Oasis: a Mapping and Integration Framework for Biomedical Ontologies

Guanglei Song, Yu Qian, Ying Liu, Kang Zhang
Department of Computer Science, University of Texas at Dallas
Richardson, Texas 75083-0688 USA
{gxs017800, yxq012100, ying.liu, kzhang }@utdallas.edu

Abstract

More and more ontologies are emerging across bioinformatics domains to represent and define domain knowledge, such as gene ontology, anatomy ontology and disease ontology. To integrate these heterogeneous ontologies is becoming critically important for applications utilizing multiple ontologies. Because the entities described in the ontology often overlap with other entities in other ontologies, a mapping between two corresponding terms is required and becoming the key to the integration of heterogeneous ontologies. Based on a previously developed schema matching algorithm, this paper presents a framework for mapping and integration of heterogeneous biomedical ontologies. The framework detects possible false mappings intelligently and provides intuitive interfaces for users to customize mappings and for applications to integrate and access biomedical ontologies.

1. Introduction

Ontology is defined as a formal explicit specification of a shared conceptualization [11]. Ontologies aim at capturing static domain knowledge in a generic way and provide a commonly agreed understanding of that domain, which may be reused and shared across applications and groups [4]. The feasibility and desirability of single comprehensive ontology for molecular biology versus several smaller task oriented ontologies have been extensively debated in the community [24]. It seemed much more efficient and effective to have several smaller or sub-domain ontologies which take less time and expertise to grow and maintain and therefore are in the position to be put to use much sooner [24]. Many such ontologies are emerging in various bioinformatics domains to represent and define domain knowledge [24]. A typical example is the widely-used Gene Ontology (GO) [3] developed by the Gene Ontology Consortium, which also supports other ontologies and makes these ontologies freely available, called Open Biomedical Ontologies (OBO) [37]. For microarray experiments, some ontologies are available at the Web site of Microarray Gene Expression Data Society (MGED).

More and more applications ([8], [13], [21], [30] and [31]) are in need to utilize multiple heterogeneous ontologies across various biomedical domains, such as anatomy, drug, gene and disease ontologies. To facilitate such applications, it is urgent to reuse and interoperate heterogeneous ontologies. As summarized by Pinto and Martins [28], there are two reuse processes [26]: *merge* and *integration*. Significant amount of research on both merge ([10], [22], [23], [29] and [32]) and integration operations ([1], [24] and [27]) have been conducted. Tools and methodologies to help in the merge and integration process are now available ([17], [23] and [28]).

Both merge and integration however produce a static ontology based on the existing ontologies. The result ontology is relatively hard to evolve with the change of existing ontologies. Once the result is generated, any minor change will result in a new merge or integration process. The biomedical research community constantly generates new knowledge from latest research results and thus corresponding ontologies are updated frequently. The evolving characteristics of biomedical ontologies present a great challenge for ontology reuse, and a dynamic integration is desirable to meet the requirements. For example, text mining of biomedical publications may obtain assistance from biomedical ontologies to infer high quality unidentified knowledge, such as gene to disease relationship [30]. Concepts in different domains may use different terms [32]. When two ontologies need to communicate or exchange information, the prerequisite is that a consensus has to form between them, i.e. mapping between two ontologies. For a drug-disease relationship inference system, minor errors or divergence in mapping of drug to disease ontology may cause the inference process astray. For practical applications, careful domain expert interventions are needed to ensure accuracy of mapping. Because no algorithms can produce a perfect mapping, the mapping generation process is inherently semi-automatic. Thus a user-friendly and systematic approach to mapping generation and integration is

desirable.

This paper proposes an ontology warehouse, called Oasis, and provides interface for applications to access heterogeneous ontologies from the biomedical research community. Oasis provides an ontology mapping tool to generate mappings among ontologies. The warehouse also enables incremental update of terms and mappings of ontologies in response to knowledge update. Oasis provides a systematic approach to mapping, integration and storage of heterogeneous ontologies.

2. Framework Design

2.1 System Structure

Oasis is based on the existing database for Open Biomedical Ontologies (OBO). Ontologies in Oasis are mapped to each other through mappings stored in a table. Oasis also provides a mapping generation tool, called Interactive Ontology Mapping Generator (IOMG), for users to generate and customize the mappings.

