Supplementary - Leveraging Heterogeneous Network Embedding for Metabolic Pathway Prediction

Abdur Rahman M. A. Basher and Steven J. Hallam

October 1, 2020

1 Definitions

In this section, we provide several additional definitions used in the paper. We first re-state the definitions of a heterogeneous information network and a meta-path as:

**Definition 1.1. Heterogeneous Information Network** A heterogeneous information network is defined as a graph \( G = (V, E) \), where \( V \) and \( E \) denote the set of nodes and edges (either directed or undirected), respectively [10]. Each \( v \in V \) is associated with an object type mapping function \( \phi(v) : V \rightarrow O \), where \( O \) represents a set of object types. Each edge \( e \in E \subseteq V \times V \) includes multiple types of links, and is associated with a link type mapping function \( \phi(e) : E \rightarrow R \), where \( R \) represents a set of relation types. In particular, when \(|O| + |R| > 2\), the graph is referred to as a heterogeneous information network.

**Definition 1.2. Meta-Path** A meta-path \( P \in \mathcal{P} \) is a path over \( G \) in the form of \( O_1 \xrightarrow{R_1} O_2 \xrightarrow{R_2} O_3 \xrightarrow{R_3} \ldots \xrightarrow{R_j} O_{j+1} \), which defines an aggregation of relationships \( U = R_1 \circ R_2 \circ \ldots \circ R_j \) between type \( O_1 \) and \( O_{j+1} \), where \( \circ \) denotes the composition operator on relationships and \( O_i \in O \) and \( R_k \in R \) are object and relation type, respectively.

The following is the definition of a first-order random walk as implemented in DeepWalk [9], extending a nodes immediate neighbors to include nodes that are locally connected.

**Definition 1.3. Random Walk** [9]. A random walk \( W \) of length \( l \), rooted at node \( v \), is a stochastic process with random variables \( v^1, v^2, ..., v^l, v^{l+1} \) such that \( v^{l+1} \) is a vertex sampled at random from the neighbors of vertex \( v^j \) for all \( 1 \leq j \leq l \) according to the following distribution:

\[
p(v^{j+1}|v^j) = \begin{cases} \alpha \pi_{v^{j+1}|v^j} & \text{if } (v^j, v^{j+1}) \in E \\ 0 & \text{otherwise} \end{cases} \quad (1.1)
\]

where \( \pi_{v^{j+1}|v^j} \in \mathbb{R}^{|V|\times|V|} \) is an unnormalized transition probability, indicating the probability of a random walker visiting a node \( v^{j+1} \) conditioned on the current node being \( v^j \), \( Q \) is a normalizing term, and \( \alpha \in [0, 1] \) is a prior probability.

The above definition does not address in-depth and in-breadth graph exploration. Therefore, node2vec [6] was proposed where the process of node2vec random walks can be defined by manipulating \( \alpha \) in Def. 1.3 according to:

\[
\alpha_{s,h}(j - 1, j + 1) = \begin{cases} \frac{1}{2} & \text{if } \beta_{j-1,j+1} = 0 \\ 1 & \text{if } \beta_{j-1,j+1} = 1 \\ \frac{1}{\beta_{j-1,j+1}} & \text{if } \beta_{j-1,j+1} = 2 \end{cases} \quad (1.2)
\]

where \( \beta_{j-1,j+1} \in \{0, 1, 2\} \) denotes the distance between the previously visited node \( v^{j-1} \) and the next neighbor node \( v^{j+1} \).

First-order and second-order random walks were initially proposed for homogeneous graphs, but can be readily extended to heterogeneous information networks. This lead to developing metapath2vec [4] where meta-path based walks can be defined as:

**Definition 1.4. Meta-Path based Random Walk** [4]. Given a meta-path \( P \in \mathcal{P} \) and \( G \), a meta-path based random walk \( W \) of length \( l \), rooted at node \( v \in O \) as dictated by \( P \), is a stochastic process with random variables \( v^1, v^2, ..., v^l, v^{l+1} \) such that \( v^{l+1} \) is a vertex of type \( O_{k+1} \) sampled randomly from the neighbors of vertex \( v^j \in O_k \) for all \( 1 \leq j \leq l \) according to the following distribution:

\[
p(v^{j+1}|v^j, P) = \begin{cases} \frac{1}{|N_{v^j}(v^{j+1})|} & \text{if } (v^j, v^{j+1}) \in E, \phi(v^{j+1}) = k + 1 \\ 0 & \text{if } (v^j, v^{j+1}) \in E, \phi(v^{j+1}) \neq k + 1 \\ 0 & \text{if } (v^j, v^{j+1}) \notin E \end{cases} \quad (1.3)
\]
Definition 1.5. Jump and Stay based Random Walk (JUST). Given a set of domain types \( \mathcal{O} \), a graph \( G \), a queue \( M \) of size \( m \), and an initial stay probability \( \alpha \in [0, 1] \), a JUST-based random walk \( W \) of length \( l \), rooted at node \( v \in \mathcal{O} \), is a stochastic process with random variables \( v^1, v^2, \ldots, v^l, v^{l+1} \) such that \( v^{l+1} \) is selected according to the two following consecutive steps:

1. Predict the stay probability \( p_{\text{stay}} \) as:
   
   \[
   p_{\text{st}}(v^j) = \begin{cases} 
   0 & \text{if } S(v^j) = \emptyset \\
   1 & \text{if } J(v^j) = \emptyset \\
   \alpha^c & \text{if otherwise}
   \end{cases}
   \]  
   
   where \( c \) is the number of nodes consecutively visited in the same domain as \( v^j \) and the remaining terms \( S(v^j) \) and \( J(v^j) \) are:
   
   \[
   S(v^j) = \{v^{j+1} | (v^j, v^{j+1}) \in E \land \phi(v^j) = \phi(v^{j+1})\}
   \]
   
   \[
   J(v^j) = \{v^{j+1} | (v^j, v^{j+1}) \in E \land \phi(v^j) \neq \phi(v^{j+1})\}
   \]  

2. Sample \( v^{j+1} \) either: i)- from the same domain as \( v^j \) or ii)- apply the equation below, if \( p_{\text{st}}(v^j) = 0 \) or \( p_{\text{st}}(v^j) = 1 - \alpha^c \):

   \[
   H(v^j) = \begin{cases} 
   \{k| k \in \mathcal{O} \land k \notin M, J(v^j) \neq \emptyset\} & \text{if empty} \\
   \{k| k \in \mathcal{O}, \, k \neq \phi(v^j), J(v^j) \neq \emptyset\} & \text{if otherwise (1.5)}
   \end{cases}
   \]

   where \( M \) is a queue of size \( m \) that stores \( m \) previously visited types.

If the set \( S(v^j) \) is empty, meaning no edges exist between \( v^j \) and any nodes in \( \mathcal{V} \) that share the same domain type as \( v^j \), then a node is sampled from different types based on Eq. 1.5 which suggests to select randomly any domains not included in \( M \). If, however, the latter condition is not satisfied then simply choose one domain that is different from the current node type. If \( J(v^j) \) is empty, i.e., no heterogeneous edges connected to \( v^j \), then the random walker is forced to stay in the same domain. Finally, if both homogeneous and heterogeneous edges are connected to \( v^j \) then the walker may choose to either stay with \( \alpha^c \) or jump with \( 1 - \alpha^c \), where \( \alpha \) value decays exponentially by \( c \) that stores the number of nodes sequentially visited in the same type of \( v^j \). This can misrepresent graph structure in two ways: i)- explorations within domain because the last visited consecutive \( c \) nodes may enforce sampling from another domain, or ii) jumping deep towards nodes from other domains because \( M \) is constrained. To alleviate these problems we develop a novel random walk algorithm, RUST, adopting a unit-circle equation to sample node pairs that generalize previous representational learning methods.

Definition 1.6. Unit-Circle based Jump and Stay Random Walk (RUST). Given a set of domain types \( \mathcal{O} \), a graph \( G \), a queue \( M \) of size \( m \), and two hyper-parameters \( s \) and \( h \), a RUST-based random walk \( W \) of length \( l \), rooted at node \( v \in \mathcal{O} \), is a stochastic process with random variables \( v^1, v^2, \ldots, v^l, v^{l+1} \) such that \( v^{l+1} \) is chosen in two steps:

1. Estimate domain types transition probabilities given \( v^j \):

   \[
   \pi_{\text{dom}}_{j,j+1} = \begin{cases} 
   \frac{h \beta_j \tilde{\pi}_{j-1,j+1}}{Q} & \text{if } S(v^j) = \emptyset \\
   \frac{s \beta_j \tilde{\pi}_{j-1,j+1}}{Q} & \text{if } J(v^j) = \emptyset
   \end{cases}
   \]  

   where \( Q \) is a normalizing term, \( \beta_j \in [0, 1] \) is a domain weight hyperparameter of \( v^j \) added to give more weights, if necessary, to some domains, and \( \pi_{j-1,j} \) is an unnormalized transition probability from previous node \( v^{j-1} \) to the current node \( v^j \). The remaining terms:

   \[
   S(v^j) = \{v^{j+1} | (v^j, v^{j+1}) \in E \land \phi(v^j) = \phi(v^{j+1})\}
   \]
   
   \[
   J(v^j) = \{v^{j+1} | (v^j, v^{j+1}) \in E \land \phi(v^j) \neq \phi(v^{j+1})\}
   \]
2. Sample a domain type $k$ at random from $\pi_{dom}^{j,j+1}$ according to:

$$H(v^j) = \{k | k \in \mathcal{O}, \alpha_k \pi_{dom}^{j,j+1}\}$$  \hspace{1cm} (1.7)

where $\alpha_k = 1/e^{c_k}$ and $c_k$ is the number of nodes with type $k$ that is stored in $M$. Finally, select randomly a node $v^{j+1}$ based on $H(v^j)$.

The two hyper-parameters $s$ and $h$ constitute a unit circle, i.e., $h^2 + s^2 = 1$, where $h \in [0, 1]$ indicates how much exploration is needed within a domain while $s \in [0, 1]$ defines the in-depth search towards other domains such that $s > h$ encourages the walk to explore more domains and vice versa. Consequently, RUST blends both semantic associations and local/global structural information for generating walks without restricting the number of memorized domains $m$ while the $\alpha_k$ serves as a function of node size having the type $k$ as stored in $M$. Algorithm 1 presents the pseudocode of RUST based random walk.

2 Parameter Sensitivity of RUST

Fig. 1a indicates that RUST performance tends to saturate when the memorized domains are concentrated around $m = 5$ and $h = 0.55$, indicating a preference to explore more domain types. By fixing $m = 3$ and $h = 0.55$ the optimal results of NMI score w.r.t. the number of embedding dimensionality was found to be at 80 and 128 (Fig. 1b). Beyond this value RUST performance deteriorated. A similar trend was also observed when the context neighborhood size was increased beyond $q > 5$ (Fig. 1c). Based on these observations, the following settings $m = 3$, $h = 0.55$, $d = 80$ or $d = 128$, and $q = 5$ provide the most efficient and accurate clustering outcomes using MetaCyc with uec option. For comparative purposes, we set $d = 128$.

3 Node Clustering

Fig. 2 indicates node clustering results for node2vec, metapath2vec, JUST and RUST. node2vec, JUST and RUST exhibited similar performance across all configurations, indicating that these methods are less likely to extract semantic knowledge, characterizing node domains, from MetaCyc. However, RUST performed optimally better than node2vec and JUST in learning representations. In the case of metapath2vec, the random walk follows a predefined meta-path scheme, capturing the necessary relational knowledge for defining node types. For example, nitrogenase (EC-1.18.6.1), which reduces nitrogen gas into ammonium, is exclusively linked to the nitrogen fixation I (ferredoxin) pathway (Fig. 3). Without a predefined relation, a walker may explore more local/global structure of $G$, hence, become less efficient in exploiting relations between these two nodes. Among the four walks, only metapath2vec is able to accurately group those nodes, according to their classes. Despite the advantages of metapath2vec, it is biased to a scheme, as described in (7), which is explicitly observed for the case of “uec+r” (Fig. 2d). Under these conditions, both isolated nodes and links among ECs are discarded, resulting in a reduced number of nodes that are more easily traversed by a meta-path walker. metapath2vec++ exhibited trends similar to metapath2vec because they share the same walks. However, metapath2vec++ is
trained using normalized Skip-Gram. Therefore, it is expected to achieve good NMI scores, yielding over 0.41 on uec+full content (in Fig. 3), which is also similar to RUST-norm NMI score (\(\sim 0.38\)). This is interesting because RUST-norm employs RUST based walks but the embeddings are learned using normalized Skip-Gram.

