Serdica J. Computing **6** (2012), 309–332

Serdica
Journal of Computing

Bulgarian Academy of Sciences
Institute of Mathematics and Informatics

AN ALGORITHMIC APPROACH TO INFERRING CROSS-ONTOLOGY LINKS WHILE MAPPING ANATOMICAL ONTOLOGIES

Peter Petrov, Milko Krachounov, Ernest A. A. van Ophuizen, Dimitar Vassilev

Abstract. Automated and semi-automated mapping and the subsequently merging of two (or more) anatomical ontologies can be achieved by (at least) two direct procedures.

The first concerns syntactic matching between the terms of the two ontologies; in this paper, we call this direct matching (DM). It relies on identities between the terms of the two input ontologies in order to establish cross-ontology links between them.

The second involves consulting one or more external knowledge sources and utilizing the information available in them, thus providing additional information as to how terms (concepts) from the two input ontologies are related/linked to each other. Each of the two ontologies is aligned to an external knowledge source and links representing synonymy, is-a parent-child, and part-of parent-child relations, are drawn between the ontology and the knowledge source. These links are then run through a set of simple logical rules in order to come up with cross-ontology links between the two input ontologies. This method is known as semantic matching. It proves useful

ACM Computing Classification System (1998): J.3.

Key words: ontology, anatomical ontology, ontology mapping, anatomical ontology mapping, probability, scoring, external knowledge source, algorithm, graph, directed acyclic graph.

and reasonably accurate; in this paper, we call it the source matching predictions (SMP) procedure.

Not all cross-ontology links that semantically (i.e., from a biological/anatomical standpoint) exist between the two input ontologies will be discovered by either DM or SMP. To improve the discovery of cross-ontology links we propose a novel algorithmic procedure which involves a probability-like scoring scheme. This procedure is called the child matching predictions (CMP) procedure. Describing the DM, SMP, CMP procedures, and particularly the CMP procedure in formal terms is the main goal of this paper.

1. Introduction. Ontologies are formal models for knowledge representation and knowledge modeling. A widely adopted definition is that "*an ontology is an explicit and formal specification of a conceptualization of a domain of interest*" [5]. Two main aspects are highlighted by this definition – first, that the specification is formal, which implies that automatic reasoning can be performed on it, and secondly, that it is practically oriented towards a particular domain of interest. Another informal definition can be found in [6]; it states that "*an ontology grasps the entities which exist within a given portion of the world at a given level of generality, it includes a taxonomy of the types of entities and relations that exist in that portion of the world seen from within a given perspective".* This definition focuses again on two aspects – first, that an ontology models only a portion of the world, which implies its specifici, and second, that an ontology has a formal structure (called taxonomy) that includes the entities that exist (in the portion of the world that is being modeled) and the relations which exist among them.

Important problems in the research area which deals with ontologies are those of ontology integration or mediation [1]. The two terms, integration and mediation, are pretty much synonymous but the latter is preferred for the purposes of this paper as it has already been adopted by most authors. Ontology ***mediation*** concerns integrating ontologies that model identical or similar domains but which have different origin. The importance of the ontology mediation problem comes from the fact that ontologies are designed and developed by different parties (research groups, business organizations) and it cannot be expected that these parties will ever agree on using a common ontology even though the domain being modeled is similar or even identical.

As noted in [1], two principal types of ontology mediation exist – ***ontology mapping*** and ***ontology merging***. Mapping is about establishing links/bridges between two (or more) ontologies without altering them. The result of the mapping process of several input ontologies is, in principle, not an ontology but a

set of semantic links/bridges/correspondences between the ontologies. That result doesn't replace the original ontologies, but supplements them and is stored separately of the input ontologies. Merging is about taking two ontologies and generating a single ontology from them that unifies/unites the knowledge contained in the input ontologies. The result of the merging process is a single output ontology that could be used as a replacement of the two input ontologies.

Another important concept related to ontology mediation is ***ontology alignment*** – the process of automatic or semi-automatic discovery of links between ontologies [1], as opposed to manual discovery of these links. In particular, special attention should be paid to the cases of alignment of heterogeneous ontologies based on different conceptualizations of the same problem domain [3, 4]. For the purposes of this work, it is assumed that two given ontologies can be aligned to each other, but also to some external knowledge sources (which may or may not be ontologies themselves).

For solving the general ontology mediation problem, various efforts have been made in the last decade that usually produce theoretical models, which then serve as a basis for practical program or framework implementations. We list here only the most prominent or popular ones: (i) ontology mapping – MAFRA [8], RDTF [9], and IF-Map [10], (ii) ontology merging – PROMPT [11], and OntoMerge [12], (iii) ontology alignment – Anchor-PROMPT [13], GLUE [14], QOM [15, 16], S-Match [17, 18]. Excellent surveys of the ontology mediation research field can be found in [1], [2], and [19].

In this work, we deal with an ontology mapping and merging problem within a very specific, practical context. This is the problem of mapping and merging **anatomical ontologies** of two or more different species/organisms. The problem is important for at least two different reasons.

First, the ability to perform cross-species automated text searches (text mining) in scientific literature can produce valuable results. It enables a researcher designing experiments in a particular model organism (e.g., mouse) to draw upon earlier findings in a different model organism (e.g., zebrafish), without needing to be an expert on both systems. Anatomical ontologies of many different species are nowadays publicly available, but no intelligent tools exist that are able to perform **intelligent cross-species text searches** (or text mining) in these ontologies or in various text sources that contain anatomical information about the different species (e.g., mouse, rat, chicken, zebrafish). What is needed is the ability to perform searches that don't rely solely on simple text identities between term names in order to report these terms as synonyms (e.g., head(mouse) = head(rat)), but which would be intelligent enough to detect cross-species synonyms whose

textual representations have nothing in common (for instance fin(zebrafish) = wing(chicken) = foreleg(mouse, rat)). Here the equality sign denotes an anatomical similarity (roughly speaking) or homology (strictly speaking) between anatomical terms of different species. It is apparent that to achieve these goals, the different **species-specific anatomical ontologies** need to be mapped onto each other and (in the ideal case) ultimately merged into a **single output anatomical super-ontology**.

Second, having two species-specific ontologies mapped onto each other and possibly merged into a common super-ontology would enable tools which currently work with the anatomical ontology of one species to support more than the species-specific ontology which they were originally designed for. That is, solving the ontology mapping problem could extend the capabilities of existing tools and could make them more intelligent and more powerful. Once the anatomical super-ontology is there, existing tools could be ported (with some effort) to the super-ontology which resulted from merging the two input species-specific anatomical ontologies. This would turn those tools from **single-species aware** to **multi-species aware**.

Due to the very specific nature of the problem, a very specific approach is presented here which does not have any claims to generality but rather to specificity and biological (in particular anatomical) adequacy of the results.

The general methods listed above usually try to map, merge or align ontologies modeling the same or similar domains of interest. In this work, the domains modeled by the input ontologies are rather similar when viewed from one angle (as they are both anatomical domains) but rather distinct when viewed from another angle (as they represent the anatomies of two different species which may or may not be closely related from an evolutionary standpoint). Due to the specific nature of the problem, it is possible to interrogate specific biomedical knowledge sources (like UMLS[1] [22], FMA[2] [23]) and to utilize their knowledge which inherently imparts certain intelligence to the software program (AnatOM [7]) that implements the algorithmic procedures presented in this paper. However, the specificity of the problem does not prevent AnatOM from also interrogating general-purpose knowledge sources (like WordNet[3] [20, 21]). Talking to such general-purpose knowledge sources proves very useful as they provide valuable additional insights to inferring links between the ontologies which are subject to mapping and ultimately to merging.

[1] http://www.nlm.nih.gov/research/umls/ (2012)
[2] http://sig.biostr.washington.edu/projects/fm/ (2012)
[3] http://wordnet.princeton.edu/ (2012)

2. An overview of the problem domain. Anatomy is a branch of biology and medicine that studies the structures of the living things (organisms, species). Three main branches of anatomy exist – (i) human anatomy, (ii) animal anatomy (zootomy), (iii) plant anatomy (phytotomy). This work deals with (ii) even though some of its ideas and methods are applicable also to (i) and (iii). Anatomy can also be divided into (a) macroscopic anatomy which studies structures that can be observed even with the naked human eye, and (b) microscopic anatomy which studies structures that the naked human eye cannot observe. Of these two, this work deals mostly with (a). The algorithmic procedures presented in this paper take two anatomical ontologies as input (e.g., **the adult mouse anatomical ontology** and the **ontology of the zebrafish anatomy and development**) and map them onto each other.

