nevertheless, in most of the real word cases, ontology alignment process mainly relies on human interventions (i.e. similarity assessments) [14], [19]. After recognition of similarity between concepts there are two more activities to perform before engaging semantic reasoners to produce entailments autonomously. First, there is a need to formalize and/or harmonize semantics of concepts with DL to ensure alignment and secondly, one should identify mapping rules, which is very complex and nonautomated task [15]. Aforementioned activities (needed to learn, enrich and/or align onotologies codified in DL) can only be performed if domain experts collaborate with skilled ontology engineers familiar with the theory and practice of knowledge representation [11].

It seems that strongly structured knowledge approach within DL-founded ontologies is certainly powerful tool for deduction in the "local field" of semantics, but at the same time it reduces capability to infer *autonomously* and *flexibly*. Actually, semantic reasoners designed to work with DL-founded ontologies showed themselves weak in making relevant entailments beyond the predefined and embedded logical formalism of deduction. The similar is also true for reasoning *flexibility* – ability to make a relevant, but quite different entailments about the same concept for semantically distant or different contexts (vocabularies) with a single set of logical inference rules and axioms on disposal. Finally, having *analytical* ability to autonomously dissolve a portion of knowledge about one concept or group of concepts from one context and apply it to quite different (semantically distant) concept or a group of concepts that are inherent to equally different context is something which appears not as strong side of richly axiomatized ontologies which rely on first-principles reasoning approach.

Often emphasized, essential weakness of ontologies, as well as all semantic models, is reflected in the fact that an increase in level of detail of the meaning to be described, significantly increases complexity and time required to create the ontology [20]. Therefore, much more ontologies that do not require a high level of semantic detail were developed.

2.2. Why to develop new semantic model?

In order to overcome obvious weaknesses of DL ontologies it becomes more attractive to explore whether the cognitive process of analogy making or analogical reasoning can serve as the basis for ontology learning and alignment processes [21], [22], [23]. In addition, growing exploration within the field of DL ontologies similarity indicates that focus is moving to analogy-based identification of semantic correspondence between ontologies [14], [17], [24]. As Forbus et al. state in [2] "analogy based reasoning is a method of last resort" for DL ontologies issues, "something to use only when other forms of inference have failed".

Familiarizing with research [25] on how humans derive knowledge from analogies between symbols, concepts, situations and events inspired us to try to create our own semantic model with better capacity to perform analogy based reasoning. We found that links structure in a semantic network needed to be changed slightly in order to determine and categorize similarities between concepts, situations and events more efficiently.

However, before we get down to describe in-house developed semantic model featured by original structure of links, we need to give a short survey on Analogies and Analogy-Based Reasoning.

2.3. Analogies

Achieving autonomy, flexibility and analyticity of semantic interpretation is considered a major current goal of all artificial intelligence methods and models, including ontologies [1]. In pursuit for solution, interest for approach where semantic interpretation of data is based on analogies reappears [26]. Research in cognitive psychology often indicate that use of analogies represents the core of cognitive process, and may be considered as primary process of cognition and communication [27]. Traditional logic distinguishes three forms of reasoning: deductive, inductive, and analogy-based reasoning (ABR). Examples of heuristics most commonly used for solving problems are determination of partial goals and reliance on analogies [28]. In the latter case, known procedure, which proved to be successful in solving previous related (similar) problems is used to solve new problem. Precondition for success of this strategy is recognition of analogy between two problems and recalling the solution applied earlier. One of the reasons why sometimes it is difficult to recognize analogy between two problems is the fact that their elements are different, although relations are the same [25].

ABR is often used to characterize methods that solve new problems based on past cases from a different domain, while typical case-based reasoning (CBR) methods focus on indexing and matching strategies for single-domain cases [29]. In general, analogy involves several subtasks including retrieving from memory the source case most similar to the target problem, mapping (or aligning) the elements of target and the source, transferring knowledge from the source to the target, evaluating what was transferred in the context of the target, and storing the target in memory [30].

Three major types of case representation are feature vector cases, structured cases, and textual cases [31]. Feature vector approach represents a case as a vector of attribute-value pairs, while structured approach as clusters of relations between the kinds of elementary objects that comprise it [32].

Case representation and the way similarity is assessed during retrieval are strongly related to each other. In some applications of ABR, similarity of stored cases is assessed in terms of their surface features, which are parts of their description typically represented using attribute-value pairs. Various methods exist: k-nearest neighbor (k-NN) based on Euclidean distance, mix neural networks [33], fuzzy logic [34], and genetic algorithms [35]. Structured cases often require knowledge intensive matching algorithms to assess structural similarity. Experiments confirmed that both surface and structural similarity assessment are necessary for sound retrieval [36], [37]. Structural features, however, have a greater impact than surface features on a problem solver's ability to use an analogue once its relevance has been pointed out [37]. Retrieval based solely on similarity has limitations. That's why similarity is increasingly being combined with other criteria to guide the retrieval process, such as adaptability of the retrieved case [38], [39].

ABR is considered the most flexible and analytical approach within the corpus of CBR systems [29], [40]. MARVIN [40] is interesting and very expressive example. It is a system for general knowledge representation in form of analogies, and graphical/tabular visualization and searching for analogies, i.e. analogy-based reasoning. What's special about this example is that the system uses XML syntax for representing and visualizing analogies. Searching for analogies in this system is based on the so-called superficial similarity of analogies (full or partial match of node's names in the structure). Additional structural search mechanism traces synonyms, hypernyms relations and other k-level generalizations in order to extend the set of potential analogies which could be used for reasoning. This approach of structural mapping, i.e. searching graphs of semantic or functional model is dominant in many other earlier or later realizations. Unfortunately, it doesn't bring, nor demonstrate, full potential and advantage of ABR approach, but often discredits it as too limited and arbitrary.

Research on analogy reasoning is concerned with mechanisms for identification and utilization of cross-domain analogies [41], [42]. The major focus has been on finding a way to transfer (or map) the solution of an identified analogue problem to the present problem. Analogical mapping has been studied in many theories of analogy, such as Proteus [30], AMBR [43], [44], LISA [45], SME [46], and ACME [47].

In order to recognize analogy between two problems, it is necessary to have insight into common elements of the solution which can be applied to new problem. This insight is actually contained in similarity and/or sameness of relations between these elements [25]. Realization of this claim is the main objective of Active Semantic Model (ASM) – to embed knowledge in semantic relations and their plexuses (not in the nodes of the semantic network), and also to try to recognize analogies by determining the similarity of semantic relations and their plexuses in order to interpret the meaning and draw conclusions. Structure of ASM [48], and approaches to semantic categorization of data [49], [50], and recognizing topological analogy in ASM semantic network were presented so far. This paper proposes to extend ASM for ABR.

ABR brought by ASM allows us to overcome the need for: 1) pre-planned conditions which have to be fulfilled in order to trigger predefined response, i.e. inference, and 2) standardization of nomenclature. It provides system with ability to make creative reactions, and to ensure relevant answer with minimal investment in preparation.

3. Active Semantic Model

ASM is a sort of semantic network model, developed in-house, aiming to capture and interpret semantics of design features related to manufacturability issues [48]. The most frequent representation of semantic network is graph notation consisting of nodes and links (or edges, arcs). Nodes usually represent concepts, objects or situations in a particular domain, while links usually represent the semantic relations between these concepts, objects or situations. More complex link structure is what distinguishes ASM from other semantic network models. Actually, decision to focus on link structure came from the thesis stating that the knowledge people have about things (visual representations, objects, situations, etc.) is contained in associations between concepts that abstractly represent those things [51]. Beside functional relation between concepts, ASM's link express also its affiliation, accuracy, and significance for specific context, and for particular instructor (user). In this way semantic link provides chunk of knowledge which is subjective and context related. Furthermore, each semantic link bears information about direction and character of associating between concepts (that is, about the way in which semantic interpretation should be made). This feature of ASM's link – to point out the pathway of inference – induced us to use the term *association* for the link instead of relation. Here, we will explain ASM in brief.

3.1. Structure

The structure of ASM is built just from associations (links of network). Each association is characterized by eleven parameters [49] among which two of them are *names of concepts* (cpt_i, cpt_j). Considering that these parameters can belong to more than one association, they represent junctions of associations, i.e. virtual nodes of network. The explicit knowledge related to concepts and their instances is also linked to these virtual nodes of ASM's network via associations. On the other side of the association which connects concept of ASM network with a chunk of explicit knowledge about it is a pointer to a certain local computer or internet depository where the knowledge is stored. These pointers are named *concept bodies*, because they point out to some kind of knowledge embodiments of each instance of the concept. For example, concept Blue-Color can be embodied by one or plenty of specific values of color codes and procedure to generate this color on the computer screen in accordance to its code. Thus, one concept can have several concept bodies, i.e. its real represents.

The parameter *name of the concept* in an association is used to designate human abstraction of different level of complexity, from "Something" concept to very complex spatial and time and/or abstract contexts. The "Something" concepts can be tangible (e.g. pencil) and intangible (e.g. geometrical shape, circle) objects, attributes (e.g. blue), activities (e.g. cause, use), or abstract ideas (e.g. number, below). The spatial and time contexts can be different situations and events. The abstract context can be e.g. differential equation. There can be only one concept with a given name, but there can be many associations belonging to different contexts associating it with other concepts.

Beside two different *names of concept*, an association in ASM is defined by additional three sets of parameters:

Topological parameters:

roles (r_i, r_j) of concepts within association denote what functional role each of these concepts have in their mutual association (e.g. Photo associates to Photo-Album as a part to assembly).

type (t) denotes type of associating (e.g. affiliation in the aforementioned association between Photo and Photo-Album). Actually, roles and type of associating make a determinate triplet. For semantically unsymmetrical types of associations (e.g. affiliation) roles indicate the default orientation of association.

direction (d) of associating (←, ↔, →) denotes whether both concepts linked by association associate to each other or just one of them associates to another and not vice versa (e.g. in the association between Triangle and Geometric-Shape, Triangle almost always associates to Geometric-Shape, but Geometric-Shape doesn't associate necessarily to Triangle). It should be noted that direction is not a parameter which indicates the direction of deduction process (graph routing).

character (c) of associating (+, -) denotes how both concepts in an association associate to each other (positive character denotes that concepts associate to each other affirmatively, like Ball and Oval – when we think about the Ball we think that it is Oval, while negative character denotes that concepts associate to each other, but negatively, like Ball and Cubical – Ball is something which is not Cubical).

