Anc2vec: embedding Gene Ontology terms by preserving ancestors relationships

Alejandro A. Edera[1]* Diego H. Milone[1] Georgina Stegmayer[1]

[1]Research Institute for Signals, Systems and Computational Intelligence, sinc(i), FICH-UNL/CONICET, Ciudad Universitaria, Santa Fe, Colectora Ruta Nacional No 168 km. 0, Paraje El Pozo, Santa Fe, 3000, Argentina.

1 Introduction

Gene products or proteins are sophisticated molecules that play many functional roles essential for life on Earth. Protein functions are generally described by using the Gene Ontology (GO) that is amenable for being processed by computers [Consortium, 2019]. Technically, the GO consists of a structured and controlled vocabulary of about 40,000 terms that represent biological entities suited for describing protein functions. The structure of the ontology consists of terms graphically represented as nodes, which are hierarchically organized into three directed acyclic graphs or sub-ontologies called Cellular Component (CC), Biological Process (BP) and Molecular Function (MF). This structure encodes different types of hierarchical relations between GO terms, such that ancestors nodes represent more abstract entities than their descendants. Two common hierarchical relationships are *is_a* and *part_of*. For example, if term a *is_a* term b, then term a is a subtype of term b; while if term a is *part_of* term c then term a is part of a whole defined by term c. Here, terms b and c are the ancestors of term a. Note that, in this example, the ancestors of term a are defined by two different types of hierarchical relationships: *is_a* and *part_of*. Considering only the ancestor of either of these relationships may exclude a significant piece of information to correctly interpret the biological meaning of term a.

The topological structure induced by these hierarchical relationships provides the foundation to compare the semantic similarity between terms, which is fundamental for assessing the functional similarity between proteins according to their GO annotations. However, how to perform such comparisons is still an open research area, because there is not yet an effective method for quantifying the semantic information of terms [Pesquita et al., 2009, Zhao and Wang, 2018]. The semantic similarity is usually calculated by using the information content (IC), estimated from the term frequencies observed in a corpus of annotated gene products [Sousa et al., 2020, Guzzi et al., 2012, Mazandu et al., 2017]. The Resnik similarity measure is a well known IC method [Resnik, 1995, 1999], but better performances are achieved by newer IC methods, such as AIC [Song et al., 2014] and the one proposed in Wang et al. [2007].

More recent, IC methods have been significantly outperformed by methods based on neural networks [Ristoski and Paulheim, 2016, Zhong et al., 2019, Ali et al., 2019, Kulmanov et al., 2021, Alshahrani et al., 2021]. These methods find vector representations, or embeddings, of terms in a low-dimensional Euclidean space, in such a way that the similarity of two terms is encoded by the Euclidean distance of their corresponding embeddings. The use of vector representations has shown dramatic improvements on diverse biological tasks [Sabando et al., 2021, Liu et al., 2021] but especially in semantic comparisons of terms [Smaili et al., 2018a,b] and predicting protein-protein interactions [Zhong et al., 2019, Duong et al., 2020, Zhao et al., 2020]. However, some structural features of the GO are not yet fully encoded by existing embeddings. This is the case of the ancestors hierarchy of a term, a crucial structural feature for semantic similarity tasks [Song et al., 2014, Zhao and Wang, 2018, Wang et al., 2007, Mazandu and Mulder, 2012, Zhang et al., 2018]. For example, Onto2Vec embeddings are built only using direct relationships between terms [Smaili et al., 2018a], thereby a single embedding is incapable of capturing the whole ancestry of its term. Similarly, the stochasticity of random-walk techniques, such as GO2Vec [Zhong et al., 2019],

*Corresponding author: aedera@sinc.unl.edu.ar

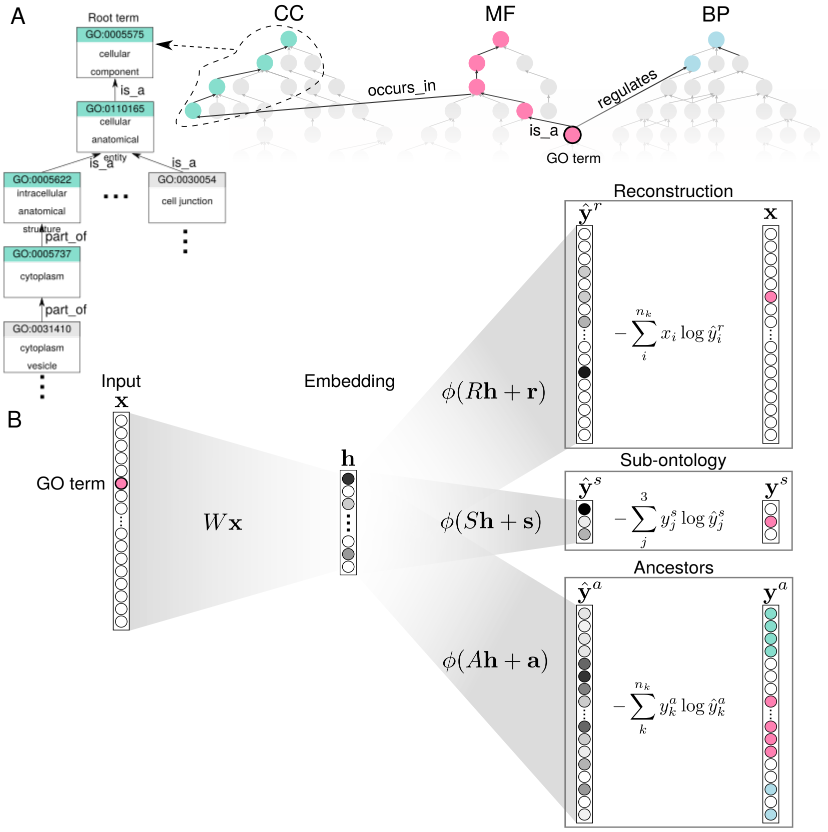

Figure 1: The GO and the architecture of anc2vec. A) Structure of the GO. It is composed of three sub-ontologies: BP, CC and MF. Colored nodes show the ancestors of a GO term. B) Anc2vec architecture. The GO term is encoded as a vector $\mathbf{x}$ and transformed into a vector $\mathbf{h}$, which is mapped into three vectors used to optimize anc2vec weights.

makes a single embedding to capture, only, some arbitrary ancestral relations rather than the full set of such relationships.

