

GoBean: a Java GUI application for visual exploration of GO term enrichments

Sanghyuk Lee^{1,2}, Ji-Young Cha³, Hyeonjin Kim⁴ & Ungsik Yu^{3,*}

¹Ewha Research Center for Systems Biology, Division of Molecular Life Sciences, Ewha Womans University, Seoul 158-710, ²Korean Bioinformation Center, Korea Research Institute of Bioscience and Biotechnology, Daejeon 305-600, ³Department of Molecular Medicine, College of Medicine, Gachon University, Incheon 406-840, ⁴Department of Medical Sciences, College of Medicine, Seoul National University, Seoul 151-742, Korea

We have developed a biologist-friendly, Java GUI application (GoBean) for GO term enrichment analysis. It was designed to be a comprehensive and flexible GUI tool for GO term enrichment analysis, combining the merits of other programs and incorporating extensive graphic exploration of enrichment results. An intuitive user interface with multiple panels allows for extensive visual scrutiny of analysis results. The program includes many essential and useful features, such as enrichment analysis algorithms, multiple test correction methods, and versatile filtering of enriched GO terms for more focused analyses. A unique graphic interface reflecting the GO tree structure was devised to facilitate comparisons of multiple GO analysis results, which can provide valuable insights for biological interpretation. Additional features to enhance user convenience include built in ID conversion, evidence code-based gene-GO association filtering, set operations of gene lists and enriched GO terms, and user -provided data files. It is available at <http://neon.gachon.ac.kr/GoBean/>. [BMB reports 2012; 45(2): 120-125]

INTRODUCTION

GO term enrichment analysis - the identification of GO terms that are significantly overrepresented in a given set of genes, is a method frequently used to acquire initial insights into the potential biological meaning of 'interesting' gene lists obtained from various high-throughput biological studies. Many enrichment analysis tools are available (1) that can be used to confirm existing knowledge concerning the underlying biology or the generation of new hypotheses, based on functional group-

ings of genes and terms, and the way they are connected. However, these tools are still in an active state of growth and improvement, with no unified methods or established standards (2, 3). A common practice in GO analysis is to try several applications of different functional aspects to reach biological insights.

Tools with seemingly similar capabilities and functionalities can often lead to quite different conclusions when applied to experimental data. Factors that can cause such wildly differing results for the same input data include the method(s) used to map gene identifiers, the sources and versions of annotation files, the method of annotation propagation, statistical testing methods, the actual mathematical formulas used for calculations, and the multiple testing correction methods. Minor differences or unsophisticated implementation of the above factors can lead to quite different analysis results, but only a few tools actually state their implementation details explicitly (4). Comparing and merging the results from multiple analysis tools is quite challenging due to the various differences.

Currently, due to the limitations of current enrichment analysis methodologies, the enrichment analysis process is more of an exploratory procedure for the identification of functional groups and relevant biological trends. It is guided by enrichment P-values, prior biological knowledge relevant to the experiments, and the data collected through exploration of the genes and annotation categories (3). A graphic representation of the analysis results in the hierarchical context of GO provides another means to assess the functional coherence of a gene set beyond relying on P-values alone. Extensive interactive graphic manipulations are desirable for effective visual exploration of analysis results because the graphic representations are usually quite complex, even for small numbers of significant terms. Most current GO tools with graphic output features deliver graphic results in a fixed format and do not support user interactions or modifications.

A need exists to compare the GO enrichment status from different experiments. New insights missed by analyzing each experiment alone may be uncovered by identifying commonly and specifically enriched GO terms in each experiment. Knowing the relationships between different GO terms would

*Corresponding author. Tel: +82-32-899-6052; Fax: +82-32-899-6039; E-mail: ungsik@gachon.ac.kr
<http://dx.doi.org/10.5483/BMBRep.2012.45.2.120>

Received 11 October 2011, Revised 31 October 2011,
Accepted 3 November 2011

Keywords: Bioinformatics, Gene list analysis, Gene Ontology, Java GUI application

help biologists better understand their experimental results from a functional point of view. Comparing results from 2 different programs or even from the same program with different input parameters typically involves a painstaking process of manual inspection. Comparison becomes much simpler if the results can be viewed in the context of a directed acyclic graph (DAG) structure, but only a few tools offer such features, even in a limited way.