Weight parameters:

accuracy (h) of an association for the given context (0; 0.25; 0.5; 0.75; 1) denotes how accurate are the values of all other parameters. 0 accuracy denotes that association (values of the other parameters) is untrue and 1 denotes that instructor (user) is convinced in absolute correctness of other parameters' values of association (of course for the given context). Untrue associations are important because these associations indicate misapprehensions which are, sometime, very important for inference process. These associations indicate what segments of network should be ignore by inference engine.

significance (s) of an association for the given context (0; 0.25; 0.5; 0.75; 1) denotes how significant this association is for semantic interpretation of related concepts in a given context. For example, in a context related to the Raven's problem shown in Fig. 1 the associations which can exist and are used to describe that Circle has no Corners are not as significant as the associations which are used to describe Circle spatial location with regard to other geometric shapes. This parameter of association can help ASM inference engine to categorize Circle from two pictures in the first row as similar to Triangle from the picture in the second row, even though the associations that are used to describe geometric features of Triangle (e.g. it has Corners and it is not Oval) are quite opposite to the associations which describe geometric features of Circle.

Affiliation parameters:

context id is a parameter which denotes to which context an association belongs, i.e. in regard to which context the values of association's parameters are valid. This feature allows instructor (user) to describe and ASM's inference engine to interpret the very same concept in semantically quite different way for different contexts.

instructor id or *user id* is a parameter which denotes who has created the association (Fig. 2). Like association affiliation to context, the origin of an association with regard to its creator allows instructors (users) to describe and, later, AMS's inference engine to interpret the semantics of the same concept in a different way. That is how ASM provides possibility to add subjectivity to semantics of a concept. After all, we should never forget that values for aforementioned parameters are assigned based on the instructor's (user's) subjective assessment.

Beside associations (links) as the basic structure of ASM's network, the *association plexuses* have very important role in inference process that ASM carries out. Actually, *association plexus* (PLX) is a term used to denote a set of at least two associations connected by the mutual virtual node(s) (*name(s) of concepts*). In general, association plexus doesn't need to have specific abstract meaning, neither instructor (user) needs to define it (like an association). However, perceiving the semantic network of ASM segregated not just by its basic elements – associations, but also by plexuses facilitates the identification of similarity or analogy of topology between different segments of the semantic network which drives analogy-based reasoning process in the core of inference engine of the ASM. Besides the association plexuses which are not created by instructor and may or do not have any abstract meaning, there is a possibility for instructor (user) to create (usually) more complex plexuses that serve to describe the semantics of complex concepts (like a concept that represents an activity, e.g. Cause), situations (time independent) or events (time dependent). These kind of plexuses is designated as *contexts* (CTX) just to emphasize the difference between plexuses with and without abstract meaning. Each context is defined by its *name* and its *creator* (*instructor* (*user*)) and is used to define affiliation of each association in ASM network, that is, its relevancy. *General context* is defined and built in ASM structure independently of the instructor (user), while other *particular contexts* are created by the user. All the associations from particular contexts are assigned to the general one, but usually with different parameters.

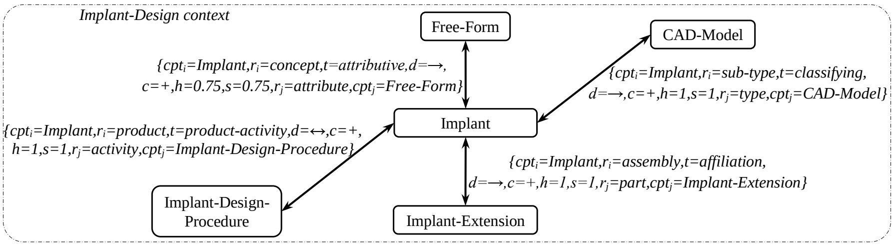

Fig. 2. ASM association structure: Several associations with specified parameters belonging to a context

ASM structure is not domain-specific and can be used for knowledge representation in diverse fields. Knowledge from specific domain should be represented through context(s), while associations as semantic relations between contexts allow knowledge from one context to be applicable to others.

4. Topologically Analogous Association Plexuses

The most common and probably the most significant case of semantic content similarity between different association plexuses or contexts is called *topological analogy* (similarity) (Fig. 3). Topologically analogous association plexuses or contexts have the same type of topology (combination of appropriate values of topological parameters of associations) and the same structure. Associations belonging to two different association plexuses or contexts, with similar values of weight parameters and the same values of topological parameters are called *topologically correspondent associations* (TCA) (associations represented by the same type of line in Fig. 3). Concepts belonging to TCA-s of two different association plexuses or contexts, which have the same role in these TCA-s are called *topologically correspondent concepts* (TCC) (concepts represented by the same background pattern in Fig. 3). Two types of topologically analogous association plexuses or contexts are distinguished: *semantically distant* (association plexuses or contexts do not share concepts, nor are their concepts similar, synonyms or connected over series of up to four associations) and *semantically close* (association plexuses or contexts share one or more concepts, or have concepts which are similar, synonyms or connected over series of up to four associations).

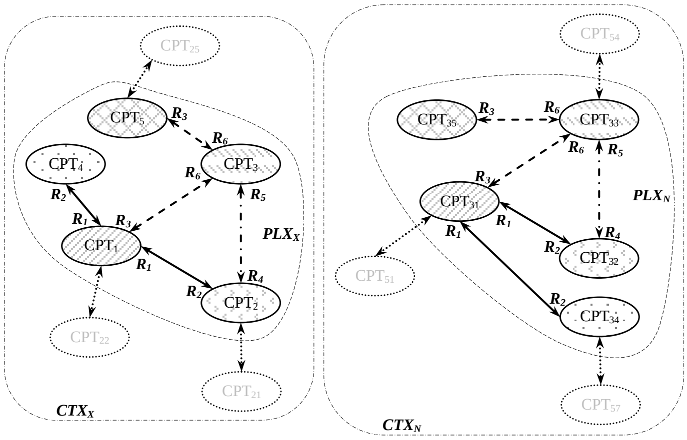

Fig. 3. Association plexuses PLX_X and PLX_N are topologically analogous

4.1. Analogy and Similarities in ASM (Multi-level recognition of similarity)

In the core of its process of data semantics interpretation ASM employs specially developed algorithms for recognizing topological similarity, i.e. analogy between different parts of the semantic network. Analogy of semantics that can be recognized between semantically (more or less) different concepts is essentially related to the similarity between topology of subgraphs built by links (associations) of these concepts in the semantic network. Depending on scope of focus in the process of topology similarity recognizing, ASM uses two main algorithms for recognizing topological similarity:

1. Contiguous, algorithm for determination of similarity between associations of two concepts that are not directly connected, but over one layer of intermediate concepts [50], and

2. Wide, algorithm for determination of similarity between plexuses (subgraphs) of associations (Determining the topological analogy between association plexuses – aimed for “semantically distant” concepts).

Actually, both algorithms are designed to determine degree and class of semantic similarity, i.e. semantic correspondence between two concepts and both of them are based on determination of similarity of topological parameters of associations. Weight parameters are used to refine associations that have to be considered in the similarity determination process by their semantic relevance regarding particular context, instructor/user and his motivation. Contiguous algorithm is simpler and more explicit (clearly defined) in similarity determination, but its application is limited to semantically close concepts. Wide algorithm is aimed for determination of semantic correspondence between two semantically distant concepts and for that it performs determination of similarity between topology of different plexuses (subgraphs) of associations.

To create inference autonomously, ASM employs algorithm for association plexus upgrading which is executed in three “attempts” (see section 5). Actually, the paper is mainly focused on describing this algorithm.

ASM encompasses additional two self-learning algorithms: 1) algorithms for creation of heuristics, and 2) algorithm for knowledge “crystallization” (weakly structured knowledge is crystallized into strongly structured knowledge, i.e. logic formalisms). Both algorithms are aimed to provide ASM with capability to learn, i.e. to formalize knowledge gained from experience with analogy-based reasoning and human interventions. In relation to learning (reasoning in acquiring new knowledge) capabilities of ASM, it should be mentioned that ASM is designed to perform data semantic interpretation in regard to user motivation context as a reference framework. Fulfillment of user actual motivation appears as the most important criteria for learning analogies from experience. However, describing these algorithms is out of the scope of this paper.

In ASM there are also CASE procedures (a sort of simplified set of predicate logic rules) that can be regarded as generic logic formalism that will be employed if the case of semantic graph is recognized to be suitable for triggering predefined logic rule. This set of rules can change (i.e. in case learning process) enabling ASM to learn and improve its reasoning performance over time. If case of “unknown” association plexus (a sort of an ontology portion) is recognized as very similar or same as one from Experience Set, then it is more efficient to apply strict logic formalism which is very likely to be truthful for that kind of association plexus topology.

5. Association Plexus Upgrading Procedure

Every association plexus can be observed as a part of the semantic network connected to other parts of the semantic network by associations involving other concepts. In general, it is very difficult to distinguish where one association plexus “ends”, and where others “begin”. User introduces new association plexus (which represents new or unknown situation) to ASM, usually by creating associations between concepts of which some or all are known to ASM, i.e. were added to ASM semantic network earlier (Fig. 4).

ASM responds to input by recognizing topological analogy between new and known association plexuses (from the narrowed semantic network space) and upgrading new association plexus modeled on remainder of the context (whose subset is recognized as topologically analogous association plexus). The response is being formulated through creating new associations between concepts from new association plexus and known concepts in the network.

Association plexus upgrading procedure is based on similarity between new and known association plexuses. New association plexus concepts will be connected modeled on their TCC-s in similar association plexuses.

