ARTICLE
BIOINFORMATICS

DOI: 10.5504/BBEQ.2012.0010

AN INTELLIGENT SYSTEM APPROACH FOR INTEGRATING ANATOMICAL ONTOLOGIES

Peter Petrov[1], Milko Krachunov[1], Elena Todorovska[2] and Dimitar Vassilev[2]
[1]Sofia University "St. Kliment Ohridski", Faculty of Mathematics and Informatics, Sofia, Bulgaria
[2]AgroBioinstitute, Bioinformatics Group, Sofia, Bulgaria
Correspondence to: Dimitar Vassilev
E-mail: jim6329@gmail.com

ABSTRACT

Recent years have seen a vast amount of data generated by various biological and biomedical experiments. The storage, management and analysis of this data, is done by means of the modern bioinformatics applications and tools. One of the bioinformatics instruments used for solving these tasks, are ontologies and the apparatus they provide. Ontology as a modeling tool is a specification of a conceptualization meaning that an ontology is a formal description of the concepts and relationships that can exist for a given software system or software agent (8, 10). Anatomical (phenotypic) ontologies of various species nowadays typically contain from few thousands to few tens of thousands of terms and relations (which is a very small number compared to the count of objects and the amount of data produced by biological experiments at the molecular level, for example) but usually the semantics employed in them is enormous in scale. The major problem when using such ontologies is that they lack intelligent tools for cross-species literature searches (text mining) as well as tools aiding the design of new biological and biomedical experiments with other (not yet tested) species/organisms, based on available information about experiments already performed on certain model species/organisms.

This is where the process of merging anatomical ontologies comes into use. Using specific models and algorithms for merging of such ontologies is a matter of choice. In this work a novel approach for solving this task, based on two directed acyclic graph (DAG) models and three original algorithmic procedures is presented. Based on them, an intelligent software system for merging two (and possibly more) input/source anatomical ontologies into one output/target super-ontology was designed and implemented. This system was named AnatOM (an abbreviation from "Anatomical Ontologies Merger").

In this work a short overview of ontologies is provided describing what ontologies are and why they are widely used as a tool in bioinformatics. The problem of merging anatomical ontologies of two or more different organisms is introduced and some effort has been put into explaining why it is important. A general outline is presented of the models and the method that have been developed for solving the ontologies merging problem. A high-level overview of the AnatOM program implemented by the authors as part of this work is also provided.

To achieve the degree of intelligence that is needed, the AnatOM program utilizes the large amount of high-quality data (knowledge) available in several widely popular and generally recognized knowledge bases such as UMLS, FMA, and WordNet. The last one of these is a general-purpose i.e. non-specialized knowledge source. The first two are biological/biomedical ones. Their choice was based on the fact that they provide a very good foundation for building an intelligent system that performs certain comparative anatomy tasks including mapping and merging of anatomical ontologies (23).

Biotechnol. & Biotechnol. Eq. 2012, **26**(4), 3173-3181

Keywords: anatomical ontology, merging, algorithm, software, artificial intelligence

Introduction

The concept of ontology

The classical meaning of the term ontology originates from philosophy. The word has a Greek origin (όντος – of being, -λογία – study, science, theory) and can be literally translated as "the study of being". In philosophy it denotes the study of existence (or reality in general), together with the basic categories of being and the relations which exist among them. Even though the word is Greek, the first existing record of it is the Latin form 'ontologia' which appeared in the works of the German philosophers Jacob Lorhard (1561 – 1609) and Rudolf Goeckel (1547 - 1628) at the beginning of the 17th century. In philosophy ontology is viewed as part of the major branch known as metaphysics. It deals with questions concerning what entities exist or can be said to exist, how such entities can be grouped, how they can be organized within a hierarchy, and divided or subdivided according to the similarities and/or the differences between them (26).

Different definitions of ontology can be found in a non-philosophical sense. One of the largest dictionaries of the (American) English language (24), provides two definitions of ontology: *1. a science or study of being: specifically, a branch of metaphysics relating to the nature and relations of being; a particular system according to which problems of the nature of being are investigated; first philosophy; 2. a theory* *concerning the kinds of entities and specifically the kinds of abstract entities that are to be admitted to a language system.*

A short but at the same time explanatory enough modern definition of the ontology concept was given by Grenon et al. (7). The definition does not pertain to computer or information sciences only: *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.*

A slightly longer definition of the ontology concept is that of Gruber (8). This definition is specifically constrained within the context of computer science (in general) and artificial intelligence and knowledge representation (in particular): *An ontology is an explicit specification of a conceptualization. The term is borrowed from philosophy, where an ontology is a systematic account of Existence. For knowledge-based systems, what "exists" is exactly that which can be represented. When the knowledge of a domain is represented in a declarative formalism, the set of objects that can be represented is called the universe of discourse. This set of objects, and the describable relationships among them, are reflected in the representational vocabulary with which a knowledge-based program represents knowledge. Thus, we can describe the ontology of a program by defining a set of representational terms. In such an ontology, definitions associate the names of entities in the universe of discourse (e.g., classes, relations, functions, or other objects) with human-readable text describing what the names are meant to denote, and formal axioms that constrain the interpretation and well-formed use of these terms.*

In a contemporary bioinformatics sense, an ontology can be considered to be composed of (i) a controlled vocabulary system of biological or biomedical terms, (ii) a formal representation of the relations which exist among these terms, (iii) a logical reasoning/inference system which can infer/deduce new knowledge from the knowledge that is explicitly contained in (i) and (ii). Therefore, for the purposes of this work, (i) and (ii) are viewed as an ontology's static part, and (iii) as an ontology's dynamic part.

