Biomath Forum

BIOMATH

http://www.biomathforum.org/biomath/index.php/biomath/

Predicting and Scoring Links in Anatomical Ontology Mapping

Peter Petrov*, Milko Krachounov*, Ognyan Kulev*, Maria Nisheva*, Dimitar Vassilev†
*Faculty of Mathematics and Informatics, Sofia University St. Kliment Ohridski
5 James Bourchier Blvd., 1164 Sofia, Bulgaria
†Bioinformatics group, AgroBioInstitute
8 Dragan Tsankov Blvd., 1164 Sofia, Bulgaria
Email: jim6329@gmail.com

Received: 12 July 2012, accepted: 11 November 2012, published: 22 December 2012

***Abstract*—The paper presents a work performed in the area of automatic and semi-automatic ontology mapping. A method for inferring additional cross-ontology links while mapping anatomical ontologies is described and the results of some experiments performed with various external knowledge sources and scoring schemes are discussed as well.**

***Keywords*-ontology; graph; directed acyclic graph; ontology mediation; ontology mapping; ontology merging; scoring scheme; probability; knowledge sharing; knowledge reuse; interoperability**

I. Introduction

The term *ontology* comes from Philosophy and has been applied in Information Systems, Information Retrieval etc. to represent the formalization of a body of knowledge describing a given domain. Ontologies have become increasingly popular because they help to realize many of the most challenging problems in the IT field like interoperability, information/knowledge sharing and knowledge reuse.

Information sources (and ontologies in particular), even from the same problem domain, are usually heterogeneous. In order to enable interoperation between such information sources (ontologies) and to integrate the information/knowledge from multiple sources, one needs to build mappings between ontologies. These mappings establish the semantic correspondence between concepts and relations in different ontologies. As we have noted in [10] there are some terminological differences pertaining to the integration of ontologies within the ontology mapping/merging/matching (OM) community. Those terminological differences are mostly between the terminology adopted in [1] on one side, and in [11] on the other. In our works, we adopt the terminology of [1]. In the sense of [1], *ontology mapping* is the process of taking two input ontologies and generating semantic links between their concepts/terms. The generated links are not part of the two input ontologies; they are stored separately from them. Two other terms are related to ontology mapping: *ontology aligning* and *ontology merging*. Ontology aligning [1] can be viewed as an automatic or semi-automatic ontology mapping; it denotes the process of discovery of cross-ontology links by a computer program. Again, these links are stored separately from the two input ontologies. Ontology merging [1] is the ultimate goal when integrating/mediating two input ontologies; it comes down to taking two input ontologies and generating an output ontology that unifies the knowledge contained in them. It is usually a process which follows the processes of mapping/aligning and which utilizes the intermediate results produced by them; during this process, some pairs of terms (one from each of the two input ontologies) are merged into single nodes of the output ontology, while other input terms are not paired but are just copied unchanged to the output ontology.

This paper discusses some issues in automatic map-

Citation: P Petrov, M Krachounov, O Kulev, M Nisheva, D Vassilev, Predicting and Scoring Links in Anatomical Ontology Mapping, Biomath 1 (2012), 1211117, http://dx.doi.org/10.11145/j.biomath.2012.11.117

ping or aligning of species-specific anatomical ontologies by utilization of various knowledge sources.

II. Problem formulation

Given the anatomical ontologies of two different species (model organisms) e.g. mouse and zebrafish, our goal is to establish semantic links between the terms of the two ontologies such that: (i) these links are of one of the following types: $R_1 =$ ***synonymy***, $R_2 =$ ***hypernymy***, $R_3 =$ ***hyponymy***, $R_4 =$ ***holonymy***, $R_5 =$ ***meronymy***, and (ii) each of these links has some *degree of certainty* or *degree of confidence* or *confidence score* which is a real number in the interval [0, 1]. The semantic relation types R_k that we refer to here are well-known and are widely utilized in the areas of linguistics, knowledge representation and ontology engineering. That is why we don't provide any formal or informal definitions for them here.

The two input ontologies are represented in the form of OBO files. OBO stands for "Open Biomedical Ontology“ and denotes an *ontology language* and an *ontology file format* [2] for defining ontologies. It has been used mostly for defining ontologies in the biomedical domain. Nowadays OBO is adopted by the GO project [2], [3], the OBO Foundry initiative [4], and other communities.