The two input ontologies are encoded in OBO [24, 25] which is a formal language for representing ontologies (like OWL [27] and RDF-Schema [28]). The OBO ontology language is used mainly in the biomedical sciences and in bioinformatics; its computer representation is a plain text file format which is also known as OBO. This plain text file format is easily readable by both humans and computer programs; it allows for describing the terms/concepts from the domain that is modeled together with the relations that exist among these terms/concepts. For the purposes of this work, the ontologies originally encoded in OBO are first translated to mathematical (graph theoretical) forms, and the procedures presented below work on these mathematical forms. The algorithmic procedures themselves are also described in mathematical terms and not in pseudo-code or in some practical programming language.

3. Formal definition of the problem.

3.1. The two input ontologies. Two input ontologies are given in the form of OBO files. For the purposes of this work each of these ontologies is viewed as a directed acyclic graph (DAG) [7] together with an edge-coloring function. The two ontologies used as examples here are the mouse $\boldsymbol{O_1 = O_M}$ and the zebrafish $\boldsymbol{O_2 = O_Z}$ anatomical ontologies but the method presented below is applicable to other couples of species-specific anatomical ontologies, e.g., (mouse, rat), (mouse, chicken), (chicken, zebrafish).

In the text below the following notations are used.

$$\boldsymbol{O_1 : DAG_1 = (V_1, E_1); F_1 : E_1 \to C = \{c_1, c_2, \ldots, c_n\}}$$

$$\boldsymbol{O_2 : DAG_2 = (V_2, E_2); F_2 : E_2 \to C = \{c_1, c_2, \ldots, c_n\}}$$

Here $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ are the two input ontologies each of which is considered as composed of a **directed acyclic graph** $\boldsymbol{DAG_k}$ and an **edge-coloring function** $\boldsymbol{F_k}$. Also here, $\boldsymbol{C = \{c_1, c_2, \ldots, c_n\}}$ is the set of colors, $\boldsymbol{F_1}$ and $\boldsymbol{F_2}$ are two coloring functions which are associated with the two directed acyclic graphs $\boldsymbol{DAG_1}$ and $\boldsymbol{DAG_2}$. Each color represents one inner-ontology relation of subsumption of certain kind (inverse generalization, i.e., specialization; inverse aggregation, i.e., membership; etc.). The relations **is-a** (specialization) and **part-of** (membership) are the two typical examples of such inner-ontology relations defined within OBO ontologies and within anatomical OBO ontologies in particular. Therefore, for the purposes of this work, it can be assumed that $\boldsymbol{n = 2}$, $\boldsymbol{c_1}$ = **is-a**, $\boldsymbol{c2}$ = **part-of**.

In the notation introduced above, $\boldsymbol{V_1}$ is the set of anatomical terms/concepts in the mouse anatomical ontology and $\boldsymbol{V_2}$ is the set of anatomical terms/concepts in the zebrafish anatomical ontology.

$\boldsymbol{V_1 = \{v_{11}, v_{12}, \ldots, v_{1n_1}\}, |V_1| = n_1}$

$\boldsymbol{V_2 = \{v_{21}, v_{22}, \ldots, v_{2n_2}\}, |V_2| = n_2}$.

Each ontology term $\boldsymbol{v_{ij}}$ has two components which are both strings $\boldsymbol{(id_{ij}, name_{ij})}$, where $\boldsymbol{id_{ij}}$ is the ***identifier (the id)*** of the term/concept $\boldsymbol{v_{ij}}$, and $\boldsymbol{name_{ij}}$ is the textual ***name*** of the term/concept $\boldsymbol{v_{ij}}$.

In general, the term ids are unique within the ontology bounds but are not globally unique. Theoretically, if two different ontologies are given, it is possible that there exist two terms, one term from the first ontology and the other one from the second ontology, which are distinct but whose ids are equal. Practically, in our case, all mouse term ids begin with the string ***"MA"*** and all zebrafish term ids begin with the string ***"ZFA"*** so it is impossible to have two terms (one from mouse, one from zebrafish) sharing the same term id. In the two ontologies $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ each term $\boldsymbol{t = (id, name)}$ may optionally also have a set of alternative names or what is called inner-ontology synonyms.

Within the first ontology, the edge $\boldsymbol{e_1 = (v_{1i}, v_{1j}) \in E_1}$ if and only if the term $\boldsymbol{v_{1i}}$ is a child of the term $\boldsymbol{v_{1j}}$ in the graph $\boldsymbol{DAG_1}$. The same applies to the second ontology, i.e., the edge $\boldsymbol{e_2 = (v_{2i}, v_{2j}) \in E_2}$ if and only if the term $\boldsymbol{v_{2i}}$ is a child of the term $\boldsymbol{v_{2j}}$ in the graph $\boldsymbol{DAG_2}$. Here, "child" is a generalized concept meaning either an **is-a** or a **part-of** child. Throughout this text we refer to $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ as the two input ontologies.

3.2. The three external knowledge sources. Also given are several large external knowledge sources (biomedical or general-purpose ontologies) which contain anatomical terms and relations (**is-a**, **part-of**, others) among those terms. In particular, three concrete external knowledge sources are used for the purposes of this work. These are $\boldsymbol{T_1 = UMLS}$, $\boldsymbol{T_2 = FMA}$, $\boldsymbol{T_3 = WordNet}$. Although

questionable if these knowledge sources are indeed ontologies (in the strict sense), they are viewed and used as such for the purposes of this work. Formally put, each of these knowledge sources $\boldsymbol{T_s}$ $\boldsymbol{(s = 1, 2, 3)}$ contains the following information.

- Set of terms

  $\boldsymbol{M_s = \{t_{s1}, t_{s2}, \ldots, t_{sm_s}\}}$, where
  $\boldsymbol{t_{sk} = (id_{sk}, name_{sk})}$ and
  $\boldsymbol{id_{sk}}$ is the identifier (the id) of the term/concept $\boldsymbol{t_{sk}}$,
  $\boldsymbol{name_{sk}}$ is the textual name of the term/concept $\boldsymbol{t_{sk}}$,
  $\boldsymbol{m_s}$ is the count of terms in the knowledge source $\boldsymbol{T_s}$.

It should be noted at this stage that: **i)** it is sometimes possible that $\boldsymbol{t_{si} \neq t_{sj}}$ but $\boldsymbol{name_{si} = name_{sj}}$ (same names, different ids); **ii)** it is sometimes possible that $\boldsymbol{t_{si} \neq t_{sj}}$ but $\boldsymbol{id_{si} = id_{sj}}$ (same ids, different names); **iii)** in this notation, the ids and the names are strings and the equalities (or inequalities) above express identity (or lack of identity) between the strings involved.

- Relations of subsumption

Each knowledge source $\mathbf{T_s}$ also defines (at least) the following two relations:

$$\boldsymbol{R'_{T_s} = R^{is\_a}_{T_s} \subseteq M_s \times M_s}$$
$$\boldsymbol{R''_{T_s} = R^{part\_of}_{T_s} \subseteq M_s \times M_s}$$

These two are the **is-a** and **part-of** relations (again) but in the way they are defined by the knowledge source $\boldsymbol{T_s}$. Additional relations are usually also defined within $\boldsymbol{T_s}$ but the **is-a** and **part-of** are of greatest interest for the purposes of this work.

3.3. The problem goal. Using the available knowledge sources $\boldsymbol{T_1 = UMLS}$, $\boldsymbol{T_2 = FMA}$, $\boldsymbol{T_3 = WordNet}$ and the **is-a** and **part-of** relations which they define between their own terms, a set of reliable (authentic, trustworthy) semantic relations between the terms of the two input ontologies $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ has to be found. These semantic relations should be biologically (anatomically, evolutionary) justified and should be of one of the following types.