Ontology elements

In bioinformatics and in computer science, ontologies are formal systems composed of several building blocks regardless of the particular formal language that is used to represent them. The ontology building blocks (25) are listed below divided into two groups based on our view about their importance.

The primary (most widely used) ontology components are:

- individuals – these are also called instances (or objects, terms, concepts);
- classes – sets or collections of objects where objects are grouped by some common feature which all objects falling within the class share among themselves;
- attributes – properties (features, characteristics) of the objects;
- relations – logical links between individuals or between classes describing how these individuals or classes are related to each other.

The secondary (less well-known, not so widely-used) ontology components are:

- function terms – structures formed by certain relations which can replace individual terms (objects) within more complex expressions;
- restrictions – formal statements describing what must hold true in order for some assertion to be considered as valid;
- rules – if-then statements (sentences) describing the logical inferences that can be drawn from an assertion;
- axioms – assertions (which may include rules) in a logical form, which constitute the theory that the ontology describes in the particular application domain that is being modeled;
- events – the changing (the process or the act of changing) of attributes and relations.

It is to be outlined that this categorization of the ontology components into two groups (primary and secondary) is a relative and subjective one. So it is not to be taken literally and accepted as the only possible one. It is based on our view as to which components are more/less commonly used when exploring ontologies in general and their applications in computer science and bioinformatics in particular. The primary components are the most common ones and they are those, which people come across first when they come to the field of ontologies. The secondary ones are more specific and not so widely used.

Most of the components presented above are static by their nature. The rules, the axioms and the events bring some dynamics to the ontology models. It is assumed throughout this paper that an ontology's static side is all about the structure which is being modeled within the particular domain under study; its dynamic side is about reasoning, making inferences and deducing new facts from the already known ones.

Ontologies in bioinformatics and in artificial intelligence

Bioinformatics is sometimes also referred to as computational biology even though there is a difference between the two and many authors acknowledge that difference. Bioinformatics is an interdisciplinary field and is concerned with the usage of methods, tools and techniques from various scientific fields and disciplines such as mathematics, statistics, computer science, artificial intelligence, chemistry and biochemistry, for solving biological problems. These problems may exist at various levels of detail e.g. molecular level, tissue level, organ level, organism level and even at certain super-organism levels.

The recent decades have seen an explosion in the amount of data produced by experiments in life sciences such as biology and molecular biology in particular. Various academic, business, and non-profit organizations have developed their own software systems for collecting, storing, analyzing, interpreting, processing and presenting that data to the end users – researchers, experts, lab workers (11, 16, 19). Interoperability and integration between these systems has been ignored for a long time but has recently become a necessity, as it became apparent that even the most powerful hardware and software systems cannot perform all necessary tasks on their own but need to cooperate and communicate with already existing systems, and to reuse the functionality and the data that are already there. This trend has also been implied by the recent developments in information technology (IT) where huge monolithic multi-functional (all-in-one) systems have been replaced by multiple, smaller and highly-specialized systems (services, agents) each of which communicates with others of the same kind, in order to provide the necessary functionality and results to its end users. These trends of transition from centralized to decentralized form of computing have led to an increased interest towards data and system integrations. Ontologies turn out to be a useful tool for approaching and solving these kinds of problems.

Another reason for putting ontologies to work in bioinformatics and computational biology is the need to have standard controlled vocabularies and to reuse them between different scientific organizations and workgroups. This has at least two important benefits. First, scientists are able to share uniform scientific terminology which eases the understanding between them and minimizes the risk of ambiguities and misunderstandings (20). Second, scientists are able to perform intelligent searches in existing scientific literature which is usually referred to as text mining.

For the purposes of this work, the second one of the two benefits noted above was of main interest. To arrive at this stage (performing intelligent, e.g. synonym-based, searches), the problem of taking two or more source ontologies as input and producing one target ontology as output needs to be solved (3, 5). This process implies merging or mapping or aligning the input/source ontologies. The output/target ontology that is produced by this process is what is denoted as the super-ontology.

Enabling different scientific organizations and groups to share common structured and controlled vocabularies naturally leads to a shift towards artificial intelligence (AI) systems. These systems can benefit from this usage of ontologies just as humans do. Ontologies can be used as an instrument for enabling higher integration, easier communication, more efficient knowledge interchange and knowledge sharing between multiple AI systems and agents (9).

Ontologies are formal systems for representing and organizing explicit knowledge and for allowing reasoning (inference) of implicit knowledge (implicit is knowledge which is not explicitly contained/declared in the ontology model itself). The former is viewed as the static side of an ontology – this is a set of objects/classes and a set of relations which exist among them (both these sets are stated/declared explicitly); it is static as it is just a description of a structure together with knowledge explicitly stated/declared to be contained in that structure (14). The latter is viewed as the dynamic side of an ontology as it is about the ability to infer even more knowledge which is implicitly contained (in the ontology) but which is not explicitly stated/declared in the static part of the ontology model.

Ontologies have emerged from earlier, less formal knowledge representation systems developed for use in AI systems and AI agents. Almost all modern ontological systems are based on description logics (DLs in short) (1). Description logics are a family of formal, mathematical logical systems for representing explicit knowledge and for inferring implicit knowledge from the explicit knowledge that is given/known up-front.