4 Manifold visualization

We examined the limitations of meta-path based random walks. In this case, 80 pathways were identified, having no enzymatic reactions, with their 109 pathway neighbors, as shown in Fig. 4a. From Fig. 4, we observe that, in contrast to node2vec, JUST, RUST, and RUST-norm, pathway nodes are skewed incorrectly in both metapath2vec and metapath2vec++ and (with lesser degree). This demonstrates the rigidity of meta-path based methods that follow a defined scheme that limits their capacity to exploit local structure in learning embeddings. Interestingly, RUST-norm, based on RUST walks, is the only method that combines structural and semantic information as indicated in Fig. 4g.

5 Dataset Characteristics

The datasets employed in this work are: i)- golden tier 1 dataset, composed of six databases, retrieved from biocyc website: EcoCyc (v21), HumanCyc (v19.5), AraCyc (v18.5), YeastCyc (v19.5), LeishCyc (v19.5), and TrypanoCyc (v18.5), and are refined to include only those pathways that cross-intersect with the MetaCyc database (v21) [1]; and ii)- BioCyc (v20.5 tier 2 & 3) [2], which consists of 9255 PGDBs (Pathway/Genome Databases) with 1463 distinct pathway labels and is constructed using the Pathway Tools software [8]. The detailed characteristics of the datasets are summarized in Table 1. For each dataset \( \mathcal{S} \), we use \(|\mathcal{S}|\) and \(L(\mathcal{S})\) to represent the number of instances and pathway labels, respectively. In addition, we also present some characteristics of the multi-label datasets, which are denoted as:

1. Label cardinality (\(L\text{Card}(\mathcal{S}) = \frac{1}{n} \sum_{i=1}^{n} \sum_{j=1}^{t} \mathbb{I}[Y_{i,j} \neq -1]\)), where \(\mathbb{I}\) is an indicator function. It denotes the average number of pathways in \(\mathcal{S}\).
2. Label density (\(L\text{Den}(\mathcal{S}) = \frac{L\text{Card}(\mathcal{S})}{L(\mathcal{S})}\)). This is simply obtained through normalizing \(L\text{Card}(\mathcal{S})\) by the number of total pathways in \(\mathcal{S}\).
3. Distinct label sets (\(D\text{L}(\mathcal{S})\)). This notation indicates the number of distinct pathways in \(\mathcal{S}\).
4. Proportion of distinct label sets (\(P\text{D}(\mathcal{S}) = \frac{D\text{L}(\mathcal{S})}{|\mathcal{S}|}\)). It represents the normalized version of \(D\text{L}(\mathcal{S})\), and is obtained by dividing \(D\text{L}(\cdot)\) with the number of instances in \(\mathcal{S}\).

The notations \(R(\mathcal{S}), R\text{Card}(\mathcal{S}), R\text{Den}(\mathcal{S}), D\text{R}(\mathcal{S}), \) and \(P\text{D}(\mathcal{S})\) have similar meanings as before but for the enzymatic reactions \(\mathcal{E}\) in \(\mathcal{S}\). Finally, \(P\text{LR}(\mathcal{S})\) represents a ratio of \(L(\mathcal{S})\) to \(R(\mathcal{S})\).

6 Metabolic Pathway Prediction

For this case study, we report the performance of mlLGPR (elastic-net) using the default hyperparameter settings, after concatenating features from each learning method, on 6 benchmark datasets, as described in
Section 5. The mlLGPR framework is based on a multi-label classification approach that uses logistic regression and feature vectors inspired by the work of Dale and colleagues [3] to predict metabolic pathways for individual genomes as well as more complex cellular communities (e.g. microbiomes). In the original settings of mlLGPR, five feature sets were applied: i)- enzymatic reactions abundance (AB) that corresponds a set of ECs with abundance information for each example in a dataset; ii)- reactions evidence (RE) that describes the properties of enzymatic reactions for each example); iii)- pathways evidence (PE) that captures information about metabolic pathways in a given example; iv)- pathway common (PC) that recognizes (mis-)matches between a list of ECs of each example and the true mappings of pathways to ECs; and v)- possible pathways (PP) that indicates probable presence/absence of pathways in each example. However, to learn mlLGPR using embeddings, we abandoned these 5 feature sets and trained using multi-label 1-vs-All approach which decomposes the multi-label problem into a set of binary classification problems, one for each pathway class [11]. This technique of training was applied in the previous mlLGPR framework. Table 2 shows the comparative performances of mlLGPR (with embeddings) against other pathway prediction algorithms using four evaluation metrics: Hamming loss, Micro precision, Micro recall, and Micro F1 score.
Figure 3: Node clustering results of metapath2vec++ (cm2v) and RUST-norm (crt) based on NMI metric using MetaCyc data.