In this work, we propose anc2vec as a novel protocol based on neural networks for constructing embeddings of terms exclusively using the GO structure. Unlike existing methods, anc2vec uses three structural features of a GO term: its ontological uniqueness, its ancestors hierarchy and its membership to sub-ontologies. Experiments show that anc2vec is effective in capturing these features, allowing it to achieve better performance than existing embeddings on diverse biological tasks involving large-scale, real-world data annotated with GO terms.

2 Building anc2vec embeddings

To encode the three proposed structural features, we define a protocol for building embeddings of GO terms such that: 1) they are as unique as their corresponding terms are within the gene ontology, and 2) their distances reflect the semantic similarity between their corresponding GO terms. We define the semantic similarity using the GO structure such that two terms are similar if they belong to the same sub-ontology and also share similar ancestors. The true path rule, also known as the annotation propagation rule, is used to find the ancestors of a given term, as illustrated in Fig. 1A. Unlike some previous works, the true path rule used here not only includes *is_a* relations but also *part_of*, *regulates*, *negatively_regulates*, *positively_regulates*, *occurs_in*, *ends_during* and *happens_during* relations. As it will be demonstrated in the experimental section, the use of these additional relations allows anc2vec to capture more structural features.

To build such embeddings, we designed a neural network architecture called anc2vec, which is schematically shown in Fig. 1B. It receives an input term $\mathbf{x}$ that is transformed into an embedding $\mathbf{h}$. Next, three vectors are built from the embedding $\mathbf{h}$ for model weight optimization, which attempts to match them as well as possible with the proposed structural features of the input term.

This architecture is formalized as follows. Let $\mathbf{x} \in \{0,1\}^{n_x}$ denote a one-hot vector representing an input term. A weight matrix $W \in \mathbb{R}^{n_h \times n_x}$ is used to transform $\mathbf{x}$ into an embedding $W\mathbf{x} = \mathbf{h} \in \mathbb{R}^{n_h}$. By defining $n_h \ll n_x$, the resulting embedding is low-dimensional and the size of W is also drastically reduced. The vector $\mathbf{h}$ is then used for building the three vectors for weight optimization

$$\begin{aligned} \hat{\mathbf{y}}^r &= \phi(R\mathbf{h} + \mathbf{r}) \\ \hat{\mathbf{y}}^s &= \phi(S\mathbf{h} + \mathbf{s}) \\ \hat{\mathbf{y}}^a &= \phi(A\mathbf{h} + \mathbf{a}), \end{aligned}$$

where $R \in \mathbb{R}^{n_x \times n_h}$, $S \in \mathbb{R}^{3 \times n_h}$; and $A \in \mathbb{R}^{n_x \times n_h}$ are additional weight matrices with their corresponding bias vectors $\mathbf{r}$, $\mathbf{s}$ and $\mathbf{a}$, respectively. The softmax function $\phi(\cdot)$ guarantees that vectors represent probability distributions.

The total number of weights of anc2vec is $3n_x n_h + 3n_h + 2n_x + 3$, where $3n_x n_h$ accounts for the weights of W, R and A, $3n_h$ for S and $2n_x + 3$ for the biases $\mathbf{r}$, $\mathbf{a}$ and $\mathbf{s}$. These weights are optimized by minimizing the following loss function

$$Loss = -\sum_{i}^{n_k} x_i \log(\hat{y}_i^r) - \sum_{j}^{3} y_j^s \log(\hat{y}_j^s) - \sum_{k}^{n_k} y_k^a \log(\hat{y}_k^a),$$

where $\mathbf{y}^s \in \{0,1\}^3$ is a vector encoding the sub-ontology of the input term (BP, CC or MF) and the binary vector $\mathbf{y}^a \in \{0,1\}^{n_x}$ represents the true ancestor terms of $\mathbf{x}$. Subscripts indicate vector components.

This loss function uses three cross-entropy losses aimed to preserve the three structural features of a term $\mathbf{x}$. The first cross-entropy loss focuses on the ontological uniqueness information by comparing how similar the term $\hat{\mathbf{y}}^r$ is with the input term $\mathbf{x}$. The second cross-entropy loss focuses on the sub-ontology membership information by measuring the similarity between the predicted sub-ontology $\hat{\mathbf{y}}^s$ and the expected sub-ontology $\mathbf{y}^s$. The third cross-entropy loss focuses on the ancestors information by comparing how similar the predicted ancestors $\hat{\mathbf{y}}^a$ are with respect to the expected ancestors $\mathbf{y}^a$. Supp. Fig. 1 illustrates the optimization of this loss function for embeddings with $n_h = 2$ dimensions.

To further understand the contribution of preserving ancestors, we carried out ablation experiments by designing a method named neigh2vec. To construct neigh2vec embeddings, vectors $\mathbf{y}^a$ simply encode, instead of ancestors, immediate neighbors, which are defined as the union of the children and parents of a term.

3 Data processing

Gene ontology. We used the release 2020-10-06[1] of the GO as reference. It was processed to remove terms tagged as *obsolete* while *alternative* terms were replaced by their primary ID. This processing resulted in a total of $N = 44,261$ GO terms, where 8,888, 11,177, and 4,196 of them belonged to BP, MF, and CC, respectively. A dataset was created from this ontology to train anc2vec. Each of the n_x terms was included in the dataset and labeled with its corresponding ancestor terms and sub-ontology. A similar dataset was prepared for neigh2vec, where terms were labeled with neighbors (instead of ancestors).

Ancestors. Using the reference gene ontology, a dataset of ancestor relationships was created. It contained 1,767,518 pairs of GO terms. Half of the pairs were labeled as related whereas the other half as unrelated. Related pairs were generated by pairing each GO term with each of its ancestor terms defined by the true path rule. Unrelated pairs were generated by pairings GO terms randomly such that the resulting pairs were not among the related pairs.

[1] http://geneontology.org/page/download-ontology

Protein function. This dataset contained GO annotations for proteins from diverse species available in the SwissProt and TrEMBL UniProtKB release 2021_02[2]. GO annotations were extracted from the GO annotation file [Huntley et al., 2014] v200[3] when having EXP, IDA, IMP, IGI, IEP, TAS, or IC evidence codes, as recommended by the Critical Assessment of Functional Annotation (CAFA) [Zhou et al., 2019]. Although TAS and IC are not experimental evidence codes, results are not significantly affected when excluded. In addition, GO terms found as *obsolete* under the reference ontology were excluded, and those found as *alternative* were replaced by their corresponding primary terms. The resulting dataset contained 130,957 entries containing one or more GO terms representing protein annotations.