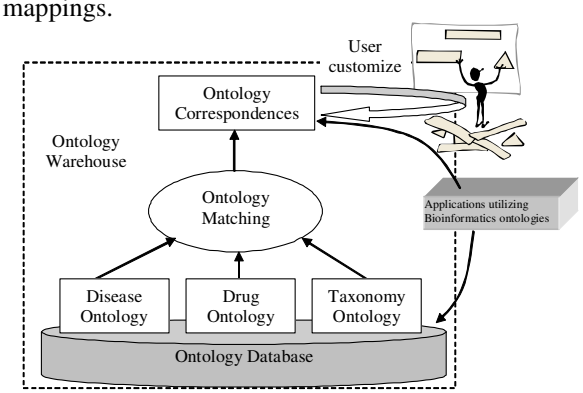

Figure 1 Ontology Mapping and Integration Framework

Figure 1 illustrates the architecture of Oasis, the ontology warehouse. In Oasis, users can customize semantic correspondences between classes and provide semantic guidance for applications, such text mining system, to induce knowledge. The Oasis system consists of two components:

1. Ontology database that consists of tables to store terms, definitions and associations of any ontology;
2. Mapping tables with a mapping generation tool that constructs and stores semantic correspondences between ontology concepts, and hence effectively glues independent ontologies together.

The warehouse provides two interfaces:

1. Graphical user interfaces to customize semantic correspondences between ontologies;
2. Query interfaces for other applications, such as biomedical literature mining system, to access the ontology warehouse.

The ontology warehouse defines a set of tables for heterogeneous ontologies of various biomedical domains. These tables define the superset of OBO ontology so that it is easy to import OBOs. Some existing OBOs, such as Pathway ontology [37] and Disease ontology [36], have been imported into the warehouse.

Semi-automatically and interactively, IOMG generates correspondences between two terms based on a reached consensus that they define the same concept, for example Parkinson disease in the Pathway ontology and Paralysis agitans in the Disease ontology. Mapping generation process of IOMG will be described in more details in section 2.3.

2.2 Ontology Database Design

As shown in Figure 2, the ontology database includes three major parts: OBO tables, proprietary tables, and mapping tables.

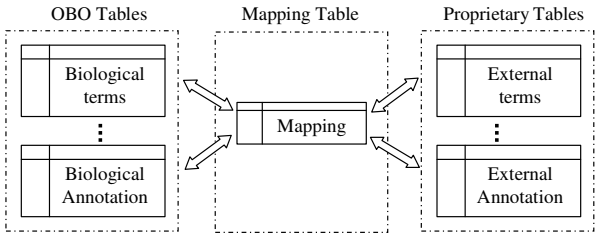

Figure 2 Ontology Database Tables

Inherited from GO, OBO style ontologies are organized as a graph. The GO or OBO terms are represented with nodes, while the relationships, such as is-a and has-a relationships, between them are arcs. In the Go database, the graph is stored in two tables: term and term2term, respectively [34]. GO terms are stored in the table term, while term-to-term relationships are in the table term2term.

As the time of writing, GO includes [34]:
19413 terms, 95.2% with definitions;
10341 biological_processes;
1680 cellular_components;
7392 molecular_functions.

Similarly, one of OBOs, Disease Ontology Version 2.1 contains 19136 concept nodes, and is going to grow to 90k nodes in Version 3. Another related ontology is Pathway ontology, which currently contains 427 terms and describes various kinds of biological networks, the relationships between them and the alterations or malfunctioning of such networks within a hierarchical structure.

These ontology concepts, prefixed with short names, are stored in table term.

Other than OBOs, various formats of vocabulary are being developed across the Internet in the biomedical domain, such as those listed by MGED. These ontologies may have specific ontology structure and definitions. To integrate and utilize these ontologies, a set of proprietary tables are designed. According to requirements of these ontologies, the set of proprietary tables can be expanded with new tables for such ontologies.

A concept in the biomedical domain may be defined with different terms in different ontologies, such as Parkinson disease in disease ontology and pathway ontology. Mapping between terms is therefore important for applications that interoperate multiple ontologies. These mappings are stored in a mapping table, which inherits from the design of GO database mapping table.

2.3 Mapping Generation

These mappings between terms in the ontology warehouse are generated by a mapping generation tool, i.e. the IOMG. The IOMG is based on a previously developed interactive schema matching algorithm with some adaptations for ontology mapping generation. The tool automatically generates mappings to reduce the effort of manually creating mappings and also provides a graphical user interface for customizing results under careful investigation of experts. With the help of a detection algorithm, the tool prompts likely false mappings to users for further investigation and confirmation.