Building AI systems and AI agents poses the question of knowledge representation and knowledge inference. When considering simple, limited application domains and developing AI systems and AI agent for them, the choice of a knowledge representation model is not that important as it is easy to come up with a consistent terminology and simple procedures for knowledge inference and decision making. When it comes to complex domains, such as, for example, predicting financial trends, computer-aided disease diagnosis, or controlling a robot in a complex environment, the choice of a knowledge representation model and a knowledge inference model become crucial. More general and more flexible forms of knowledge representation and knowledge inference need to be developed and utilized. Representing abstract/general AI concepts like actions, beliefs, facts, time, physical obstacles which occur in many real-world application domains is what is usually called 'ontology engineering' (17). This is where ontology models come into use – for representing explicit knowledge, for formally representing abstract/general concepts, and for inferring implicit knowledge from the explicit knowledge base.

Objective of this work

This work deals with anatomical ontologies as its main subject. For its purposes, the anatomical ontologies published by the OBO Foundry Project (18) were used as these are nowadays widely recognized and are a *de facto* standard in the biomedical domain. OBO stands as an abbreviation of Open Biomedical Ontologies. The OBO Foundry Project is an open collaborative effort to standardize the design, development and publication of biomedical ontologies by researchers worldwide.

Anatomical ontologies consist of anatomical/phenotypic concepts/terms (e.g. anatomic region, organ system, head, head organ, nervous system, central nervous system, brain, etc.) and the relations which exist among these concepts/terms (e.g. the brain *is part of* the central nervous system; the central nervous system *is part of* the nervous system; the nervous system *is an* organ system; the brain *is a* head organ; the head organ *is part of* the head).

Our research task involved taking two (or more) species-specific/organism-specific anatomical source ontologies as input and algorithmically generating a single generalized anatomical ontology as output (a super-ontology), thus integrating the source ontologies into a single ontology model. For this purpose, the adult mouse anatomical ontology and the zebrafish anatomical ontology were used, as these organisms are widely adopted and recognized as important model organisms in biological lab research. The integration of the anatomical ontologies of two separate organisms (mouse and zebrafish in particular) is crucial when various intelligent text searching (or text mining) tasks for finding cross-organism/cross-species synonyms need to be performed (in scientific literature for instance).

Materials and Methods

For solving this task, two novel models based on graph theory are developed and outlined here. The graph theory models were processed algorithmically by consulting/interrogating several external knowledge sources for the goal of merging nodes and relations from the input ontologies in a biologically meaningful way, and for defining nodes and relations in the output ontology again in a biologically meaningful way.

The formal computer-readable language used for representing ontologies in this work was OBO. It is one of the most common, standard ways of representing ontologies in bioinformatics even though other ontology representation languages exist and are being used (OWL, RDF and RDF-Schema, and others). OBO is the language (and the accompanying file format) defined and promoted by the Gene Ontology (GO) project (4, 20) and adopted by the OBO Foundry initiative (18).

The three external knowledge sources which were used in this work, and which the AnatOM program communicates with, are UMLS, FMA and WordNet.

The Unified Medical Language System (UMLS) is a system developed and maintained by the US National Library of Medicine (NLM). The UMLS aims to integrate and distribute key terminology, classification and coding standards, and associated resources to promote creation of more effective and interoperable biomedical information systems and services (2, 21). Presently, it is one of the largest knowledge sources (if not the largest one) in the biomedical domain documenting several million biomedical concepts and the relations between them. Over 20 natural languages are used in UMLS of which English is the most widely used one.

The Foundational Model of Anatomy (FMA) developed by the University of Washington is a domain-specific ontology that documents the anatomy of the human body. The FMA makes available anatomical information in symbolic (non-graphical) form to knowledge modelers and other developers of applications for education, clinical medicine, electronic health record, biomedical research and all areas of health care delivery and management. Currently the FMA contains approximately 75,000 classes and over 120,000 terms. Over 2.1 million relationship instances from over 168 relationship types link the FMA's classes into a coherent symbolic model. The FMA is also one of the largest computer-based knowledge sources in the biomedical domain (15, 22).

WordNet is a large lexical database of the English language developed and maintained by the Princeton University. Of the three knowledge sources used in this work (UMLS, FMA, WordNet), it is the only general-purpose one, i.e. WordNet is not specialized in biology, anatomy, or biomedicine (6, 12, 13).

Ontology modeling

It is our understanding which is adopted here and underlined throughout this work that ontologies can be viewed as higher-order (or upper-level) models composed of two lower-level sub-models. The first one deals with the static side of the knowledge – it is about representing the explicit knowledge that is there (knowledge representation). The second one is about inferring implicit knowledge (knowledge inference). The first (static) part is about graph theory; the second (dynamic) part is about logic and logical inference. The static graph theory model was of main interest for the purposes of this work, as it provides the basis for solving the problem or merging two (or more) anatomical ontologies into one single anatomical super-ontology.