RESULTS

GoBean performs popular GO term enrichment statistical analyses in conjunction with an intuitive visual display to present GO terms, trees, and paths in a manner that facilitates biological interpretation. GoBean supports many unique features for the visual explorative analysis of enrichment results, such as customizable views in table, tree, and graph forms, elaborate filtering of enriched GO terms, graphic comparison of multiple enrichment results, and detailed GO term enrichment status information. Fig. 1 presents a sample screenshot of the GoBean application showing a graphic comparison of 3 enrichment analysis results.

Analysis result views

Each analysis result is presented as a new tab in the main display area, with table and tree views. The views consist of the enriched GO terms, coverage, and enrichment P-values. In the tree view, each node can be expanded or collapsed as needed. When the number of enriched GO terms is small (<15), the graph view is activated for the enriched terms. Table rows, tree rows, and graph nodes are color-coded according to their GO categories and adjusted P-values. Each node in the graph view

shows the GO ID, GO name, and corresponding adjusted P-value. The graph edge is color-coded according to the relationship between the connecting nodes. Features for graph view manipulation, such as toggling and moving nodes, adjusting edges, panning, and zooming of the graph, are provided. A graph with as few as 5-10 significant GO terms can be quite complex, depending on the GO level of the GO terms, because many intermediate GO terms in a GO hierarchy are required for GO DAG creation. Graph view manipulation features permit elaborate interactive manipulations for user customization, and the visibility of complex graphs can be greatly enhanced. The analysis result can be exported into a table, a tree, or a graphic image. The relevant annotated genes are included in the table and tree exports.

Analysis result filtering

The program supports extensive filtering of enriched GO terms for focused investigation of analysis results. Various combinations of GO category, P-value criteria, coverage, and the include/exclude term list can be used for filtering. Hypothesis-driven queries - testing of whether some set of GO terms is enriched in a particular set of genes, are possible by specifying the set of terms as an "include term list" in a filtering condition. Functional categorizations - categorization of genes based on a relatively small set of pre-defined high-level GO terms, such as GO slim (5), are also made possible by specifying the GO terms as an include term list. Multiple testing corrections are not relevant for hypothesis-driven queries and functional categorizations because the number of tests is small. Coverage-based filtering can be used to filter out terms with many assigned gene IDs that are too general, or terms with too few assigned gene IDs, that are susceptible to experimental devia-