STRING. Interacting protein pairs from diverse species were extracted from the STRING database v11.0 [Szklarczyk et al., 2018] whenever their protein sequences appeared in SwissProt and TrEMBL UniProtKB release 2021_02, and also had annotations in the protein function dataset. A total of 3,742,248 protein pairs were extracted, encompassing 70,081 unique proteins, and labeled as positives. An equal number of pairs labeled as negative were generated by randomly pairing the 70,081 proteins such that the resulting pairs were not positive. Following the same procedure, we additionally created five species-specific datasets of protein-protein interactions for the top-5 species with the largest numbers of interacting protein pairs: *Arabidopsis thaliana* (3702), *Danio rerio* (7955), *Drosophila melanogaster* (7227), *Homo sapiens* (9606) and *Mus musculus* (10090).

4 Experimental setup

4.1 GO term embeddings

The embeddings of GO terms were constructed by optimizing the weights of anc2vec (and neigh2vec) using the reference gene ontology (training details are fully provided in the publicly available source code). The weights obtained in the best train loss were used for embedding construction, as is usually done in similar approaches [Mikolov et al., 2013]. As a trivial baseline for the proposed method, the terms were also represented as one-hot encoding vectors in $\{0,1\}^{n_x}$.

Onto2Vec [Smaili et al., 2018a] and GO2Vec [Zhong et al., 2019] were selected as fair, representative competitor methods, because they build embeddings of terms exclusively using the GO structure without additional data sources. For example, some approaches use the "textual descriptions" of GO terms as additional data source, whose words are represented with vectors previously built from a large corpus of biomedical abstracts [Smaili et al., 2018b]. For Onto2Vec, we used its public source code[4], and included its variant [Smaili et al., 2018a] named here as Onto2Vec*. This variant simply enlarges the GO by including new terms representing sets of GO terms and relations between them and each of the terms in their sets. These new terms were defined using the protein function dataset. No source code is available for GO2Vec, hence this method was implemented following authors indications which consist in adapting the publicly available node2vec[5]. Following common choices of dimensionality [Smaili et al., 2018a], and to ensure a fair comparison with the other methods, all the embeddings used in this study were 200-dimensional and were built using the same reference GO.

Anc2vec is fully implemented on Python 3.6 using TensorFlow 2. The source code, along with installation requirements, examples and the datasets employed in this study, are publicly available[6].

4.2 Semantic similarities

To compare two terms, a and b, the cosine similarity between their corresponding vector representations, $\mathbf{h}^a$ and $\mathbf{h}^b$, was calculated by

$$s_{cos}(a,b) = \frac{\langle \mathbf{h}^a, \mathbf{h}^b \rangle}{||\mathbf{h}^a||||\mathbf{h}^b||}.$$

[2] ftp.uniprot.org/pub/databases/uniprot/current_release/knowledgebase

[3] https://www.ebi.ac.uk/GOA/downloads

[4] https://github.com/bio-ontology-research-group/onto2vec

[5] https://github.com/aditya-grover/node2vec

[6] https://github.com/aedera/anc2vec

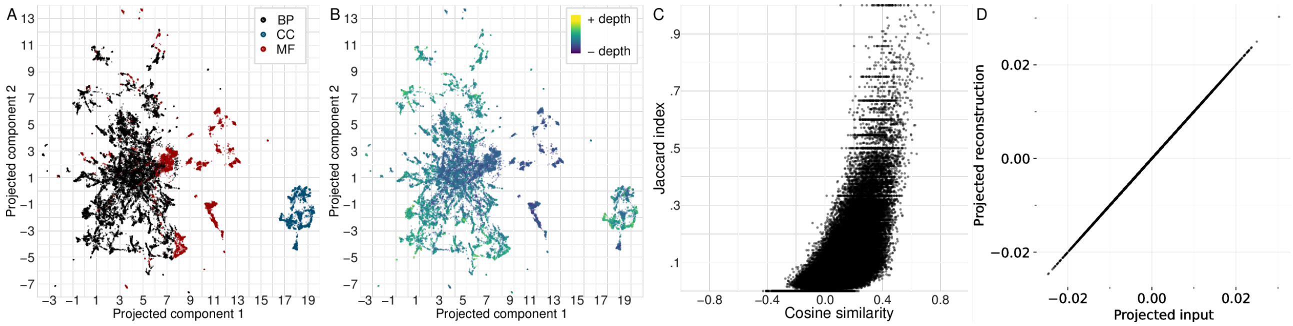

Figure 2: Anc2vec embeddings of GO terms in the three sub-ontologies. A) Points depict embeddings of terms. B) Distribution of depths. Colors plot the depth of each term in the GO hierarchy. C) The Jaccard index and the semantic similarity between the embeddings of terms. D) Correlation between one-hot inputs and their representations from anc2vec embeddings.

The numerator is a dot product, and $||\cdot||$ is the Euclidean norm. The similarity $s_{cos}(a,b) \in [-1,1]$, where -1 and 1 indicate low and high semantic similarity, respectively.

The semantic similarity between GO terms can be also calculated by the aggregated information content (AIC) [Song et al., 2014]:

$$\text{AIC}(a,b) = 2\frac{\sum_{t\in\{anc(a)\cap anc(b)\}} sw(t)}{sv(a)+sv(b)},$$

where $anc(a)$ returns a set containing all the ancestors of term a, including the term a itself. AIC is the sum of the semantic weight $sw(t)$ of the common ancestor t of a and b. This sum is normalized by the sum of the semantic values of the terms defined as: $sv(a) = \sum_{t\in anc(a)} sw(t)$, where the semantic weight of a term is $sw(a) = 1/(1+\exp(-1/IC(a)))$. Here, $IC(a) = -\log p(a)$ is the information content, where $p(\cdot)$ is calculated from a corpus and is the frequency of a term and its descendants divided by the frequency of its root term. AIC ranges from 0 to 1.

In contrast to AIC, Wang et al. Wang et al. [2007] calculates the semantic similarity exclusively using the structure of the gene ontology:

$$\text{SimWang}(a,b) = \frac{\sum_{t\in\{anc(a)\cap anc(b)\}} sc_a(t) + sc_b(t)}{si(a)+si(b)}.$$

This metric estimates the semantic information of terms a and b, defined as $si(a) = \sum_{t\in anc(a)} sc_a(t)$, where $sc_a(\cdot)$ is the semantic contribution of an ancestor t on the term a. We used the implementation of this similarity available in Goatools [Klopfenstein et al., 2018]. Recently, SimWang has been extended in a metric named GOGO Zhao and Wang [2018], in which the number of children of a term is additionally included. We used in experiments the publicly available implementation of GOGO.