TABLE 1

Semantic relations – definitions and examples

Relation	Definition	Examples
Holonym	'X' is a holonym of 'Y' if Ys are parts of (members of) Xs	'forelimb' is holonym of 'arm', of 'elbow' and of 'hand'; 'heart' is a holonym of 'heart atrium', 'heart endocardium', 'myocardium layer', 'heart septum'
Meronym	'X' is a meronym of 'Y' if Xs are parts of (members of) Ys	'arm', 'elbow', 'hand' are meronyms of 'forelimb'; 'heart atrium', 'heart endocardium', 'myocardium layer', 'heart septum' are meronyms of 'heart'
Hyponym	'X' is hyponym of 'Y' if all Xs are also Ys but X represents a more specific concept than Y	aorta, arteriole, artery are hyponyms of 'arterial blood vessel'; 'brain', 'eye', 'ear' are hyponyms of 'head organ'
Hypernym	'X' is hypernym of 'Y' if all Ys are also Xs but X represents a more general concept than Y	'arterial blood vessel' is a hypernym of aorta, arteriole, artery; 'head organ' is a hypernym of 'brain', 'eye', 'ear'

Static side. Graphs

Graph theory is a branch of discrete mathematics which studies graph models (graphs) and their properties. Graphs are mathematical network-like models composed of two sets – V (set of vertices/nodes) and E (set of edges/arcs). The set V contains elements which are called vertices (or nodes). The set $E \subseteq V \times V$, contains elements called edges (or arcs), each edge connecting two vertices from the set V. Apparently E defines a binary relation on the set V. The edges may be undirected (symmetric) or directed (asymmetric) depending on whether the relation E is symmetric or not. When edges are directed the graph is called a directed graph; respectively when edges are undirected the graph is called an undirected graph. Directed edges (also called arcs or arrows) make a distinction between their start (first) node and their end (last) node; undirected edges do not make that distinction between start and end vertices (both ends may be viewed as either start-point or end-point of the undirected edge). Each edge which has the node *v* as its start, is called an outgoing edge with respect to *v*; each edge which has the node *v* as its end is called an incoming edge with respect to *v*. A path in a graph is a sequence of edges e_1, e_2,..., e_n, such that the end node of each edge e_k coincides with the start node of the next edge e_{k+1}. A cycle is a path in which the end node of the last edge e_n coincides with the start node of the first edge e_1. A directed acyclic graph (DAG) is a directed graph with the special property that no sequence of edges forms a cycle. A tree is a DAG with the additional property that there is at most one incoming edge associated with each node. In a way a DAG can be viewed as generalization of a tree. DAG is a well-known and widely applicable type of graph in graph theory and its applications.

Graph theory and graph theoretical models have long history in being used for knowledge representation and modeling in AI systems so it is natural that their apparatus was employed for the purposes of this work. The static side of an ontology is, by its nature, equivalent to a DAG structure (as defined above), in which nodes/vertices represent objects/terms/concepts (from the ontology), and edges represent relations between these objects/terms/concepts. Therefore throughout this text, DAGs have been used for modeling anatomical ontologies of different species/organisms and for solving the particular problem of mapping/merging them.

An edge from the static DAG model of an ontology represents a relation between the two nodes of the DAG that it connects. These relations are typically used in linguistics: synonyms, hypernyms, hyponyms, meronyms, holonyms. **Table 1** summarizes these semantic relations and provides examples of them.

In **Fig. 1** an excerpt is shown from the DAG of the adult mouse anatomical ontology as published and maintained by the OBO Foundry Project. The DAG contains nodes (ovals) which represent terms/concepts from the adult mouse anatomical ontology as well as directed edges which represent relations (is_a, part_of) that exist among them.

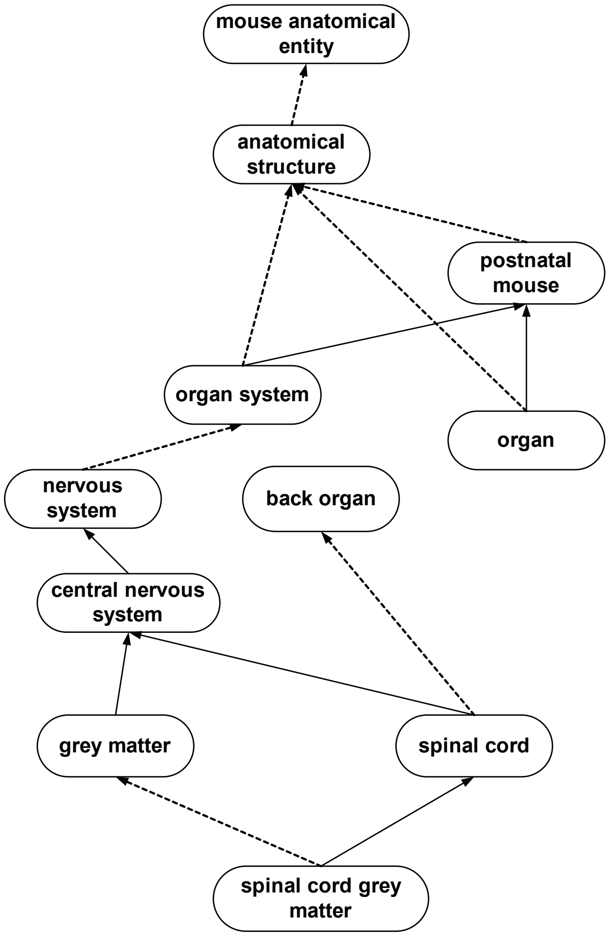

Fig. 1. An excerpt from the graph model (the DAG) of the mouse anatomy ontology which illustrates several anatomical terms and the relations which exist among them. Terms/concepts (ovals); 'part-of' relations (solid arrows); 'is-a' relations (dotted arrows); arrow heads point from the child term to the parent term.