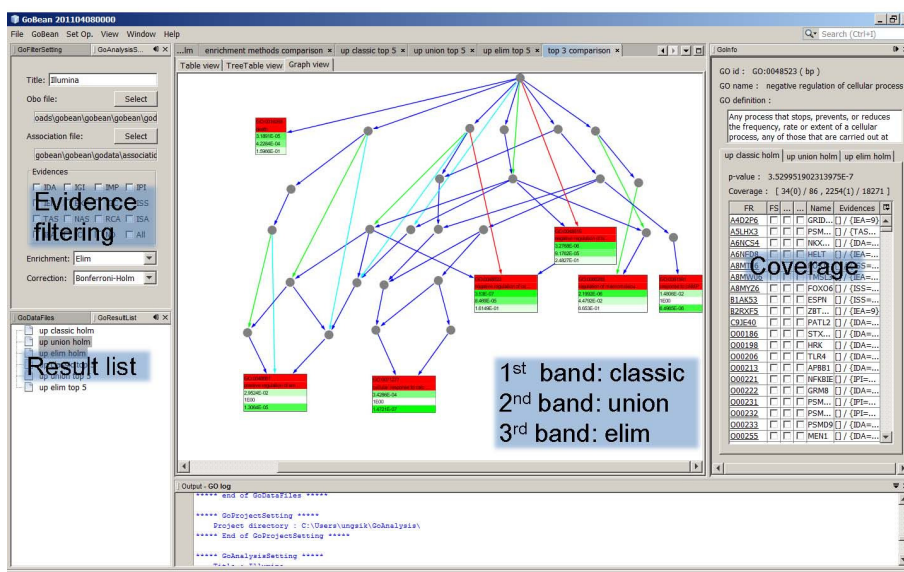


Fig. 1. Sample screenshot showing a graphic comparison of 3 enrichment analysis results. The main window shows a cross-comparison of enrichment results from classic, Parent-Child-Union, and Topology-Elim algorithms in a graph view. The top 3 significant GO terms from each result are merged and shown as a DAG. Significant GO terms are shown as square nodes, while intermediate GO terms are toggled off as gray dots. For each toggled-on graph node, P-values are shown in a colored stripe with color saturation given by the corresponding P-value. The differences in P-values are evident. Edges are colored according to the relationship between the connected nodes. The Goinfo window on the far right shows detailed information concerning the selected enriched GO term in the graph view.

tions. GO term filter settings can be saved and loaded for later use. Manual selection of enrichment GO terms to create a new result view is possible in the table view. For example, to view only a certain number of custom selected significantly enriched GO terms, the table view can be sorted according to the adjusted P-value, and the desired GO terms can be selected to create a new view.

Cross-comparison of analysis results in a graphic context

Cross-comparison of the GO enrichment status of multiple enrichments, one of the unique features of GoBean, is readily available. The comparison result can be examined in table and/or tree views. The comparison result can be filtered, as described for single enrichment analysis. A graph view is activated when the number of the resulting GO terms is small (<15). In the graph view, the adjusted P-values of an enriched GO term are shown in the GO term node as colored stripes with color saturations given by the P-value. Graph node and graph edge manipulations can also be performed to clarify the view.

Enriched term information

When an enriched GO term is selected in one of the 3 views, detailed information about the enriched term, such as GO definition, coverage, and P-value, is shown in the adjacent window panel. Coverage shows the reference IDs and study IDs assigned to the GO term and corresponding evidence codes. These can be used to judge the reliability of the enrichment. For example, enriched GO terms with a reference ID count <5 can be considered less reliable because a statistical analysis is not appropriate for questions with such a small sample size. Significant enrichment terms with assigned IDs mostly

with IEA (inferred from electronic annotation) evidence code may also be viewed as less reliable. Link-out capabilities for GO terms and gene IDs are provided for detailed information about a GO term or gene ID where applicable.

Usability features

GoBean also includes many miscellaneous features for user convenience. A built-in system for conversion of gene ID into the supported ID type is provided for organisms including human, mouse, rat, chicken, zebra fish, fruit fly, and yeast using the Ensembl Biomart Martview service (6). The ID conversion result is divided into 3 groups consisting of unique, multiple, and failed (unmapped and missing) mappings, for visual checking of conversion results. The mapping from one type of ID to another is not always one-to-one, and multiply mapped IDs usually occur. Converted IDs can be saved as a file for later use. Analysis conditions, such as obo file name, association file name, evidence codes, enrichment algorithm, and multiple testing correction procedures, can be saved and loaded for convenience. Set operations such as 'and,' 'or,' 'intersection,' and 'union' are available for enriched GO terms and IDs. The reference gene ID list or study gene ID list can be modified in the graphic window and saved for future analyses or records.

