USING REASONING TO GUIDE ANNOTATION WITH GENE ONTOLOGY TERMS IN GOAT

MICHAEL BADA[1], DANIELE TURI[1], ROBIN MCENTIRE[2], ROBERT STEVENS[1]
{mbada|dturi|robert.stevens}@cs.man.ac.uk, Robin.A.McEntire@gsk.com

[1]University of Manchester
Department of Computer Science
Kilburn Building, Oxford Road
Manchester M13 9PL UK

[2]GlaxoSmithKline
709 Swedeland Road
King of Prussia, PA 19406-0939 USA

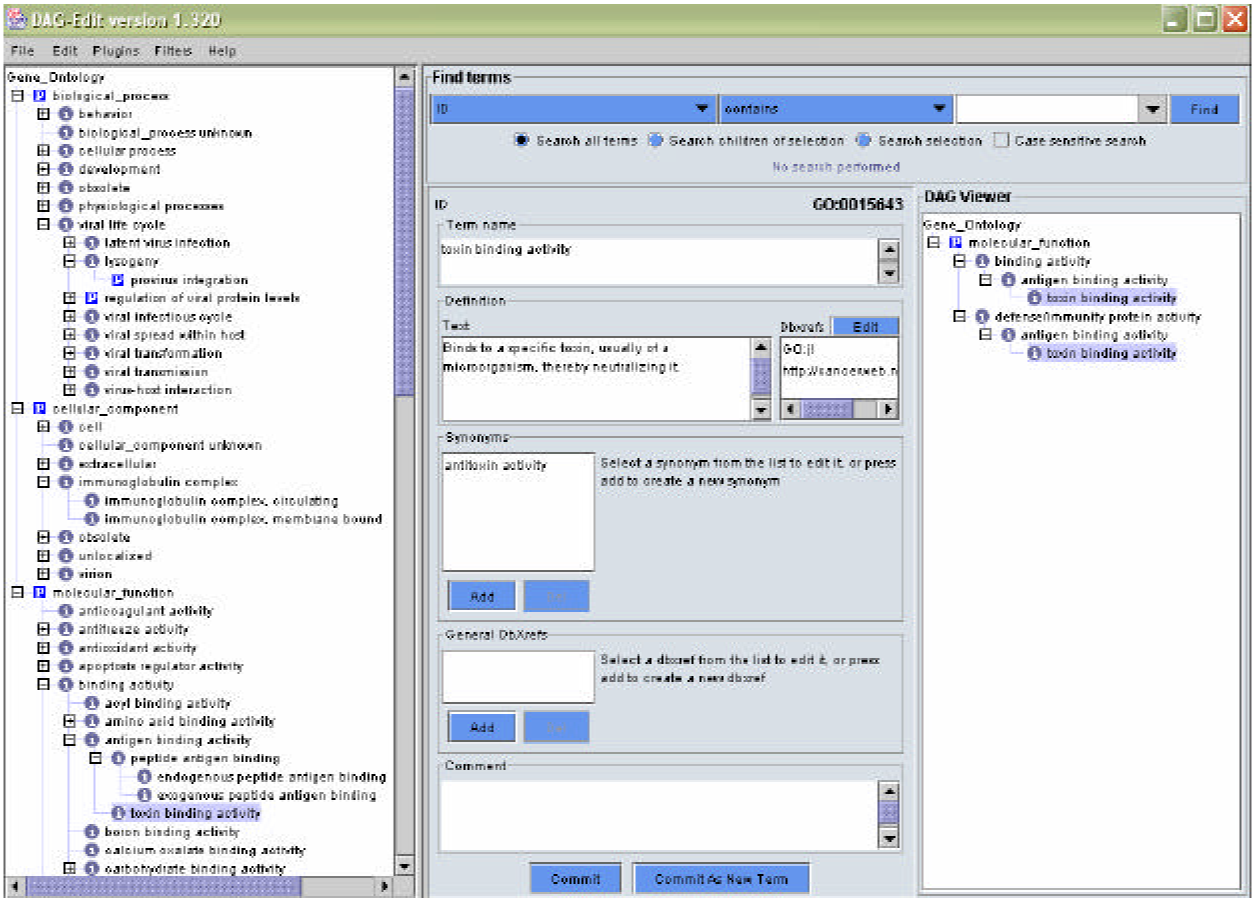
Abstract

High-quality annotation of biological data is central to bioinformatics. Annotation using terms from ontologies provides reliable computational access to data. The Gene Ontology (GO), a structured controlled vocabulary of nearly 17,000 terms, is becoming the *de facto* standard for describing the functionality of gene products. Many prominent biomedical databases use GO as a source of terms for functional annotation of their gene-product entries to promote consistent querying and interoperability. However, current annotation editors do not constrain the choice of GO terms users may enter for a given gene product, potentially resulting in an inconsistent or even nonsensical description. Furthermore, the process of annotation is largely an unguided one in which the user must wade through large GO subtrees in search of terms. Relying upon a reasoner loaded with a DAML+OIL version of GO and an instance store of mined GO-term-to-GO-term associations, GOAT aims to aid the user in the annotation of gene products with GO terms by displaying those field values that are most likely to be appropriate based on previously entered terms. This can result in a reduction in biologically inconsistent combinations of GO terms and a less tedious annotation process on the part of the user.

Introduction

There now exist many biological databases containing enormous quantities of entries of genes and gene products along with descriptions and data about a wide variety of their functional properties. However, the synonymy and polysemy of the descriptive terms and the lack of explicit relationships among them hampers consistent, reliable querying of and interoperability between these databases. In response to this, the Gene Ontology (GO) [1] (Figure 1), a structured controlled vocabulary of nearly 17,000 terms, has been (and is being) developed to be used to functionally describe the gene products of various organisms, for which it is becoming the *de facto* standard. GO is divided into three subontologies of terms (most of which also have natural-language definitions) which may be used to annotate gene products in terms of the molecular functions they possess, the higher-level biological processes in which they are involved, and the cellular locations in which they are active. Each term of each of these subontologies is related to each respective parent term via an *is-a* or a *part-of* relationship.