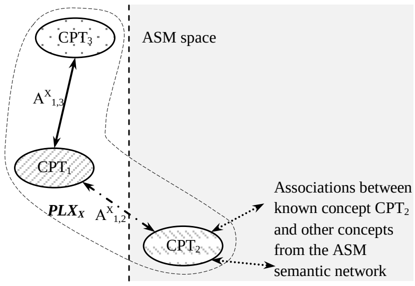

Fig. 4. Introducing new association plexus PLX_X to ASM. Concept CPT_2 is known to ASM

In the case when new association plexus PLX_X is topologically analogous to certain known association plexus PLX_N (the more TCA-s they have, the better), regardless of whether they are semantically close or semantically distant, ASM will use the logic of topologically analogous association plexus upgrading (element $\asymp$ denotes topological correspondence (for associations and concepts) or topological analogy (for contexts and association plexuses); element $\leftrightarrow$ denotes association between concepts):

If

$$\left\{A_{i,j}^{PLX_X}\right\} \asymp \left\{A_{k,l}^{PLX_N}\right\} \wedge \left\{CPT_i^{PLX_X}\right\} \asymp \left\{CPT_k^{PLX_N}\right\} \\ \left(\Rightarrow PLX_X \asymp PLX_N\right) \tag{1}$$

where $A_{i,j}^{PLX_X} = A_{CPT_i \leftrightarrow CPT_j}^{PLX_X}, A_{k,l}^{PLX_N} = A_{CPT_k \leftrightarrow CPT_l}^{PLX_N}, PLX_N \subseteq CTX_N$,

then it is possible that there exists context CTX_X, whose subset is new association plexus PLX_X, which is topologically analogous to known context CTX_N:

$$\exists CTX_X \mid CTX_X \supseteq PLX_X \wedge CTX_X \asymp CTX_N \tag{2}$$

Therefore, new association plexus PLX_X should be upgraded to context CTX_X, modeled on the remainder of the known context CTX_N (Fig. 5).

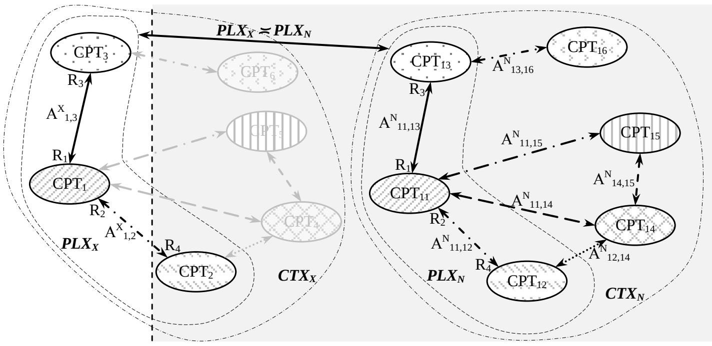

Fig. 5. Topologically analogous association plexus upgrading logic

Logic of topologically analogous association plexus upgrading is carried out through three attempts (sub-procedures). First and second attempt have several iterations.

Each iteration for every attempt is followed by iteration of the process of determining semantic similarity of concepts, which can also result in the creation of association(s) between concepts. This procedure is presented in detail in [50].

5.1. First Attempt

The first attempt is carried out through several iterations. The procedure for each iteration is identical. First attempt ends in situation when ASM is not able to add new association to known association plexus.

The same example will be used independently to illustrate first attempt procedure for semantically distant and semantically close TCC-s.

Semantically Distant TCC-s. ASM first recognizes semantically distant TCC-s of new and known association plexus:

$$\exists CPT_i \mid \{CPT_i\} \in PLX_X \wedge \exists CPT_j \mid \{CPT_j\} \in PLX_N \qquad \textbf{(3)}$$

such that:

1. $CPT_i \asymp CPT_j$

   $CPT_i \neq CPT_j, \nexists A^{CTX_0}_{CPT_i \leftrightarrow CPT_j} \mid$

2. $t\left(A^{CTX_0}_{CPT_i \leftrightarrow CPT_j}\right) = \text{similarity} \vee$

   $t\left(A^{CTX_0}_{CPT_i \leftrightarrow CPT_j}\right) = \text{synonymous}$

3. $\exists A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}} \mid \left\{A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}}\right\} \in CTX_N \wedge$

   $\left\{A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}}\right\} \notin PLX_N, PLX_N \subseteq CTX_N$

If such concepts are found, ASM searches for all associations in the semantic network involving concepts from new association plexus, which are topologically correspondent to associations from known association plexus involving their TCC-s, and adds these associations if TCC-s have the same roles in them (Fig. 6):

$$\begin{gathered}
\exists A^{CTX_M}_{CPT_i \leftrightarrow CPT_k} \mid \left\{ A^{CTX_M}_{CPT_i \leftrightarrow CPT_k} \right\} \in CTX_M \wedge \\
A^{CTX_M}_{CPT_i \leftrightarrow CPT_k} \asymp A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}}, r\left(CPT_i\right) = r\left(CPT_j\right); \\
\left\{ A^{CTX_X}_{CPT_i \leftrightarrow CPT_k} \right\} : t\left(A^{CTX_X}_{i,k}\right) = t\left(A^{CTX_M}_{i,k}\right) \\
c\left(A^{CTX_X}_{i,k}\right) = c\left(A^{CTX_M}_{i,k}\right), d\left(A^{CTX_X}_{i,k}\right) = d\left(A^{CTX_M}_{i,k}\right) \\
h\left(A^{CTX_X}_{i,k}\right) = h\left(A^{CTX_M}_{i,k}\right), s\left(A^{CTX_X}_{i,k}\right) = s\left(A^{CTX_M}_{i,k}\right)
\end{gathered} \qquad \textbf{(4)}$$

where $A^{CTX_X}_{i,k} = A^{CTX_X}_{CPT_i \leftrightarrow CPT_k}, A^{CTX_M}_{i,k} = A^{CTX_M}_{CPT_i \leftrightarrow CPT_k}$.

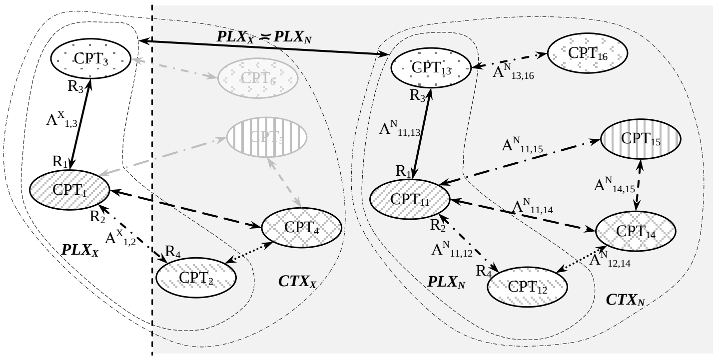

Fig. 6. Association plexus upgrading in first attempt (semantically distant TCC-s). TCC-s (CPT_1, CPT_{11}) and (CPT_2, CPT_{12}) are semantically distant

If several associations, involving concept CPT_1 and topologically correspondent to association between concepts CPT_{11} and CPT_{14}, are found in the semantic network, ASM analyzes if second concept in these associations is involved in the same or similar associations as concept CPT_{14} (e.g. concept CPT_4 is involved in association with concept CPT_{15} or similar concept, which is topologically correspondent to association between concepts CPT_{14} and CPT_{15}). ASM finally adds only associations that meet this condition.

In many situations during the upgrading procedure in first and second attempt, *structure* of the known context CTX_N has to be taken into account. One example is the addition of the association between concepts CPT_2 and CPT_4. In this situation ASM adds association between concept CPT_2 and "existing" concept CPT_4 (it is assumed that association between concepts CPT_1 and CPT_4 was previously added) which is topologically correspondent to association between concepts CPT_{12} and CPT_{14}.

Semantically Close TCC-s. Recognition of semantically distant TCC-s is followed by the recognition of semantically close TCC-s of new and known association plexus. First ASM recognizes TCC-s of new and known association plexus (or contexts of which they are a subset) which are *identical* (denoted by $\equiv$) or are *synonyms* (fifth class of similarity (denoted by $5.\sim$): absolute value of the difference of accuracy and significance for all association pairs connecting these concepts have to be less than 0.25; all association pairs connecting these concepts (through the same connectional concepts) have to have the same type of associating (and the same corresponding concept roles) and the same characters and directions of associating) or *similar* (fourth class of similarity (denoted by $4.\sim$): absolute value of the difference of accuracy and significance for all association pairs connecting these concepts have to be less than 0.5; all association pairs connecting these concepts have to have the same type of associating (and the same corresponding concept roles) and the same characters and directions of associating) in general context (semantically close TCC-s):

$$\exists CPT_i \mid \{CPT_i\} \in PLX_X \wedge \exists CPT_j \mid \{CPT_j\} \in PLX_N \qquad \textbf{(5)}$$

such that:

1. $CPT_i \asymp CPT_j$
2. $CPT_i \equiv CPT_j \vee CPT_i \overset{5.\sim}{\leftrightarrow} CPT_j \vee CPT_i \overset{4.\sim}{\leftrightarrow} CPT_j$ in general context
3. $\exists A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}} \mid \left\{A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}}\right\} \in CTX_N \wedge$
   $\left\{A^{CTX_N}_{CPT_j \leftrightarrow CPT_{j+1}}\right\} \notin PLX_N, PLX_N \subseteq CTX_N$

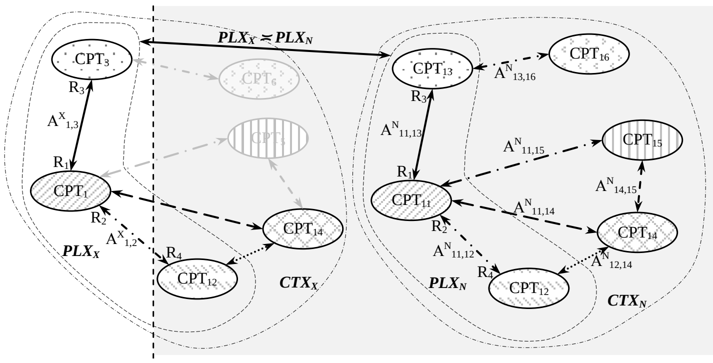

Fig. 7. Association plexus upgrading in first attempt (semantically close TCC-s). TCC-s (concept CPT_{12}) of new and known association plexus are identical

If such concepts are found, ASM adds associations of known association plexus involving found TCC-s, except that the concept from known association plexus will be replaced by its TCC in new association plexus (Fig. 7):

$$\begin{aligned}&\left\{A_{CPT_i \leftrightarrow CPT_{j+1}}^{CTX_X}\right\}: t\left(A_{i,j+1}^{CTX_X}\right)=t\left(A_{j,j+1}^{CTX_N}\right)\\ &c\left(A_{i,j+1}^{CTX_X}\right)=c\left(A_{j,j+1}^{CTX_N}\right), d\left(A_{i,j+1}^{CTX_X}\right)=d\left(A_{j,j+1}^{CTX_N}\right)\\ &h\left(A_{i,j+1}^{CTX_X}\right)=h\left(A_{j,j+1}^{CTX_N}\right), s\left(A_{i,j+1}^{CTX_X}\right)=s\left(A_{j,j+1}^{CTX_N}\right)\end{aligned} \quad \textbf{(6)}$$

where $A_{i,j+1}^{CTX_X} = A_{CPT_i \leftrightarrow CPT_{j+1}}^{CTX_X}, A_{j,j+1}^{CTX_N} = A_{CPT_j \leftrightarrow CPT_{j+1}}^{CTX_N}$.