III. Formalization of the problem

In mathematical terms, each of the two input anatomical ontologies can be considered as a directed acyclic graph together with a function colouring the graph's edges. The colours model the relations defined within the input ontologies (like ***is_a*** and ***part_of***) which we call inner-ontology relations. Typically, there are other inner-ontology relations except those two. These additional relations usually pertain to the development of the particular organism and not just to its adult/gross anatomy. Such relations are for example ***start_stage***, ***end_stage***, ***develops_from*** but practically we don't deal with them as we are mainly concerned with the organism's adult/gross anatomy, not with the organism's growth and development. We shall use further the following notation:

$$O_1 = O_M : DAG_1 = (V_1, E_1),$$
$$F_1 : E_1 \to C = \{c_1, c_2, ..., c_n\};$$
$$O_2 = O_Z : DAG_2 = (V_2, E_2),$$
$$F_2 : E_2 \to C = \{c_1, c_2, ..., c_n\}.$$

Here O_1 and O_2 are the two input anatomical ontologies; DAG_1, DAG_2 are their corresponding directed acyclic graphs; V_1 and V_2 are the sets of terms of the two input ontologies (each term has an identifier and a name); E_1 and E_2 are the relations defined within the two input ontologies; F_1 and F_2 are the edge-colouring functions. Two terms u_1 and u_2 are connected with an edge e if and only if the pair of terms (u_1, u_2) belongs to the relation represented by e.

The relations ***is_a*** (specialization/generalization) and ***part_of*** (membership/aggregation) are the two typical examples of inner-ontology relations defined within the ontologies O_1 and O_2. In our notation, we map relations to colours (through F_1 and F_2), and we deal only with two relations (is_a, $part\_of$). So it can be assumed that $n = 2$, $c_1 =$ ***is_a***, $c_2 =$ ***part_of***. Thus, if for example, $u_1 =$ ***"brain"***, $u_2 =$ ***"central nervous system"***, $u_1, u_2 \in V_1$, then there usually exists an edge e between u_1 and u_2 such that $F_1(e) = part\_of$ (because the brain is part of the central nervous system and anatomical ontologies of most organisms usually declare this fact explicitly).

Also given are several (typically large) external knowledge sources which might be either biomedical ones or general-purpose ones. They contain anatomical terms and relations (***is_a***, ***part_of***, others) between their own terms. Three concrete external knowledge sources have been used for the purposes of this work. These are $T_1 =$ ***UMLS***, $T_2 =$ ***FMA***, $T_3 =$ ***WordNet***. ***UMLS*** [5], [14] and ***FMA*** [6], [15] are biomedical knowledge sources, and ***WordNet*** [7], [8], [16] is a general purpose knowledge source. Formally stated, each of these knowledge sources T_s, $s = 1, 2, 3$, contains the following information:

- *Terms.* $M_s = \{t_{s1}, t_{s2}, ..., t_{sm_s}\}$ is the set of terms in the knowledge source T_s. Here $t_{sk} = (id_{sk}; name_{sk})$; id_{sk} is the identifier within T_s of the term t_{sk}; $name_{sk}$ is the textual name within T_s of the term t_{sk}; m_s (usually $10^6 \le m_s \le 10^7$) is the number of terms in the knowledge source T_s.
- *Relations.* These are the ***is_a*** and ***part_of*** relations defined within the external knowledge source T_s:

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

Typically other relations are also defined within the external knowledge source T_s but only these two are relevant to our work.

Each knowledge source $src = T_s$, $s = 1, 2, 3$, is up-front assigned a score $f(src)$ which is based on its preciseness in predicting synonymy and parent-child (***is_a***, ***part_of***) relations between terms of the two input ontologies. Details on this evaluation (of the three knowledge sources that we use) can be found in [9].

