GO-terms Semantic Similarity Measures

Guangchuang Yu
College of Life Science and Technology
Jinan University, Guangzhou, China
email: guangchuangyu@gmail.com

June 9, 2013

1 Introduction

Functional similarity of gene products can be estimated by controlled biological vocabularies, such as Gene Ontology (GO). GO comprises of three orthogonal ontologies, i.e. molecular function (MF), biological process (BP), and cellular component (CC).

Four methods have been presented to determine the semantic similarity of two GO terms based on the annotation statistics of their common ancestor terms (Resnik [1], Jiang [2], Lin [3] and Schlicker [4]). Wang [5] proposed a new method to measure the similarity based on the graph structure of GO. Each of these methods has its own advantages and weaknesses. *GOSemSim* package [6] is developed to compute semantic similarity among GO terms, sets of GO terms, gene products, and gene clusters, providing both five methods mentioned above. I have developed another package, *DOSE*, for measuring semantic similarity among DO terms and gene products at disease perspective.

To start with *GOSemSim* package, type following code below:

```
library(GOSemSim)
help(GOSemSim)
```

2 Citation

Please cite the following articles when using *GOSemSim*.

G Yu, F Li, Y Qin, X Bo, Y Wu, S Wang. GOSemSim: an R package for measuring semantic similarity among GO terms and gene products. *Bioinformatics*. 2010,26(7):976-978.

3 Semantic Similarity Measurement Based on GO

Four methods proposed by Resnik [1], Jiang [2], Lin [3] and Schlicker [4] are information content (IC) based, which depend on the frequencies of two GO terms involved and that of their closest common ancestor term in a specific corpus of GO annotations. The information content of a GO term is computed by the negative log probability of the term occurring in GO corpus, and is defined as $IC(t) = -\log p(t)$. A rarely used term contains a greater amount of information.

At present, *GOSemSim* supports analysis on many species. We used the following Bioconductor packages to calculate the information content.

- org.At.tair.db for *Arabidopsis*
- org.Ag.eg.db for *Anopheles*
- org.Bt.eg.db for *Bovine*
- org.Cf.eg.db for *Canine*
- org.Gg.eg.db for *Chicken*
- org.Pt.eg.db for *Chimp*
- org.Sco.eg.db for *Coelicolor*
- org.EcK12.eg.db for *E coli strain K12*
- org.EcSakai.eg.db for *E coli strain Sakai*
- org.Dm.eg.db for *Fly*
- org.Hs.eg.db for *Human*
- org.Pf.plasmo.db for *Malaria*
- org.Mm.eg.db for *Mouse*
- org.Ss.eg.db for *Pig*
- org.Rn.eg.db for *Rat*
- org.Mmu.eg.db for *Rhesus*
- org.Ce.eg.db for *Worm*
- org.Xl.eg.db for *Xenopus*
- org.Sc.sgd.db for *Yeast*
- org.Dr.eg.db for *Zebrafish*

The information content will update regularly.

As GO allow multiple parents for each concept, two terms can share parents by multiple paths. We take the most informative common ancestor (MICA), where there is more than one shared parents.

- Resnik's method is defined as:

$$sim_{Resnik}(t_1, t_2) = IC(MICA)$$

- Lin's method is defined as:

$$sim_{Lin}(t_1, t_2) = \frac{2IC(MICA)}{IC(t_1) + IC(t_2)}$$

- The Relevance method, which was proposed by Schlicker's method, combine Resnik's and Lin's method, and is defined as:

$$sim_{Rel}(t_1, t_2) = \frac{2IC(MICA)(1 - p(MICA))}{IC(t_1) + IC(t_2)}$$

- Jiang and Conrath's method is defined as:

$$sim_{Jiang}(t_1, t_2) = 1 - \min(1, d_{Jiang}(t_1, t_2))$$

  where

$$d_{Jiang}(t_1, t_2) = IC(t_1) + IC(t_2) - 2IC(MICA)$$

Graph-based methods using the topology of GO graph structure to compute semantic similarity. Formally, a GO term A can be represented as $DAG_A = (A, T_A, E_A)$ where T_A is the set of GO terms in DAG_A, including term A and all of its ancestor terms in the GO graph, and E_A is the set of edges connecting the GO terms in DAG_A.