5.2. Second Attempt

The second attempt is carried out in several iterations. The procedure for each iteration is identical. Second attempt ends in situation when ASM is not able to add new association to known association plexus. Complete first attempt is carried out between second attempt iterations. The second attempt will continue from the situation illustrated in Fig. 6 (first attempt for semantically distant TCC-s).

ASM searches for concepts in the semantic network which are similar to concepts from new association plexus in specific context, and are involved in associations which are topologically correspondent to associations from known association plexus. It is necessary to find the concepts in the semantic network which are similar to concepts from new association plexus in at least third class of similarity (denoted by $\geq 3.\sim$; absolute value of the difference of accuracy and significance for all association pairs connecting these concepts have to be less than 0.5; all association pairs connecting these concepts have to have the same type of associating (and the same corresponding concept roles) and the same characters of associating):

$$\exists CPT_i \mid \left\{CPT_i\right\} \in CTX_X \wedge \exists CPT_j \mid \left\{CPT_j\right\} \in CTX_K \quad \textbf{(7)}$$

such that:

1. $CPT_i \overset{\geq 3.\sim}{\leftrightarrow} CPT_j$
2. $\exists A_{CPT_j \leftrightarrow CPT_{j+1}}^{CTX_K} \mid \left\{A_{CPT_j \leftrightarrow CPT_{j+1}}^{CTX_K}\right\} \in CTX_K \wedge$
   $A_{CPT_j \leftrightarrow CPT_{j+1}}^{CTX_K} \asymp A_{CPT_k \leftrightarrow CPT_{k+1}}^{CTX_N}$

If such concepts are found, ASM adds associations involving them, which are topologically correspondent to associations from known association plexus, except that found concept will be replaced by its similar concept in new association plexus (Fig. 8):

$$\begin{aligned}&\left\{A_{CPT_i \leftrightarrow CPT_{j+1}}^{CTX_X}\right\}: t\left(A_{i,j+1}^{CTX_X}\right)=t\left(A_{j,j+1}^{CTX_K}\right)\\ &c\left(A_{i,j+1}^{CTX_X}\right)=c\left(A_{j,j+1}^{CTX_K}\right), d\left(A_{i,j+1}^{CTX_X}\right)=d\left(A_{j,j+1}^{CTX_K}\right)\\ &h\left(A_{i,j+1}^{CTX_X}\right)=h\left(A_{j,j+1}^{CTX_K}\right), s\left(A_{i,j+1}^{CTX_X}\right)=s\left(A_{j,j+1}^{CTX_K}\right)\end{aligned} \quad \textbf{(8)}$$

where $A_{i,j+1}^{CTX_X} = A_{CPT_i \leftrightarrow CPT_{j+1}}^{CTX_X}, A_{j,j+1}^{CTX_K} = A_{CPT_j \leftrightarrow CPT_{j+1}}^{CTX_K}$.

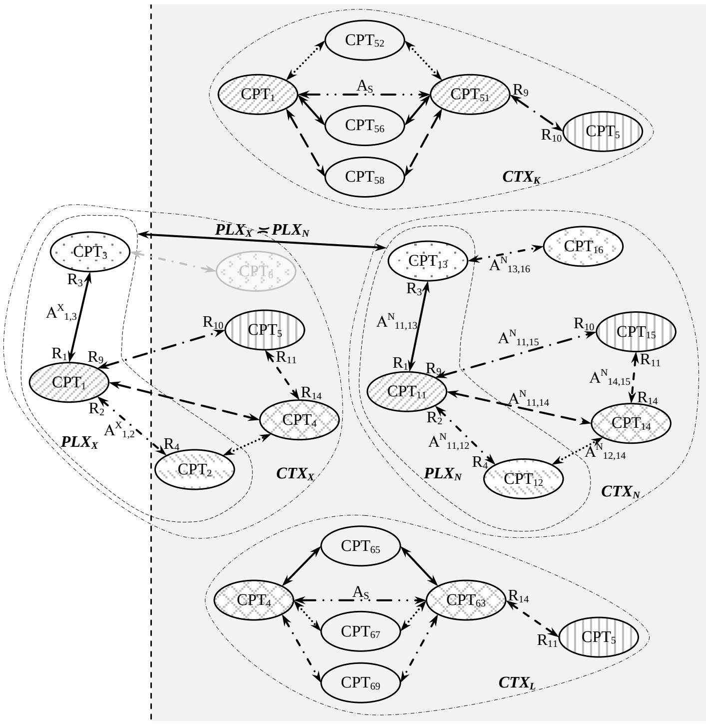

Fig. 8. Association plexus upgrading in second attempt. Concepts CPT_1 and CPT_{51} are similar in context CTX_K, while concepts CPT_4 and CPT_{63} are similar in context CTX_L

5.3. Third Attempt

Third attempt does not have iterations. After the third attempt is carried out, the user, depending on whether he is satisfied with the results, decides whether to complete the upgrading procedure or to carry it out from the beginning (from the first attempt).

The goal of the third attempt is to find *candidate* concepts in the semantic network which should be connected with the remaining concepts (concept CPT_3) from new association plexus. Candidate concepts and their corresponding concepts (concept CPT_{16}) from known association plexus are usually semantically distant. Focus of the third attempt is the similarity between associations involving candidate concepts and associations involving their corresponding concepts from known association plexus.

In the third attempt ASM recognizes concepts (concept CPT_{16}) involved in associations from context whose subset is known association plexus, which do not have TCC-s in the context whose subset is new association plexus. After that ASM identifies all association plexuses with associations involving recognized concepts, as well as their topologically analogous association plexuses. In the last step ASM identifies TCC-s of the recognized concepts which are involved in the same or similar set of TCA-s in recognized topologically analogous association plexuses (Fig. 9).

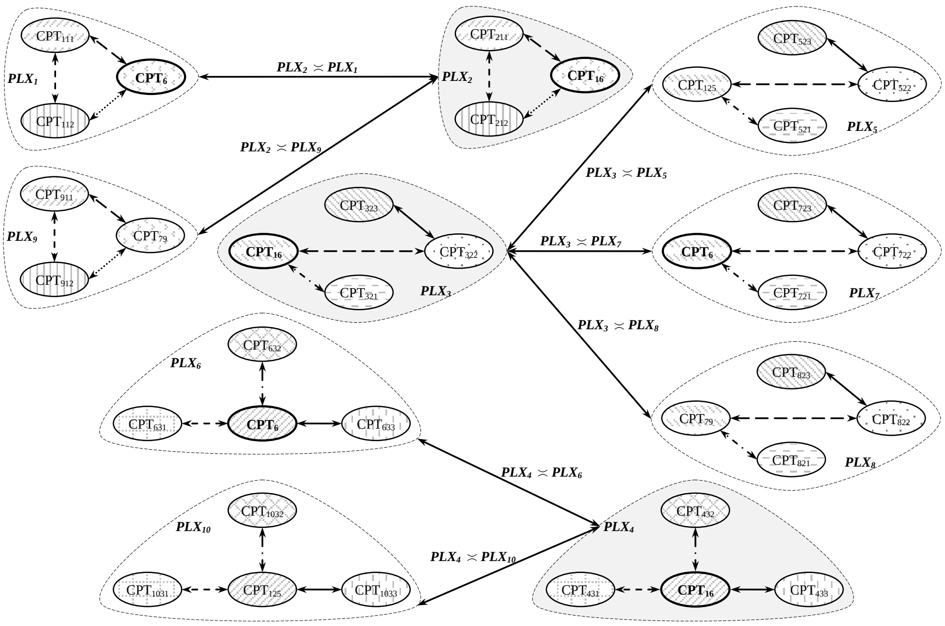

Fig. 9. Recognizing *candidate* concept(s) in the semantic network which should be connected with the concept CPT_3. Concepts CPT_6 and CPT_{16} are TCC-s in most cases of identified topologically analogous association plexuses

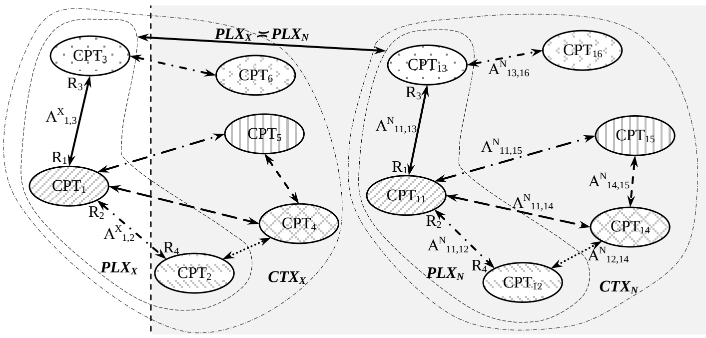

Fig. 10. Association plexus upgrading in third attempt

If such TCC-s are found ASM adds associations between these concepts (concept CPT_6) and corresponding concepts (concept CPT_3) from new association plexus which will have the same parameters as associations from known association plexus recognized at the beginning of the attempt (association between concepts CPT_{13} and CPT_{16}) (Fig. 10).

6. Case

Association plexus upgrading procedure is demonstrated through solving one Raven's Progressive Matrix which was Angelina's task (Fig. 1).

Raven's Progressive Matrices (RPM) is a multiple choice test of abstract reasoning introduced by dr John C. Raven in 1936. They are often used as a test of so-called general intelligence [52], which was also one of Raven's motives when constructing the test. Each RPM problem is presented as a 2x2 or 3x3 matrix of pictures following a pattern. Bottom right position in each matrix is left blank and solver's task is to choose the missing picture for that cell from provided list with eight possible solutions. The first and most common set of RPM are the Standard Progressive Matrices (SPM), consisting of 60 matrices, developed in 1936 [53] and published in 1938.

Why Raven's Progressive Matrices?: The RPM is a type of problem which is familiar to almost everyone, since most people faced an IQ test and it is very clear what the task is. Describing data semantic interpretation process is already too complex task by itself and would be even more difficult to understand if we try to introduce it through ASM cognitive process applied upon some domain specific problem (e.g. choosing the most suitable CAD procedure for reverse modeling of a free-form shape like sternum i.e. chest bone, which actually was our real task to solve). On the other side, introducing new research product (such as ASM) usually needs more space and details, which is always an issue.