Having the notation introduced above, we now seek to find a set of predictions (a set of 4-tuples):

$$D = \{(v_{1k}, v_{2k}, r_k, s_k) \mid k = 1, 2, ..., |D|\},$$

such that $v_{1k} \in V_1, v_{2k} \in V_2, r_k \in \{R_1, R_2, R_3, R_4, R_5\}$ and $s_k \in (0, 1]$. Here, for each k, v_{1k} is a term from the input ontology O_1, v_{2k} is a term from the input ontology O_2, r_k are automatically (i.e. *in silico*) predicted cross-ontology links from one of the five types defined in the previous section, and s_k is a real number denoting the confidence score of the prediction that the terms v_{1k} and v_{2k} are related/linked by a cross-ontology link of the type r_k. Requiring that $s_k \in (0, 1]$, we basically imply that the set D which we seek, is in fact a set of cross-ontology predictions or a set of predicted cross-ontology links between O_1 and O_2 where each score is probabilistic-based (modeling, given the information we have in the input ontologies and also in the available knowledge sources, the probability that the corresponding prediction is actually true).

IV. Algorithmic procedures

Three algorithmic procedures are applied to the graph structures that were described formally in the previous section. Each of them adds more links to the set D that is being sought. These three procedures are detailed in [12], here we mention them only briefly.

Within the first procedure, the two input ontologies are scanned for identity matches between the names of their terms. If $t_1 \in V_1$ and $t_2 \in V_2$ have the same names, they are marked as synonyms predicted by what we call the ***direct matching (DM) procedure***. The cross-ontology links discovered/predicted this way are assigned the highest possible scores of 1.0 as these predictions come from information contained entirely in the two input ontologies.

During the second procedure, using the information (the terms and the relations) in the external knowledge sources, and identity matches between term names of the two input ontologies and term names of the three external knowledge sources, we build a graph model/structure which aligns each of the two input ontologies to each of the three external knowledge sources. This model contains a set of semantic links (of the types R_k, $k = 1, 2, ..., 5$, that were defined above) between the two input ontologies on the one side, and the three external knowledge sources on the other side. Then a set of logical rules is applied, and conclusions are drawn for the semantic relations that exist between terms $t_1 \in V_1$ and $t_2 \in V_2$ of the two input ontologies. The following rules are applied at this stage:

- *Rule (A).* If two terms $t_1 \in V_1$ and $t_2 \in V_2$ have been detected as synonyms of the same term $t \in T_s$, then t_1 and t_2 are marked as predicted cross-ontology synonyms of each other;
- *Rule (B).* If $t_j \in V_j$ has been detected as a synonym of $t \in T_s$ $(s = 1, 2, 3)$, and if the term $t_{3-j} \in V_{3-j}$ has been detected as an $(is\_a/part\_of)$ child/parent of t, then t_j is marked as predicted cross-ontology $(is\_a/part\_of)$ parent/child of t_{3-j} (here $j = 1$ or $j = 2$).

The application of these rules is what we call the ***source matching predictions (SMP) procedure***. Rule (A), when applied, finds the synonymy relations (i.e. the relations of type R_1) between terms from the two input ontologies. Rule (B) is a composite (generalized) version of four separate rules (two options for ***is_a/part_of*** by two options for ***child/parent*** makes four options in total). These four rules which originate from rule (B), when applied, find the ***hypernymy***, ***hyponymy***, ***holonymy*** and ***meronymy*** relations (i.e. the relations of types R_2, R_3, R_4, R_5) between terms of the two input ontologies. All links predicted through SMP are given the score $f(src)$, where src is the knowledge source confirming/implying these predictions.

Finally, we run a procedure that we denote as the ***child matching predictions (CMP) procedure***. This one tries to find R_1, R_2, R_3, R_4 and R_5 links between terms of the two input ontologies, $t_1 \in V_1$ and $t_2 \in V_2$, for which no links have been predicted either by DM or by SMP. The approach CMP takes is to consider patterns of cross-ontology connectivity (found by DM and SMP) between $t_1 \in V_1$ (parent term 1), $t_2 \in V_2$ (parent term 2), and the child terms of the two parent terms t_1 and t_2. Three separate patterns of connectivity are considered by CMP:

(i) $t_1 \in V_1 \longleftarrow t_{ch1} \in V_1 \longleftrightarrow t_{ch2} \in V_2 \longrightarrow t_2 \in V_2$
(we call this an **U Pattern**);

(ii) $t_1 \in V_1 \longleftarrow t_{ch2} \in V_2 \longleftrightarrow t_{ch1} \in V_1 \longrightarrow t_2 \in V_2$
(we call this an **X Pattern**);

(iii) $t_1 \in V_1 \longleftarrow t_{ch1} \in V_1 \longrightarrow t_2 \in V_2$ or
$t_1 \in V_1 \longleftarrow t_{ch2} \in V_2 \longrightarrow t_2 \in V_2$
(we call these two patterns **V Patterns**).