Type 1. Synonyms – $\boldsymbol{R_1 = R_{syn}}$ – terms with similar or identical meaning are called synonyms.

Type 2. Hypernyms – $\boldsymbol{R_1 = R_{hyper}}$ – generalization – a hypernym is a term whose semantic range includes that of another term (its hyponym) – Fig. 1.

Type 3. Hyponyms – $\boldsymbol{R_1 = R_{hypo}}$ – specialization – a hyponym is a term whose semantic range is included within that of another term (its hypernym) – Fig. 1.

Type 4. Holonym – $\boldsymbol{R_1 = R_{holo}}$ – aggregation – term X is a holonym of term Y, if Ys are parts of (members of) X – Fig. 2.

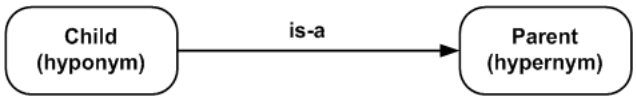

Fig. 1. An is-a parent-child relation

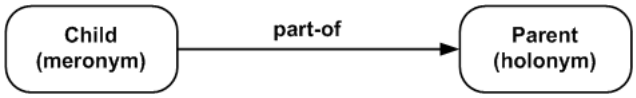

Fig. 2. A part-of parent-child relation

Type 5. Meronym – $\boldsymbol{R_1 = R_{mero}}$ – membership – term Y is a meronym of term X, if Ys are parts of (members of) X – Fig. 2.

The goal here is to establish relations of the types just described (from 1 to 5) such that $\boldsymbol{R_i \subseteq (V_1 \times V_2) \bigcup (V_2 \times V_1)}$, for $\boldsymbol{i = 1, 2, 3, 4, 5}$ and such that these relations are authentic (make sense, i.e., are biologically valid, i.e., are evolutionary justified) based on the knowledge that is available in the external knowledge sources $\boldsymbol{T_1 = UMLS}$, $\boldsymbol{T_2 = FMA}$, $\boldsymbol{T_3 = WordNet}$. Of greatest interest is establishing the relations of Type 1 (the synonymy relations or $\boldsymbol{R_1 = R_{syn}}$ as these allow for mapping the two input ontologies $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ onto each other, and ultimately for merging them into one common output **super-ontology** which we denote as $\boldsymbol{O_{super}}$.

4. The algorithmic solution. In this paper an integrated algorithmic approach to solving the problem is proposed. The method consists of three main stages which we briefly outline here.

• **Stage 1:** Generate the thesauri

Within this stage from the mouse ontology $\boldsymbol{O_1}$ its thesaurus $\boldsymbol{Th_1}$ is built, and analogically from the zebrafish ontology $\boldsymbol{O_2}$ its thesaurus $\boldsymbol{Th_2}$ is built.

• **Stage 2:** Align the two input ontologies to the three knowledge sources

Within this stage each of the two input ontologies $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ is aligned to each of the three knowledge sources available $\boldsymbol{T_1 = UMLS}$, $\boldsymbol{T_2 = FMA}$, $\boldsymbol{T_3 = WordNet}$. In fact, within this stage not the ontologies themselves, but the thesauri $\boldsymbol{Th_1}$ and $\boldsymbol{Th_2}$ that have been generated from them, are aligned to the three external knowledge sources. Still, we usually say that the two input ontologies are aligned to the three external knowledge sources.

• **Stage 3:** Infer cross-ontology synonymy links/relations, and cross-ontology parent-child (is-a/part-of) links/relations.

This stage consists of three phases which we outline here.

• **Phase 3.1:** Synonymy links are drawn for syntactic or direct matches between the terms from $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$. This is what we denote as the ***direct matching (DM)*** procedure.

• **Phase 3.2:** Using the results from Stage 2 (the alignments performed there) and a set of simple logical rules, cross-ontology synonyms are predicted. This is what we call the ***source matching predictions (SMP)*** procedure.

• **Phase 3.3:** For pairs of terms $\boldsymbol{t_1} \in \boldsymbol{V_1}$ and $\boldsymbol{t_2} \in \boldsymbol{V_2}$ for which no synonymy relation has been discovered so far, the relations hereto predicted between $\boldsymbol{t_1}$'s and $\boldsymbol{t_2}$'s children are used, in order to infer additional predictions about how $\boldsymbol{t_1}$ and $\boldsymbol{t_2}$ are related. That's what we call the ***child matching predictions (CMP)*** procedure. This procedure infers new predictions about relations which seem to exist between $\boldsymbol{t_1}$ and $\boldsymbol{t_2}$ even though these relations don't directly originate from the knowledge contained in the three external knowledge sources.

In the next subsections, the three stages which were only briefly outlined here, are described in more details.

4.1. Stage 1 – Generating the thesauri. The thesauri $\boldsymbol{Th_1}$ of $\boldsymbol{O_1}$ and $\boldsymbol{Th_2}$ of $\boldsymbol{O_2}$ are simple dictionary-like tabular structures. For the id of any term $\boldsymbol{t} \in \boldsymbol{V_1}$, the thesaurus $\boldsymbol{Th_1}$ maintains a list $\boldsymbol{Th_1[t.id]}$ that contains the primary name and all the alternative names (if any) of the term $\boldsymbol{t} \in \boldsymbol{V_1}$ with identifier $\boldsymbol{id}$. In the same way, for the id of any term $\boldsymbol{t} \in \boldsymbol{V_2}$, the thesaurus $\boldsymbol{Th_2}$ maintains a list $\boldsymbol{Th_2[t.id]}$ containing the primary name and the alternative names (if any) of the term $\boldsymbol{t} \in \boldsymbol{V_2}$ with identifier $\boldsymbol{id}$. The lists $\boldsymbol{Th_i[t.id]}$ $\boldsymbol{(i = 1, 2)}$ are simple lists of strings. Their members are all the names (as defined by the input ontologies) of the term $\boldsymbol{t}$ with the given identifier $\boldsymbol{id}$. Building the thesauri from the two input ontologies is a fairl straightforward process.

4.2. Stage 2 – Aligning the input ontologies to the knowledge sources. In this stage each of the two input ontologies (each of the two thesauri) is aligned to the three external knowledge sources. Below is described how the ontology $\boldsymbol{O_1}$ (i.e., its thesaurus $\boldsymbol{Th_1}$) gets aligned to the external knowledge source $\boldsymbol{T_1}$. The other alignments are performed analogically.

<u>**Phase 2.1:**</u> For each term id $\boldsymbol{k} \in \boldsymbol{O_1}$ do → get the list $\boldsymbol{L} = \boldsymbol{Th_1[k]}$ from the pre-built thesaurus $\boldsymbol{Th_1}$.

<u>**Phase 2.2:**</u> For each term name $\boldsymbol{s} \in \boldsymbol{L}$ do → get from $\boldsymbol{T_1}$ all distinct ids ($\boldsymbol{T_1}$'s term ids) which correspond to the term name $\boldsymbol{s}$, i.e., get

$$\boldsymbol{RS_1 = \{(t^I.id) \mid t^I \in T_1 \text{ and } t^I.name = s\}}$$

<u>Step 2.2.1:</u> For each id from $\boldsymbol{RS_1}$ do → get from $\boldsymbol{T_1}$

$$\boldsymbol{RS_2 = \{(t^{II}.id, t^{II}.name) \mid t^{II} \in T_1 \text{ and } t^{II}.id = t^I.id\}}$$

Having performed this step, the result is that the synonyms of $\boldsymbol{s}$ (as they are defined by $\boldsymbol{T_1}$) are now known. These are also denoted as the $\boldsymbol{T_1}$-synonyms of $\boldsymbol{s}$ – technically this is the set

$$\boldsymbol{RS_2^* = \{t^{II}.id \mid t^{II} \in T_1 \; and \; t^{II}.id = t^I.id\}}$$

composed of the first components of the ordered couples contained in $\boldsymbol{RS_2}$.