GO has been a success in that its terms are being used to functionally annotate genes and gene products in a number of prominent biological databases. However, as GO continues to increase in size, users find it increasingly difficult to find the terms they wish to use for annotation. Furthermore, although a large vocabulary is provided, the terms have no links to each other apart from those relationships that form the three taxonomic/partonomic hierarchies. Thus, beyond this hierarchical information, there are no constraints within GO that can be used to indicate which terms should or should not be used together in the annotation of a given gene product. It is possible (though unlikely) that an annotator, in describing a protein, could willfully associate the terms "viral life cycle", "amino-acid biosynthesis", and "extracellular matrix" to that protein; it is more likely that he would accidentally do so. In either case, this is likely to be biologically nonsensical. Good annotation relies upon the domain expertise of the annotator and the usability of the annotation tool. We seek to improve upon the latter by creating formal relationships between pairs of GO terms (as well as between GO terms and gene-product types) mined from biological databases and building an application that, relying upon these relationships, can dynamically retrieve and present those GO terms that are most likely to be applicable for a given gene product based on the GO terms and the gene-product type already entered by the user for that gene product. Thus, if an annotator has already selected “viral life cycle” as a biological-process term and then indicated that she wanted to add a molecular-function term, she would be presented with those molecular-function terms that have been used as annotating terms along with “viral life cycle” (as well as those terms’ descendants). In this paper we introduce the Gene Ontology Annotation Tool (GOAT), which aims to use description logic, associated reasoning, and a store of GO-term associations to guide the annotation process.



Find terms: ID, contains

● Search all terms ○ Search children of selection ○ Search selection □ Case sensitive search

No search performed

ID GO:0015643

Term name: toxin binding activity

Definition

Text: Binds to a specific toxin, usually of a microorganism, thereby neutralizing it.