In this notation, the $\longrightarrow$ and $\longleftarrow$ arrows denote sets of non-CMP parent-child links (the arrows always point from child to parent). These are asymmetrical links. The $\longleftrightarrow$ arrows denote sets of non-CMP synonymy links These are symmetrical links. The t_{ch1} and t_{ch2} are child terms from the two input ontologies. Each occurrence of any of these patterns between t_1 and t_2 (the two parent terms) we call a pattern instance. All arrows within one pattern instance represent either ***is_a*** or ***part_of*** links (we don't allow mixing these two within a single pattern instance).

Based on these patterns of connectivity, new cross-ontology links (CMP links) are introduced (one CMP link per pattern instance) between t_1 and t_2. We call these links individual CMP links. To assign scores to the individual CMP links, the concepts ***score of a set of non-CMP links between two terms*** and ***score of a pattern instance (or score of an individual CMP link)*** are defined below. Also, we introduce two functions, ***Conj*** and ***Disj***, with $N \geq 2$ parameters each, which, provided that the probabilities $p_1, p_2, ..., p_N$ of N events are given, define the probabilities of (i) all these events occurring at the same time (***Conj***), and (ii) at least one of these events occurring (***Disj***). We call the ***Conj*** and ***Disj*** functions ***accumulation functions*** as they accumulate scores of non-CMP links to produce a score of an individual CMP link. Finally, all individual CMP links between t_1 and t_2 are aggregated through what we call an **aggregation function** (which can be e.g. the max of $N \geq 1$ numbers). Next, we define in some more detail the concepts which we just introduced in relation to CMP.

Definition 1 (*Conj*): ***Conj*** is a function which takes N arguments (each of them in [0, 1]) and returns a result in [0, 1]. We discuss a possible implementation for it below.

Definition 2 (*Disj*): ***Disj*** is a function which takes N arguments (each of them in $[0, 1]$) and returns a result in $[0, 1]$. We discuss possible implementations for it below.

Definition 3 (score of a non-CMP link): The score of a non-CMP link between any two terms (which could be from the same ontology or not) is defined as follows:

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

Here IO stands for *inner-ontology*, DM stands for *direct matching* and SMP stands for *source matching predictions*; s_{ij} is one single non-CMP link (i.e. one single evidence); the $\boldsymbol{I}$ and $\boldsymbol{D}$ are constants (typically having the values of 1.0).

Definition 4 (score of a set of non-CMP links): The score of a set of non-CMP links (score of an evidence set) is defined as follows:

$$score(\overline{S_i}) = \boldsymbol{Disj}_{k=1}^{m}(score(s_{ik})),$$

where ***Disj*** is the function from Definition 2, s_{ik} are non-CMP (i.e. either IO or DM or SMP) links, and the ***Disj*** is taken over all non-CMP links taking part in the evidence set $\overline{S_i}$.

Definition 5 (score of an individual CMP link): The score of an individual CMP link e is defined as:

$$score(e) = p \cdot \boldsymbol{Conj}_{i=1}^{n}(score(\overline{S_i})),$$

where $p \in [0, 1]$ is a CMP penalty constant, ***Conj*** is the function from Definition 1, and the ***Conj*** is taken over all evidence sets $\overline{S_i}$ that take part in the pattern instance, which the link e originates from (note that $n = 2$ for the **V** patterns and $n = 3$ for the **X** and **U** patterns).

Definition 6 (aggregation function): Let $\boldsymbol{K}$ be the number of all individual CMP links drawn between $t_1 \in V_1$ and $t_2 \in V_2$. An aggregation function is a known function F_{agg} which takes the scores of all these $\boldsymbol{K}$ individual CMP links and produces a single number $P_{CMP}(t_1, t_2) \in [0, 1]$, which we call *score of the aggregated (final) CMP link* drawn between t_1 and t_2.

As a final result from the CMP procedure, this aggregated CMP link is drawn between any two terms t_1 and t_2 for which at least one pattern (of any of the three types **X**, **U**, **V**) is found. The score of this link is calculated in the way shown above.

V. Comparison of alternative scoring schemes

We have produced several distinct scoring schemes by varying the functions ***Conj***, ***Disj*** and ***Fagg*** which were defined above.