Step 2.2.2: For each id from $\boldsymbol{RS_1}$ do → get from $\boldsymbol{T_1}$ the set

$$\boldsymbol{RS_3 = \{(t^{III}.id) \mid t^{III} \in T_1 \; and \; ((t^{III}, t^I) \in R_{T_1}^{is\_a} \; or \; (t^{III}, t^I) \in R_{T_1}^{part\_of})\}}$$

Having performed this step, the result is that the followings sets are now known

• the meronyms of $\boldsymbol{s}$ as defined by $\boldsymbol{T_1}$. These are also called the $\boldsymbol{T_1}$-meronyms of $\boldsymbol{s}$ – technically this is the set

$$\boldsymbol{RS_{3,1} = \{t^{III}.id \mid (t^{III}, t^I) \in R_{T_1}^{part\_of}\}}$$

• the hyponyms of $\boldsymbol{s}$ as defined by $\boldsymbol{T_1}$. These are also called the $\boldsymbol{T_1}$-hyponyms of $\boldsymbol{s}$ – technically this is the set

$$\boldsymbol{RS_{3,2} = \{t^{III}.id \mid (t^{III}, t^I) \in R_{T_1}^{is\_a}\}}$$

Is should be noted that $\boldsymbol{RS_{3,1} \cup RS_{3,2} = RS_3}$, $\boldsymbol{RS_{3,1} \cap RS_{3,2} = \emptyset}$.

Step 2.2.3: For each id from $\boldsymbol{RS_1}$ do → get from $\boldsymbol{T_1}$ the set

$$\boldsymbol{RS_4 = \{(t^{IV}.id) \mid t^{IV} \in T_1 \; and \; ((t^I, t^{IV}) \in R_{T_1}^{is\_a} \; or \; (t^I, t^{IV}) \in R_{T_1}^{part\_of})\}}$$

Having performed this step, the result is that the following sets are now known

• the holonyms of $\boldsymbol{s}$ as defined by $\boldsymbol{T_1}$. These are also called the $\boldsymbol{T_1}$-holonyms of $\boldsymbol{s}$ – technically this is the set

$$\boldsymbol{RS_{4,1} = \{t^{IV}.id \mid (t^I, t^{IV}) \in R_{T_1}^{part\_of}\}}$$

• the hypernyms of $\boldsymbol{s}$ as defined by $\boldsymbol{T_1}$. These are also called the $\boldsymbol{T_1}$-hypernyms of $\boldsymbol{s}$ – technically this is the set

$$\boldsymbol{RS_{4,2} = \{t^{IV}.id \mid (t^I, t^{IV}) \in R_{T_1}^{is\_a}\}}$$

Again, it should be noted that $\boldsymbol{RS_{4,1} \cup RS_{4,2} = RS_4}$, $\boldsymbol{RS_{4,1} \cap RS_{4,2} = \emptyset}$.

To summarize all this in plain words—the steps 2.2.1, 2.2.2, and 2.2.3 find in the external knowledge source $\boldsymbol{T_1}$ the following sets of $\boldsymbol{T_1}$-terms:

• Step 2.2.1—synonyms of the original ontology-defined term $\boldsymbol{s}$ with id $\boldsymbol{k}$;

• Step 2.2.2—meronyms and hyponyms of the original ontology term $\boldsymbol{s}$ with id $\boldsymbol{k}$;

• Step 2.2.3—holonyms and hypernyms of the original ontology term $\boldsymbol{s}$ with id $\boldsymbol{k}$;

These steps complete the process of aligning the input ontology $\boldsymbol{O_1}$ (i.e., its thesaurus $\boldsymbol{Th_1}$) to the external knowledge source $\boldsymbol{T_1}$. Then, in an identical manner, $\boldsymbol{O_2}$ is aligned to $\boldsymbol{T_1}$. Finally $\boldsymbol{O_1}$ and $\boldsymbol{O_2}$ are separately aligned to $\boldsymbol{T_2}$ and $\boldsymbol{T_3}$ (i.e., four more alignments are performed) by applying the exact same procedure as described here.

4.3. Stage 3 – Inferring cross-ontology synonymy and cross-ontology parent-child (is-a and part-of) links/relations. In this stage three separate algorithmic procedures are applied, which are denoted as $\boldsymbol{DM}$, $\boldsymbol{SMP}$ and $\boldsymbol{CMP}$. They are described here in full details.

Phase 3.1: Within this phase (called **DM**) textual/syntactical/direct matches/predictions for cross-ontology synonyms are found by checking for textual identities between the term names in the two ontologies. This procedure is straightforward, the algorithm just iterates through all terms $\boldsymbol{t_1} \in \boldsymbol{V_1}$ and $\boldsymbol{t_2} \in \boldsymbol{V_2}$ and tests if $\boldsymbol{t_1.name = t_2.name}$. Whenever such matches are found, the terms $\boldsymbol{t_1}$ and $\boldsymbol{t_2}$ are marked as synonyms and it is noted (memorized) that this synonymy prediction comes from direct matching (DM).

Here is a simple example: In $\boldsymbol{O_1}$ (the mouse anatomy ontology) there exists a term $\boldsymbol{t_1 = (id = "MA0000168", name = "brain")}$, while in $\boldsymbol{O_2}$ (the zebrafish anatomy ontology) there exists a term $\boldsymbol{t_2 = (id = "ZFA0000008", name = "brain")}$. So by doing the checks in this step, it is easily found that their names are identical ("brain") and so these terms are marked as cross-ontology synonyms coming from **DM**.

Phase 3.2: Within this phase (called **SMP**) more predictions are inferred for synonymy links and for parent-child links between the terms of the two input ontologies. As the two input ontologies have already been aligned to the external knowledge sources available, a set of logical rules is applied which results in inferring/predicting what is called source matching synonymy and source matching parent-child (**is-a** and **part-of**) predictions. The rules applied in this phase are as follows.

Rule A: If two terms $t_M \in O_1$ and $t_Z \in O_2$ have been detected as synonyms of the same term $t \in T_i$ (by step 2.2.1) we mark t_M and t_Z as a predicted (by SMP) cross-ontology synonyms of each other.

Rule B1: If $t_M \in O_1$ has been detected as synonym of $t \in T_i$ (by 2.2.1) and if term $t_Z \in O_2$ has been detected as (*is-a/part-of*) child of t (by 2.2.2), we mark t_M as a predicted (by SMP) cross-ontology (*is-a/part-of*) parent of t_Z.

Rule B2: If $t_Z \in O_2$ has been detected as synonym of $t \in T_i$ (by 2.2.1) and if term $t_M \in O_1$ has been detected as (*is-a/part-of*) child of t (by 2.2.2), we mark t_Z as a predicted (by SMP) cross-ontology (*is-a/part-of*) parent of t_M.

Rule C1: If $t_M \in O_1$ has been detected as synonym of $t \in T_i$ (by 2.2.1) and if term $t_Z \in O_2$ has been detected as (*is-a/part-of*) parent of t (by 2.2.3), we mark t_M as a predicted (by SMP) cross-ontology (*is-a/part-of*) child of t_Z.

Rule C2: If $t_Z \in O_2$ has been detected as synonym of $t \in T_i$ (by 2.2.1) and if term $t_M \in O_1$ has been detected as (*is-a/part-of*) parent of t (by 2.2.3), we mark t_Z as a predicted (by SMP) cross-ontology (*is-a/part-of*) child of t_M.

By applying the above described rules, a set of cross-ontology relations (synonymy and parent-child) is drawn (established) between the nodes of $\boldsymbol{DAG_1}$ and $\boldsymbol{DAG_2}$ (i.e., between the terms of the two ontologies). These predicted links or relations are said to come from source matching inference (**SMP**) because the evidence of their existence originates from the information stored in the external knowledge sources that are used. It should also be noted that for the so-inferred parent-child links, the information whether these are **is-a** or **part-of** links is also stored. This completes the description of the **SMP** procedure.

Before proceeding with the formal description of phase 3.3 (the so-called child matching predictions (**CMP**) procedure), here is a short recap itulation of what has been done so far. Several new notations and definitions are introduced here which are going to help us in describing the **CMP** procedure (the last phase 3.3 of stage 3).