Dbxrefs: GO:jl, http://sanoerweb.n

Synonyms: antitoxin activity

Select a synonym from the list to edit it, or press add to create a new synonym

General DbXrefs

Select a dbxref from the list to edit it, or press add to create a new dbxref

Comment

Figure 1. A screenshot of a portion of GO in DAG-Edit [2], a tool for editing controlled vocabularies that are represented as directed acyclic graphs. The left pane displays the GO hierarchies, from which "toxin binding activity" has been selected. The lower center and right panes show information for this term, including its natural-language definition, synonym, external-database reference, and all of its ancestor terms. A term related to a parent via an *is-a* relationship is shown with a circled "i" to the left of the term, and a term related to a parent via a *part-of* relationship is shown with a squared "p" to the left of the term.

Approach

GOAT is closely related to another project at the University of Manchester named GONG (Gene Ontology Next Generation) [3]. The goal of GONG is to convert the present GO into a description-logic-based ontology (specifically, in DAML+OIL [4]) and then to further enrich it with formally represented biological knowledge. Our DAML+OIL version of GO is loaded into FaCT [5], a classifier of description-logic-based ontologies, allowing us to reason easily with its component terms.

We also make use of an instance store [6] to hold associations between GO terms. The instance store is an effort towards reasoning over instances in a description-logic representation by using a database to store assertions about instances as well as information inferred from a reasoner, which can reduce the amount of reasoning needed. While each concept (*i.e.*, class) of our DAML+OIL GO ontology represents a GO term, each instance of the instance store is an association record for a corresponding GO-term concept in the ontology. Each association record refers to its corresponding GO term and to the set of other GO terms with which that term is associated. Most of the associations are between pairs of GO terms in different subontologies of GO (*e.g.*, between a molecular-function term and a biological-process term). However, there are also many links between pairs of terms within the same subontology (*e.g*, between two cellular-component terms). This latter type of association would be used when a user wanted to assign more than one term to a given GO attribute (*e.g.*, for a multifunctional gene product or for one that is potentially active in multiple cellular locations).

The GO-term-to-GO-term associations were mined from the complete version of GOA (Gene Ontology Annotation) [7], a database holding all GO-term annotations of entries in the databases of UniProt (a comprehensive resource for information about proteins) and Ensembl (a project that maintains information about large genomes). We examined each GO-term annotation in GOA that represents neither an unknown term (*e.g.*, "unknown biological process") nor an obsolete term and that has an evidence code that we deem reliable. (Each annotation is given an evidence code by the annotator that indicates the type of evidence that she cites in assigning the annotation. The evidence codes that we assessed as relatively unreliable are IEA (inferred from electronic annotation), NAS (nontraceable author statement), ND (no biological data available), and NR (not recorded); thus, any annotation having one of these evidence codes was ignored.) We compiled associations of GO terms in the sense that the two terms that make up each associative pair (*i.e.*, a given GO term and (one of) its associated GO term(s)) have been used together as annotating terms in at least one UniProt entry of GOA. Both terms of the association must satisfy all three of the aforementioned criteria. In addition, for each association in which the given GO term and its associated term are located in the same subontology, neither of these terms can be taxonomically or partonomically more specific than the other term, since such associations (*e.g.*, a term and a term it subsumes) are trivial and are already explicitly represented or can be inferred from GO.

We further grouped these associations into records such that, given a GO term, its association record contains, in addition to the GO term to which it refers, all other GO terms associated with it in the manner described above. None of the terms from an association record can similarly be taxonomically or partonomically more specific than any other term in the record; thus, any such annotating term, if encountered, is excluded from the record. Following all of the previously mentioned constraints, we have represented over 600,000 GO-term-to-GO-term associations. Each of these association records is represented in the instance store as an instance with its corresponding description in DIG [8], an XML-based logical interface language for various description-logic reasoners, including FaCT. DIG does not provide extended expressiveness but is rather a common representation that can be used to communicate with these reasoners; thus, since we use DIG, we could plug in a different reasoner. With DIG, one can make conceptual expressions (*e.g.*, top, conjunction, existential restriction), tell (*e.g.*, to clear a knowledge base, to define a concept, that one concept implies another), or ask (*e.g.*, for all concepts of a knowledge base, if a concept is satisfiable, for the parents of a concept).