OBO terms are organized in structures called Directed Acyclic Graphs (DAG), where a child term can have multiple parents. OBO represents two types of relationships: is-a and part-of relationships. We utilize similar representation of OBO for our mapping generation tool to represent other ontologies. Mappings between two ontologies consist of a set of one-to-one mappings between a pair of terms, each from one of the input ontologies. A successful mapping implies that two terms may define the same concept.

Mappings between two ontologies are represented by a set of bi-directional and dotted links, each of which defines correspondence of two nodes from the two ontologies. These links together with two DAGs constitute a mapping graph, as shown in Figure 3.

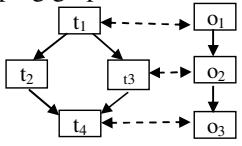

Figure 3 A Mapping Example

Based on the graphical representation of ontologies and mappings, the mapping generation process proceeds as follows: calculating similarities between terms, generating initial mapping pairs, and selecting and customizing false mappings.

2.3.1 Calculating Term Similarities. By comparing terms of ontologies, IOMG firstly generates similarity values between two terms based on certain metrics, such as linguistic, definitions and structures. Through each metrics IOMG obtains a similarity value according to certain characteristics of a pair of elements. Similarity values are in the range [0, 1]. The larger the value is, the more similar the two elements are. Our prototype relies on the following metrics:

Linguistic similarity: linguistic similarity between every pair of terms is based on their names. A string match is utilized in this case. The synonymy table of OBO is also considered helpful for the calculation.

Definition similarity: certain terms in the ontology warehouse have definitions attached, for example 95.2% of terms in GO have definitions. The definition for each term is a reliable source for calculating its similarity value with other terms. By employing a text classification algorithm, a similarity value between two terms is obtained by comparing the two definitions.

Neighbor similarity: neighbors of a term are parents and children of the node. The probability of a term is similar to another one is high if the neighbors of both are mapped. The neighbor similarity is combination of similarity values from parents and children.

IOMG computes a similarity table based on these similarity values by the three metrics. Assume we have n similarity tables, each of which has similarity values $Sim_i\ (a, b),\ i = 1..n$, for any pair of elements *(a, b)*. For each pair of elements *(a, b)* from ontologies A and B, the overall similarity *Sim (a, b)* can be calculated by:

$$Sim(a,b) = \frac{\sum_{i=1}^{n}(Sim_i(a,b) \times w_i)}{n}, \text{ where } \sum_{i=1}^{n}(w_i) = 1 \cdot$$

The weight w_i of each similarity table reflects the preference and importance of the table. Domain experts can adjust weights to achieve high accuracy.

The approach can also utilize result from other mapping algorithms by a conversion, i.e. a table entry (a, b) =0 if a and b are mapped, or (a, b) = 1 otherwise. The mechanism greatly increases the IOMG's extensibility and customizability, and allows the IOMG to exploit different mapping algorithms for other domains to improve the performance.

2.3.2 Generating Mapping Pairs. Based on the similarity values, the IOMG generates a set of initial mapping pairs via a selection algorithm.

The initial mapping pair selection process is closely related to the well-known matching problem of bipartite graphs [15]. In the graph matching literature, a matching is defined as a mapping with cardinality, i.e., in a set of edges no more than two are incident on the same node. A bipartite graph is the one whose nodes form two disjoint parts such that no edge connects any two nodes in the same part. Thus, a mapping can be viewed as an undirected weighted bipartite graph [16].

Our selection algorithm is based on the Kuhn's Hungarian assignment algorithm [14]. The algorithm attempts to maximize the total similarity value of matched elements.

To reduce computation, we adapt the idea of k-nearest neighbor algorithm [9] used in the data mining community. The algorithm only chooses the highest k number of similarity values from the similarity table for a single term to reduce the number of comparisons for similarity values.

Given a k, two ontologies, O_1 and O_2, with m and n terms respectively, and a table Sim [m][n] to represent the similarity values between terms of O_1 and O_2 *of*: $O_1 \times O_2 \rightarrow R$, where R $\in$ [0, 1]. The algorithm eliminates similarity values other than those k highest similarity values for a single term. Then the algorithm tries to find the bijection function $f: O_1 \rightarrow O_2$ such that maximizes the output of the following objective function:

$$\arg\max_{f}\left(\sum_{i=1}^{m} Sim[i][f(i)]\right)$$

In some cases, input ontologies have few terms that can be matched, while the matching algorithm tries to maximize the number of matched elements and therefore produces many false mappings. To reduce such false mappings, the algorithm chooses mapping pairs with similarity values above a predefined threshold, T, i.e. $\forall i(Sim[i][f(i)] >= T)$.