The two original (input) graphs $\boldsymbol{DAG_1}$ and $\boldsymbol{DAG_2}$ defined above are available. The cross-ontology links which have been inferred so far (in 3.1 – $\boldsymbol{DM}$ and in 3.2 – $\boldsymbol{SMP}$) are also available. Now, the two original graphs together with the links established by $\boldsymbol{DM}$ and $\boldsymbol{SMP}$ can be thought of as one single graph $\boldsymbol{G = (V, E)}$, where $\boldsymbol{V = V_1 \cup V_2}$ and $\boldsymbol{E = S_{IO} \cup S_{DM} \cup S_{SMP}}$, where

- $\boldsymbol{S_{IO}}$ is the set of all inner-ontology links in $\boldsymbol{DAG_1}$ and $\boldsymbol{DAG_2}$;
- $\boldsymbol{S_{DM}}$ is the set of all links inferred in phase 3.1, i.e., by direct matching (DM);
- $\boldsymbol{S_{SMP}}$ is the set of all links inferred in phase 3.2, i.e., by the source matching predictions (SMP) procedure.

The properties of each of these types of links are summarized in the table on Fig. 3.

- The **IO links** are the original links from the two input ontologies. These are always parent-child links and are colored/labeled either with ***is-a*** or with ***part-of***. These are unidirectional links as the parent-child relations are not symmetrical.

Link Type	Synonymy or Parent-child	Color/Label	Symmetry
IO Links	Only parent-child	Either is-a or part-of	Unidirectional
DM Links	Only synonymy	No color/label	Bidirectional
SMP Parent-child Links	Parent-child	Either is-a or part-of	Unidirectional
SMP Synonymy Links	Synonymy	No color/label	Bidirectional

Fig. 3. Links and link types

• The **DM links** are cross-ontology links which by their definition (phase 3.1) are always synonymy links and as such they are colored neither with ***is-a*** nor with ***part-of***. As the synonymy is a symmetrical relation, these are bidirectional links, i.e., we may think of each **DM** link (t_1, t_2) or $t_1 \longleftrightarrow t_2$ as a pair of two links $t_1 \rightarrow t_2$ and $t_1 \leftarrow t_2$.

• The **SMP links** are either parent-child links (steps 2.2.2 and 2.2.3 of phase 2.2) or synonymy links (step 2.2.1 of phase 2.2). As with the **IO** and **DM** links: the **parent-child SMP links** are colored either with ***is-a*** or with ***part-of*** and are unidirectional; the **synonymy SMP links** have no color/label and are bidirectional. All **SMP links** are cross-ontology ones by their definition (steps 2.2.1, 2.2.2, 2.2.3 from phase 2.2 and rules A, B1, B2, C1, C2 from phase 3.2).

All this having been said, the single graph G (as defined above), which has been produced from DAG_1 and DAG_2 by the **DM** and **SMP** procedures, can now be considered.

Phase 3.3: The description of the CMP procedure is what follows next. This description is intermixed with several definitions which allow us to arrive at one final number that we call ***final/aggregated CMP score*** of the ***aggregated CMP link*** that gets drawn between any two nodes $v_1 \in E_1$ and $v_2 \in E_2$ that are involved in certain patterns of connectivity within the graph G.

Definition 1. *Constant* I *– reliability score of an inner-ontology (**IO**) link. Typically* $I = 1$ *but this value could be varied/adjusted if needed.*

Definition 2. *Constant* D *– reliability score of a direct matching (**DM**) link. Typically* $D = 1$ *but this value could be varied/adjusted.*

Definition 3. *Constants* $f(UMLS)$, $f(FMA)$, $f(WordNet)$ *– reliability scores of the three available knowledge sources. We require that:* $0<f(T_i)<1$, *for* $i = 1, 2, 3$.

Definition 4. *Constant* $p \in (0, 1)$ *– the **CMP** score penalty.*

It should be noted here that the **CMP** procedure is a probabilistic-like procedure in the sense that it deals with scores (evidence scores or link scores

or prediction scores) which are all real numbers from the interval **[0, 1]**. The above-defined $\boldsymbol{p}$ constant aims to lower the **scores of the pattern instances** and the **final CMP score** (these two are to be defined later in Definition 9 and Definition 11) due to the sole fact that the predictions/links drawn in this phase 3.3, do not directly originate from the knowledge contained in the three available knowledge sources (UMLS, FMA, WordNet), i.e., due to the fact that the **CMP** links/predictions come from **CMP** and not through the other means described so far (**DM**, **SMP**). This matches the intuitive observation that **CMP links** should be given lower score than links coming from **DM** or **SMP**.

Next, two variable-argument probabilistic-like functions are defined which are denoted as $\boldsymbol{Conj}$ (short for conjunction) and $\boldsymbol{Disj}$ (short for disjunction). They are used for defining the ***individual CMP links*** and their confidence scores.

Definition 5. *The* $\boldsymbol{Conj}$ *function is defined recursively as follows.*

5.1: $Conj(A_1,\ A_2) = A_1 \cdot A_2$

5.2: $Conj(A_1, A_2, \ldots, A_N) = Conj(Conj(A_1, A_2, \ldots, A_{N-1}), A_N)$, *for* $N \geq 3$.

It is required here that all arguments $\boldsymbol{A_k}$ *are within the internal* **[0, 1]**. *It is easy to prove that if this condition holds true, then* $\boldsymbol{Conj(A_1, A_2, \ldots, A_N)}$ *is also within the interval* **[0, 1]** *which means that this recursive definition is logically correct i.e., that there is no problem at the recursive step 5.2. Instead of* $\boldsymbol{Conj(A_1, A_2, \ldots, A_N)}$ *sometimes also* $\boldsymbol{Conj\ \overset{N}{\underset{i=1}{}}\ (A_i)}$ *can be written for short. Note that the* $\boldsymbol{Conj}$ *function models the probability of two or more independent events occurring simultaneously.*

Definition 6. *The* $\boldsymbol{Disj}$ *function is defined recursively as follows.*

6.1: $Disj(A_1,\ A_2) = A_1 + A_2 - A_1 \cdot A_2$

6.2: $Disj(A_1, A_2, \ldots, A_N) = Disj(Disj(A_1, A_2, \ldots, A_{N-1}), A_N)$, *for* $N \geq 3$.

It is required here that all arguments $\boldsymbol{A_k}$ *bewithin the internal* **[0, 1]**. *It is easy to prove that if this condition holds true, then* $\boldsymbol{Disj(A_1, A_2, \ldots, A_N)}$ *is also within the interval* **[0, 1]**, *which means that this recursive definition is logically correct, i.e., that there is no problem at the recursive step 6.2. Instead of* $\boldsymbol{Disj(A_1, A_2, \ldots, A_N)}$ *sometimes also* $\boldsymbol{Disj\ \overset{N}{\underset{i=1}{}}\ (A_i)}$ *can be written for short. Note that the* $\boldsymbol{Disj}$ *function models the probability of at least one of two or more independent events occurring.*

The **CMP** procedure scans the graph $\boldsymbol{G}$ and looks for three different types of patterns of connectivity within $\boldsymbol{G}$. The patterns involve a few (3 or 4) nodes,

some of which are children and some of which are parents. The procedure takes into account all links involved in the pattern and infers an additional/new link (***an individual CMP link***) between the parents involved. This individual CMP link is then assigned a reliability score which (by definition) equals to the score of the particular ***pattern instance*** that is currently being considered.

Finally, for each two parent nodes, the scores of all the pattern instances in which these two parents are involved, are aggregated in order to produce one ***final/aggregated CMP link*** and its score which we call ***final CMP score*** of the two parent nodes (terms) which are under consideration. The three different types of patterns are presented in the three figures that follow.