Dynamic side. Reasoning and inference

For the purposes of the method presented here, it is not necessary to go into much detail about the dynamic part of an ontology. Therefore it is mentioned here only briefly. The dynamic side of an ontology is based on a formal logical system for inferring new (implicitly contained) knowledge from the already existing (explicitly stated) knowledge. There are several widely used logical systems in AI and the most widely-known and well-studied ones are propositional logic/calculus and first-order predicate logic. Ontology engineering and ontology knowledge modeling are based on another set of formal logical systems known as description logics (DLs).

Description logics are systems which are more general (wider, more expressive) than propositional logic but less general (narrower, less expressive) than first order predicate logic. When it comes to reasoning and knowledge inference, the questions of decidability of the subsumption problem (Does A belong to B?) and the instance problem (Is A an instance of B?), in a given logical system, become very important. It has been proven that in the worst-case (widest) sense these problems are intractable for description logics as they are non-deterministic polynomial (NP) complete. Still, it has been shown that the intractability of the two problems in their most general sense does not prevent DLs from being useful in practice for building ontologies and for applying reasoning procedures on them. That practical usefulness is achieved by narrowing the expressiveness of the DL languages and also by applying various optimization techniques when implementing reasoning in DL systems.

Algorithmic solution

The method outlined here for generating an output (target) super-ontology from two input (source) ontologies, is composed of two main phases: 1) mapping of the two input ontologies onto each other; 2) merging the two input ontologies into a super-ontology.

The mapping phase consists of establishing the semantic links (synonymy, parent-child) between the two anatomical ontologies and between their nodes/terms in particular. It does that by applying three algorithmic procedures of different complexity which are listed here from most straightforward to most intelligent: 1) syntactical/direct matching of nodes; 2) matching of nodes based on the knowledge available in the external knowledge sources; 3) matching (parent) nodes of the two input ontologies based on patterns of matches (patterns of cross-ontology connectivity) already discovered in 1) and 2) between the children of these (parent) nodes.

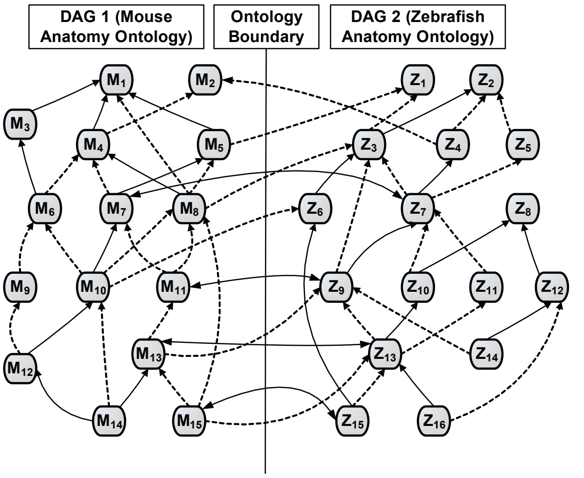

Fig. 2. Model #1 – the two ontologies mapped onto/linked with each other. Mouse (M_k) and zebrafish (Z_k) anatomy terms; virtual border between the two source ontologies being mapped (vertical solid line); 'part-of' relations within the source ontologies (inner-ontology solid lines); 'is-a' relations within each of the source ontologies (dashed inner-ontology lines); 'synonymy' relations between the nodes of the source ontologies (cross-ontology solid lines, and so are symmetrical/bidirectional); 'parent-child' relations between the nodes of the source ontologies (cross-ontology dashed lines, and therefore are asymmetrical/unidirectional); all unidirectional links (arrows) point from child term to parent term.

The mapping phase results in building a semantically rich initial model – Model #1 which is denoted here as 'the two ontologies mapped onto each other'. This model contains all cross-ontology links/connections which are discovered by the three algorithmic procedures outlined above. The cross-ontology links established here are of the following types: 'is_a' parent-child links, 'part_of' parent-child links, 'synonymy' links (**Fig. 2**).

The merging phase consists of introducing/defining new terms/concepts (generalized concepts or super-concepts) from the input ones and drawing the hierarchical edges ('part-of', 'is-a', etc.) between the newly defined super-concepts. This is done based on the inner-ontology links given and the cross-ontology links found. The main result of the merging phase is what is called Model #2 or 'the super-ontology'. Having the super-ontology generated implies that an important side-product – a cross-species thesaurus, is also generated. The thesaurus semantically translates terms/concepts from source anatomy #1 (the mouse anatomical ontology) to terms/concepts of source anatomy #2 (the zebrafish anatomical ontology) (**Fig. 3**).