An example instance of an association record can be seen in Figure 2. Using the DIG description of this figure, if a user entered "cell morphogenesis checkpoint" as a biological-process term and then indicated that he wished to add a molecular-function term, GOAT would query the instance store for the association record for "cell morphogenesis checkpoint", which would return the description of Figure 2. The tool would then parse this description for any associated molecular-function terms, resulting in this case in "protein tyrosine kinase". The reasoner would then subsequently be queried to retrieve all taxonomic and partonomic descendants of "protein tyrosine kinase" for display to the user as additional plausible choices for molecular-function annotation.

The three GO subontologies are formed from a mixture of taxonomy and partonomy, as each child can be related to each of its parents via *is-a* or *part-of*. Thus, when we refer to taxonomically or partonomically more specific terms of a given GO term, this includes the term's subsumptive children, its direct parts, its subsumptive children's children, its subsumptive children's direct parts, its direct parts' children, its direct parts' direct parts, and recursively onward. When GOAT displays the associated terms of a given GO term, it presents not only the terms that have been explicitly associated with the term (in the term's association record) but also all of those terms' taxonomic and partonomic descendants, as we encourage the user to annotate the given gene product as specifically as possible. Given a term, we can easily retrieve these descendants by querying the FaCT reasoner into which our DAML+OIL version of GO is loaded as part of GOAT. Thus, by not representing these more specific associated terms in the instance store, we reduce the size of the instance store and instead further reuse our ontology.

The second type of these associations is that between GO terms and gene-product types (*i.e.*, types of biological molecules), which were obtained from the various prominent organism-specific databases that use GO terms to annotate their gene-product entries (*e.g.*, the *Saccharomyces* Genome Database [9], which concentrates on the yeast *Saccharomyces cerevisiae*). The entries of most of these databases do not have structured fields that classify them into gene-product types, and thus, there is no easy way to automatically mine for this type of association. Instead, the databases were manually searched and examined, resulting in a small set of existential restrictions (added directly to our DAML+OIL version of GO) for the most general terms to which each gene-product type was found to be associated.

```
<and>
<some>
<ratom name=”is+association+set+for”/>
<catom name=”cell+morphogenesis+checkpoint”/>
</some>
<some>
<ratom name=”has+associated+nonsubsumed+GO+molecular+function+in+GOA”/>
<catom name=”protein+tyrosine+kinase”/>
</some>
</and>
```