- Wang's method

  To encode the semantics of a GO term in a measurable format to enable a quantitative comparison between two term's semantics, Wang firstly defined the semantic value of term A as the aggregate contribution of all terms in DAG_A to the semantics of term A, terms closer to term A in DAG_A contribute more to its semantics. Thus, defined the contribution of a GO term *t* to the semantics of GO term A as the S-value of GO term *t* related to term A. For any of term *t* in $DAG_A = (A, T_A, E_A)$, its S-value related to term A. $S_A(t)$ is defined as:

$$\begin{cases} S_A(A) = 1 \\ S_A(t) = \max\{w_e \times S_A(t') | t' \in children\,of(t)\}\ if\ t \neq A \end{cases}$$

  where w_e is the semantic contribution factor for edge $e \in E_A$ linking term *t* with its child term *t*'. Wang defined term A contributes to its own as one. After obtaining the S-values for all terms in DAG_A, the semantic value of GO term A, SV(A), is calculated as:

$$SV(A) = \sum_{t \in T_A} S_A(t)$$

Thus, given two GO terms A and B, the semantic similarity between these two terms, $sim_{Wang}(A,B)$, is defined as:

$$sim_{Wang}(A,B) = \frac{\sum\limits_{t \in T_A \cap T_B} S_A(t) + S_B(t)}{SV(A) + SV(B)}$$

where $S_A(t)$ is the S-value of GO term *t* related to term A and $S_B(t)$ is the S-value of GO term *t* related to term B.

This method proposed by Wang [5] determines the semantic similarity of two GO terms based on both the locations of these terms in the GO graph and their relations with their ancestor terms.

On the basis of semantic similarity between GO terms, *GOSemSim* can also compute semantic similarity among sets of GO terms, gene products, and gene clusters.

We implemented four methods which called *max*, *avg*, *rcmax*, and *BMA* to combine semantic similarity scores of multiple GO terms. The similarities among gene products and gene clusters which annotated by multiple GO terms were also calculated by the same combine methods mentioned above.

Given two GO terms sets $GO_1 = \{go_{11}, go_{12} \cdots go_{1m}\}$ and $GO_2 = \{go_{21}, go_{22} \cdots go_{2n}\}$ annotated for tow genes g_1 and g_2, four combine methods for calculating gene similarity are defined as follows:

- The *max* method calculates the maximum semantic similarity score over all pairs of GO terms between these two sets.

$$sim_{max}(g_1, g_2) = \max_{1 \le i \le m, 1 \le j \le n} sim(go_{1i}, go_{2j})$$

- The *avg* calcuate the average semantic similarity score over all pairs of GO terms.

$$sim_{avg}(g_1, g_2) = \frac{\sum\limits_{i=1}^{m} \sum\limits_{j=1}^{n} sim(go_{1i}, go_{2j})}{m \times n}$$

- Similarities among GO terms form a matrix, and method *rcmax* use the maximum of RowScore and ColumnScore as the similarity, where RowScore (or ColumnScore) is the average of maximum similarities on each row (or column).

$$sim_{rcmax}(g_1, g_2) = \max\left(\frac{\sum\limits_{i=1}^{m} \max\limits_{1 \le j \le n} sim(go_{1i}, go_{2j})}{m}, \frac{\sum\limits_{j=1}^{n} \max\limits_{1 \le i \le m} sim(go_{1i}, go_{2j})}{n}\right)$$

- The *BMA* method, used the best-match average strategy, calculates the average of all maximum similarities on each row and column, and defined as:

$$sim_{BMA}(g_1, g_2) = \frac{\sum_{1=i}^{m} \max_{1\leq j\leq n} sim(go_{1i}, go_{2j}) + \sum_{1=j}^{n} \max_{1\leq i\leq m} sim(go_{1i}, go_{2j})}{m+n}$$

4 Examples

GOSemSim implemented multiple functions for calculate semantic similarities:

- goSim for calculate semantic similarity between two GO terms.
- mgoSim for calculate semantic similarity among multiple GO terms.
- geneSim for calculate semantic similarity between two gene products.
- mgeneSim for calculate semantic similarity among multiple gene products.
- clusterSim for calculate semantic similarity between two gene clusters.
- mclusterSim for calculate semantic similarity among multiple gene clusters.