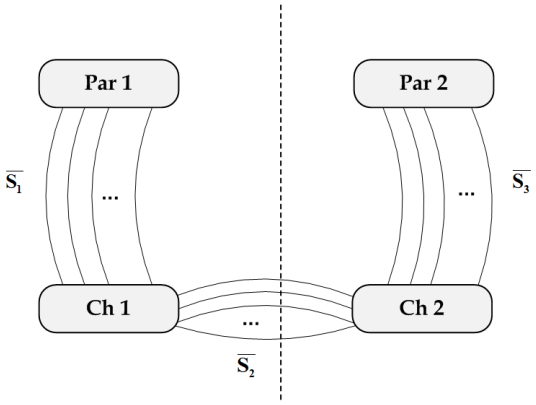

Fig. 4. The U-Pattern

In the figures above, the following notations have been used:

1) the vertical dashed line marks the boundary between the two input ontologies: what's on the left of the line belongs to $\boldsymbol{O_1}$, what's on the right of the line belongs to $\boldsymbol{O_2}$; the solid lines represent edges/links from the graph $\boldsymbol{G}$;

2) the solid lines crossing the dashed line are cross-ontology edges/links, they were inferred either by **DM** or by **SMP**; the solid lines not crossing the dashed line are **IO** links which were there in the two original/input graphs $\boldsymbol{DAG_1}$ and $\boldsymbol{DAG_2}$;

3) $\boldsymbol{\overline{S_1},\ \overline{S_2},\ \overline{S_3}}$ are sets of links from the graph $\boldsymbol{G}$ (it should be noted that the links from $\boldsymbol{G}$ are also called ***supporting evidences***); so $\boldsymbol{\overline{S_1},\ \overline{S_2},\ \overline{S_3}}$ are sets of supporting evidences;

4) $\boldsymbol{\overline{S_i} = \{s_{i1}, s_{i2}, \ldots, s_{im_i}\}}$ for $\boldsymbol{i = 1, 2, 3}$, where $\boldsymbol{s_{ij}}$ is one single link, i.e., one single supporting evidence, and $\boldsymbol{m_i = |S_i|}$ is the count of links in the set $\boldsymbol{S_i}$;

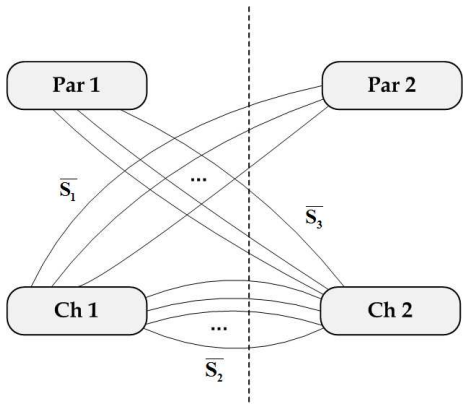

Fig. 5. The X-Pattern

5) $\boldsymbol{Par_1}$ stands for ***parent 1***; $\boldsymbol{Ch_1}$ stands for ***child 1***; $\boldsymbol{v_1 = Par_1 \in V_1}$, $\boldsymbol{Ch_1 \in V_1}$;

6) $\boldsymbol{Par_2}$ stands for ***parent 2***; $\boldsymbol{Ch_2}$ stands for ***child 2***; $\boldsymbol{v_2 = Par_2 \in V_2}$, $\boldsymbol{Ch_2 \in V_2}$;

7) In the U-pattern and in the X-pattern

7.1) $\overline{\boldsymbol{S_2}}$ is a set of cross-ontology synonymy links;

7.2) $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_3}}$ are sets of parent-child links of the same color (i.e., either all links from $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_3}}$ are ***is-a*** links or all links from $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_3}}$ are ***part-of*** links;

8) In the V-pattern

8.1) $\overline{\boldsymbol{S_1}}$ is a set of inner-ontology parent-child links and $\overline{\boldsymbol{S_2}}$ is a set of cross-ontology parent-child links;

8.2) $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_2}}$ are sets of parent-child links of the same color (i.e., either all links from $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_2}}$ are ***is-a*** links or all links from $\overline{\boldsymbol{S_1}}$ and $\overline{\boldsymbol{S_2}}$ are ***part-of*** links.

It is important to note here that the items 7.1, 7.2, 8.1, 8.2 are in fact conditions for the respective connectivity patterns to be considered during the ***CMP*** scan. In other words, these are necessary conditions for those patterns to be processed by the ***CMP*** procedure. If these conditions are not met completely, the respective pattern is not being considered as one of the valid connectivity patterns that ***CMP*** is looking for and so the pattern is not processed at all by the ***CMP*** procedure.

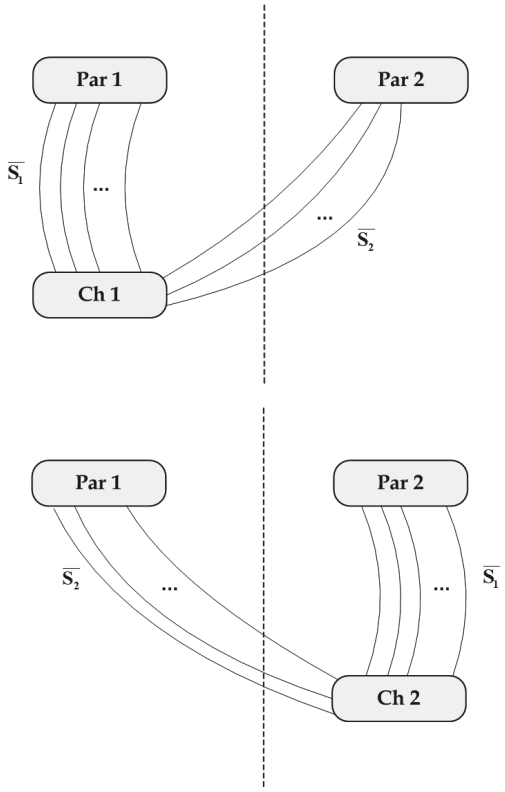

Fig. 6. The two flavors of the V-Pattern

For each of the three kinds of patterns introduced above, the CMP procedure then goes ahead and draws a new ***cross-ontology synonymy CMP link*** between $\boldsymbol{Par_1}$ and $\boldsymbol{Par_2}$. As with ***DM*** and ***SMP*** synonymy links, this one is also a bidirectional link $\boldsymbol{Par_1} \longleftrightarrow \boldsymbol{Par_2}$ (Fig. 7).

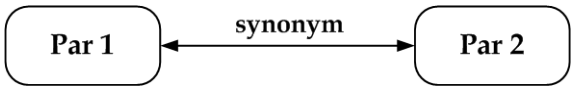

Fig. 7. A bidirectional CMP link

This newly drawn CMP link is called ***an individual CMP link*** between $\boldsymbol{Par_1}$ and $\boldsymbol{Par_2}$. The ***CMP*** procedure assigns a reliability score to the newly drawn individual ***CMP*** link. The score of the newly drawn link is defined as a function of the scores of the already present links from $\boldsymbol{\overline{S_1}}$, $\boldsymbol{\overline{S_2}}$, $\boldsymbol{\overline{S_3}}$.

Definition 7. *The score of an individual (non-CMP) link is defined as follows*

$$\boldsymbol{score}(s_{ij}) = \begin{cases} \boldsymbol{I} & \text{if } \boldsymbol{s_{ij}} \text{ is an IO link,} \\ \boldsymbol{D} & \text{if } \boldsymbol{s_{ij}} \text{ is a DM link,} \\ \boldsymbol{f(src)} & \text{if } \boldsymbol{s_{ij}} \text{ is an SMP link which came from} \\ & \text{the source } \boldsymbol{src} \in \{\boldsymbol{UMLS,\ FMA,\ WordNet}\}. \end{cases}$$

Definition 8. *The score of a set of links (an evidence set* $\overline{\boldsymbol{S_i}}$*) is defined as follows* $\boldsymbol{score}(\overline{\boldsymbol{S_i}}) = \mathbf{Disj}\ _{\boldsymbol{j=1}}^{\boldsymbol{m_i}}\ (\boldsymbol{score}(\boldsymbol{s_{ij}}))$, $\boldsymbol{i = 1, 2, 3}$, *where* $\boldsymbol{Disj}$ *is the function from Definition* 6.

Having these two definitions in place, the score of the ***individual CMP link*** and also of the ***pattern instance*** which produced it is given by the following definition.