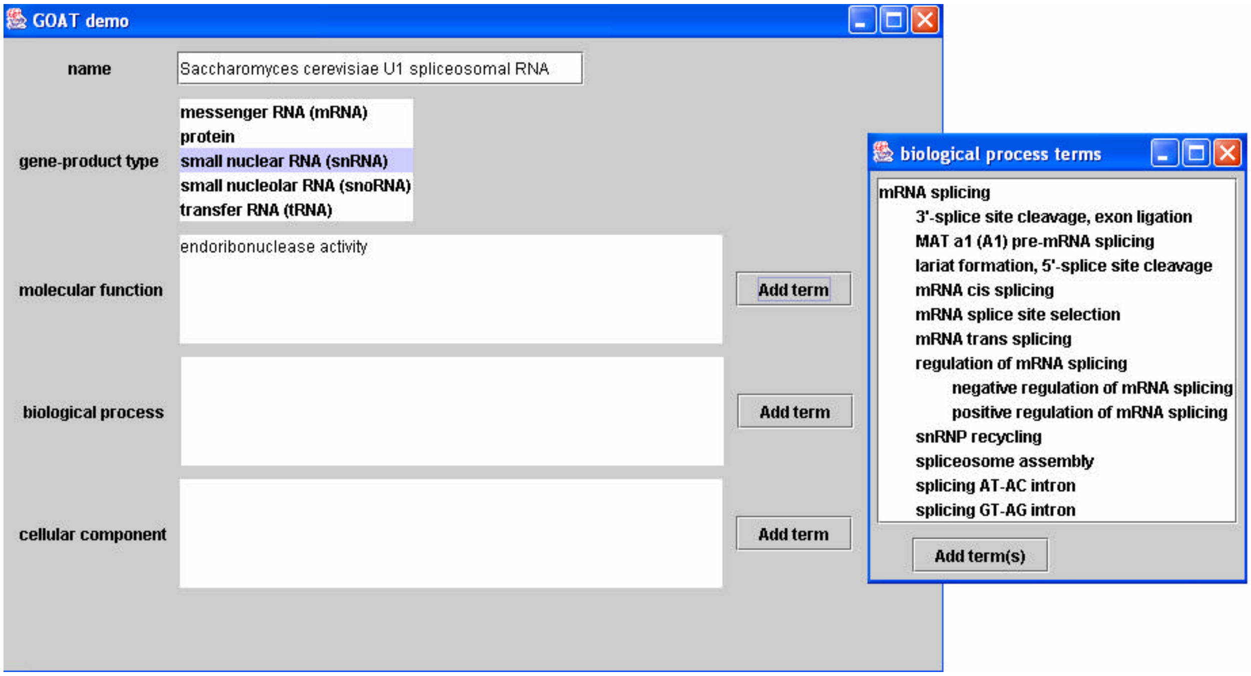
Figure 2. DIG description of the instance that is the association record for the GO biological-process term “cell morphogenesis checkpoint”. This is a conjunction of two existential restrictions, the first representing the fact that this is the association record for “cell morphogenesis checkpoint” and the second detailing its most general associated term, namely one molecular-function term (“protein tyrosine kinase”). In predicate logic, this description is equivalent to ∃x is association set for(cell morphogenesis checkpoint(x)) ∧ ∃y has associated nonsubsumed GO molecular function in GOA(protein tyrosine kinase(y)). The namespaces and namespace delimiters of the role and concept names have been omitted for space, and the role and concept names themselves have been URL-encoded. This is but a simple example, as an instance’s description can have any number of associated terms from any of the three subontologies so long as they meet the criteria described in the text.

We assumed that proteins can be annotated with almost any GO term and instead concentrated on finding terms associated with other types of molecules (*e.g.*, tRNAs). These types of macromolecules have more restricted functions (and processes and cellular locations) that can be used to pare a given GO subontology down to a more manageable size for presentation to the user. It turns out that this is nicely complementary: We have no restrictions linking GO terms to “protein”, but there is a very large number of GO-term-to-GO-term associations resulting from specific protein annotations in the instance store. Conversely, there are no annotations for anything other than proteins in GOA and thus no resulting GO-term-to-GO-term associations for these molecules in the instance store, but we do have specific GO-term-to-gene-product-type restrictions for them.

Discussion

Translation to a description logic and augmentation with formal term definitions and relationships among the terms can result in a richer, more consistent GO that is open to machine reasoning. Tools driven by this formally represented knowledge can then be built to guide users in specific tasks. Such an example is GOAT, which uses a reasoner loaded with this DAML+OIL version of GO and an instance store of association records encoded in a logical formalism to guide biomedical researchers in the annotation of gene products with GO terms. Specifically, we plan to guide these users by presenting those terms that are most likely appropriate to enter for a given field given the values that have been entered or chosen for the fields up to that point. Currently, life scientists lack such tools and instead must largely rely upon their own expertise and slow traversal of large GO hierarchies.

Figure 3 shows a screenshot of GOAT as currently implemented. Each gene-product annotation can include a free-text name, a gene-product type selected from an enumerated list, and a set of one or more GO terms for each of the three GO subontologies. It is in these GO-subontology fields where users can be aided in the process of annotation. For example, as shown in Figure 3, the user has entered “endoribonuclease activity” in the molecular-function field and has chosen “small nuclear RNA (snRNA)” from the menu of gene-product types. Upon indicating that she wishes to enter a value for the biological-process field, GOAT will dynamically attempt to retrieve the association record for “endoribonuclease activity” from the instance store by forming and submitting DIG queries. It also asks the reasoner directly for GO-term concepts that are explicitly associated with “small nuclear RNA” in the loaded DAML+OIL version of GO. A set of biological-process terms found to be associated with all of this information is determined, and these terms’ descendants are dynamically retrieved from the ontology. This subset of GO terms is shown as a tree (in the familiar GO format rather than that of our DAML+OIL version) in a pop-up window, from which she may choose one or more terms as field values. Thus, instead of searching for the appropriate term(s) through all of GO, the user is presented only with values likely to be appropriate.