2.3.3 Detecting False Mappings As shown in Figure 4, each node represents a term and connected to other terms by directional links representing is-a or part-of relationships.

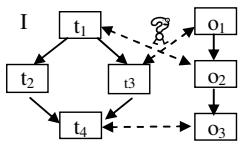

Figure 4 A Mapping Cycle Example

As a DAG is acyclic, a mapping graph should also be acyclic. Consider the example mapping graph in Figure 4 that has a cycle due to the mappings between t_1 and o_2 and between t_3 and o_1. Suppose DAGs of OBOs represent is-a relationships, semantically, the cycle implies that a parent, or a more general term, inherits from a child, or a less general term. A cycle in a mapping graph therefore indicates contradiction to the definition of is-a relationship. These mappings marked with question marks that cause cycles are not desirable in practical applications. This characteristic of the mapping graph is utilized to help identifying possible false mappings and assists users to refine result mappings.

```
Bool Cycle(MappingGraph g, MappingPairs pairs)

1:      For each pair p(ai, bj) in pairs;
2:        If ((descendent s (ai) X ancestors (bj)) U
3:             (ancestors (ai) X descendent s(bj)))
4:          ∩ pairs ≠ empty then
5:        Return true;
6:      End for

End
```
Figure 5 Detecting Cycles

We propose a fast cycle detection algorithm as described in Figure 5. A cycle exists if an element's ancestors match other term's descendents. The ancestor set and descendent set of an element are computed during the conversion of input ontologies and can be used thereafter since the input ontologies are unchanged throughout ontology matching to improve the performance.

3. Results

We provide a warehouse design for heterogeneous ontologies, such as OBOs and other proprietary ontologies. These proprietary ontologies can be imported into the warehouse through proprietary designated interface programs, while tools are available for importing OBOs into the warehouse. To interoperate these ontologies, mapping tables were designed to store the correspondences between terms of these ontologies.

The IOMG is implemented to generate mappings among the ontologies retrieved from the warehouse and detects possible false mappings for domain experts to confirm. Figure 6 shows the IOMG mapping generation interface.

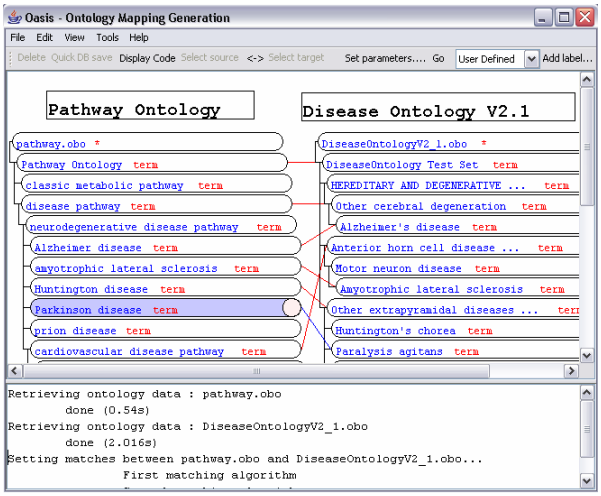

Figure 6 Oasis Ontology Mapping

As highlighted, Parkinson disease is successfully mapped to Paralysis agitans, which is synonym to Parkinson disease. The automatic generation may produce some false mappings too, or miss some true mappings. Result mappings should be verified by biomedical experts before being saved into the ontology warehouse. Through a friendly user interface, domain experts can correct mappings by editing the mappings directly in the graph.

4. Discussion

Other than GO, many biomedical ontologies are freely available. For example, Disease Ontology [36] was designed to facilitate the mapping of diseases and associated conditions to particular medical codes such as ICD9CM, SNOMED and others. The Pathway Ontology [37] captures various kinds of biological networks, the relationships between them and the alterations or malfunctioning of such networks within a hierarchical structure.

These biomedical ontologies have been widely used within biomedical research community. Utilizing ontologies to guide text and data mining for biomedical research has recently achieved encouraging progress. N. Tiffin and colleagues successfully used a controlled vocabulary of anatomical terms, the eVOC Anatomical System ontology, to match tissues associated with diseases to genes expressed in those tissues [30]. Based on the gene ontology variant, Textpresso is a new text-mining system for scientific literature whose capabilities go far beyond those of a simple keyword search engine [21]. In a similar way, many other techniques have been reported successfully for ontology application in text and data mining for bioinformatics ([8], [13] and [31]).