Figure 4: 2D UMAP projections of 80 pathways that have no enzymatic reactions, indicated by the blue color, with 109 corresponding pathway neighbors, represented by the grey color. n2v: node2vec, m2v: metapath2vec, jt: JUST, rt: RUST, cm2v: metapath2vec++, and crt: RUST-norm.

7 Scalability

Here, we analyzed training times (after 3 epochs) under full+uec MetaCyc, where node2vec (n2v), in Fig 5, was observed to be scalable to thousands of nodes without requiring prior knowledge on meta-paths, while metapath2vec++ (cm2v) and RUST-norm (crt) are less likely to scale on a larger graph.

8 Similarity Search

We conducted cosine similarity search to determine the distance between the query pathway and the rest of pathways using metapath2vec++ (any other method discussed in this paper can be employed). For this, we selected a total of 23 nitrogen metabolism pathways, including pathway variants in Table 3. For illustration purposes, Table 4 lists only the top 5 results for querying the 7 pathway ids. One can observe that for the query id “DENITRIFICATION-PWY” (nitrate reduction I (denitrification)), for example, cm2v returns pathways that are similar to it, such as “PWY-5674” (nitrate reduction IV (dissimilatory)) and “PWY-5675” (nitrate reduction V (assimilatory)). Similar results can also be recovered when querying other pathways (or ECs).

References

[1] Ron Caspi, Richard Billington, Luciana Ferrer, Hartmut Foerster, Carol A. Fulcher, Ingrid M. Keseler, Anamika Kothari, Markus Krummenacker, Mario Latendresse, Lukas A. Mueller, Quang Ong, Suzanne Paley, Pallavi Subhraveti, Daniel S. Weaver, and Peter D. Karp. The metacyc database of metabolic pathways and enzymes and the biocyc collection of pathway/genome databases. *Nucleic Acids Research*, 44(D1):D471–D480, 2016.

[2] Ron Caspi, Richard Billington, Hartmut Foerster, Carol A Fulcher, Ingrid Keseler, Anamika Kothari, Markus Krummenacker, Mario Latendresse, Lukas A Mueller, Quang Ong, et al. Biocyc: Online resource for genome and metabolic pathway analysis. *The FASEB Journal*, 30(1 Supplement):lb192–lb192, 2016.
Table 1: Characteristics of the experimental datasets. The notations $|S|$, L($S$), LCard($S$), LDen($S$), DL($S$), and PDL($S$) represent number of instances, number of pathway labels, pathway labels cardinality, pathway labels density, distinct pathway labels set, and proportion of distinct pathway labels set for $S$, respectively. The notations R($S$), RCard($S$), RDen($S$), DR($S$), and PDR($S$) have similar meanings as before but for the enzymatic reactions $E$ in $S$. PLR($S$) represents a ratio of L($S$) to R($S$). The last column denotes the domain of $S$.