In addition, RPM can be very useful and simple example for comparing different approaches in describing and interpreting semantics of figures presented in RPM. Considering the universality, RPM seems as very appropriate domain to compare learning capabilities of different not domain-specific knowledge representation models. Finally, within this kind of domain different reasoning engines can be compared by how many elementary activities in preparation i.e. customization of reasoner an instructor should do to make it capable for relevant entailments.

As of today the cognitive and computational characteristics of RPM aren't yet well understood [54] and no general algorithm for solving them in their entirety has been developed.

In our approach the missing picture (solution) is not chosen from the provided alternatives, but rather built (generated), making the task much harder than in conventional RPM solving.

Two pictures in the first row of the matrix presented in Fig. 1 are semantically described in the context CTX_1 (Fig. 11).

One picture in the second row of the matrix presented in Fig. 1 is semantically described in the context CTX_2 (Fig. 12).

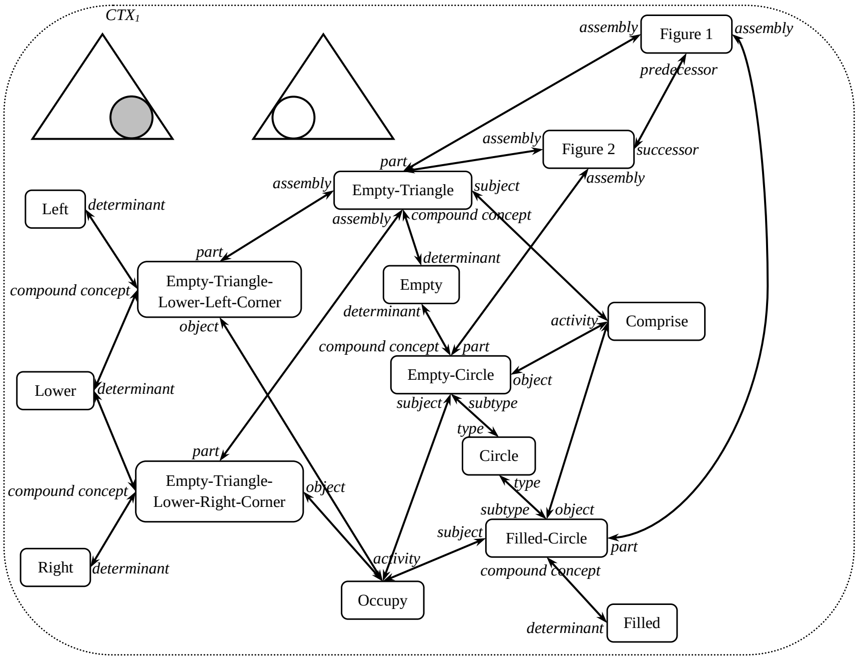

Fig. 11. Semantic description of the two pictures ("Figure 1" (left) and "Figure 2" (right)) in the first row of the matrix. In first picture we have empty triangle which comprises filled circle, while filled circle occupies empty triangle's lower right corner. In second we have empty triangle which comprises empty circle, while empty circle occupies empty triangle's lower left corner

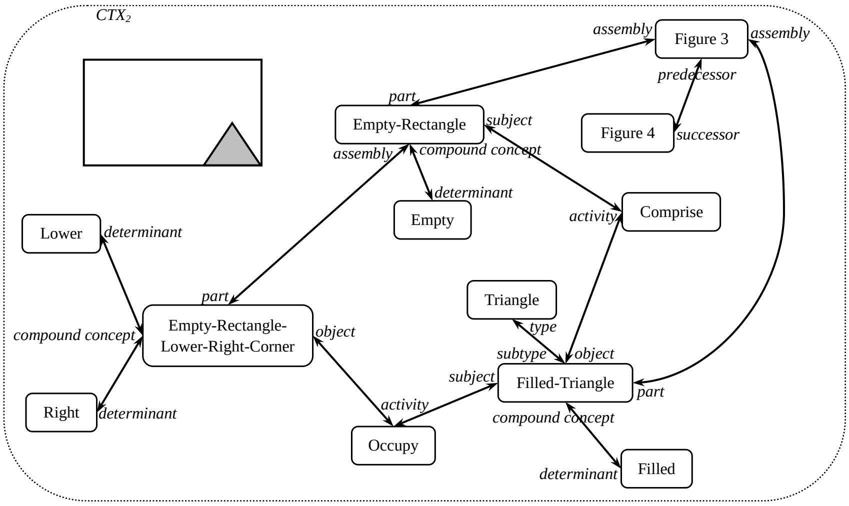

Fig. 12. Semantic description of the picture ("Figure 3") in the second row of the matrix. "Figure 4" will eventually be the missing picture (solution)

General context contains, among other things, knowledge about the geometric shapes from the pictures of the matrix (Fig. 13).

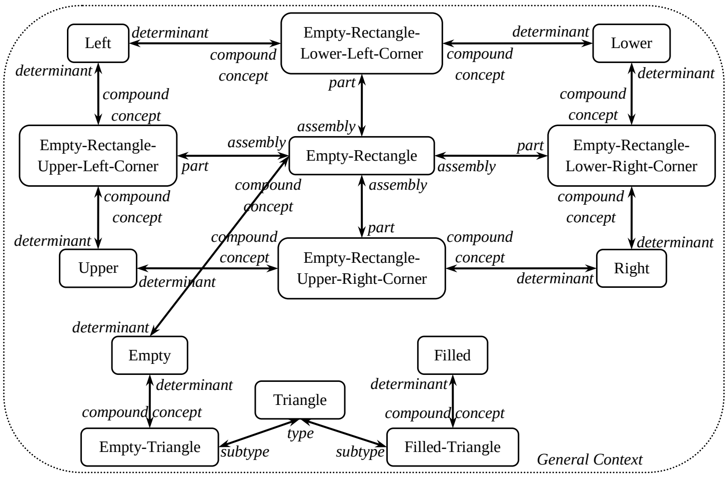

Fig. 13. Knowledge about the geometric shapes from the pictures of the matrix

Association plexuses representing knowledge about the picture in the second row of the matrix and the first picture in the first row of the matrix are topologically analogous and semantically close (Fig. 14). TCA of these two association plexuses are represented by the same type of line, while TCC are represented by the same background pattern.

ASM tries to upgrade new association plexus (representing knowledge about one picture in the second row of the matrix) through several iterations. In the first iteration ASM recognizes semantically distant TCC-s of two association plexuses: 1) “Empty-Rectangle” and “Empty-Triangle”, and 2) “Triangle” and “Circle”. ASM searches for associations in the semantic network involving concept “Empty-Rectangle” which are topologically correspondent to association between concepts “Empty-Triangle” and “Empty-Triangle-Lower-Left-Corner”. If several of them are found (e.g. in *General context*), ASM selects the one whose second concept is involved in the same or similar associations as concept “Empty-Triangle-Lower-Left-Corner” (Fig. 15 up). The same approach is used (applied) for concept “Triangle” (Fig. 15 up). The next step is upgrading of new association plexus through creating new associations: 1) between concepts “Empty-Rectangle” and “Empty-Rectangle-Lower-Left-Corner”, and 2) between concepts “Triangle” and “Empty-Triangle” (Fig. 15 down).

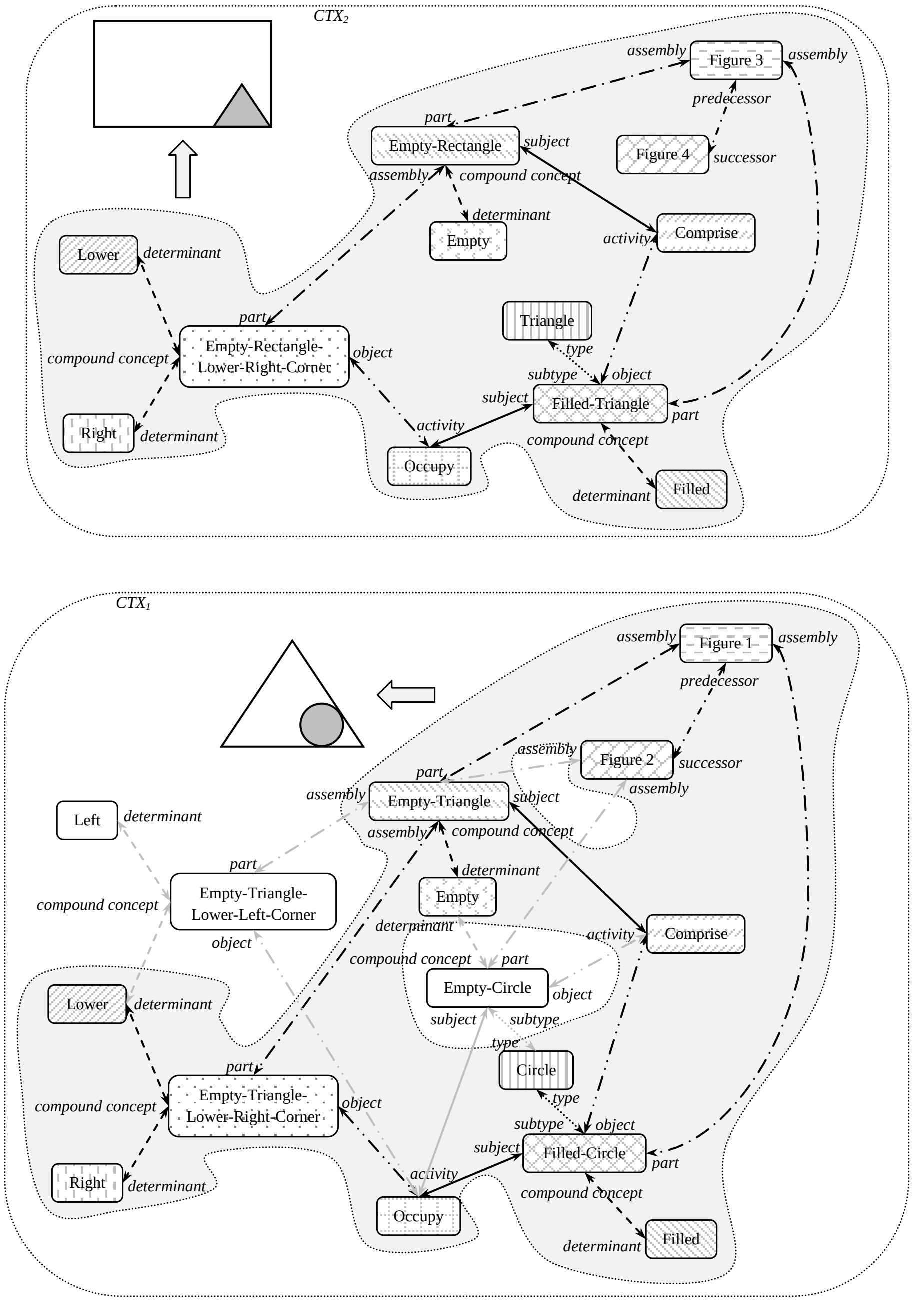

Fig. 14. Recognized topologically analogous and semantically close association plexuses – Subsets of CTX_1 and CTX_2 contexts. TCA are represented by the same type of line, while TCC are represented by the same background pattern. Associations from the context CTX_1 which don't belong to recognized topologically analogous association plexus are grayed out