To compare the semantic similarity between two sets of GO terms, the best match average (BMA) is one of the most widely used methods [Azuaje et al., 2005, Pesquita et al., 2008]. Let A and B be two sets of GO terms, BMA calculates the average of two sums

$$BMA(A,B) = \frac{1}{2|A|}\sum_{a\in A}\max_{b\in B} sim(a,b) + \frac{1}{2|B|}\sum_{b\in B}\max_{a\in A} sim(a,b).$$

Each sum is over the semantic values for each term in one input set and a term in the other set selected to yield maximum similarity. In our experiments, AIC and SimWang were used as $sim(a,b)$, since GOGO works for sets of GO terms.

4.3 Protein-protein interaction

To assess semantic performance, embeddings of terms were used to predict interacting protein pairs in the STRING dataset. For this task, a protein was represented by summing the embeddings corresponding to each annotated GO terms. Two experimental scenarios were designed to make such predictions. In the first scenario, the cosine similarity between embeddings representing proteins were used to discriminate between interacting and non-interacting pairs.

In the second scenario, a neural network classifier was used to make predictions. It was constructed from pairs of embeddings representing proteins. Given two embeddings, this classifier linearly transforms them into a single score normalized in the range from 0 (no interaction) to 1 (interaction), by using a sigmoid activation function. A standard binary cross-entropy loss was used for training, and the best parameters were selected using a validation set randomly drawn from the training set. A 3-fold cross validation was used to assess classifier performance.

4.4 Performance metrics

Jaccard Index. Let A and B be two sets of GO terms, the Jaccard index is $J(A, B) = |A \cap B|/|A \cup B|$. It measures the degree of match between the two sets, as the ratio of the number of GO terms shared by both sets to the number of all terms in both sets. To calculate this index, terms were propagated with the true path rule.

1-Wasserstein distance. It is often used to measure the dissimilarity between two discrete distributions $p \in \mathbb{R}^m$ and $q \in \mathbb{R}^n$ [Kolouri et al., 2017]:

$$W_1(p, q) = \min_{\gamma \in \mathbb{R}_+^{m \times n}} \sum_{i,j} \gamma_{i,j} |p_i - q_j|,$$

where γ is a matrix where the sum of its rows and columns are equal to the input empirical distributions, respectively. The sum is over the absolute distance between elements of the empirical distributions. The optimum value of the Wasserstein distance is found by solving a linear programming problem [Bassetti et al., 2020]. The higher the 1-Wasserstein distance, the more different the distributions are.

Predictive performance metrics. To assess the performance of predictive methods, we used the precision-recall and receiver operating characteristic (ROC) curves, for imbalanced and balanced datasets respectively [Saito and Rehmsmeier, 2015]. The precision-recall curve shows the tradeoff between the precision p and recall or sensitivity r when varying a threshold (or cutoff) to binarize method outputs into negative and positive predictions. The precision measures how many positive predictions are true positives whereas the recall measures how many true positives are correctly retrieved by predictions. Formally, $p = TP/(TP + FP)$ and $r = TP/(TP + FN)$, where TP, FP and FN are the number of true positives, false positives and false negatives, respectively. The F1 value is equal to $2(r \cdot p)/(r + p)$ and its maximum value (F1-max) is often used to summarize a precision-recall curve into a single value. By contrast, the ROC curve shows the tradeoff between the recall and the false positive rate: $1 - (TN/(TN + FP))$, and is often summarized by calculating the area under it (AUROC).

5 Results

5.1 Exploring the embeddings space topology

To assess whether the embeddings built by anc2vec are able to encode the proposed structural features, we studied how they were arranged in their vector space by non-linearly projecting them onto a 2-D space with UMAP [McInnes et al., 2018]. The projected embeddings were well separated into two large clusters, corresponding to the BP and CC sub-ontologies, plotted as black and blue points in Fig. 2A, respectively. In addition, the projected embeddings also formed a less defined cluster corresponding to terms of the MF sub-ontology (red points). This result shows that anc2vec embeddings are capturing very well the three sub-ontologies of the GO. Notably, the MF and BP clusters showed some regions overlapped, indicating that some of the embeddings belonging to MF (red) could be sharing information with others belonging to BP (black). A further analysis of the GO structure revealed that about 47% of the MF terms have, at least, one ancestor in BP, supporting the observed overlapping between the BP and MF clusters. Interestingly, the majority of these relationships between BP and MF terms are of the type *occurs_in*. This demonstrates that the use of additional ontological relationships is advantageous for capturing further structural features of the GO.

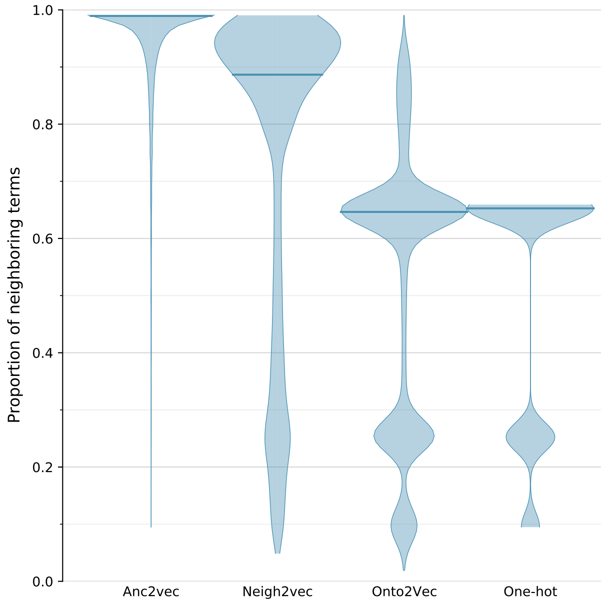

Figure 3: Sub-ontology membership preserved by embeddings. Violin plots distributions of relative numbers of neighboring terms sharing the same sub-ontology of the central term. Horizontal line segments show the medians of distributions.

sinc(*i*) Research Institute for Signals, Systems and Computational Intelligence (sinc.unl.edu.ar)
A. Edera, D. H. Milone & G. Stegmayer; "Anc2vec: embedding gene ontology terms by preserving ancestors relationships"
Briefings in Bioinformatics, feb, 2022.