Definition 9. *The score of a* **pattern instance** *(of* **an individual CMP link***) is defined as follows*

9.1: *For U-patterns and X-patterns*

$\boldsymbol{score(ptrn) = Conj(score(\overline{S_1}),\ score(\overline{S_2}),\ score(\overline{S_3}),\ p)}$

9.2: *For V-patterns*

$\boldsymbol{score(ptrn) = Conj(score(\overline{S_1}),\ score(\overline{S_2}),\ p)}$

Here $\boldsymbol{Conj}$ *is the function from Definition 5;* $\boldsymbol{p}$ *is the* **CMP** *penalty constant introduced in Definition 4;* $\boldsymbol{ptrn}$ *is one particular instance (one particular occurrence) of the pattern (either* $\boldsymbol{U}$ *or* $\boldsymbol{V}$ *or* $\boldsymbol{X}$*) within the graph* $\boldsymbol{G}$.

Having defined the score of ***an individual CMP link***, it should now be considered that two nodes $\boldsymbol{v_1 = Par_1} \in \boldsymbol{V_1}$ and $\boldsymbol{v_2 = Par_2} \in \boldsymbol{V_2}$ may be (and usually are) involved in many pattern instances/occurrences discovered by the ***CMP*** procedure. So in such cases, many individual synonymy ***CMP*** links $\boldsymbol{e_1, e_2, \ldots, e_L}$ will be drawn between $\boldsymbol{Par_1}$ and $\boldsymbol{Par_2}$ by the ***CMP*** procedure. As defined, the scores of these CMP links are equal to the scores of the pattern instances which they originate from (Definition 9).

The goal now is to get rid of these multiple ***individual CMP links*** between $\boldsymbol{v_1 = Par_1}$ and $\boldsymbol{v_2 = Par_2}$ and to replace them with one ***final/aggregated CMP link*** denoted by $\boldsymbol{e_{CMP}(v_1, v_2)}$ between the nodes $\boldsymbol{Par_1}$ and $\boldsymbol{Par_2}$. To achieve this, the only thing left to do is to provide a way to aggregate the scores of $\boldsymbol{e_1, e_2, \ldots, e_L}$ and to replace these links and their scores with the ***final/aggregated CMP link*** $\boldsymbol{e_{CMP}(v_1, v_2)}$ and with its score. This is done by Definition 11, which is given below.

Definition 10. *Let us have* $\boldsymbol{MAX}$ *denote the maximal of* $\boldsymbol{N}$ *given real numbers* $\boldsymbol{N \geq 1}$.

Definition 11. *Let* $\boldsymbol{v_1 = Par_1 \in V_1}$ *and* $\boldsymbol{v_2 = Par_2 \in V_2}$ *be two terms from the two input ontologies. Let* $\boldsymbol{G}$ *be the graph produced from* $\boldsymbol{DAG_1}$ *and* $\boldsymbol{DAG_2}$ *after all the* **DM** *and* **SMP** *links have been inferred/generated (by phases 3.1 and 3.2 of stage 3). Let also:*

11.1: $\boldsymbol{u = \{u_1, u_2, \ldots, u_{N_u}\}}$ *be the set of all U-patterns in which* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *are involved as parents;* $\boldsymbol{N_u \geq 0}$*;*

11.2: $\boldsymbol{x = \{x_1, x_2, \ldots, x_{N_x}\}}$ *be the set of all X-patterns in which* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *are involved as parents;* $\boldsymbol{N_x \geq 0}$*;*

11.3: $\boldsymbol{w = \{w_1, w_2, \ldots, w_{N_w}\}}$ *be the set of all V-patterns in which* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *are involved as parents;* $\boldsymbol{N_w \geq 0}$*;*

11.4: $\boldsymbol{PIS(v_1, v_2) = u \cup x \cup w}$ *(here* $\boldsymbol{PIS}$ *denotes the pattern instance set for the nodes* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$*, i.e., the set of all patterns instances in which* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *are involved as parents);*

11.5: $\boldsymbol{|PIS(v_1, v_2)| = N_u + N_x + N_w > 0}$.

The number defined by $\boldsymbol{score_{CMP}(v_1, v_2) = MAX}$ $_{\forall p \in PIS(v_1, v_2)}$ $(score(p))$ *is what we call the* **final/aggregated CMP score** *for the terms* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$*. Here* $\boldsymbol{p}$ *stands for pattern, i.e., the* $\boldsymbol{MAX}$ *is taken over all patterns which* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *are involved in (as parents). This is the* **final CMP score** *of the* **final/aggregated CMP link** $\boldsymbol{e_{CMP}(v_1, v_2)}$ *that is drawn between the nodes* $\boldsymbol{v_1}$ *and* $\boldsymbol{v_2}$ *as a final result of the* **CMP** *procedure.*

This definition completes our description of phase 3.3 (the ***CMP*** procedure) from stage 3. It was shown how ***individual CMP synonymy links*** (which are cross-ontology links) can be drawn between two terms $\boldsymbol{v_1 \in V_1}$ and $\boldsymbol{v_2 \in V_2}$. The reliability score for each of these individual CMP links was defined. The multiple individual CMP links were aggregated and one ***final/aggregated CMP link*** was drawn between the two nodes $\boldsymbol{v_1 \in V_1}$ and $\boldsymbol{v_2 \in V_2}$. At the end, one final number $\boldsymbol{score_{CMP}(v_1, v_2)}$ called ***final CMP score*** was defined as score of the ***final/aggregated CMP link*** $\boldsymbol{e_{CMP}(v_1, v_2)}$.

This completes the description of the whole ontology mapping algorithm composed of the three procedures ***DM***, ***SMP***, and ***CMP***.

5. Summary and discussion. In this paper we have presented an integrated algorithmic solution for the problem of mapping the anatomical ontologies of two distinct species/organisms. The two ontologies were modeled as two ***DAGs*** with their edges colored in different colors based on the inner-ontology relations that these edges represent. Several external knowledge sources have been used as references during the process of mapping the two input anatomical ontologies.

Three separate algorithmic procedures have been utilized – ***DM***, ***SMP***, ***CMP*** – listed here from simplest to most complex, which run on the two given ***DAGs*** and predict cross-ontology links between them. The ***DM*** procedure doesn't consult any external knowledge sources but uses information that is purely internal with respect the two input ontologies. ***DM*** predicts synonymy links/relations only. The ***SMP*** is the procedure which consults the external knowledge sources in order to predict various semantic cross-ontology links/relations between the two input ontologies (synonyms, hypernyms, hyponyms, holonyms, meronyms). The ***CMP*** procedure then uses the outputs from ***DM*** and ***SMP*** (i.e., the cross-ontology links generated by them) and infers additional cross-ontology links/relations that hadn't yet been discovered either by ***DM*** or by ***SMP***.

The CMP procedure is based on three ***patterns of connectivity*** (denoted as ***U***, ***X***, and ***V***) within the graph produced after ***DM*** and ***SMP*** have finished their execution, and a ***probabilistic scoring scheme*** based on the ***Conj*** and ***Disj*** functions defined in this paper. These two functions model the probabilities of: (i) several independent events occurring at the same time (Conj), and (ii) at least one of several independent events occurring (Disj). These functions were chosen for two reasons: (1) for the purposes of this work the three external knowledge sources were considered independent; (2) the choice of Conj and Disj in the way described in this paper aligns well with the general theory of weighted graphs (in which edge weights represent probabilities) and of weighting routes/paths in such graphs.

Further improvements and extensions of the algorithmic procedures presented in this paper can be made in at least in three different directions: (1) as noted in [26] further improvements of the scoring scheme (defined in this paper) are possible by amending the functions ***Conj*** and ***Disj*** (from Definitions 5 and 6), and the aggregation function ***MAX*** (used in Definition 11). These amendments turn out to be useful because assuming that the three external knowledge sources are independent, is not the most flexible and realistic approach; in reality the external knowledge sources do have certain dependencies (between each

other) but evaluating those is a complex matter; (2) special care needs to be taken to ensure that the graph produced after applying the DM, SMP, and CMP procedures contains no cycles (is a DAG too) but this might require an additional cycle elimination algorithm or the need to involve a curator (a human, an anatomy specialist) at that point; (3) the connectivity patterns that ***CMP*** looks for may be extended to span not just across child nodes but also across grand-child and grand-grand-child nodes in the graphs representing the two ontologies. More generally the CMP procedure may look for $\boldsymbol{k_1}$ levels above the current node, and $\boldsymbol{k_2}$ levels below the current node (topological sort assumed on the DAG), i.e., the procedure may consider all nodes which fall within that $[\boldsymbol{-k_1, k_2}]$ node range around the current node. If we denote that generalized ***CMP*** procedure as $[\boldsymbol{-k_1, k_2}]$-***CMP*** then it is logical to expect that the generalized ***CMP*** would be more sensitive and more local context aware than the standard ***CMP*** (the $[\mathbf{1, 1}]$-***CMP***) which was described in the current paper. Still, if that extended $[\boldsymbol{-k_1, k_2}]$-***CMP*** is to be used, special care has to be taken for making sure that not too much noise is introduced in the generated predictions.