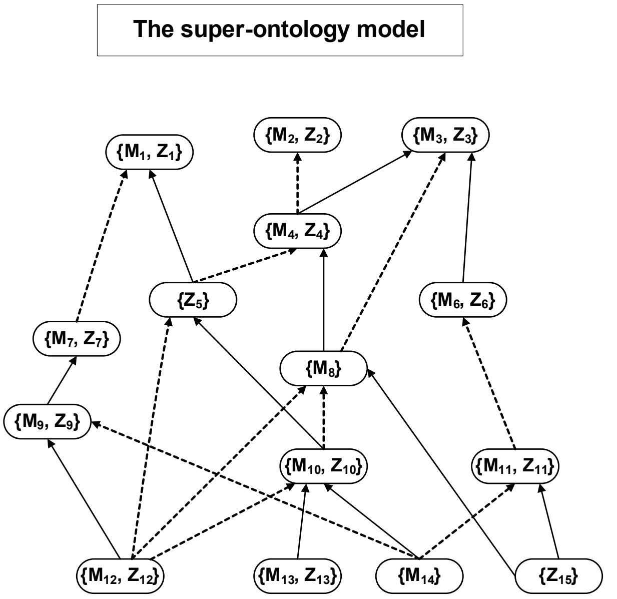

Fig. 3. Model #2 – the super-ontology – the two ontologies merged. Super-ontology terms are presented: some of them originate from mouse e.g. $\{M_8\}$, some, from zebrafish, e.g. $\{Z_5\}$, and some, from both source ontologies, e.g. the node $\{M_7, Z_7\}$; 'part-of' relations within the super ontology (solid lines); 'is-a' relations within the super-ontology (dashed lines); all links point from child term to parent term**.**

Results and Discussion

The models and the method outlined above were implemented in a software program named AnatOM. It is a platform-independent program implemented in Python which uses three separate MySQL databases representing the three external knowledge sources which AnatOM communicates with (UMLS, FMA, WordNet). AnatOM is a typical GUI-based program and not a command line tool. It was tested to work under both Linux and Windows. In **Fig. 4** an overview of the AnatOM's program/process flow is presented.

The user is first given the option to choose the two input anatomical ontologies. The program then loads and parses them thus generating two DAGs which are stored in memory.

The program applies three algorithmic procedures on the two DAGs to come up with a mapping of the two input ontologies onto each other, and to finally merge them into a single super-ontology (a single DAG).

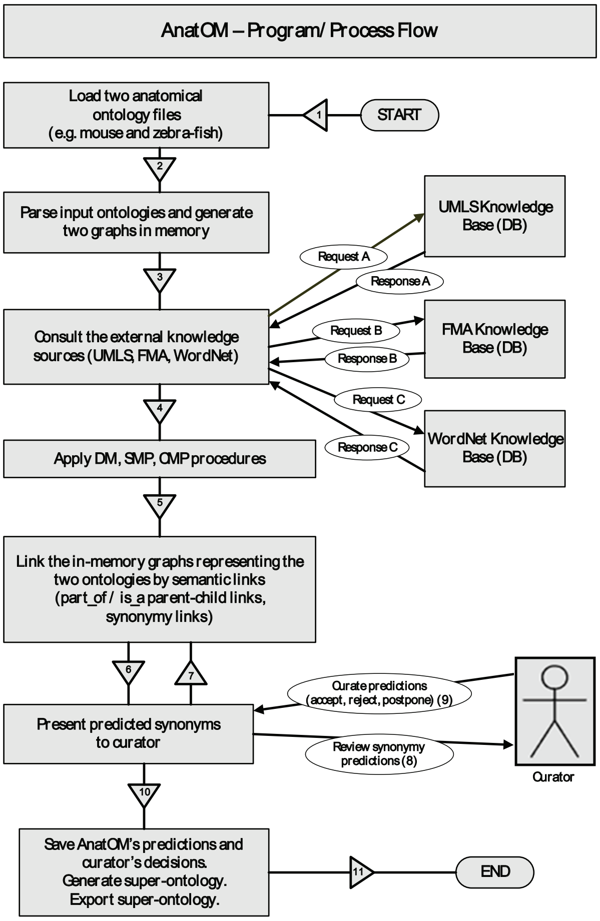
AnatOM – Program/ Process Flow
Load two anatomical ontology files (e.g. mouse and zebra-fish)
1
START
2
Parse input ontologies and generate two graphs in memory
UMLS Knowledge Base (DB)
Request A
Response A
3
Consult the external knowledge sources (UMLS, FMA, WordNet)
Request B
FMA Knowledge Base (DB)
Response B
Request C
4
WordNet Knowledge Base (DB)
Response C
Apply DM, SMP, CMP procedures
5
Link the in-memory graphs representing the two ontologies by semantic links (part_of / is_a parent-child links, synonymy links)
6
7
Curate predictions (accept, reject, postpone) (9)
Present predicted synonyms to curator
Review synonymy predictions (8)
Curator
10
Save AnatOM's predictions and curator's decisions. Generate super-ontology. Export super-ontology.
11
END

Fig. 4. An overview of AnatOM's process/program flow.

The first procedure applied is what is called textual, or syntactical, or direct matching (DM), procedure. This procedure looks for plain text matches between terms from the two input ontologies in order to draw cross-ontology links (synonymy, is-a parent-child, part-of parent-child) between those terms. To give an example, if the first input ontology contains the term 'brain', and the second input ontology also contains the term 'brain', it is natural to predict that these two terms would be synonyms (without even communicating with any external knowledge sources). Such direct matches are assigned scores (as are all predictions made by the AnatOM program) and are marked as originating from DM and not from any of the two procedures which are described next. Direct matches are assigned a score and this score is a constant/property which is configurable in AnatOM. The direct matches though, are pretty rare and the program cannot rely solely on them in order to make a complete and accurate mapping between the two input ontologies. For that to be achieved some more intelligence is needed.