The projected embeddings were also distributed according to the depths of their terms, measured as the length of the longest path between a term and the root of the sub-ontology. Visualizing depth information along with the projected embeddings showed a radial-like pattern resembling the tree-structure of the GO (Fig. 2B). Here, hierarchically shallow embeddings (blue) were frequently surrounded by embeddings encoding their ancestors (yellow), indicating that hierarchical relationships are being encoded. To further assess this finding, the original vector space built by anc2vec was also analyzed. To this end, each GO term was paired with 500 randomly sampled terms to calculate the cosine similarity of their embeddings and the Jaccard index of their propagated terms. An important correlation was found between the cosine similarity and the Jaccard index (Fig. 2C), suggesting that semantically similar embeddings generally share similar ancestors.

To analyze whether the ontological uniqueness was correctly encoded, one-hot inputs were qualitatively compared with their reconstructions built by anc2vec. By projecting these two vectors onto a 1-dimensional space with PCA, a strong linear correlation was found (Fig. 2D), indicating that anc2vec embeddings encode enough information to uniquely identify their terms.

When further investigating the semantic of anc2vec embeddings, an extremely high cosine similarity was found between the embeddings corresponding to the terms "*ER ubiquitin ligase complex*" (GO:0000835) and "*Hrd1p ubiquitin ligase ERAD-M complex*" (GO:0000838). Note that GO:0000835 is the grandfather of GO:0000838. The consistence of this semantic similarity was assessed by solving the following word analogy task [Mikolov et al., 2013]: "GO:0000835 is to GO:0000838 as v_1 is to v_2", where v_i are other two terms. When using "*mannan polymerase I complex*" (GO:0140498) as v_1, the embedding of the term "*mannan polymerase complex*" (GO:0000136) was found as v_2, which is actually the parent of GO:0140498. This shows that the embedding pair (GO:0000136, GO:0140498) closely resembles the relation between the embedding pair (GO:0000835, GO:0000838). This demonstrates that anc2vec embeddings are able to capture fine-grained hierarchical structure.

5.2 Sub-ontology membership encoded by embeddings

To further evaluate whether the sub-ontology of each term was captured by the embeddings, we first analyzed if terms from the same sub-ontology were actually embedded close to each other. To this end, we estimated for each embedding the relative number of neighbors in the vector space belonging to the same sub-ontology. Neighbors were defined as those embeddings within a hypersphere in which the embedding

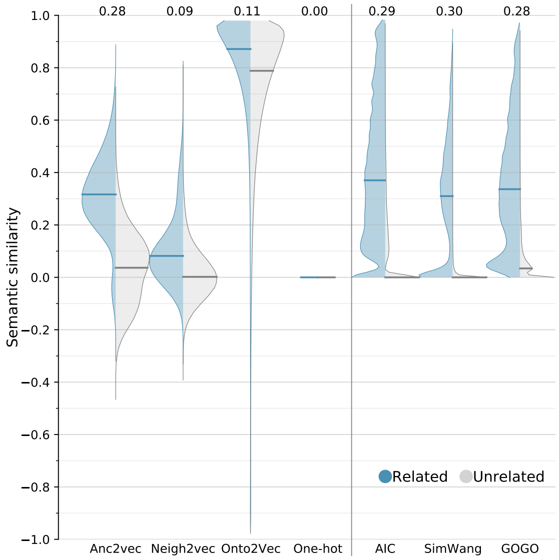

Figure 4: Discriminating ancestors. Violin plots semantic similarities between pairs of terms related (or not) by ancestors. Methods not using embeddings are on the right. 1-Wasserstein distances are shown on the top.

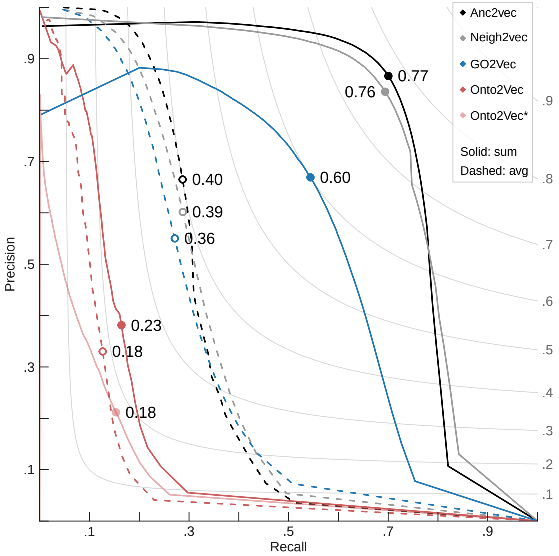

Figure 5: Decomposability of embeddings. Colored lines depict precision-recall curves and gray iso-curves show F1 values. Points indicate F1-max values.

of a given term was situated at the center. The radius of this hypersphere was the distance between the center and the farthest embedding corresponding to one of the children or parents of the term at the center. Then, the sub-ontology membership of terms in each neighborhood was used to calculate the proportion of embeddings that had the same sub-ontology as the central term.

The results are shown in Fig. 3 where anc2vec shows the highest performance with a median around 1.0. This indicates that anc2vec embeds terms close to each other when they share the same sub-ontology. In comparison to neigh2vec, the performance of anc2vec is higher, indicating that ancestors information is beneficial for capturing the sub-ontology structure. By contrast, the performance of Onto2Vec is much lower and very similar to that of the one-hot encodings. Since the latter are orthonormal vectors in the nonnegative orthant, all the one-hot encodings are equidistant, and thus their distances and positions completely lack sub-ontology information. The absence of the sub-ontology structure is also reflected in the shapes of the distributions of values of Onto2Vec and one-hot encodings, both following the uneven distribution of GO terms among the three sub-ontologies (∼65, ∼25, and ∼10% of the 44,261 terms belong to BP, MF, and CC, respectively). Taken together, these results demonstrate that the sub-ontology structure of the gene ontology is being captured by the embedding space built by anc2vec.

5.3 Discriminating ancestors

We evaluated whether ancestors relationships can be discriminated from the embeddings of term pairs in the ancestors dataset. The performance of our propose was compared against existing embeddings and three well-known ontology-based metrics used as baselines: AIC, SimWang and GOGO. For calculating the information content values used by AIC, the entire protein function dataset was used as a corpus.

