Word and sentence embedding tools to measure semantic similarity of Gene Ontology terms by their definitions

Dat Duong [1,*], Wasi Uddin Ahmad [1], Eleazar Eskin [1,3], Kai-Wei Chang [1], Jingyi Jessica Li [2,3,*]

[1] Department of Computer Science, University of California, Los Angeles, United States of America. [2] Department of Statistics, University of California, Los Angeles, United States of America. [3] Department of Human Genetics, University of California, Los Angeles, United States of America.

1 Appendix

1.1 Resnik method

The most basic node-based method introduced by Resnik in 1999 relies on the information content (IC) of a GO term (Resnik, 1999). The IC of a GO term t is computed as $\text{IC}(t) = -\log(p(t))$ where $p(t)$ is the probability of observing a term t in the ontology. $p(t)$ is computed as $p(t) = \frac{\text{freq}(t)}{\text{freq}(root)}$. $\text{freq}(t)$ is defined as the cumulative count of term t and its descendants, where $\text{freq}(t) = \text{count}(t) + \sum_{c \in \text{child}(t)} \text{freq}(c)$. $\text{count}(t)$ is the number of genes annotated with the term t, and $\text{child}(t)$ are the children of t. Based on this definition, $\text{IC}(root) = 0$, and a node near the leaves has higher IC than nodes at upper levels. To compute a similarity score of the GO terms a and b, one finds the most informative common ancestor of these two terms.

$$\text{Resnik}(a, b) = \max_{p \in \{\text{par}(a) \cap \text{par}(b)\}} \text{IC}(p) \,, \tag{1}$$

where $\text{par}(t)$ denotes all the ancestors of term t. $\text{Resnik}(a, b)$ ranges from 0 to ∞ because the probability $p(t)$ ranges from 0 to 1.

In this model, the similarity score between a GO term t and itself is not 1. Second, when a, b have only the $root$ as a common ancestor, then $\text{Resnik}(a, b) = 0$. This is problematic because leaf nodes are more informative than other types of nodes. Consider the example in Song et al. (2014). Here, the $root$ is the only common ancestor of the pair a, b and the pair c, d. Next, suppose that a, b are leaf nodes, c is the parent of a, d is the parent of b, and $root$ is the parent of both c, d. One would then expect that $\text{Resnik}(a, b) < \text{Resnik}(c, d)$; however, one would obtain $\text{Resnik}(a, b) = \text{Resnik}(c, d) = 0$.

1.2 Aggregate Information Content (AIC) Method

The AIC method by Song et al. (2014) amends the two problems in the Resnik method. To encode the fact that leaf nodes are more informative, AIC defines a knowledge function of term t as $k(t) = 1/\text{IC}(t)$ which is used to measure its semantic weight $sw(t) = 1/(1 + \exp(-k(t)))$. Here $sw(root) = 1$. Semantic value $sv(t)$ of t is then defined as $sv(t) =$

$\sum_{p\in\text{path}(t)} sw(p)$. Function path$(t)$ contains every ancestor of t and the term t itself. Usually, $sv(a) < sv(b)$ when term a is nearer to the root than b. Song et al. (2014) define their similarity score of two GO terms a, b as

$$\text{AIC}(a,b) = \frac{2\sum_{p\in\{\text{path}(a)\cap\text{path}(b)\}} sw(p)}{sv(a)+sv(b)}\,. \tag{2}$$

AIC(a,b) ranges from 0 to 1. In this model, AIC$(a,a) = 1$. When a, b have only the $root$ as the common ancestor, then AIC$(a,b) = 2/(sv(a)+sv(b))$ which depends on where a, b are on the GO tree.

1.3 Graph-based Similarity Measure (GraSM)

GraSM is an extension of Resnik, and can be classified as an edge-based method. GraSM analyses more than just the most informative common ancestor of two GO terms, by looking at their disjunctive common ancestors (Couto et al., 2007). For one GO term a, two of its ancestors are disjunctive if there are different paths from both ancestors to the GO term.

$$\text{DisjAnc}(a) = \{c_1, c_2 \mid \exists\text{path}(a, c_1) \text{ not containing } c_2 \text{ AND } \exists\text{path}(a, c_2) \text{ not containing } c_1\} \tag{3}$$

For two GO terms a and b , suppose the term c_1 is in the union set U=DisjAnc(a)$\cup$ DisjAnc(b). c_1 is a common disjunctive ancestor of a, b if for each common ancestor c_2 of a, b where IC$(c_1) <$ IC(c_2) we have both $c_1, c_2 \in U$. The similarity measurement for a, b is

$$\text{GraSM}(a,b) = \text{mean IC}(c) \text{ where } c \text{ is common disjunctive ancestor of } a, b \tag{4}$$

We use the software GOssTo to implement GraSM (Caniza et al., 2014).

1.4 Random Walk Contribution (RWC)

We briefly describe RWC's key idea. Unlike many other methods which inspect the ancestors of two given GO terms, RWC is an edge-based approach that analyzes the shared children of two GO terms a and b (Yang et al., 2012). In brief, in the RWC paradigm, GO terms with more common children are more deemed to be more similar.

Define N_c as the number of genes annotated by term c. In RWC, the random walker moves from the parent node p to its direct child c with probability $\mathbb{P}(p \to c) = N_c / \sum_{u:\exists p\to u} N_u$. As the random walker moves for a very long time, we can denote the probability of ending at a node i from a to be $W^a_\infty(i)$. Let L be the set of all leaf nodes in the GO tree. The RWC for two terms a and b is

$$\text{RWC}(a,b) = \sum_{i,j\in L} W^a_\infty(i) W^b_\infty(j) \text{score}(i,j) \tag{5}$$

score(i,j) is a generic place holder. For example, score(i,j) can be Resnik(i,j). Yang et al.

(2012) uses their RWC to improve score(a, b) by taking the average

$$\text{score}_{\text{RWC}}(a, b) = \frac{1}{2}(\text{RWC}(a, b) + \text{score}(a, b)) \tag{6}$$

In this paper, we use the software GOssTo to implement RWC with score(a, b) being Resnik and GraSM (Caniza et al., 2014). Currently, GOssTo is unable to take any generic score function as its argument.

References

Caniza, H., Romero, A. E., Heron, S., et al. (2014). Gossto: a stand-alone application and a web tool for calculating semantic similarities on the gene ontology. Bioinformatics, **30**(15), 2235–2236.

Couto, F. M., Silva, M. J., and Coutinho, P. M. (2007). Measuring semantic similarity between gene ontology terms. Data & knowledge engineering, **61**(1), 137–152.

Resnik, P. (1999). Semantic similarity in a taxonomy: An information-based measure and its application to problems of ambiguity in natural language. J. Artif. Intell. Res.(JAIR), **11**, 95–130.

Song, X., Li, L., Srimani, P. K., et al. (2014). Measure the semantic similarity of GO terms using aggregate information content. IEEE/ACM Transactions on Computational Biology and Bioinformatics, **11**(3), 468–476.

Yang, H., Nepusz, T., and Paccanaro, A. (2012). Improving go semantic similarity measures by exploring the ontology beneath the terms and modelling uncertainty. Bioinformatics, **28**(10), 1383–1389.