The following example demonstrated the function calls of these function, details about the arguments can refer to the manuals (eg `?geneSim`).

```
goSim("GO:0004022", "GO:0005515", ont="MF", measure="Wang")
```

```
 [1] 0.158
```

```
go1 = c("GO:0004022","GO:0004024","GO:0004174")
go2 = c("GO:0009055","GO:0005515")
mgoSim(go1, go2, ont="MF", measure="Wang", combine="BMA")
```

```
 [1] 0.192
```

```
geneSim("241", "251", ont="MF", organism="human", measure="Wang", combine="BMA")
```

```
$geneSim
 [1] 0.207

$GO1
 [1] "GO:0005515" "GO:0004051" "GO:0004364" "GO:0004602"
 [5] "GO:0047485" "GO:0050544"

$GO2
 [1] "GO:0004035"
```

```
mgeneSim(genes=c("835", "5261","241", "994"),
                        ont="MF", organism="human", measure="Wang",
                        verbose=FALSE)
```

```
        835  5261   241   994
835  1.000 0.132 0.536 0.491
5261 0.132 1.000 0.221 0.108
241  0.536 0.221 1.000 0.313
994  0.491 0.108 0.313 1.000
```

```
gs1 <- c("835", "5261","241", "994", "514", "533")
gs2 <- c("578","582", "400", "409", "411")
clusterSim(gs1, gs2, ont="MF", organism="human", measure="Wang", combine="BMA")
x <- org.Hs.egGO
hsEG <- mappedkeys(x)
set.seed <- 123
clusters <- list(a=sample(hsEG, 20), b=sample(hsEG, 20), c=sample(hsEG, 20))
mclusterSim(clusters, ont="MF", organism="human", measure="Wang", combine="BMA")
```

5 Case Study

In [7], we proposed a method for measuring functional similarity of microRNAs. This method was based on semantic similarity of microRNAs' target genes, and was calculated by *GOSemSim*. We furhter analyzed viral microRNAs using this method [7] and compared significant KEGG pathways regulated by different viruses' microRNAs using *clusterProfiler* [8].

6 Session Information

The version number of R and packages loaded for generating the vignette were:

- R version 3.0.1 (2013-05-16), `x86_64-apple-darwin10.8.0`
- Locale: `C/UTF-8/C/C/C/C`
- Base packages: base, datasets, grDevices, graphics, methods, parallel, stats, utils
- Other packages: AnnotationDbi 1.22.6, Biobase 2.20.0, BiocGenerics 0.6.0, BiocInstaller 1.10.1, DBI 0.2-7, DO.db 2.6.0, DOSE 1.99.0, GO.db 2.9.0, GOSemSim 1.19.0, RSQLite 0.11.4, Rcpp 0.10.3, cacheSweave 0.6-1, clusterProfiler 1.9.1, filehash 2.2-1, ggplot2 0.9.3.1, org.Hs.eg.db 2.9.0, stashR 0.3-5
- Loaded via a namespace (and not attached): IRanges 1.18.1, KEGG.db 2.9.1, MASS 7.3-26, RColorBrewer 1.0-5, colorspace 1.2-2, dichromat 2.0-0, digest 0.6.3, grid 3.0.1, gtable 0.1.2, igraph 0.6.5-2, labeling 0.1, munsell 0.4, plyr 1.8, proto 0.3-10, qvalue 1.34.0, reshape2 1.2.2, scales 0.2.3, stats4 3.0.1, stringr 0.6.2, tcltk 3.0.1, tools 3.0.1

References

[1] Resnik Philip. Semantic similarity in a taxonomy: An Information-Based measure and its application to problems of ambiguity in natural language. *Journal of Artificial Intelligence Research*, 11:95–130, 1999.

[2] Jay J. Jiang and David W. Conrath. Semantic similarity based on corpus statistics and lexical taxonomy. *Proceedings of 10th International Conference on Research In Computational Linguistics*, 1997.

[3] Dekang Lin. An Information-Theoretic definition of similarity. *In Proceedings of the 15th International Conference on Machine Learning*, pages 296—304, 1998.

[4] Andreas Schlicker, Francisco S Domingues, JÃűrg RahnenfÃijhrer, and Thomas Lengauer. A new measure for functional similarity of gene products based on gene ontology. *BMC Bioinformatics*, 7:302, 2006. PMID: 16776819.

[5] James Z Wang, Zhidian Du, Rapeeporn Payattakool, Philip S Yu, and Chin-Fu Chen. A new method to measure the semantic similarity of go terms. *Bioinformatics (Oxford, England)*, 23:1274–81, May 2007. PMID: 17344234.

[6] Guangchuang Yu, Fei Li, Yide Qin, Xiaochen Bo, Yibo Wu, and Shengqi Wang. Gosemsim: an r package for measuring semantic similarity among go terms and gene products. *Bioinformatics*, 26:976–978, 2010. PMID: 20179076.

[7] Guangchuang Yu, Chuan-Le Xiao, Xiaochen Bo, Chun-Hua Lu, Yide Qin, Sheng Zhan, and Qing-Yu He. A new method for measuring functional similarity of micrornas. *Journal of Integrated OMICS*, 1(1):49–54, February 2011.

[8] Guangchuang Yu, Le-Gen Wang, Yanyan Han, and Qing-Yu He. clusterprofiler: an r package for comparing biological themes among gene clusters. *OMICS: A Journal of Integrative Biology*, 16:in press, 2012.