| Dataset       | |S| | L(S) | LCard(S) | LDen(S) | DL(S) | PDL(S) | R(S) | RCard(S) | RDen(S) | DR(S) | PDR(S) | Domain                                      |
|---------------|----------------|-----|------|--------|--------|-------|--------|------|----------|---------|-------|--------|---------------------------------------------|
| Synth2        | 15000 | 6800202 | 1937508 | 0.00007 | 2026  | 0.1084 | 3486636 | 228130224 | 0.00001 | 3690  | 0.2453 | 0.2001 | synthetically generated (corrupted) Escherichia coli K-12 sub-str. MG1655 |
| EcoCyc       | 1 | 307  | 307 | 1 | 307  | 307 | 1134 | 1134 | 1 | 719  | 719 | 0.2707 | Homo sapiens                                      |
| HumanCyc     | 1 | 279  | 279 | 1 | 279  | 279 | 1177 | 1177 | 1 | 693  | 693 | 0.2370 | Arabidopsis thaliana Saccharomyces cerevisiae Leishmania major Trypanosoma brucei |
| AraCyc       | 1 | 510  | 510 | 1 | 510  | 510 | 2182 | 2182 | 1 | 1034 | 1034 | 0.2337 | Homo sapiens                                      |
| YeastCyc     | 1 | 229  | 229 | 1 | 229  | 229 | 966  | 966  | 1 | 544  | 544 | 0.2371 | Homo sapiens                                      |
| LeishCyc     | 1 | 87   | 87  | 1 | 87   | 87  | 363  | 363  | 1 | 292  | 292 | 0.2397 | Homo sapiens                                      |
| TrypanCyc    | 1 | 175  | 175 | 1 | 175  | 175 | 743  | 743  | 1 | 512  | 512 | 0.2355 | Homo sapiens                                      |
| BioCyc       | 9255 | 1804003 | 1949220 | 0.0001 | 1463 | 0.1581 | 8845714 | 9561689 | 0.0001 | 2705 | 0.2923 | 0.2039 | BioCyc version 20.5 (tier 2 & 3)              |

Figure 5: Scalability measured in seconds ($\times 10^3$) under uec+full configuration. n2v: node2vec, m2v: metapath2vec, cm2v: metapath2vec++, and crt: RUST-norm.

[3] Joseph M Dale, Liviu Popescu, and Peter D Karp. Machine learning methods for metabolic pathway prediction. *BMC bioinformatics*, 11(1):1, 2010.

[4] Yuxiao Dong, Nitesh V Chawla, and Ananthram Swami. metapath2vec: Scalable representation learning for heterogeneous networks. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 135–144. ACM, 2017.

[5] Robert R Eady. Structure-function relationships of alternative nitrogenases. *Chemical reviews*, 96(7):3013–3030, 1996.

[6] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 855–864. ACM, 2016.
[7] Rana Hussein, Dingqi Yang, and Philippe Cudré-Mauroux. Are meta-paths necessary?: Revisiting heterogeneous graph embeddings. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 437–446. ACM, 2018.

[8] Peter D Karp, Mario Latendresse, Suzanne M Paley, Markus Krummenacker, Quang D Ong, Richard Billington, Anamika Kothari, Daniel Weaver, Thomas Lee, Pallavi Subhraveti, et al. Pathway tools version 19.0 update: software for pathway/genome informatics and systems biology. *Briefings in bioinformatics*, 17(5):877–890, 2016.

[9] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 701–710. ACM, 2014.

[10] Yizhou Sun, Jiawei Han, Xifeng Yan, Philip S Yu, and Tianyi Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proceedings of the VLDB Endowment*, 4(11), 2011.

[11] Min-Ling Zhang and Zhi-Hua Zhou. A review on multi-label learning algorithms. *IEEE transactions on knowledge and data engineering*, 26(8):1819–1837, 2014.
| Methods               | Hamming Loss ↓ | Micro F1 Score ↑ |
|-----------------------|----------------|-----------------|
|                       | EcoCyc | HumanCyc | AraCyc | YeastCyc | LeishCyc | TrypanoCyc | EcoCyc | HumanCyc | AraCyc | YeastCyc | LeishCyc | TrypanoCyc |
| PathoLogic            | 0.0610  | 0.0633   | 0.1188  | 0.0424   | 0.0368   | 0.0424     | 0.7230  | 0.6695   | 0.7011  | 0.7194   | 0.4803   | 0.5480     |
| MinPath               | 0.2257  | 0.2530   | 0.3266  | 0.2482   | 0.1615   | 0.2561     | 0.9902  | 0.9713   | 0.9843  | 1.0000   | 1.0000   | 1.0000     |
| mlLGPR                | 0.0804  | 0.0633   | 0.1069  | 0.0550   | 0.0380   | 0.0590     | 0.7700  | 0.5556   | 0.6620  | 0.6723   | 0.4159   | 0.4076     |
| mlLGPR+n2v            | 0.0558  | 0.1021   | 0.1706  | 0.0768   | 0.0424   | 0.0883     | 0.7993  | 0.5873   | 0.7898  | 0.6581   | 0.3983   | 0.4105     |
| mlLGPR+m2v            | 0.0558  | 0.0998   | 0.1742  | 0.0740   | 0.0412   | 0.0926     | 0.7747  | 0.6014   | 0.6732  | 0.6949   | 0.3840   | 0.3924     |
| mlLGPR+cm2v           | 0.0586  | 0.1041   | 0.1742  | 0.0744   | 0.0420   | 0.0867     | 0.8000  | 0.6014   | 0.6635  | 0.6560   | 0.4146   | 0.4113     |
| mlLGPR+jt             | 0.0550  | 0.1041   | 0.1738  | 0.0724   | 0.0459   | 0.0895     | 0.7889  | 0.6014   | 0.6635  | 0.6560   | 0.4146   | 0.4113     |
| mlLGPR+rt             | 0.0554  | 0.0990   | 0.1746  | 0.0752   | 0.0428   | 0.0855     | 0.7074  | 0.5745   | 0.6109  | 0.5632   | 0.5862   | 0.5429     |
| mlLGPR+cm2v           | 0.0554  | 0.1091   | 0.1615  | 0.0760   | 0.0439   | 0.0855     | 0.7074  | 0.5745   | 0.6109  | 0.5632   | 0.5862   | 0.5429     |