The resulting semantic similarities are shown in Fig. 4. For each method, these values are grouped in two distributions according to whether the term pairs are either related (blue) or not (gray) by ancestors. Unlike the other embeddings, anc2vec obtains almost not overlapped distributions, as shown by the difference between their medians (horizontal segments). In addition, the median of related terms is higher than that of unrelated ones. These results indicate a good ancestors discrimination because term pairs are more semantically similar when they are related by ancestors. By contrast, the distributions of neigh2vec and Onto2Vec are much more overlapped, resulting in a worst discrimination. In particular, the distributions of the one-hot encodings are totally collapsed in zero, reflecting the fact that their

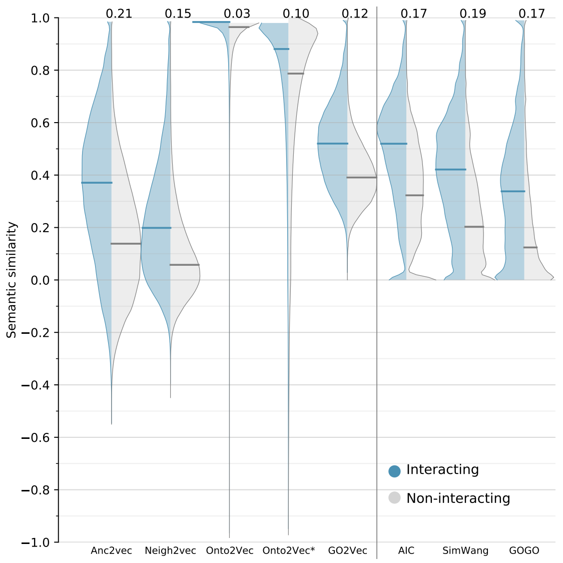

Figure 6: Semantic similarity of GO terms on protein-protein interactions. Violin plots distributions of semantic similarities between interacting and non-interacting proteins. Numbers in the top indicate 1-Wasserstein distances.

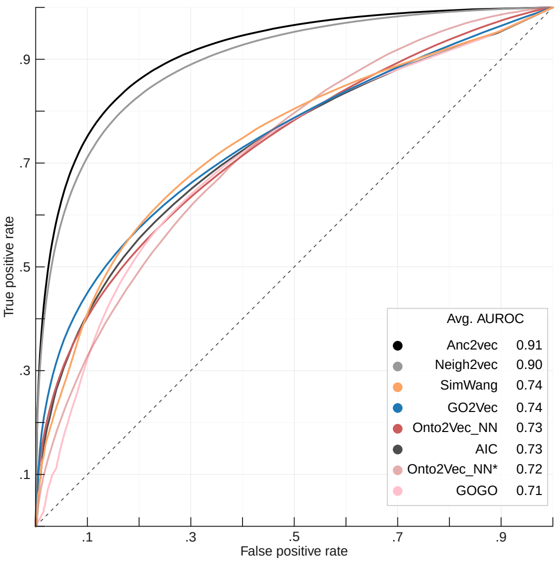

Figure 7: Average performance on predicting protein-protein interactions. Lines plot the performance of methods assessed by their ROC curves using 3,740,874 interactions and 70,081 proteins of diverse species.

orthogonality results in null (cosine) similarities. On the other hand, since the ontology-based metrics have complete access to the GO structure when calculating semantic similarities, their distributions are more separated. However, note that these metrics are totally unable to construct vector representations.

It is worth noting that Fig. 4 also shows that the median of the distribution of anc2vec is slightly higher than that of neigh2vec for unrelated terms, as shown by the horizontal lines in the gray areas. This indicates that anc2vec is finding some pairs of unrelated terms more semantically similar than neigh2vec. Given how both models were defined, this difference could be caused by the use of the information of ancestors. A further analysis confirmed this showing that the majority of the pairs of unrelated terms indeed share a certain number of common ancestors. Because the embeddings built by neigh2vec encode only immediate neighbors, they are unable to capture such higher order relationships between terms, resulting in semantic similarities less than zero. In contrast, since terms sharing common ancestors result in anc2vec embeddings encoding similar information (Fig. 2C), the semantic similarities of anc2vec are larger than zero for such cases, shifting the median for unrelated terms upward. This illustrates the representational advantages of exploiting ancestor information for building embeddings of GO terms.

To assess more quantitatively the discriminative performance of each method, the 1-Wasserstein distance was calculated between its distributions. Larger distance means better discrimination. The resulting distances are shown on the top of Fig. 4. The 1-Wasserstein distances show that anc2vec (0.28) is better that the other embeddings (0.00-0.11) and it is also as discriminative as the ontology-based metrics are (0.28-0.30). This demonstrates the benefits of building embedings encoding ancestors information, and also shows that anc2vec embeddings are able, on its own, to discriminate ancestors.

5.4 Decomposability of embeddings

In practice, numerous bioinformatics applications assess the functional similarity between two proteins by comparing the semantic similarity of their GO annotations. Since proteins may carry out multiple functions, a single protein is often annotated with multiple terms. When representing terms as vectors, a simple approach for representing multiple terms is by summing their corresponding embeddings. This approach is straightforward when using a simple one-hot encoding. Because these encodings are orthogonal vectors, they lead to no information degradation when used for composing and decomposing sets of terms. However, we wondered whether the use of low-dimensional embeddings for representing sets of terms could lead to information degradation, impeding the correct identification of the individual terms in a set. Therefore, we evaluated embeddings for their decomposability by predicting the individual terms present in embeddings constructed by aggregating the vector representations of multiple terms.

For this evaluation, we used the protein function dataset, which contains 130,957 proteins annotated with multiple terms experimentally validated in practice. Each protein was represented as the sum of the embeddings corresponding to their annotated GO terms. Alternatively, a protein was also represented by averaging the embeddings of its terms. We used cross validation on this dataset (70-30% train-test) to build and evaluate a simple classifier to predict which terms were aggregated in an input embedding. This classifier linearly transforms an input embedding and then uses a sigmoid function to obtain an output vector in $[0, 1]^{n_x}$.

The performances of the classifiers are shown in Fig. 5 as precision-recall curves. This figure also shows gray iso-curves showing the F1 values, and points depicting the F1-max values. Here, the F1-max values of anc2vec and neigh2vec are very similar to each other but substantially better than those of GO2Vec, Onto2Vec and Onto2Vec* when using embeddings aggregated by either the sum (solid curves) or the average (dashed curves). Notably, the performance of all methods is increased when using the sum instead of the average. This is because the sum preserves information about the number of individual terms.