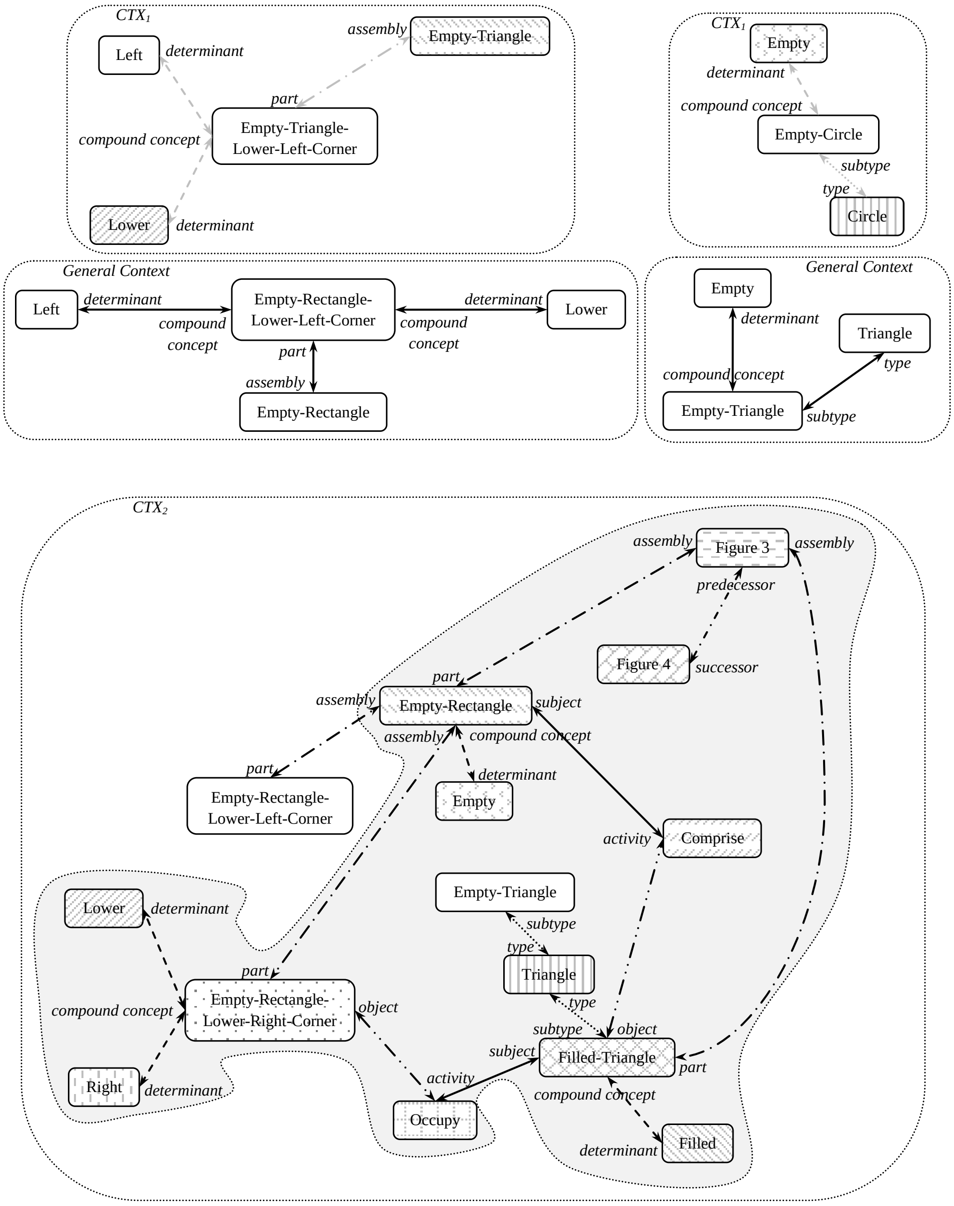

Fig. 15. First iteration of upgrading new association plexus – Creation of new associations: 1) between concepts "Empty-Rectangle" and "Empty-Rectangle-Lower-Left-Corner", and 2) between concepts "Triangle" and "Empty-Triangle" (down). Explanation for choosing the appropriate associations (up)

In second iteration ASM recognizes semantically close TCC-s of two association plexuses (concepts "Lower", "Empty", "Occupy", and "Comprise"). Instead of adding associations of context CTX_1 involving found semantically close TCC-s (like association between concepts “Lower” and “Empty-Triangle-Lower-Left-Corner”), ASM analyzes the structure of context CTX_1, and creates appropriate associations between existing concepts (association between concepts “Lower” and “Empty-Rectangle-Lower-Left-Corner”) (Fig. 16 up). The same approach is used (applied) for other recognized semantically close TCC-s. ASM upgrades new association plexus through creating five new associations (Fig. 16 down).

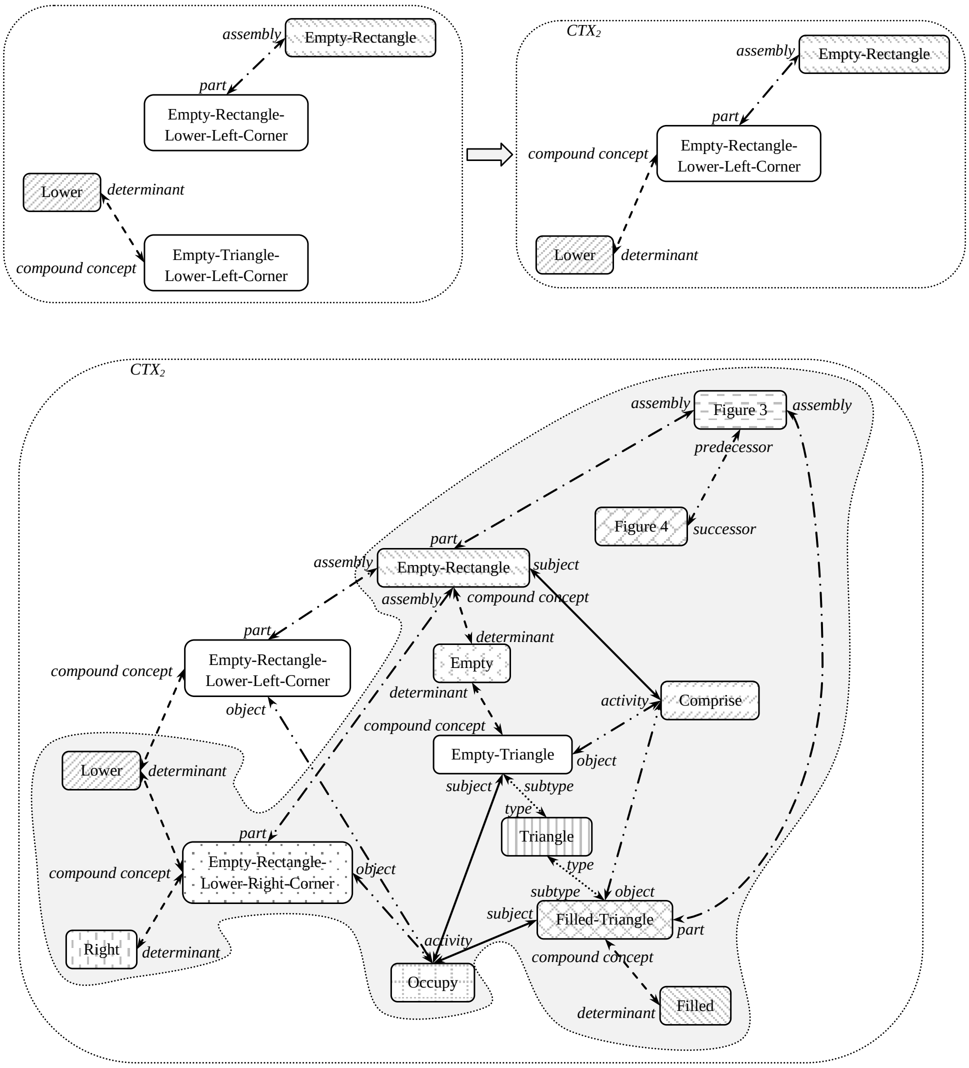

Fig. 16. Second attempt of upgrading new association plexus – Creation of five new associations (down). Decisions are made based on the analysis of the structure of context CTX_1 (up)

In third iteration ASM recognizes semantically distant TCC-s of two association plexuses: 1) “Empty-Rectangle-Lower-Left-Corner” and “Empty-Triangle-Lower-Left-

Corner”, 2) “Empty-Rectangle” and “Empty-Triangle”, and 3) “Empty-Triangle” and “Empty-Circle”. Association between concepts “Empty-Rectangle-Lower-Left-Corner” and “Left” was found in *General context*, and is topologically correspondent to association between concepts “Empty-Triangle-Lower-Left-Corner” and “Left”. As for the remaining two TCC-s ASM analyzes the structure of context CTX_1, and creates two appropriate associations between existing concepts in context CTX_2. ASM finally upgrades new association plexus through creating three new associations (Fig. 17).