- **Scheme #1:**

(1a) $\boldsymbol{Conj}(s_1, s_2) = s_1 s_2$;
$\boldsymbol{Conj}(s_1,s_2,...,s_N) = \boldsymbol{Conj}(\boldsymbol{Conj}(s_1,s_2,...,s_{N-1}),s_N)$
(1b) $\boldsymbol{Disj}(s_1, s_2) = s_1 + s_2 - s_1 s_2$;
$\boldsymbol{Disj}(s_1,s_2,...,s_N) = \boldsymbol{Disj}(\boldsymbol{Disj}(s_1,s_2,...,s_{N-1}),s_N)$
(1c) $\boldsymbol{Fagg}(s_1, s_2, ..., s_N) = \max(s_1, s_2, ..., s_N)$

- **Scheme #2:**

(2a) $\boldsymbol{Conj}(s_1, s_2) = s_1 s_2$;
$\boldsymbol{Conj}(s_1,s_2,...,s_N) = \boldsymbol{Conj}(\boldsymbol{Conj}(s_1,s_2,...,s_{N-1}),s_N)$
(2b) $\boldsymbol{Disj}(s_1, s_2) = s_1 + s_2 - s_1 s_2$;
$\boldsymbol{Disj}(s_1,s_2,...,s_N) = \boldsymbol{Disj}(\boldsymbol{Disj}(s_1,s_2,...,s_{N-1}),s_N)$
(2c) $\boldsymbol{Fagg}(s_1, s_2, ..., s_N) = \boldsymbol{Disj}(s_1, s_2, ..., s_N)$

- **Scheme #3:**

(3a) $\textbf{\textit{Conj}}(s_1, s_2) = s_1 s_2;$

$\textbf{\textit{Conj}}(s_1,s_2,\ldots,s_N)=\textbf{\textit{Conj}}(\textbf{\textit{Conj}}(s_1,s_2,\ldots,s_{N-1}),s_N)$

(3b) $\textbf{\textit{Disj}}(s_1,s_2) = \alpha(s_1+s_2-s_1s_2)+(1-\alpha)\max(s_1,s_2);$

$\textbf{\textit{Disj}}(s_1,s_2,\ldots,s_N) = \textbf{\textit{Disj}}(\textbf{\textit{Disj}}(s_1,s_2,\ldots,s_{N-1}),s_N)$

(3c) $\textbf{\textit{Fagg}}(s_1, s_2, \ldots, s_N) = \textbf{\textit{Disj}}(s_1, s_2, \ldots, s_N)$

The ***Disj*** functions from schemes #1 and #2 are identical to the formula for calculating the probability of the union of two (and respectively N) independent events.

Fagg from scoring scheme #1 corresponds to the probability of the union of two events such that one is completely dependent on the other. ***Fagg*** from scoring scheme #2 coincides with ***Disj*** from the same scoring scheme, which equals the probability of the union of two independent events.

Therefore in a probabilistic model the expression $s_1 + s_2 - s_1 s_2$ is a good choice for combining two independent scores, while $max(s_1, s_2)$ is a good choice for combining scores when one score is completely dependent on the other.

In scoring scheme #3 we design a scoring function whose values are between the values of the first two scoring functions (#3 is a linear combination of #1 and #2). The main objective behind the use of this third scoring function is to account for the dependencies between the knowledge sources (***UMLS***, ***FMA***, ***WordNet***) without completely ignoring the fact that, if more than one of them confirm certain prediction, that usually improves the odds that this prediction is correct. In scheme #3, $\alpha \in [0, 1]$ is a parameter of the linear combination defined in (3b). It varies depending on the knowledge source or the combination of knowledge sources, which confirm the predictions whose scores we accumulate in (3b). The α parameter acts as a buffer to prevent the score from growing too fast when adding up cumulative predictions (i.e. when the predictions being accumulated are confirmed by several knowledge sources): when α equals 0.0, the value is growing the quickest (as it should for independent scores); when α equals 1.0, the value is limited by the maximum score of the scores being accumulated.