5.5 Predicting protein-protein interactions

Different studies have shown that proteins found in similar cellular locations or participating in related biological processes are more likely to interact with each other [Kanehisa et al., 2020]. The basic hypothesis is that such protein-protein interactions should be reflected as relationships between the GO terms annotating the involved proteins. We used this to evaluate embeddings for their ability to discriminate proteins as interacting and non-interacting. This evaluation was performed on two experimental scenarios using the STRING dataset (details in Experimental Setup section). It should be highlighted that STRING represents not only physical but also functional interactions between proteins.

The results of the first experimental scenario are shown in Fig. 6. It shows violins plotting the resulting semantic similarities calculated from protein pairs. For each method, these values are grouped into two distributions according to whether protein pairs are interacting (blue) or not (gray). In comparison to all the methods, the results show that anc2vec exhibits the best discriminative power, as indicating by the gap between the medians of the interacting and non-interacting distributions. Similar to the results with unrelated terms in Fig. 4, the anc2vec median for non-interacting pairs is slightly higher than that of neigh2vec. This may suggest the presence of interacting pairs among the non-interacting ones, because the last ones were synthetically generated. Nevertheless, anc2vec is substantially better than neigh2vec for discriminating interacting pairs (blue areas), which are experimentally validated. To quantitatively assess the discriminative power, the 1-Wasserstein distance between the distributions is shown on the top of Fig. 6. The resulting distances show that anc2vec achieves the best performance (0.21), as compared to neigh2vec (0.15), Onto2Vec (0.03), Onto2Vec* (0.10) and GO2Vec (0.12), as well as the baseline methods (0.17, 0.17 and 0.19). Note that anc2vec (0.21) is also better than neigh2vec (0.15). This indicates that ancestors information plays an important role for this task.

Fig. 7 shows the results of the second experimental scenario. Here, the ROC curves show the average predictive performance of classifiers trained upon the embeddings for discriminating interacting protein pairs. The highest AUROC indicates the best prediction performance that is achieved by anc2vec (0.91) and is followed by neigh2vec (0.90). In contrast, existing embeddings showed lower AUROC values: GO2Vec (0.74), Onto2Vec_NN (0.73) and Onto2Vec_NN* (0.72). Note that the classifier trained with the Onto2Vec(*) embeddings is the method known as Onto2Vec_NN in [Smaili et al., 2018a]. The ontology-based metrics also achieved lower AUROCs: SimWang (0.74), AIC (0.73) and GOGO (0.71). In order to analyze the influence of embedding dimension on the performance, anc2vec was also tested for $n_h \in \{10, 50, 100, 200, 300, 400, 500, 1000\}$. As expected, its discriminative performance improved for higher dimensions up to a point where a large increase in dimension is required to obtain relatively small improvements in performance (Supp. Fig. 2). In addition, similar results were obtained when making predictions by exclusively using protein pairs annotated with specific STRING scores or number of GO terms (Supp. Fig. 3). Likewise, anc2vec outperformed competitors when using the dataset released by the authors of Onto2Vec [Smaili et al., 2018a] (Supp. Fig. 4) as well as when using binary interactomes for well and not-so-well characterized organisms (Supp. Fig. 5) available in the curated database APID [Alonso-López et al., 2019]. These results demonstrate the advantages of using anc2vec embeddings for predicting protein-protein interactions.

Finally, the second experimental scenario was further explored by evaluating the predictive performance of individual classifiers when trained with the five species-specific datasets of protein interactions. The resulting AUROC values are shown in Table **??**. Regardless species, the highest AUROC values are obtained by anc2vec, followed again by neigh2vec. Interestingly, this result shows that protein-protein interactions can be reliably predicted with "generic" embeddings, that is, not encoding any type of species-specific information. Note that, in comparison to neigh2vec, the higher AUROC values of anc2vec point out that the contribution of ancestors information is very important for predicting protein interactions, particularly for species whose protein-protein interactions are well characterized, such as *Homo sapiens* (9606). Taken together, the superior predictions obtained by anc2vec support the semantic relevance of the three structural features proposed here for constructing embeddings of GO terms.

6 Conclusions

A novel neural network model named anc2vec is presented, for constructing embeddings of GO terms. Anc2vec preserves three structural features of the GO in the embedding of a term: the uniqueness of the term, its ancestors and the sub-ontology to which it belongs. Anc2vec has proven useful for data visualization, sub-ontology prediction, inference of structurally related terms, retrieval of terms from aggregated embeddings, and prediction of protein-protein interactions. Results on large-scale, real-world data show that anc2vec embeddings can encode much more semantic information than existing embeddings.

7 Funding

This work was supported by ANPCyT (PICT 2018 #3384) and UNL (CAI+D 2020 115).

8 Author contributions statement

A.E., D.M., and G.S. conceived experiments; A.E. conducted experiments and prepared figures; A.E., D.M., and G.S. analyzed results. A.E., D.M., and G.S. wrote and reviewed the manuscript.

9 Conflict of interests

The authors declare no competing interests.

10 Acknowledgments

We also acknowledged the support of NVIDIA Corporation for the donation of a Titan Xp GPU used for this research.

11 Key points

- A novel neural network model named anc2vec is presented, for constructing embeddings of GO terms.
- Anc2vec preserves three structural features of the GO in the embedding of a term: the uniqueness of the term, its ancestors and the sub-ontology to which it belongs.
- Anc2vec has proven useful for data visualization, sub-ontology prediction, inference of structurally related terms, retrieval of terms from aggregated embeddings, and prediction of protein-protein interactions.
- Results on large-scale, real-world data show that anc2vec embeddings can encode much more semantic information than existing embeddings.

References

Gene Ontology Consortium. The gene ontology resource: 20 years and still going strong. *Nucleic acids research*, 47(D1):D330–D338, 2019.

Catia Pesquita et al. Semantic similarity in biomedical ontologies. *PLoS Comput Biol Computational Biology*, 5(7):e1000443, jul 2009. doi: 10.1371/journal.pcbi.1000443.

Chenguang Zhao and Zheng Wang. GOGO: An improved algorithm to measure the semantic similarity between gene ontology terms. *Scientific Reports*, 8(1), oct 2018. doi: 10.1038/s41598-018-33219-y.

Rita T Sousa, Sara Silva, and Catia Pesquita. Evolving knowledge graph similarity for supervised learning in complex biomedical domains. *BMC bioinformatics*, 21(1):1–19, 2020.

Pietro H Guzzi, Marco Mina, Concettina Guerra, and Mario Cannataro. Semantic similarity analysis of protein data: assessment with biological features and issues. *Briefings in bioinformatics*, 13(5): 569–585, 2012.