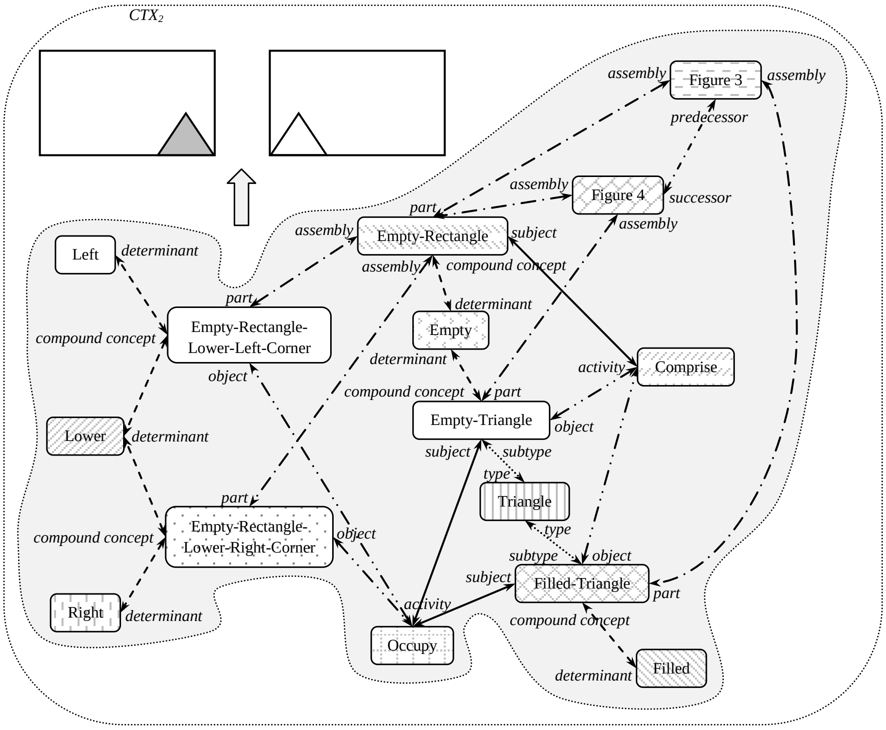

Fig. 17. New association plexus is finally upgraded through creating three new associations

Experimental evaluation of upgrading procedure was done for the following cases: product quality assessment in the early stages of product design [55]; automation of choosing and composing manufacturing process for free-form design parts [56]; exception detection in business process management systems [57]. Presented approach is also being evaluated in the area of digital reconstruction of free-form objects.

7. Implementation and Evaluation

AcSeMod web application, implementing ASM structure and accompanying cognitive data processing algorithms, has been developed for testing purposes and dissemination. The web application was developed using Java programming language. Apache Tomcat v6 was used as application server. Associations and other elements of the ASM structure are stored using MySQL Community Server v5 relational database management system. Presented association plexus upgrading procedure is one of the cognitive data processing algorithms, and is implemented on database level through stored procedures and views.

Performance of the association plexus upgrading procedure was assessed on desktop computer with Windows 7 Enterprise operating system and the following hardware specifications: Intel Core i5 quad-core processor with 3.0 GHz clock speed and 6 MB of cache memory; 8 GB DDR3 RAM memory; hard disk with 1 TB capacity and 32 MB of cache memory. As for the software specifications, MySQL Community Server v5.5.27 relational database management system, Apache Tomcat v6.0.37 application server, and Google Chrome v34.0.1847.131 m web browser were used. Semantic network contained seven contexts with 98 associations. All three iterations were finished in three or less seconds.

8. Conclusion

As it is shown, ASM brings original approach in realization of ABR in semantic network. The core of the ABR process and semantic interpretation of data is performed by ASM’s association plexus upgrading procedure which is based on recognition and determining the similarity of association plexuses. Determining the similarity of association plexuses is performed by the recognition of topological analogy between association plexuses. Relaying on this approach ASM responds to an unpredicted input, which is defined through input association plexuses, by upgrading that association plexus modeled on remainder of the context whose subset is recognized as topologically analogous association plexus. ABR process designed in this way enables autonomous, flexible and analytic semantic interpretation of data described in the semantic network.

Limitations: First users involved in testing or incorporating knowledge can be discouraged by difficulty to understand ASM operating mode. The degree of meaningfulness of ASM responses depends to a large extent on the way in which user described the request for semantic interpretation. Request description is currently the biggest functional problem of ASM. Of course, before describing request some knowledge should be incorporated in ASM. Testing showed that for some successful, meaningful conclusions ASM needed only small portions of knowledge in the network. This can be considered as advantage (like with the case presented in this paper).

Another weakness of ASM is the imprecision (inaccuracy) of the responses. This is also the problem of all approaches which base semantic interpretation of data on analogies. Related to this problem is also the problem of “imaginative” period of work in ASM. In situations when there is little knowledge incorporated in the network, ASM responses can be characterized by “imagination”, and ASM can relate concepts that can’t be related.

Since responses of ASM are based on knowledge previously incorporated in the network, there is a risk of indoctrination. Incorrect indoctrination of ASM, leading to creation (generation) of incorrect conclusions, can appear when multiple users are incorporating their knowledge about the same domain. In this case there is a risk of making semantic content inconsistent (if their interpretations are extremely inconsistent).

Future work on ASM could include testing and adjustment of data structure (semantic network) and set of algorithms for processing of data. Development of intuitive interface, which will enable other software applications from various fields to connect with ASM could be second direction of ASM development. As for the enhancement of ASM functionality there is a need for developing structural elements for incorporation and semantic categorization of *events*, i.e. contexts which string one after another in the time sequence of discrete time instants.

The ability to recognize analogy between semantically very distant situations is considered as one of the essential characteristics of creativity. Creative conclusions usually start by recognizing similarity between apparently semantically disconnected elements and arise by creating new semantic relations between these elements or ideas. According to another stand [25], creative conclusions arise by creating new context-suitable semantic relations between elements or ideas which are already connected by some “old” semantic relations, which are not applicable for the actual context. In ASM topologically correspondent associations from completely semantically distant contexts can be used for drawing conclusions. In this way knowledge from one context can be applied in situations which belong to other completely different contexts enabling ASM to demonstrate creativity. Associations between the same concepts, belonging to different contexts (and having different parameters), participate in the decision making process in a completely different way, depending on the context they belong to, which makes ASM more flexible and productive in capturing and interpreting semantics of data compared to existing semantic models [49].

One can see ASM as a kind of layer above the DL ontologies layer (strongly structured knowledge, i.e. richly axiomatized discourse) which helps semantic interpretation.

Acknowledgment. The paper represents a summary about a part of the research that is conducted within the project “Virtual human osteoarticular system and its application in preclinical and clinical practice” (project id III 41017) which is funded by the Ministry of Education, Science and Technological Development of the Republic of Serbia for the period 2011-2014.

References

1. Daconta, M.C., Obrst, L.J., Smith, K.T.: The Semantic Web: A Guide to the Future of XML, Web Services, and Knowledge Management. Wiley Publishing, Inc., Indianapolis, USA. (2003)
2. Forbus, K.D., Mostek, T., Ferguson, R.: An Analogy Ontology for Integrating Analogical Processing and First-principles Reasoning. In Proceedings of the 14th Conference on Innovative Applications of AI, IAAI-02. AAAI Press, Menlo Park, California, USA. (2002)
3. Corcho, O., Gomez-Perez, A.: Evaluating Knowledge Representation and Reasoning Capabilities of Ontology Specification Languages. In Proceedings of the Workshop on Applications of Ontologies and Problem Solving Methods, 14th European Conference on Artificial Intelligence, ECAI'00. IOS Press, Amsterdam, The Netherlands. (2000)
4. Dentler, K., Cornet, R., ten Teije, A., de Keizer, N.: Comparison of reasoners for large ontologies in the OWL 2 EL profile. Semantic Web – Interoperability, Usability, Applicability, Vol. 2, No. 2, 71-87. (2011)
5. Pan, Z.: Benchmarking DL Reasoners Using Realistic Ontologies. In Proceedings of the 1st OWL: Experiences and Directions Workshop. Galway, Ireland. (2005)

6. W3C Recommendation – RIF Production Rule Dialect (Second Edition) (2013). Available: http://www.w3.org/TR/2013/REC-rif-prd-20130205/ (current February 2013)
7. Nikitina, N., Rudolph, S., Blohm, S.: Refining Ontologies by Pattern-Based Completion. In Proceedings of the Workshop on Ontology Patterns, WOP 2009. Washington, DC, USA. (2009)
8. Simperl, E., Burger, T., Hangl, H., Worgl, S., Popov, I.: ONTOCOM: A reliable cost estimation method for ontology development projects. Web Semantics: Science, Services and Agents on the World Wide Web, Vol. 16, 1-16. (2012)
9. Hitzler, P., van Harmelen, F.: A Reasonable Semantic Web. Semantic Web – Interoperability, Usability, Applicability, Vol. 1, No. 1-2, 39-44. (2010)
10. Horrocks, I.: Description Logic Reasoning (tutorial). 11th International Conference on Logic for Programming Artificial Intelligence and Reasoning, LPAR 2004. Montevideo, Uruguay. Available: http://www.cs.ox.ac.uk/people/ian.horrocks/Seminars/download/lpar05.ppt (2005)
11. Lehmann, J., Volker, J.: An Introduction to Ontology Learning. In: Lehmann, J., Volker, J. (eds.): Perspectives on Ontology Learning. IOS Press, Amsterdam, The Netherlands. (2014)
12. Shamsfard, M., Barforoush, A.A.: The state of the art in ontology learning: a framework for comparison. The Knowledge Engineering Review, Vol. 18, No. 4, 293-316. (2003)
13. Hazman, M., El-Beltagy, S.R., Rafea, A.: A Survey of Ontology Learning Approaches. International Journal of Computer Applications, Vol. 22, No. 9, 36-43. (2011)
14. Arnold, P., Rahm, E.: Enriching ontology mappings with semantic relations. Data & Knowledge Engineering, Vol. 93, 1-18. (2014)
15. Lu, Y., Panetto, H., Ni, Y., Gu, X.: Ontology alignment for networked enterprise information system interoperability in supply chain environment. International Journal of Computer Integrated Manufacturing, Vol. 25, No. 12, 1148-1159. (2012)
16. Petrakis, E.G.M., Varelas, G., Hliaoutakis, A., Raftopoulou, P.: Design and Evaluation of Semantic Similarity Measures for Concepts Stemming from the Same or Different Ontologies. In Proceedings of the 4th Workshop on Multimedia Semantics, WMS'06. Chania, Crete, Greece. (2006)
17. Rahm, E., Arnold, P., Raunich, S.: Semantic ontology mappings: How to determine and use them. Colloquium Talk, University of Paris-Sud. Available: http://dbs.uni-leipzig.de/file/paris-Octob2014.pdf (2014)
18. Zdravkovic, M., Panetto, H., Trajanovic, M., Aubry, A.: An approach for formalising the supply chain operations. Enterprise Information Systems, Vol. 5, No. 4, 401-421. (2011)
19. Borgida, A., Walsh, T.J., Hirsh, H.: Towards Measuring Similarity in Description Logics. In Proceedings of the 2005 International Workshop on Description Logics, DL2005. Edinburgh, Scotland, UK. (2005)
20. Kim, K.-Y., Manley, D.G., Yang, H.: Ontology-based assembly design and information sharing for collaborative product development. Computer-Aided Design, Vol. 38, No. 12, 1233-1250. (2006)
21. d'Amato, C., Fanizzi, N., Esposito, F.: Reasoning by Analogy in Description Logics through Instance-based Learning. In Proceedings of the 3rd Italian Semantic Web Workshop Scuola Normale Superiore. Pisa, Italy. (2006)
22. Raad, E., Evermann, J.: Is Ontology Alignment like Analogy? — Knowledge Integration with LISA. In Proceedings of the 29th Annual ACM Symposium on Applied Computing, SAC'14. Gyeongju, Korea. (2014)
23. Raad, E., Evermann, J.: The role of analogy in ontology alignment: A study on LISA. Cognitive Systems Research, Vol. 33, 1-16. (2015)
24. Janowicz, K., Raubal, M., Kuhn, W.: The semantics of similarity in geographic information retrieval. Journal of Spatial Information Science, No. 2, 29-57. (2011)
25. Kostić, A.: Kognitivna psihologija. Zavod za udžbenike i nastavna sredstva, Beograd, Srbija. (2006)
26. Gentner, D., Holyoak, K.J., Kokinov, B.N.: The Analogical Mind: Perspectives from Cognitive Science. The MIT Press, Cambridge, USA. (2001)