To experimentally show that the choice of ***Disj*** from (3b) is a reasonable one, we have generated a set of observations on two dependent random variables x_1, x_2 with Boolean (1/0 i.e. true/false) truth values, and we have confirmed that if we substitute the scores s_1 and s_2 in (3b) with the probabilities $P(x_i = true)$ $(i = 1, 2)$, and α with the modulus of the correlation coefficient between the two random variables, we get a very good approximation for the probability $P(z = true)$ of their Boolean disjunction $z = (x_1 \; or \; x_2)$.

VI. Results and discussion

Let us consider the following two figures which illustrate how the scores generated by the three scoring schemes are related to each other and demonstrate the advantages of scheme #3.

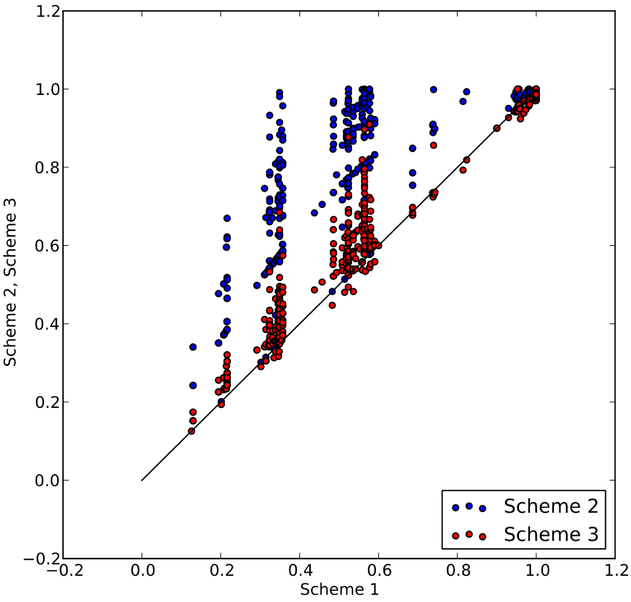

Figure 1. *Scatter plot: scheme #1 vs schemes #2 and #3*

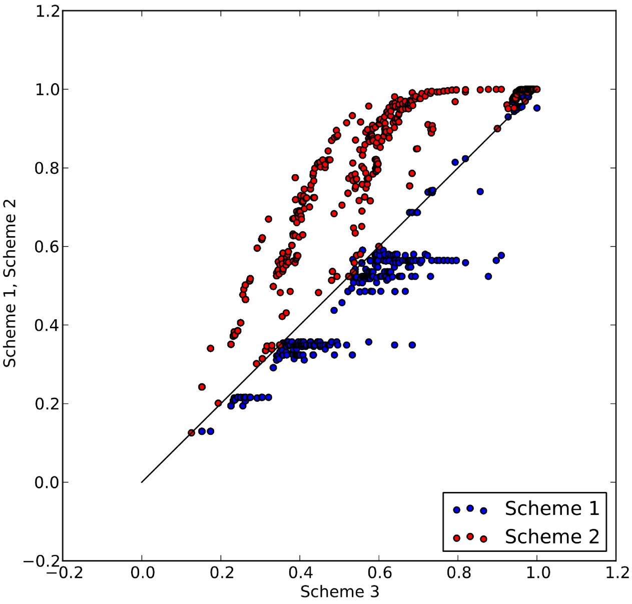

Figure 2. *Scatter plot: scheme #3 vs schemes #1 and #2*

It can be seen on Fig.1 that the data in scheme #1 appear clustered around the configured values for knowledge source scores (and combinations of these), because there isn't anything to account for the amount of available evidence gathered from each source (e.g. the number of patterns confirming a prediction). Compared to scheme #1, both schemes #2 and #3 scatter the clusters because the $\boldsymbol{F_{agg}}$ values are growing when more patterns are confirming a given prediction. As $\boldsymbol{F_{agg}}$ in scheme #3 is limited through the α parameter, it causes a more moderate scattering as seen on Fig. 1, while scheme #2 causes a very rapid increase.

The main advantage of scheme #3 is that it allows us to control the speed at which additional patterns increase the score, while scheme #2 gives control only over the initial value of that score. Within scheme #2, when having one pattern confirming the prediction, the scores start somewhere around the configured CMP score value (defined by the penalty constant), and grow with the same speed up to 1.0. Within scheme #3 this growth can be slowed down and controlled through the α parameter. The difference between the schemes #2 and #3 can be seen on Fig. 2 in red, and it clearly shows how easily some scores approach the value 1.0 when scheme #2 is used. The ***Disj*** function from Scheme #3 also causes a softening effect on the score when there are multiple knowledge sources and algorithmic procedures (DM, SMP, CMP) confirming the prediction, because it allows us to control the speed at which the score grows and even to use the actual correlation coefficient between the distinct knowledge sources. This is not directly visible on the figures at this scale, because it largely produces local shifts in the position of the clusters and has the biggest effect on data predicted by the knowledge sources (SMP) which constitute the cluster around score=1.0.