Comparison of GoBean with related tools

Table 1 illustrates comparisons of GoBean with similar GO enrichment tools, such as GoMiner (7), GOEAST (8), GOstat (9), DAVID (10), FatiGO (11), Onto-Express (12), Ontologizer (13), Gorilla (14), and GOFFA (15). Evaluations of the available implementations and associated documentation as of June 2011 are used for the comparison. GoBean supports 3 powerful statistical algorithms with established multiple testing correction

Table 1. Comparison of statistical analysis and data handling capabilities of GoBean with similar GO enrichment tools

Tool	Statistical model	Enrichment methods	Multiple testing corrections	Local obo, association file support	Evidence code based filtering	'NOT' qualified association removal
GoBean	Hypergeometric	Classic, Parent-Child, Topology-elim	Yekutieli, Hochberg, Holm, Bonferroni	Both	Y	Y
GoMiner	Fisher's	Classic	q-value	Annotation	Y	-
GOEAST	Hypergeometric, Fisher's, Chi-square	Classic, Topology-weighted	Yekutieli, Hochberg, Bonferroni, Hommel	Annotation	-	-
GOstat	Fisher's	Classic	Yekutieli, Benjamini, Holm	Annotation	-	Y
DAVID	Modified Fisher's	Classic	Yekutieli, Benjamini, Bonferroni	Annotation	-	-
FatiGO	Fisher's	Inclusive analysis	Yekutieli, Hochberg, Westfall-Young	Annotation	-	Y
Onto-express	Hypergeometric, Fisher's, binomial, Chi-square	Classic	Benjamini, Bonferroni, Holm, Sidak	Annotation	-	Y
Ontologizer	Hypergeometric	Many	Many	Both	-	Y
GOrilla	Minimum hypergeometric	Classic	Not applicable	-	-	-
GOFFA	Fisher's	Classic	E-value	-	-	-

Table 2. Comparison of explorative analysis capabilities of GoBean with similar GO enrichment tools

Tool	Visualization	Interactive DAG	Result filtering	Multiple results comparison
GoBean	Table, tree, DAG	Y	Extensive	Y
GoMiner	Table, tree, DAG	-	-	-
GoEAST	Table, DAG	-	P-value	Y (max.: 3)
GoStat	Table	-	P-value	-
DAVID	Table	-	-	-
FatiGO	Table, DAG	-	P-value	-
Onto-Express	Table, DAG	Y	Extensive	-
Ontologizer	Table, DAG	-	P-value	-
GOrilla	Table, DAG	-	P-value	-
GOFFA	Table, tree, DAG	Y	P-value, E-value	-

methods. GoBean covers a wide variety of input parameter choices. It supports full obo files, custom association files, evidence code-based association filtering, and custom reference gene IDs. The ability to use custom annotations provides a short cut to overcome outdated or incomplete annotation information. The other tools, except Ontologizer, do not support local ontology files, and the specific version of ontology in use may be unclear. Because a computationally derived gene-GO association is regarded as less reliable than a curated one, including or excluding certain types of annotations may be important for certain types of analyses. GoBean and GoMiner can limit associations to specific evidence codes without changing the association file by changing input parameters, which can be useful in certain cases. Other tools do not support such filtering even if the use of a local annotation file is possible, and users have to prepare an annotation file consisting of only annotations with desired evidence codes. GoBean also properly handles the 'NOT' qualified annotations.

Table 2 summarizes a comparison of the explorative analysis capabilities of GoBean with other tools. Some tools, such as GOEAST, GoMiner, Ontologizer, topGO (16), Gorilla, and GOFFA, do provide graph views of enrichment results in the context of GO DAG. Most of the tools providing graph views use the Graphviz tool (17). The graph view using Graphviz is a static image and can provide quite limited, if any, interactivity and other features. Only GoBean, OntoExpress, and GOFFA provide interactive graph views. Many tools provide simple P-value-based filtering of enrichment results, but the filtering condition is enacted when submitting an analysis. GoBean and Onto-Express provide extensive filtering capabilities of enrichment results for further analyses. Cross-comparison of the GO enrichment status of multiple enrichments enables the discovery of information missed by analyzing each experiment alone. GoBean and Multi-GOEAST provide such multiple result graphic cross-comparisons, but Multi-GOEAST is limited to 3 results. GoBean can cross-compare

more than 3 results and the resulting graph can be interactively manipulated. Additionally, the graph is simpler than with Multi-GOEAST, because fewer nodes are displayed.