27. Forbus, K.D.: Exploring Analogy in the Large. In: Gentner, D., Holyoak, K.J., Kokinov, B.N. (eds.): The Analogical Mind: Perspectives from Cognitive Science. The MIT Press, Cambridge, USA, 23-58. (2001)
28. Newel, A., Simon, H.A.: Human Problem Solving. Prentice Hall, New Jersey, USA. (1972)
29. Aamodt, A., Plaza, E.: Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches. AI Communications, Vol. 7, No. 1, 39-59. (1994)
30. Davies, J., Goel, A.K., Yaner, P.W.: Proteus: Visuospatial analogy in problem solving. Knowledge-Based Systems, Vol. 21, No. 7, 636-654. (2008)
31. Bergman, R., Kolodner, J., Plaza, E.: Representation in case-based reasoning. The Knowledge Engineering Review, Vol. 20, No. 3, 209-213. (2006)
32. Holyoak, K.J.: Analogical thinking and human intelligence. In: Sternberg, R.J. (ed.): Advances in the psychology of human intelligence. Erlbaum, Hillsdale, NJ, USA. (1984)
33. Chang, P.-C., Lin, J.-J., Dzan, W.-Y.: Forecasting of manufacturing cost in mobile phone products by case-based reasoning and artificial neural network models. Journal of Intelligent Manufacturing, Vol. 23, No. 3, 517-531. (2012)
34. Begum, S., Ahmed, M.U., Funk, P., Xiong, N., Scheele, B.V.: A case-based decision support system for individual stress diagnosis using fuzzy similarity matching. Computational Intelligence, Vol. 25, No. 3, 180-195. (2009)
35. Passone, S., Chung, P.W.H., Nassehi, V.: Incorporating domain-specific knowledge into a genetic algorithm to implement case-based reasoning adaptation. Knowledge-Based Systems, Vol. 19, No. 3, 192-201. (2006)
36. Forbus, K.D., Gentner, D., Law, K.: MAC/FAC: A Model of Similarity-based Retrieval. Cognitive Science, Vol. 19, No. 2, 141-205. (1995)
37. Holyoak, K.J., Koh, K.: Surface and structural similarity in analogical transfer. Memory & Cognition, Vol. 15, No. 4, 332-340. (1987)
38. Negny, S., Riesco, H., Le Lann, J.M.: Effective retrieval and new indexing method for case based reasoning: Application in chemical process design. Engineering Applications of Artificial Intelligence, Vol. 23, No. 6, 880-894. (2010)
39. Smyth, B., Keane, M.T.: Adaptation-guided retrieval: questioning the similarity assumption in reasoning. Artificial Intelligence, Vol. 102, No. 2, 249-293. (1998)
40. Foxwell, H.J., Menasce, D.A.: MARVIN: A Web-Based System for Representing, Retrieving, and Visualizing Analogies. World Wide Web: Internet and Web Information Systems, Vol. 7, No. 4, 385-419. (2004)
41. Hall, R.P.: Computational Approaches to Analogical Reasoning: A Comparative Analysis. Artificial Intelligence, Vol. 39, No. 1, 39-120. (1989)
42. Kedar-Cabelli, S.: Analogy - From a Unified Perspective. In Helman, D.H. (ed.): Analogical Reasoning: Perspectives of Artificial Intelligence, Cognitive Science, and Philosophy, Vol. 197. Kluwer Academic Publishers, Dordrecht, Netherlands, 65-103. (1988)
43. Kokinov, B.: Analogy is like Cognition: Dynamic, Emergent, and Context-Sensitive. In Holyoak, K., Gentner, D., Kokinov, B. (eds.): Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Comptational, and Neural Sciences. New Bulgarian University Press, Sofia, Bulgaria, 96-105. (1998)
44. Kokinov, B.N., Petrov, A.A.: Integration of Memory and Reasoning in Analogy-Making: The AMBR Model. In: Gentner, D., Holyoak, K.J., Kokinov, B. (eds.): The analogical mind: Perspectives from Cognitive Science. MIT Press, Cambridge, MA, USA. (2000)
45. Hummel, J.E., Holyoak, K.J.: LISA: A Computational Model of Analogical Inference and Schema Induction. In Proceedings of the 18th Annual Conference of the Cognitive Science Society. Lawrence Erlbaum Associates, Mahwah, NJ, USA. (1996)
46. Falkenhainer, B., Forbus, K.D., Gentner, D.: The Structure-Mapping Engine: Algorithm and Examples. Artificial Intelligence, Vol. 41, No. 1, 1-63. (1989)
47. Holyoak, K.J., Thagard, P.: Analogical Mapping by Constraint Satisfaction. Cognitive Science, Vol. 13, No. 3, 295-355. (1989)

48. Stojković, M.: Analysis of the Manufacturability Parameters Based on Semantic Structures of the Digital Product Model. Ph.D. Thesis, University of Niš, Faculty of Mechanical Engineering in Niš, Niš, Serbia. (2011)
49. Stojković, M., Manić, M., Trifunović, M., Mišić, D.: Semantic categorization of data by determining the similarities of associations of the semantic network. E-Society Journal Research and Applications, Vol. 2, No. 1, 3-13. (2011)
50. Trifunović, M., Stojković, M., Trajanović, M., Mišić, D., Manić, M.: Interpreting the meaning of geometric features based on the similarities between associations of semantic network. FACTA UNIVERSITATIS Series: Mechanical Engineering, Vol. 11, No. 2, 181-192. (2013)
51. Anderson, J.R., Bower, G. H.: Human Associative Memory. Winston, Washington, USA. (1973)
52. Snow, R., Kyllonen, P., Marshalek, B.: The topography of ability and learning correlations. In: Sternberg, R.J. (ed.): Advances in the Psychology of Human Intelligence, Vol. 2. Lawrence Erlbaum Associates, Hillsdale, New Jersey, USA, 47-103. (1984)
53. Raven, J.C.: Mental tests used in genetic studies: The performances of related individuals in tests mainly educative and mainly reproductive. Master's Thesis, University of London. (1936)
54. Kunda, M., McGreggor, K., Goel, A.: Addressing the Raven's Progressive Matrices Test of "General" Intelligence. In Proceedings of the AAAI Fall Symposium on Multi-representational Architectures for Human-level Intelligence. AAAI Press, Menlo Park, California, USA. (2009)
55. Stojkovic, M., Manic, M., Trifunovic, M., Vitkovic, N.: Semantic interpretation of the product model features in product quality assessment. In Proceedings of the 6th International Working Conference "Total Quality Management – Advanced and Intelligent Approaches". 481-484. (2011)
56. Manic, M., Stojkovic, M., Trifunovic, M.: Semantic features in computer aided manufacturing systems. In Proceedings of the International Conference "Mechanical Engineering in XXI Century". Faculty of Mechanical Engineering in Niš, Niš, Serbia, 179-182. (2010)
57. Misic, D., Stojkovic, M., Domazet, D., Trajanovic, M., Manic, M., Trifunovic, M.: Exception detection in business process management systems. Journal of Scientific and Industrial Research, Vol. 69, No. 3, 188-193. (2010)

Milos Stojkovic is assistant professor in the Department for Production, IT and Management at the Faculty of Mechanical Engineering in Nis, University of Nis. He graduated with a degree in mechanical engineering from the same University in 1996 and received his Ph.D. in 2011. His areas of research are knowledge-based engineering systems (R&D in semantic networks and cognitive data processing models for CAD/CAM systems) and bioengineering (R&D in customized tissue scaffold design and fabrication and R&D in CAD methods for reverse modeling of the human bones). He is the author of more than 40 international scientific publications, 15 of them in international journals.

Milan Trifunovic is teaching and research assistant in the Department for Production, IT and Management at the Faculty of Mechanical Engineering in Nis, University of Nis. He graduated with a degree in mechanical engineering from the same University in 2003. His areas of research are application of information technologies in manufacturing, artificial intelligence and knowledge-based engineering systems

(semantic networks and cognitive data processing models for CAD/CAM systems). He is the author of approximately 14 international scientific publications, 3 of them in international journals.

Dragan Misic is assistant professor in the Department for Production, IT and Management at the Faculty of Mechanical Engineering in Nis, University of Nis. He graduated with a degree in mechanical engineering from the same University in 1991 and received his Ph.D. in 2010. His areas of research are application of information technologies in manufacturing, simultaneous design of technological processes and integration of CAD/CAPP/CAMM systems. He is the author of approximately 21 international scientific publications, 5 of them in international journals.

Miodrag Manic is full professor in the Department for Production, IT and Management at the Faculty of Mechanical Engineering in Nis, University of Nis. He graduated with a degree in mechanical engineering from the same University in 1980 and received his Ph.D. in 1995. His areas of research are application of information technologies, including methods of artificial intelligence, computer numerically controlled machine tools and their programming and technology and manufacturing systems. He is the author of approximately 35 international scientific publications, 9 of them in international journals.

Received: November 3, 2014; Accepted: April 29, 2015.