On the other hand, many approaches for generic ontology integration and mapping have been proposed ([2], [4], [7], [18], [19] and [20]). H.L. Johnson and colleagues implemented a system to discover relationships among the Gene Ontology and three other OBO ontologies: ChEBI, Cell Type, and BRENDA Tissue [12]. Existing approaches however concentrate on algorithms rather than a systematic approach. Since matching algorithm inherently requires expert intervention, a friendly user interface with semi-automatic mapping generation mechanism greatly eases the process. Our approach is based on an ontology warehouse and concentrates on biomedical ontologies, such that it provides an ease-to-use user interface and programming interface for further applications.

5. Conclusion

Ontology is not a static model by itself such that it must have the potential to capture changes of meanings and relations. As such, mapping and evolving ontologies are part of an essential task of ontology learning and development [6]. This paper has presented a warehouse system, called Oasis, to integrate existing ontologies. Oasis provides tables to integrate heterogeneous biomedical ontologies, including OBO formats and other formats. To dynamically manage evolving ontologies and mappings, this paper presented an ontology matching tool, the IOMG, to generate mappings for heterogeneous ontologies. The algorithm significantly reduces manual effort by producing initial mappings and prompts possible adaptations to the results through a detection algorithm. IOMG also provides intuitive interfaces for users to customize the results.

The framework as a whole provides a systematic approach to the integration of heterogeneous biomedical ontologies. Recently text and data mining techniques produce great results by utilizing ontologies. Oasis will further greatly improve productivity for medical knowledge discovery and management.

References

[1] J. Arpirez-Vega, A. G´omez-P´erez, A. Lozano-Tello and H. Sofia Pinto. Reference Ontology and (ONTO) Agent: the Ontology Yellow Pages. *Knowledge and Information Systems*, 2(4): pp. 387 - 412, 2000.

[2] M. Bonifacio and P. Bouquet. Enabling Distributed Knowledge Management: Managerial and Technological Implications, *Novatica and Informatik/Informatique*, vol. 3, 2002.

[3] E. Camon, D. Barrell, V.Lee, E. Dimmer. and R. Apweiler. The Gene Ontology Annotation (GOA)