The second procedure applied is the source matching predictions (SMP) procedure. At this point, the program communicates with the three external knowledge sources which contain biomedical terms and interrogates them for synonyms, parents (is-a/part-of), and children (also is-a/part-of) of the terms from the two input ontologies. Then a set of basic rules is applied which allow AnatOM to map/link certain terms from the two input ontologies. An example for such a rule is as follows: when term A from the first input ontology is found to be synonym of term X from the knowledge source S (S is either UMLS or FMA or WordNet), and if term B from the second input ontology is also found to be synonym of term X from the knowledge source S, then the program marks A and B as synonyms predicted by SMP through the knowledge source S. This prediction is then assigned a score equal to the reliability score of the knowledge source S (23) through which the SMP prediction was made. The reliability scores of the three knowledge sources are constants configurable in AnatOM.

The intelligence of AnatOM and its ability to map the two input ontologies onto each other, come mostly from the fact that it communicates with the three external knowledge sources UMLS, FMA, and WordNet. These knowledge sources are represented for the program's purposes in the form of three MySQL relational databases. AnatOM's communication with them is shown in **Fig. 4** as Requests/Responses (A), (B), and (C). These are plain SQL requests/queries and responses/results. In fact, the (A), (B), and (C) are series/sessions of SQL requests/responses. The results returned by these SQL queries allow the AnatOM program to build cross-ontology links (synonymy links, is-a parent-child links, part-of parent-child links). The communication with the external knowledge sources is at the heart of the SMP procedure and (in a way) also of the CMP procedure which is described next and which uses the results obtained through DM and SMP as its input. Once the SMP procedure is complete, the two input ontology DAGs are practically so heavily linked together that they can be viewed (and are viewed) as one single graph.

The third, the so-called child matching predictions (CMP), procedure is applied last. It uses the cross-ontology links which DM and SMP have generated and their scores but it assumes that there might be some omissions in the links found so far. So CMP tries to find even more cross-ontology links (synonymy, parent-child), which are not discoverable (and were not discovered) either through DM or through SMP. To do so, it scans the single DAG graph that resulted from applying DM and SMP, and looks for certain patterns of cross-ontology connectivity between pairs of inner-ontology parents/children. The CMP procedure looks for parents (from both ontologies) which are not linked yet (by DM or SMP) but whose children are (relatively) heavily linked. Then it draws conclusions (makes predictions) about a possible link (CMP-predicted link) between the parent terms based on how their children are cross-ontologically linked by DM and/or by SMP. Each CMP link is also assigned a score which is a function of the scores of all the DM and SMP links involved in the pattern detected and considered when making the CMP prediction.

At the final stages of the process flow, the predictions made by DM, SMP, CMP are presented to a curator (a biologist, an anatomical domain expert) who can make the necessary adjustments/amendments/decisions by either rejecting or accepting the predictions that were auto-generated by AnatOM. Finally, the program allows for saving all auto-generated predictions as well as all decisions made on them by the curator. Based on all this knowledge (the auto-generated predictions and the curator's decisions on them), the two ontologies are merged into a single one and a super-ontology is generated (**Fig. 4**). Finally, AnatOM supports exporting the super-ontology to the same file format (OBO) in which the two input ontologies were initially passed in as input.

The mouse anatomical ontology contains about 3000 anatomical terms and the zebrafish anatomical ontology contains about 2700 anatomical terms (these figures are valid as of February 2012). With the method presented here and implemented in AnatOM, about 700 predicted synonyms were identified. Extensive review and curation of these synonymy links by a domain expert curator (the AnatOM's user) is still to be performed, but the results are biologically adequate from non-expert perspective and thus quite encouraging.

The main goal of applying the method has been not to miss any biologically meaningful cross-ontology synonymy and parent-child links rather than to minimize the cross-ontology links which were predicted in error. Further fine-tuning of the algorithm and the scoring scheme are easily possible as AnatOM is flexible-enough and all initial constants which might influence the outcome of the algorithm are easily configurable.

The current results showed that mapping and merging the anatomical ontologies of two distinct organisms/species can be greatly simplified (semi-automated) with the use of an intelligent program such as AnatOM by utilizing the extensive structured external knowledge which has been collected, extended, curated, and improved by human domain experts over several years.

The AnatOM program uses discrete graph theoretical models, and a probability-like scoring scheme. Three algorithms/procedures act upon these models – DM, SMP and CMP, in order to predict semantic links (synonymy, is-a parent-child, part-of parent-child) between the two input anatomical ontologies. Going forward, other models including non-discrete (continuous) e.g. statistical models, could also be utilized in order to provide further improvement of the predictions that are made and thus to reduce the human intervention (the curator's work) that is necessary after the auto-prediction procedures have completed execution.

Conclusions

Merging anatomical ontologies from different species is important to biologists trying to perform cross-species textual searches in the scientific literature available. That is usually done in order to find cross-species similarity patterns of anatomical nature. In this work, a method was proposed which semi-automates the process of merging two given anatomical ontologies. Manual curation is still a necessity but the amount of work that is left for the curator is greatly reduced through the use of the AnatOM program. The program was developed as part of the current work and contains the most useful modules for solving the problem at hand.

A communication module is available for querying external structured knowledge sources like UMLS, FMA, WordNet. Additional knowledge sources, e.g. the Gene Ontology (GO), might be added with some minimal effort if that turns out necessary.

A visualization module is in place allowing the user to easily navigate through the mapped ontologies and the super-ontology as well as to visualize the links to the two source ontologies and to view the scores of the predicted synonymy and parent-child links.