Table 2: Predictive performance of each comparing algorithm on 6 benchmark datasets. For each performance metric, '↓' indicates the smaller score is better while '↑' indicates the higher score is better.
| Pathway ID | Pathway name |
|-----------|-------------|
| PWY-6964  | ammonia assimilation cycle II |
| GLNSYN-PWY | L-glutamine biosynthesis I |
| AMMOXID-PWY | ammonia oxidation I (aerobic) |
| P303-PWY  | ammonia oxidation II (anaerobic) |
| PWY-2242  | ammonia oxidation III |
| DENITRIFICATION-PWY | nitrate reduction I (denitrification) |
| PWY-6748  | nitrate reduction VII (denitrification) |
| PWY-6523  | nitrite-dependent anaerobic methane oxidation |
| PWY-381   | nitrate reduction II (assimilatory) |
| PWY0-1321 | nitrate reduction III (dissimilatory) |
| PWY-5674  | nitrate reduction IV (dissimilatory) |
| PWY490-3  | nitrate reduction VI (assimilatory) |
| PWY-6748  | nitrate reduction VII (denitrification) |
| PWY0-1352 | nitrate reduction VIII (dissimilatory) |
| PWY0-1581 | nitrate reduction IX (dissimilatory) |
| PWY0-1584 | nitrate reduction X (dissimilatory, periplasmic) |
| PWY0-1573 | nitrate reduction VIIIb (dissimilatory) |
| N2FIX-PWY | nitrogen fixation I (ferredoxin) |
| PWY-7576  | nitrogen fixation II (flavodoxin) |
| PWY-1263  | taurine degradation I |
| PWY-1264  | taurine degradation II |
| TAURINEDEG-PWY | taurine degradation III |
| PWY0-981  | taurine degradation IV |

Table 3: 23 nitrogen metabolism pathways, including variants, as extracted from MetaCyc.

| Rank | Pathway ID | AMMOXID-PWY | P303-PWY | PWY-2242 |
|------|------------|-------------|----------|----------|
| 1    | PWY-6964   | PWY-5366    | PWY-7058 | PWY-1269 |
| 2    | PWY-7672   | PWY-6014    | PWY-862  | AMMOXID-PWY |
| 3    | GLUTAMINEFUM-PWY | PWY-5373    | PWY-7562 | HEMESYN2-PWY |
| 4    | CITRULBIO-PWY | PWY-7802    | PWY-5789 | PWY-6557 |
| 5    | PWY-5675   | PWY-6837    | PWY-6310 | PWY-6873 |

| Rank | DENITRIFICATION-PWY | PWY-490-3 | PWY-1264 | TAURINEDEG-PWY |
|------|---------------------|-----------|----------|----------------|---|
| 1    | PWY-5674            | PWY-5675  | PWY-5944 | PWY-5036       |
| 2    | PWY-5675            | PWY-381   | TAURINEDEG-PWY | PWY-6043 |
| 3    | PWYQT-4471          | PWY-6840  | PWY-5844 | PWY-282        |
| 4    | PWY-7405            | TRNA-CHARGING-PWY | PWY-6423 | PWY-6388      |
| 5    | PWY-6275            | PWY-6945  | PWY-1263 | PWY-7833       |

Table 4: Top 5 Pathway IDs for nitrogen metabolism.