Gaston K Mazandu, Emile R Chimusa, and Nicola J Mulder. Gene ontology semantic similarity tools: survey on features and challenges for biological knowledge discovery. *Briefings in bioinformatics*, 18 (5):886–901, 2017.

P Resnik. Using information content to evaluate semantic similarity in a taxonomy. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI). San Francisco: Morgan Kaufmann*, volume 448, page 453, 1995.

P. Resnik. 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, jul 1999. doi: 10.1613/jair.514.

Xuebo Song, Lin Li, Pradip K. Srimani, Philip S. Yu, and James Z. Wang. Measure the semantic similarity of GO terms using aggregate information content. *ACM Transactions on Computational Biology and Bioinformatics*, 11(3):468–476, may 2014. doi: 10.1109/tcbb.2013.176.

J. Z. Wang et al. A new method to measure the semantic similarity of GO terms. *Bioinformatics*, 23 (10):1274–1281, mar 2007. doi: 10.1093/bioinformatics/btm087.

Petar Ristoski and Heiko Paulheim. Rdf2vec: Rdf graph embeddings for data mining. In *International Semantic Web Conference*, pages 498–514. Springer, 2016.

Xiaoshi Zhong, Rama Kaalia, and Jagath C Rajapakse. Go2vec: transforming go terms and proteins to vector representations via graph embeddings. *BMC genomics*, 20(9):1–10, 2019.

Mehdi Ali, Charles Tapley Hoyt, Daniel Domingo-Fernández, Jens Lehmann, and Hajira Jabeen. Biokeen: a library for learning and evaluating biological knowledge graph embeddings. *Bioinformatics*, 35(18): 3538–3540, 2019.

Maxat Kulmanov, Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Semantic similarity and machine learning with ontologies. *Briefings in bioinformatics*, 22(4):bbaa199, 2021.

Mona Alshahrani, Maha A Thafar, and Magbubah Essack. Application and evaluation of knowledge graph embeddings in biomedical data. *PeerJ Computer Science*, 7:e341, 2021.

María Virginia Sabando, Ignacio Ponzoni, Evangelos E Milios, and Axel J Soto. Using molecular embeddings in qsar modeling: Does it make a difference? *Briefings in Bioinformatics*, 2021.

Jin Liu, Ran Su, Jiahang Zhang, and Leyi Wei. Classification and gene selection of triple-negative breast cancer subtype embedding gene connectivity matrix in deep neural network. *Briefings in Bioinformatics*, 2021.

Fatima Zohra Smaili, Xin Gao, and Robert Hoehndorf. Onto2vec: joint vector-based representation of biological entities and their ontology-based annotations. *Bioinformatics*, 34(13):i52–i60, jun 2018a. doi: 10.1093/bioinformatics/bty259.

Fatima Zohra Smaili et al. OPA2vec: combining formal and informal content of biomedical ontologies to improve similarity-based prediction. *Bioinformatics*, 35(12):2133–2140, nov 2018b. doi: 10.1093/bioinformatics/bty933.

Dat Duong et al. Evaluating representations for gene ontology terms. *bioRxiv*, page 765644, 2020.

Lingling Zhao et al. Conjoint feature representation of GO and protein sequence for PPI prediction based on an inception RNN attention network. *Molecular Therapy - Nucleic Acids*, 22:198–208, dec 2020. doi: 10.1016/j.omtn.2020.08.025.

Gaston K. Mazandu and Nicola J. Mulder. A topology-based metric for measuring term similarity in the gene ontology. *Advances in Bioinformatics*, 2012:1–17, may 2012. doi: 10.1155/2012/975783.

Jiongmin Zhang et al. An improved approach to infer protein-protein interaction based on a hierarchical vector space model. *BMC Bioinformatics*, 19(1), apr 2018. doi: 10.1186/s12859-018-2152-z.

Rachael P. Huntley et al. The GOA database: Gene ontology annotation updates for 2015. *Nucleic Acids Research*, 43(D1):D1057–D1063, nov 2014. doi: 10.1093/nar/gku1113.

Naihui Zhou et al. The cafa challenge reports improved protein function prediction and new functional annotations for hundreds of genes through experimental screens. *Genome biology*, 20(1):1–23, 2019.

Damian Szklarczyk et al. STRING v11: protein–protein association networks with increased coverage, supporting functional discovery in genome-wide experimental datasets. *Nucleic Acids Research*, 47 (D1):D607–D613, nov 2018. doi: 10.1093/nar/gky1131.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

D. V. Klopfenstein et al. GOATOOLS: A python library for gene ontology analyses. *Scientific Reports*, 8(1), jul 2018. doi: 10.1038/s41598-018-28948-z.

Francisco Azuaje, Haiying Wang, and Olivier Bodenreider. Ontology-driven similarity approaches to supporting gene functional assessment. In *Proceedings of the ISMB 2005 SIG meeting on Bio-ontologies*, volume 2005, pages 9–10, 2005.

Catia Pesquita et al. Metrics for GO based protein semantic similarity: a systematic evaluation. *BMC Bioinformatics*, 9(S5), apr 2008. doi: 10.1186/1471-2105-9-s5-s4.

Soheil Kolouri et al. Optimal mass transport: Signal processing and machine-learning applications. *IEEE signal processing magazine*, 34(4):43–59, jul 2017. doi: 10.1109/msp.2017.2695801.

Federico Bassetti et al. On the computation of kantorovich–wasserstein distances between two-dimensional histograms by uncapacitated minimum cost flows. *SIAM Journal on Optimization*, 30(3):2441–2469, jan 2020. doi: 10.1137/19m1261195.

Takaya Saito and Marc Rehmsmeier. The precision-recall plot is more informative than the ROC plot when evaluating binary classifiers on imbalanced datasets. *PLoS ONE*, 10(3):e0118432, mar 2015. doi: 10.1371/journal.pone.0118432.

Leland McInnes et al. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

Minoru Kanehisa et al. KEGG: integrating viruses and cellular organisms. *Nucleic Acids Research*, 49 (D1):D545–D551, oct 2020. doi: 10.1093/nar/gkaa970.

Diego Alonso-López, Francisco J Campos-Laborie, Miguel A Gutiérrez, Luke Lambourne, Michael A Calderwood, Marc Vidal, and Javier De Las Rivas. Apid database: redefining protein–protein interaction experimental evidences and binary interactomes. *Database*, 2019, 2019.