DISCUSSION

GoBean is a standalone GO enrichment analysis tool capable of extensive visual explorative analysis. It supports 3 powerful statistical algorithms for enrichment analysis with multiple testing correction methods. The result can be examined in 3 types of views: table view, tree view, and graph view, for effective exploration. Diverse sorting options and links in the table view provide detailed information on enriched GO terms. Tree and graph views, reflecting the hierarchical structure of GO categories, help users gain insights into the biological meaning of the gene lists. Unlike other GO applications, GoBean's graph view supports elaborate interactive manipulations for user customization. Versatile filtering of enrichment results is provided in all 3 views for a more focused investigation. Elaborate investigation of enrichment results is made possible by combining the versatile filtering and interactive graphic view.

Comparison of results from different GO analyses is often essential for deducing biological interpretations. GoBean provides an intuitive interface to compare several GO analysis results. It can be used to compare the enriched GO terms from different optional parameters/methods for the same gene list or for different gene lists, often by clustering analyses of microarray data.

GoBean supports many desirable features, such as local obo and annotation data files, built-in ID conversion, and set operations for gene lists. It also addresses the issue of obscure implementation differences by using a clearly defined data handling scheme. To our knowledge, it is the first application allowing direct comparison of enrichment results in the hierarchical context of GO with explorative graphic manipulations.

MATERIALS AND METHODS

GoBean was designed to be a comprehensive and flexible GUI tool for GO term enrichment analysis, combining the merits of many other programs. GoBean was developed as a self-contained standalone Java GUI application and packaged as a zip archive. GoBean is machine- and OS-independent, and operable under JRE 1.6 with no external dependency. We chose technologies such as Java, Swing, and the NetBeans Platform (18) to ensure stability, wide accessibility, usability, and extensibility.

Differences in enrichment results originate from many factors. We addressed the issue of obscure implementation differences of those factors by using a clearly defined data handling scheme. Table 3 summarizes our implementation choices for these factors.

Algorithms and statistics

P-values are calculated using the upper tail of a hypergeometric distribution probability. As Fisher's exact test is equivalent to a hypergeometric test (19) and a binomial test is a large sample

Table 3. Factors influencing enrichment analysis results and their handling in GoBean

Factor	Implementation
Major factors	
Version of ontology file and gene-GO association file	Use of the latest local data files is supported
Evidence code based association filtering	Various combinations of evidence code based filtering are possible
Statistical test	Hypergeometric test
Enrichment algorithms	Classic, Parent-Child, Topology-Elim
Multiple testing corrections	None, Bonferroni, Bonferroni-Holm, Benjamini-Hochberg, Benjamini-Yekutieli
Minor factors	
Hypergeometric probability calculation	<i>Phyper</i> function of 'R' statistical package is adapted.
Associations of gene IDs to alternative GO terms	Associations are assigned to the representative GO terms
'NOT' qualified associations	Removed
ID conversion	Built-in ID conversion
Reference gene IDs missing in association file	Removed
Study gene IDs missing in association file or reference gene IDs	Removed
Duplicated reference IDs or study IDs	Duplications are eliminated
Monotonicity of adjusted-P-values	Monotonicity is enforced by stepping down from the largest adjusted-P-value

size approximation for a hypergeometric test, Fisher's exact test and a binomial test are implicitly covered. The *phyper* algorithm for hypergeometric distribution probability calculation of the 'R' statistical package (20) was ported into Java. The implementation was fast enough for most interactive analyses. The Term-for-Term (classic) enrichment algorithm used in many popular tools was implemented. The recently introduced Parent-Child (21) and Topology-Elim (22) enrichment algorithms are also included to deal with implicit dependencies and redundancies among neighboring GO terms. For multiple testing corrections, 4 popular methods were implemented: Bonferroni, Bonferroni-Holm, Benjamini-Hochberg, and Benjamini-Yekutieli methods (23). No multiple testing correction procedure is applicable for Topology-Elim enrichment results because the tests are dependent. Monotonicity of adjusted P-values, obtained by applying the multiple testing corrections, was enforced.