name: Saccharomyces cerevisiae U1 spliceosomal RNA

gene-product type:
- messenger RNA (mRNA)
- protein
- small nuclear RNA (snRNA)
- small nucleolar RNA (snoRNA)
- transfer RNA (tRNA)

molecular function: endoribonuclease activity

biological process:

cellular component:

mRNA splicing
- 3'-splice site cleavage, exon ligation
- MAT a1 (A1) pre-mRNA splicing
- lariat formation, 5'-splice site cleavage
- mRNA cis splicing
- mRNA splice site selection
- mRNA trans splicing
- regulation of mRNA splicing
    - negative regulation of mRNA splicing
    - positive regulation of mRNA splicing
- snRNP recycling
- spliceosome assembly
- splicing AT-AC intron
- splicing GT-AG intron

Figure 3. Screenshots of GOAT. The main window is shown on the left; in it can be seen a free-text field for the gene-product name, an enumerated list of gene-product types, and a field for each of the three subontologies of GO, each of which may have one or more GO terms. The right window is displaying biological-process terms that are likely to be applicable to the gene product being annotated based on the information that the user has already entered for the gene product, shown in the main window.

The terms that GOAT determines to be most likely relevant and presents to the user are based on previous annotations, which ostensibly correspond to what is known biologically. Even though a new combination of terms could represent novel biological knowledge, this is discouraged, as a computational agent does not know that it is a valid new combination. Also, we emphasize that these are only suggested terms. We plan on adding a button for each field that lists all terms of the given subontology if the annotator cannot find the term(s) he wishes to use among the list of these suggested terms.

Most of the hard work in determining the GO terms' various types of associated terms was done by mining GOA and further processing the resulting data. We have explicitly represented all top-level associated terms for each term (rather than, for example, query FaCT repeatedly for the associated terms of a given term's ancestors) for performance reasons. FaCT is useful, however, when more than one piece of information has already been entered for a gene product and the user indicates that she would like to add another GO term: Any term should be displayed only once within the presented list of terms. However, any of the associated terms of one term might subsume any of the associated terms of another; thus, we cannot simply show all associated terms and all of their descendants. We must instead ask FaCT for a subsumption check for each permutation of two associated terms in determining the minimal set of top-level associated terms. We also use FaCT to retrieve all taxonomic and partonomic descendants of the associated terms, but this is essentially a transitive closure over the *is-a* and *part-of* relations and thus could be easily accomplished in other ways.

Our biggest issue with which to deal is the speed of the tool: It takes approximately two and a half minutes to load an DAML+OIL version of GO with 13,349 terms into FaCT (which is done upon first launching GOAT). Retrieving and presenting all 1,153 terms of the cellular-component hierarchy (which optimally requires 1,153 × 2 = 2,306 FaCT queries in order to retrieve each term's children and direct parts) also takes approximately two and a half minutes; thus, we are looking at alternate ways of doing this. We are in the process of moving from DAML+OIL to OWL, and we would also like to perform evaluations of the tool and get feedback from real users. GOAT is not yet ready as a complete annotation tool, but a prototype can be downloaded from its Web site at http://goat.man.ac.uk.

Various approaches to GO-term prediction have been attempted, including those based on natural-language processing [10], sequence analysis [11], and microarray data [12]. We differ from the large majority of these approaches in that we are deriving associations among the GO terms themselves and using these associations to predict GO terms based on terms already used in annotating a given gene product. A similar emphasis is found in the work of King *et al*., who have probabilistically modeled such associations with decision trees and Bayesian networks [13]. Correspondingly, among the information that QuickGO [14], a GO browser, displays for a given term are other GO terms that have often been used together with the term in question in gene-product annotations. GOAT seeks to expand upon such functionality by relying upon a formal reasoner and an instance store of associations derived from sophisticated mining of GOA. We aim to facilitate the annotation process by having the tool retrieve and present all appropriate terms rather than forcing the user to keep track of the potentially many associations a given set of one or more terms may have. Also, we believe it beneficial that such functionality is built directly into the annotation tool.