REFERENCES

[1] de Bruijn J. et al. Ontology Mediation, Merging, and Aligning. In: Semantic Web Technologies J. (Eds J. Davies, R. Studer, P. Warren). John Wiley and Sons, 2006, 95–113.

[2] Kalfoglou Y., M. Schorlemmer. Ontology mapping: the state of the art. *The Knowledge Engineering Review*, **18** (2003), No 1, 1–31.

[3] Zlatareva N., M. Nisheva. Alignment of Heterogeneous Ontologies: A Practical Approach to Testing for Similarities and Discrepancies. In: Proceedings of the Twenty-First International Florida Artificial Intelligence Research Society Conference (Eds D. Wilson, H. Chad Lane), Coconut Grove, Florida, May 15–17, 2008, ISBN 978-1-57735-365-2, AAAI Press, Menlo Park, California, 2008, 365–370.

[4] Nisheva–Pavlova M. Mapping and Merging Domain Ontologies in Digital Library Systems. In: Proceedings of the Fifth International Conference on Information Systems and Grid Technologies, Sofia, May 27–28, 2011, ISSN 1314-4855, Sofia, St. Kliment Ohridski Uiversity Press, 2011, 107–113.

[5] Gruber T. A translation approach to portable ontologies. *Knowledge Acquisition*, **5** (1993), No 2, 199–220.
http://ksl-web.stanford.edu/KSL_Abstracts/KSL-92-71.html

[6] Grenon P., B. Smith, L. Goldberg. Bio Dynamic ontology: applying BFO in the biomedical domain. In: Ontologies in Medicine. Studies in Health technology and Informatics (Ed. P. M. Pisannelli), Amsterdam, IOS Press 102, 20–38.

[7] Petrov P., M. Krachounov, E. Todorovska, D. Vassilev. AnatOM – An in silico solution for merging anatomical ontologies. Biotechnology and biotechnological equipment, June 2012.

[8] Maedche A., B. Motik, N. Silva, R. Volz. MAFRA — a mapping framework for distributed ontologies. In: Proceedings of the 13th European Conference on Knowledge Engineering and Knowledge Management EKAW-2002, Madrid, Spain.

[9] Omelayenko B. RDFT: A mapping meta-ontology for business integration. In: Proceedings of the Workshop on Knowledge Transformation for the Semantic Web (KTSW 2002), 15th European Conference on Artificial Intelligence, Lyon, France, 76–83.

[10] Kalfoglou Y., M. Schorlemmer. IF-Map: an ontology mapping method based on Information Flow theory. *Journal on Data Semantics*, **1** (2003), No 1, ISSN 3-540-20407-5, 98–127.

[11] Noy N. F., M. A. Musen. PROMPT: Algorithm and tool for automated ontology merging and alignment. In: Proceedings of the 17th National Conference On Artificial Intelligence (AAAI-2000), Noy NF, Musen, MA 2000, 450–455.

[12] Dou D., McDermott D., Qi P. Ontology translation by ontology merging and automated reasoning. In: Proceedings of the EKAW 2002 Workshop on Ontologies for Multi-Agent Systems, Siguenza, Spain, 30 September 2002, 3–18.

[13] Noy N. F., M. A. Musen. Anchor-PROMPT: Using non-local context for semantic matching. In: Proceedings of the Workshop on Ontologies and Information Sharing at the Seventeenth International Joint Conference on Artificial Intelligence (IJCAI-2001), Seattle, WA, USA, 2001, 63–70.

[14] Doan A., J. Madhaven, P. Domingos, A. Halevy. Ontology matching: A machine learning approach. In: Handbook on Ontologies in Information Systems(Eds S. Staab, R. Studer), Springer-Verlag, 2004, 397–416.

[15] Ehrig M., S. Staab. QOM—quick ontology mapping. In: Proceedings of the Third International SemanticWeb Conference (Eds F. van Harmelen, S. McIlraith, D. Plexousakis), Hiroshima, Japan, LNCS, Springer, Vol. **3298**, 683–696.

[16] Ehrig M., Y. Sure. Ontology mapping—an integrated approach. In: Proceedings of the First European Semantic Web Symposium (ESWS 2004), Heraklion, Greece, Lecture Notes in Computer Science, Springer-Verlag, Vol. **3053**, 76–91.

[17] Giunchiglia F., P. Shvaiko. Semantic matching. *The Knowledge Engineering Review*. **18** (2004), No 3, 265–280.

[18] Giunchiglia F., P. Shvaiko, M. Yatskevich. S-Match: an algorithm and an implementation of semantic matching. In: Proceedings of the First European Semantic Web Symposium (ESWS 2004), Heraklion, Greece, Lecture Notes in Computer Science, Springer-Verlag, Vol. **3053**, 61–75.

[19] de Bruijn J., F. Martín-Recuerda, D. Manov, M. Ehrig. (2004) State-of-the-art survey on Ontology Merging and Aligning V1.

[20] Miller G. A. WordNet: A Lexical Database for English. *Communications of the ACM*, **38** (1995), No 11, 39–41.

[21] Fellbaum Ch. WordNet: An Electronic Lexical Database. Cambridge, MA: MIT Press, 1998.

[22] Bodenreider O. The Unified Medical Language System (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, **32** (2004), 267–270.

[23] Rosse C., J. L. Mejino, Jr. A reference ontology for biomedical informatics: the Foundational Model of Anatomy. *J Biomed Inform.*, **36** (2003), No 6, 478–500.

[24] Smith B., M. Ashburner, C. Rosse, C. Bard, W. Bug, W. Ceusters, L. J. Goldberg, K. Eilbeck, A. Ireland, C. J. Mungall. The OBI Consortium, N. Leontis, P. Rocca-Serra, A. Ruttenberg, S.-A. Sansone, R. H. Scheuermann, N. Shah, P. L. Whetzel, S. Lewis. The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, **25** (2007), 1251–1255.

[25] Day-Richter J. OBO Flat File Format Specification, version 1.2. `http://www.geneontology.org/GO.format.obo-1_2.shtml`, 2006

[26] Petrov, P., M. Krachounov, O. Kulev, M. Nisheva, D. Vassilev (2012) Predicting and Scoring Links in Anatomical Ontology Mapping. Submitted for presentation at the ISAR International E-Conference on Information Technology (June 11–15, 2012).

[27] McGuinness D. L., F. van Harmelen (Eds). OWL Web Ontology Language Overview, W3C recommendation.
http://www.w3.org/TR/owl-features/, 2004

[28] D. Brickley, R. V. Guha (Eds). RDF Vocabulary Description Language 1.0: RDF Schema, W3C recommendation.
http://www.w3.org/TR/rdf-schema/, 2004

Peter Petrov
Milko Krachounov
Faculty of Mathematics and Informatics
St. Kliment Ohridski University of Sofia
5, J. Bourchier Blvd
1164 Sofia, Bulgaria
e-mail: p.a.petrov@gmail.com
e-mail: milko@3mhz.net

Ernest van Ophuizen
Laboratory of Bioinformatics
Wageningen University
The Netherlands
e-mail: ernest.vanophuizen@gmail.com

Dimitar Vassilev
Bioinformatics group
Agro Bio Institute
1164 Sofia, Bulgaria
e-mail: jim6329@gmail.com

Received March 26, 2012
Final Accepted June 7, 2012