Data handling

The OBO 1.2 ontology file format and GAF 2.0 annotation file format of the GO consortium are supported. Users can use the full ontology file, which includes cross-products, inter-ontology links, and *has-part* relationships, in addition to the filtered ontology file. Gene-GO term associations can be filtered according to evidence codes. Associations assigned to an *alt_id* GO term are mapped to the representative one. Associations with the 'NOT' qualifier are removed. Two OBO files and filtered association files for popular species from the GO consortium are included in the application. These files can be replaced with the latest ontology and local associations by the user, as updates become available.

Reference IDs can be specified with a file. The reference ID type should match the ID type of the corresponding association file. When no reference ID file is specified, all IDs from the association file are treated as reference IDs. Reference IDs that are absent in the association file are removed in the sub-

sequent analysis. Study IDs that are missing in the association file or reference IDs are also removed. A GO sub-graph, consisting of GO terms with assigned reference IDs, is built from the specified combination of obo file, associations, reference IDs, and study IDs. GO term enrichments are evaluated for all GO terms in the sub-graph.

User interface and visualization

We used the NetBeans platform as a general GUI framework. A rich client platform (RCP) provides a GUI framework in which common GUI features can be quickly and simply assembled. Various common GUI features such as window move, resize, and menu bars are systematically incorporated using the NetBeans platform. The NetBeans platform and Eclipse RCP (24) are the most popular RCPs. We chose the NetBeans platform over Eclipse RCP because the NetBeans platform is a pure Swing-based framework and the application developed can be used as long as the JRE is available. Eclipse RCP involves use of some machine- and/or OS-dependent libraries.

The SwingX library (25) was used for table and tree visualization; it provides control of table column visibility, column rearrangement, sorting, and searching in table and tree views. The NetBeans Visual library (26) was used for graphic visualizations and manipulations of enrichment results. We used the NetBeans Validation API for input validation, the Jackson JSON processor library (27) for saving and restoring analysis parameters, and the Batik SVG Toolkit (28) for SVG export of graph views.

Acknowledgements

This work was supported by a GIST systems biology infrastructure establishment grant (2011).