The functional annotation of gene products with GO terms is a vital part of enabling consistent, reliable querying of biomedical databases. Using description logic representation and reasoning, GOAT can help guide the user through the annotation process by suggesting the terms most likely to be appropriate for a gene product. In addition to a likely reduction in biologically inappropriate combinations of annotation terms, this can make the process less tedious and more satisfying by reducing the amount of verbiage through which the user must to wade to reach the appropriate terms. Improving the quality of semantic annotation and easing its production can thus facilitate the progress of biomedical computational science.

Acknowledgments

GOAT is funded by the DTI and EPSRC under the e-Science programme via ESNW. We would also like to acknowledge the collaboration of our industrial partner GlaxoSmithKline.

Bibliography

[1] The Gene Ontology Consortium (2000). Gene Ontology: tool for the unification of biology. *Nature Genetics*, **25**: 25-29.

[2] http://www.geneontology.org/GO.Curators.intro.html#dagedit

[3] Wroe, C., Stevens, R., Goble, C., and Ashburner, M. (2003). A Methodology to Migrate the Gene Ontology to a Description Logic Environment using DAML+OIL. *2003 Pacific Symposium on Biocomputing Proceedings (PSB)*, Hawaii, USA, January 2003.

[4] Horrocks, I. (2002). DAML+OIL: a Reason-able Web Ontology Language. *Proceedings of Extending Database Technology (EDBT) 2002*, Prague, the Czech Republic, March 2002.

[5] Horrocks, I. (1999). FaCT and iFaCT. In: Lambrix, P., Borgida, A., Lenzerini, M., Möller, R., and Patel-Schneider, P., eds. *Proceedings of the International Workshop on Description Logics (DL'99)*, Linköping, Sweden, July-August 1999.

[6] http://instancestore.man.ac.uk/instancestore.pdf

[7] Camon, E., Magrane, M., Barrell, D., Binns, D., Fleischmann, W., Kersey, P., Mulder, N., Oinn, T., Maslen, J., Cox, A., and Apweiler, R. (2003). The Gene Ontology Annotation (GOA) Project: Implementation of GO in SWISS-PROT, TrEMBL, and InterPro. *Genome Research*, **4**, 662-672.

[8] http://potato.cs.man.ac.uk/dig/interface1.0.pdf

[9] Issel-Tarver, L., Christie, K. R., Dolinski, K., Andrada, R., Balakrishnan, R., Ball, C. A., Binkley, G., Dong, S., Dwight, S. S., Fisk, D. G., Harris, M., Schroeder, M., Sethuraman, A., Tse, K., Weng, S., Botstein, D., and Cherry, J. M. (2002). *Saccharomyces* Genome Database. *Methods of Enzymology*, **350**, 329-346.

[10] Raychaudhuri, S., Chang, J. T., Sutphin, P. D., and Altman, R. B. (2002). Associating genes with gene ontology codes using a maximum entropy analysis of biomedical literature. *Genome Research*, **12**, 203-214.

[11] Hennig, S., Groth, D., and Lehrach, H. (2003). Automated Gene Ontology annotation for anonymous sequence data. *Nucleic Acids Research*, **31** (13), 3712-3715.

[12] Hvidsten, T. R., Komorowski, J., Sandvik, A. K., and Lægreid, A. (2001). Predicting gene function from gene expressions and ontologies. *2001 Pacific Symposium on Biocomputing Proceedings (PSB)*, Hawaii, USA, 299-310.

[13] King, O. D., Foulger, R. E., Dwight, S. S., White, J. V., and Roth, F. R. (2003). Predicting Gene Function From Patterns of Annotation. *Genome Research*, **13**, 896-904.

[14] http://www.ebi.ac.uk/ego/