VII. Conclusion

We presented in this paper an original algorithmic approach to inferring (predicting and scoring) cross-ontology links within automatic mapping of distinct species-specific anatomical ontologies. The full mapping procedure assumes that the auto-generated set of predictions will be carefully checked by a curator (a human, an anatomy expert) and his/her input will be utilized to accurately calculate the correlation coefficients between certain pairs of knowledge sources. These correlation coefficients could be used as values for the α parameters of the scoring scheme. The procedures described briefly here and detailed in [12], and the scoring schemes introduced here, are utilized in the software program AnatOM [10], [13] developed as part of our work on semi-automatic mapping and merging of anatomical ontologies.

References

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

[2] J. Day-Richter, *OBO Flat File Format Specification, version 1.2*, 2006, Available online at *http://www.geneontology.org/GO.format.obo-1_2.shtml*, Last accessed: 20 October 2012.

[3] M. Ashburner et al., *Gene ontology: tool for the unification of biology*. Nature Genetics, Vol. 25(1), 2000, pp. 25–29. http://dx.doi.org/10.1038/75556

[4] B. Smith et al., *The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration*. Nature Biotechnology, Vol. 25, 2007, pp. 1251–1255. http://dx.doi.org/10.1038/nbt1346

[5] O. Bodenreider, *The Unified Medical Language System (UMLS): integrating biomedical terminology*. Nucleic Acids Research, Vol. 32, 2004, pp. 267–270. http://dx.doi.org/10.1093/nar/gkh061

[6] C. Rosse, J. Mejino, *A reference ontology for biomedical informatics: the Foundational Model of Anatomy*. J. Biomed. Inform., Vol. 36(6), 2003, pp. 478–500. http://dx.doi.org/10.1016/j.jbi.2003.11.007

[7] G. Miller, *WordNet: A Lexical Database for English*. Communications of the ACM, Vol. 38(11), 1995, pp. 39–41. http://dx.doi.org/10.1145/219717.219748

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

[9] Ernest A.A. van Ophuizen, Jack A.M. Leunissen, *An evaluation of the performance of three semantic background knowledge sources in comparative anatomy*. J. Integrative Bioinformatics, Vol. 7, 2010, pp. 124–130.

[10] Peter Petrov, Nikolay Natchev, Dimitar Vassilev, Milko Krachounov, Maria Nisheva, Ognyan Kulev, *AnatOM An intelligent software program for semi-automatic mapping and merging of anatomy ontologies*. To appear in: Proceedings of the 6th International Conference on Information Systems & Grid Technologies (ISGT, Sofia, 1–3 June 2012), Sofia, St. Kliment Ohridski University Press, 2012.

[11] J. Euzenat, P. Shvaiko, *Ontology Matching*. Springer, Heidelberg, 2007.

[12] Peter Petrov, Milko Krachunov, Ernest A.A van Ophuizen, Dimitar Vassilev, *An algorithmic approach to inferring cross-ontology links while mapping anatomical ontologies*. To appear in Serdica Journal of Computing, ISSN 1312-6555, Vol. 6, 2012.

[13] Peter Petrov, Milko Krachunov, Elena Todorovska, Dimitar Vassilev, *An intelligent system approach for integrating anatomical ontologies*. Biotechnology and Biotechnological Equipment 26(4):3173–3181, 2012

[14] *http://www.nlm.nih.gov/research/umls/* – Web site of the Unified Medical Language System (UMLS) by the U.S. National Library of Medicine (NLM), Last accessed: 20 October 2012.

[15] *http://sig.biostr.washington.edu/projects/fm/* – Web site of the Foundational Model of Anatomy (FMA) by the University of Washington, Last accessed: 20 October 2012.

[16] *http://wordnet.princeton.edu/* – Web site of the WordNet project by the Princeton University, Last accessed: 20 October 2012.