REFERENCES

1. Gene Ontology Tools [<http://www.geneontology.org/GO-tools.shtml>].
2. Khatri, P. and Draghici, S. (2005) Ontological analysis of gene expression data: current tools, limitations, and open problems. *Bioinformatics* **21**, 3587-3595.
3. Huang da, W., Sherman, B. T. and Lempicki, R. A. (2009) Bioinformatics enrichment tools: paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res.* **37**, 1-13.
4. Rhee, S. Y., Wood, V., Dolinski, K. and Draghici, S. (2008) Use and misuse of the gene ontology annotations. *Nature reviews. Genetics* **9**, 509-515.
5. GO Slim and Subset Guide [<http://www.geneontology.org/GO.slims.shtml>].
6. Flicek, P., Amode, M. R., Barrell, D., Beal, K., Brent, S., Chen, Y., Clapham, P., Coates, G., Fairley, S., Fitzgerald, S., Gordon, L., Hendrix, M., Hourlier, T., Johnson, N., Kahari, A., Keefe, D., Keenan, S., Kinsella, R., Kokocinski, F., Kulesha, E., Larsson, P., Longden, I., McLaren, W., Overduin, B., Pritchard, B., Riat, H. S., Rios, D., Ritchie, G. R., Ruffier, M., Schuster, M., Sobral, D., Spudich, G., Tang, Y. A., Trevanion, S., Vandrovcova, J., Vilella, A. J., White, S., Wilder, S. P., Zadissa, A., Zamora, J., Aken, B. L., Birney, E., Cunningham, F., Dunham, I., Durbin, R., Fernandez-Suarez, X. M., Herrero, J., Hubbard, T. J., Parker, A., Proctor, G., Vogel, J. and Searle, S. M. (2011) Ensembl 2011. *Nucleic Acids Res.* **39**, D800-806.
7. Zeeberg, B. R., Feng, W., Wang, G., Wang, M. D., Fojo, A. T., Sunshine, M., Narasimhan, S., Kane, D. W., Reinhold, W. C., Lababidi, S., Bussey, K. J., Riss, J., Barrett, J. C. and Weinstein, J. N. (2003) GoMiner: a resource for biological interpretation of genomic and proteomic data. *Genome Biology* **4**, R28.
8. Zheng, Q. and Wang, X. J. (2008) GOEAST: a web-based software toolkit for Gene Ontology enrichment analysis. *Nucleic Acids Res.* **36**, W358-363.
9. Beissbarth, T. and Speed, T. P. (2004) Gostat: find statistically overrepresented Gene Ontologies within a group of genes. *Bioinformatics* **20**, 1464-1465.
10. Huang da, W., Sherman, B. T. and Lempicki, R. A. (2009) Systematic and integrative analysis of large gene lists using DAVID bioinformatics resources. *Nat. Protoc.* **4**, 44-57.
11. Al-Shahrour, F., Diaz-Uriarte, R. and Dopazo, J. (2004) FatiGO: a web tool for finding significant associations of Gene Ontology terms with groups of genes. *Bioinformatics* **20**, 578-580.
12. Draghici, S., Khatri, P., Bhavsar, P., Shah, A., Krawetz, S. A. and Tainsky, M. A. (2003) Onto-Tools, the toolkit of the modern biologist: Onto-Express, Onto-Compare, Onto-Design and Onto-Translate. *Nucleic Acids Res.* **31**, 3775-3781.
13. Bauer, S., Grossmann, S., Vingron, M. and Robinson, P. N. (2008) Ontologizer 2.0-a multifunctional tool for GO term enrichment analysis and data exploration. *Bioinformatics* **24**, 1650-1651.
14. Eden, E., Navon, R., Steinfeld, I., Lipson, D. and Yakhini, Z. (2009) GOrilla: a tool for discovery and visualization of enriched GO terms in ranked gene lists. *BMC Bioinformatics* **10**, 48.
15. Sun, H., Fang, H., Chen, T., Perkins, R. and Tong, W. (2006) GOFFA: gene ontology for functional analysis—a FDA gene ontology tool for analysis of genomic and proteomic data. *BMC Bioinformatics* **7**(Suppl 2), S23.
16. topGO - Enrichment analysis for Gene Ontology [<http://topgo.bioinf.mpi-inf.mpg.de/>].
17. Graphviz - Graph Visualization Software [<http://www.graphviz.org/>].
18. NetBeans Platform [<http://platform.netbeans.org/>].
19. Rivals, I., Personnaz, L., Taing, L. and Potier, M. C. (2007) Enrichment or depletion of a GO category within a class of genes: which test? *Bioinformatics* **23**, 401-407.
20. The R Project for Statistical Computing [<http://www.r-project.org/>].
21. Grossmann, S., Bauer, S., Robinson, P. N. and Vingron, M. (2007) Improved detection of overrepresentation of Gene-Ontology annotations with parent child analysis. *Bioinformatics* **23**, 3024-3031.
22. Alexa, A., Rahnenfuhrer, J. and Lengauer, T. (2006) Improved scoring of functional groups from gene expression data by decorrelating GO graph structure. *Bioinformatics* **22**, 1600-1607.
23. Noble, W. S. (2009) How does multiple testing correction work? *Nature Biotechnology* **27**, 1135-1137.
24. Eclipse RCP [<http://www.eclipse.org/home/categories/rcp.php>].
25. SwingsLabs Swing Component Extensions [<http://java.net/projects/swingx/>].
26. NetBeans Visual Library [<http://platform.netbeans.org/graph/>].
27. Jackson: High-performance JSON processor [<http://jackson.codehaus.org/>].
28. Batik SVG Toolkit [<http://xmlgraphics.apache.org/batik/>].