Database—an integrated resource of GO annotations to the UniProt Knowledgebase. In *Silico Biol*., 4, pp. 5–6. 20, 2004.
[4] S. Castano, A. Ferrara & S. Montanelli. P. Traverso. Dynamic Knowledge Discovery in Open, Distributed and Multi-Ontology Systems: Techniques and Applications. *Web Semantics and Ontology*, Idea Group Publishing, 2005.
[5] B. Chandrasekaran, J. Josepheson, and V.R. Benjamins. Ontologies: What are they? Why do we need them? *IEEE Intelligent Systems*, 14(1): pp. 20-26, 1999.
[6] Y. Ding and S. Foo. Ontology Research and Development, Part 1 – a Review of Onlotogy. *Journal of Information Science,* pp.123 – 136, 2002.
[7] A. Doan, P. Domingos, J. Madhavan and A. Halevy. Learning to map between ontologies on the semantic web. In *Proceedings of the 11th International World Wide Web Conference*, 2002.
[8] J. Freudenberg, and P. Propping, A similarity-based method for genome-wide prediction of disease-relevant human genes. *Bioinformatics*., 18, S110–S115, 2002.
[9] K. Fukunaga and M. Narendra. A branch and bound algorithm for computing k-nearest neighbors. *IEEE Trans. Computing* 24 pp. 750-753, 1975.
[10] A. Gangemi, D. Pisanelli, G. Steve. Ontology Integration: Experiences with Medical Terminologies. In N. Guarino (ed.), *Formal Ontology in Information Systems,* IOS Press, pp. 163-178, 1998.
[11] T.R. Gruber, A translation approach to portable ontology specifications, *Knowledge Acquisition* 5 pp. 199–220, 1993.
[12] H.L. Johnson, K. B. Cohen, W. A. Baumgartner Jr., Z. Lu, M. Bada, T. Kester, H. Kim, and L. Hunter. Evaluation of Lexical Methods for Detecting Relationships Between Concepts from Multiple Ontologies, *PSB Online Proceedings* 2006.
[13] P. Khatri, P. Bhavsar, G. Bawa and S. Draghici, Onto-Tools: an ensemble of web-accessible, ontology-based tools for the functional design and interpretation of high-throughput gene expression experiments. *Nucleic Acids Res.*, 32, pp. 449–456, 2004.
[14] H. Kuhn. The Hungarian Method for the Assignment Problem, *Naval Research Logistics Quarterly*, 2, pp. 83-97, 1955.
[15] W. Li and C. Clifton. SemInt: A Tool for Identifying Attribute Correspondences in Heterogeneous Databases Using Neural Network. *Data and Knowledge Engineering* 33: 1, pp. 49-84, 2000.
[16] J. Madhavan, P. Bernstein, and E. Rahm. Generic Schema Matching with Cupid. In *Proc. 27th Intl. Conference on Very Large Databases (VLDB)*, Roma, Italy, pp. 49 - 58, 2001.
[17] D. McGuiness, R. Fikes, J. Rice, and S.Wilder. An Environment for Merging and Testing Large Ontologies. In A. Cohn, F. Giunchiglia, B. Selman (eds.), *Proc. KR2000*, Morgan Kaufmann, pp. 483-493, 2000.
[18] P. Mitra, G. Wiederhold, and M. L. Kersten. A graph-oriented model for articulation of ontology interdependencies. In *Extending Database Technology*, pp. 86 – 100, 2000.
[19] P.Mitra, N.F. Noy, A.R. Jaiswal. Ontology Mapping Discovery with Uncertainty, in *Fourth International Conference on the Semantic Web (ISWC-2005),* Galway, Ireland.
[20] P. Mitra and G. Wiederhold. Resolving terminological heterogeneity in ontologies. In *Workshop on Ontologies and Semantic Interoperability at the 15th European Conference on Artificial Intelligence (ECAI)*, 2002.
[21] H.M. Muller, EE. Kenny, and P.W. Sternberg. Textpresso: an ontology-based information retrieval and extraction system for biological literature. *PLoS Biol.* 2004 Nov;2(11):e309. Epub 2004.
[22] A. Newell. The Knowledge Level. *Artificial Intelligence*, 18(1): pp. 87-127, 1982.
[23] N. Noy and M. Musen. PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment. In *Proc. AAAI2000*, AAAI Press, pp. 450-455, 2000.
[24] S. Schulze-Kremer. Ontologies for molecular biology and bioinformatics. In *Silico Biol.*;**2**: pp. 179–193, 2002.
[25] H. Sofia Pinto. Towards Ontology Reuse. *In Proc. of AAAI99's Workshop on Ontology Management*, AAAI Press, pp. 67-73, 1999.
[26] H. Sofia Pinto, A. G´omez-P´erez, J. P. Martins. Some Issues on Ontology Integration. In *Proc. of IJCAI99's Workshop on Ontologies and Problem Solving Methods: Lessons Learned and Future Trends*, 1999.
[27] H. Sofia Pinto, J.P. Martins. Reusing Ontologies. In Proc. of AAAI2000 Spring Symposium Series, *Workshop on Bringing Knowledge to Business Processes,* AAAI Press, pp. 77- 84, 2000.
[28] H.S. Pinto and J. Martins, A Methodology for Ontology Integration, *K'CAP 2001 Proceedings*, ACM Press, pp. 131-138, 2001.
[29] J. Sowa. Knowledge Representation: logical, philosophical and computational foundations. *Brooks/Cole*, 2000.
[30] N. Tiffin, JF. Kelso, AR. Powell, H. Pan, VB. Bajic, WA. Hide. Integration of text- and data-mining using ontologies successfully selects disease gene candidates. *Nucleic Acids Research* 33(5): pp. 1544-1552, 2005.
[31] F.S. Turner, D.R. Clutterbuck, and C.A. Semple, POCUS: mining genomic sequence annotation to predict disease genes. *Genome Biol.*, 4, R75, 2003.
[32] G. Wiederhold. Interoperation, Mediation and Ontologies. In *Symposium on the 5th Generation Computer Systems, Workshop on Heterogeneous Cooperative Knowledge-Bases*, 1994.
[33] L. Yue and W.C. Reisdorf, Pathway and Ontology Analysis: Emerging Approaches Connecting Transcriptome Data and Clinical Endpoints, *Current Molecular Medicine*, pp. 11-21, 2005.
[34] http://www.godatabase.org/
[35] http://viscomp.utdallas.edu/ggs.htm
[36] http://diseaseontology.sourceforge.net/
[37] http://obo.sourceforge.net/cgi-bin/detail.cgi?pathway