What could be implemented next, is a searching (text mining) module providing ability to perform intelligent text searches or text mining into various external unstructured (natural language based) knowledge sources (scientific literature, the web) by utilizing the richness of the here generated super-ontology model. On the other hand, such a module could be viewed as a separate program which uses AnatOM, calls into it and gets results back (something that is usually denoted as pipelining).

Going forward, the improved accuracy of results from cross-species text searches (text mining) of anatomical terms in non-structured, natural language based information could be the main benefit brought by AnatOM. The AnatOM project long-term goal is to provide researchers with the necessary text-mining tools for finding similar scientific results (to their own) which are already published by others but are related to different organisms. Being able to perform such text mining tasks could help researchers in extrapolating the results obtained by their own experiments (to other model organisms), or help them design new experiments on other (yet untested) model organisms.

With respect to scalability, support for more than two input ontologies could be added to AnatOM in a relatively straightforward way so that more than two species could be merged into the generated super-ontology. This is possible due to the fact that AnatOM is able to export the generated super-ontology to OBO which is the same format as the one of the two input ontologies. Therefore the program could be run multiple times and each time a new anatomical species-specific ontology could be merged into the super-ontology produced by the previous program run.

REFERENCES

1. **Baader F., Nutt W.** (2007) In: The Description Logic Handbook: Theory, Implementation, and Applications (F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Shneider, Eds.), 2nd Ed., Cambridge University Press, 45-103.

2. **Bodenreider O.** (2004) Nucleic Acids Res., **32**, 267-270.

3. **Choi N., Song I-Y., Han H.** (2006) SIGMOD Record, **35**(3), 34-41.

4. **Day-Richter J.** (2006) OBO Flat File Format Specification, version 1.2., http://www.geneontology.org/GO.format.obo-1_2.shtml (Accessed: 20 February 2012)

5. **de Bruijn J., Martín-Recuerda F., Manov D., Ehrig M.** (2004) D4.2.1 State-of-the-art survey on Ontology Merging and Aligning V1, Digital Enterprise Research Institute, University of Innsbruck, http://www.sekt-project.com/rd/deliverables/wp04/sekt-d-4-2-1-Mediation-survey-final.pdf (Accessed: 20 February 2012)

6. **Fellbaum C.** (1998) WordNet: An Electronic Lexical Database, MIT Press, Cambridge, MA, p. 422.

7. **Grenon P., Smith B., Goldberg L.** (2004) In: Ontologies in Medicine. Studies in Health Technology and Informatics (P.M. Pisannelli, Ed.), Amsterdam, IOS Press, **102**, 20-38.

8. **Gruber T.R.** (1993) Knowl. Acquis., **5**(2), 199-220.

9. **Gruber T.R.** (1995) Int. J. Hum.-Comput. Studies, **43**(4-5), 907-928.

10. **Lambrix P., Tan H.** (2006) Web Semantics: Science, Services and Agents of the World Wide Web, **4**(3), 196-206.

11. **Lambrix P., Tan H.** (2008) In: Anatomy Ontologies for Bioinformatics: Principles and Practice (A. Burger, D. Davidson, R. Baldock R., Eds.) Computational Biology Series, Springer-Verlag, 133-150.

12. **Miller G.A.** (1995) Commun. ACM, **38**(11), 39-41.

13. **Princeton University** (2012) WordNet: A lexical database for English, http://wordnet.princeton.edu/ (Accessed: 20 February 2012)

14. **Rector A.** (2007) In: The Description Logic Handbook: Theory, Implementation, and Applications (F. Baader, D. Calvanese, D. McGuinness, D. Nardi, P. Patel-Shneider, Eds.), 2nd Ed., Cambridge University Press, 436-457.

15. **Rosse C., Mejino J.L.Jr.** (2003) J. Biomed. Inform., **36**(6), 478-500.

16. **Rubin D.L., Shah N.H., Noy N.F.** (2007) Briefings in Bioinformatics, **9**(1), 75-90.

17. **Russell S.J., Norvig P.** (2010) Artificial Intelligence: A Modern Approach, 5 Ed., Prentice Hall, p. 1132.

18. **Smith B., Ashburner M., Rosse C., Bard C., Bug W., Ceusters W. et al.** (2007) Nat. Biotechnol., **25**, 1251-1255.

19. **Songmao Zhang, Bodenreider O.** (2007) International Journal on Semantic Web and Information Systems, **3**(2), 1-26.

20. **The Gene Ontology Consortium, Ashburner M., Ball C.A. et al.** (2000) Nat. Genet., **25**(1), 25-29.

21. **U.S. National Library of Medicine** (2011) Unified Medical Language System, http://www.nlm.nih.gov/research/umls/ (Accessed: 20 February 2012)

22. **University of Washington School of Medicine** (2011) Foundational Model of Anatomy, http://sig.biostr.washington.edu/projects/fm/AboutFM.html (Accessed: 20 February 2012)

23. **van Ophuizen E.A.A., Leunissen J.A.M.** (2010) Journal of Integrative Bioinformatics, **7**, 124-130.

24. **Webster's Third New International Dictionary** (2002) Merriam-Webster.

25. **Wikipedia** (2012) Ontology (Information Science), http://en.wikipedia.org/wiki/Ontology_(information_science) (Accessed: 20 February 2012)

26. **Wikipedia** (2012) Ontology, http://en.wikipedia.org/wiki/Ontology (Accessed: 20 February 2012)