SimCT: a generic tool to visualize ontology-based relationships for biological objects

Carl Herrmann1,2,∗, Sèverine Béard3 and Laurent Tichit2,4
1TAGC-U928 Inserm, 2Faculté des Sciences, Université de la Méditerranée, Campus de Luminy Case 928, Marseille, 3Université Montpellier 2, UMR AMAP, and UMR LIRMM – CNRS, 34000 Montpellier and 4IML – UMR 6206 CNRS, Marseille, France

Received on June 5, 2009; revised on August 27, 2009; accepted on September 15, 2009
Associate Editor: Alfonso Valencia

ABSTRACT
Summary: We present a web-based service, SimCT, which allows to graphically display the relationships between biological objects (e.g. genes or proteins) based on their annotations to a biomedical ontology. The result is presented as a tree of these objects, which can be viewed and explored through a dedicated java applet designed to highlight relevant features. Unlike the numerous tools that search for overrepresented terms, SimCT draws a simplified representation of biological terms present in the set of objects, and can be applied to any ontology for which annotation data is available. Being web-based, it does not require prior installation, and provides an intuitive, easy-to-use service.

Availability: http://tagc.univ-mrs.fr/SimCT
Contact: carl.herrmann@univmed.fr
Supplementary information: Supplementary data are available at Bioinformatics online

1 INTRODUCTION

The wealth of data available from large-scale experiments in recent years has made the development of efficient tools to visualize, analyze, interpret and share post-genomic data a crucial endeavor. Among these, biomedical ontologies have been increasingly based on their annotations to an ontology, in the form of the relations between biological objects (e.g. genes, proteins, etc.) and their adoption by genome databases such as MGI, Wormbase and Flybase, will ensure that they will be increasingly used in the community. In this note, we present a generic, web-based tool called SimCT (Similarity Clustering Tool) which allows the visualization of the relationships between biological objects (e.g. genes, proteins, etc.) based on their annotations to an ontology, in the form of a clustering tree. Our clustering procedure is a way to turn the ontology into a simplified tree (which is a subgraph of the ontology), which better represents the terms associated to a list of objects, therefore highlighting their relationships. This representation could neither be obtained by mapping the annotations onto the ontology, due to its complexity, nor by searching for overrepresented terms, which by definition overlooks terms that are not statistically relevant. The visualization is done using a dedicated java applet. Although many tools have been developed for GO, very few comparable tools exist for other biomedical ontologies yet.

2 METHODS

To measure the specificity of a term t in an ontology O, we have introduced the notion of precision as follows (see Supplementary Material for details, in particular the glossary for definition of terms used):

\[ p(t) = \frac{\log N_{da}(t)}{\log N_{ana}(t)} \in [0, 1] \] (1)

where \( N_{da}(t) \) represents the number of descendant terms of t, \( N_{ana}(t) \) the number of ancestor terms of t, N the total number of terms in O and \( N_{ana}(t) \) the maximal number of ancestors a term can have in O. Interestingly, our definition of precision only depends on the structure of the ontology and not on annotation statistics like in Lord et al. (2003). Therefore, it can be applied to any existing ontology. Additionally, precision differs from information content, which gives equal specificity to all leaves of the ontology (Resnik, 1999; Schlicker et al., 2006; Wang et al., 2007). Based on precision, we define the similarity of two terms as the precision of their most precise common ancestor.

Given a list of objects annotated to ontology, we consider the set of (object, ontology term) pairs. If an object has several annotations, it generates several (object/ontology term) pairs. We have implemented an aggregative clustering algorithm that builds the clustering tree based on the similarity between terms. The leaves of the resulting tree are the (object/ontology term) pairs and the internal nodes are ontology terms. We attach to each internal node a numerical index called Subtree Relevance Index (SRI):

\[ SRI(T) = p(t) \times N(T) \] (2)

where \( SRI(T) \) represents the subtree attached to it, \( p(t) \) its precision and \( N(T) \) is the number of leaves of the subtree. It measures the relevance of each term for the list of objects submitted (Supplementary Material). The topology of the tree respects that of the underlying ontology (i.e. it is included in the directed acyclic graph (DAG) of the ontology).

3 IMPLEMENTATION

SimCT can be used in two different ways, depending on the ontology the user is interested in:

(1) With GO, the user can input a list of genes/proteins, select the corresponding organism (29 are currently available) and the GO sub-ontologies. The system retrieves available annotations.

∗To whom correspondence should be addressed.
We chose a set of 69 coregulated genes extracted from Transcriptome Browser (Lopez et al., 2008) around the natural killer (NK) gene NCR3 in human. We compared the P-values of the nodes with SRI ≥ 2.5 with the P-values given by DAVID (Dennis et al., 2003) and GO:TermFinder (Boyle et al., 2004). The differences between the SimCT approach and the search for overrepresented terms are highlighted in Supplementary Table S2. In particular, although no term related to biopolymer synthesis is overrepresented, SimCT detects that five genes of the list are related to transcriptional (TRX21, CEBPD, TAFl, GFI1) or translational (EIF3B, RPS5) processes which are child terms of biopolymer synthesis. These are the effectors at the end of the cascade of NK activity, leading for instance to the production of gamma-interferon.

5 CONCLUSION

Our approach can be compared with GOSurfer (Zhong et al., 2004), GOTreePlus (Lee et al., 2008) or GO:TermFinder (Boyle et al., 2004). However, SimCT includes the possibility to work with other biomedical ontologies than GO, and is web-based. Therefore, it provides an intuitive, easy-to-use and immediately available service, which allows to draw a clear picture of the ontological terms represented in a list of biological objects annotated to ontology, for any biomedical ontology. The viewer applet helps easily exploring and annotating the resulting tree to highlight its most relevant features. As more and more ontologies are being developed, we believe that this tool will prove very useful in working with these.

ACKNOWLEDGEMENTS

We want to thank E. Choron, M. Hermes, P. Bonnaure, L. Baumgaertel and C. Lepoivre for their participation in the developments of the javaTreeViewer applet. We thank P. Rieth, C. Brun and B. Jacq for discussions and feedback.

Conflict of Interest: none declared.

REFERENCES