Improving genome-scale metabolic models of incomplete genomes with deep learning

Meine D. Boer[1,2], Chrats Melkonian[1,5], Haris Zafeiropoulos[3], Andreas F. Haas[2], Daniel Garza[3], Bas E. Dutilh[1,4]

Affiliations:

1. Theoretical Biology and Bioinformatics, Utrecht University, Padualaan 8, 3584 CH Utrecht, the Netherlands
2. Dept. Marine Microbiology and Biogeochemistry, NIOZ Royal Netherlands Institute for Sea Research, PO Box 59, 1790 AB Den Burg, Texel, The Netherlands
3. Laboratory of Molecular Bacteriology (Rega Institute), Department of Microbiology, Immunology and Transplantation, KU Leuven, Leuven, Belgium
4. Institute of Biodiversity, Faculty of Biological Sciences, Cluster of Excellence Balance of the Microverse, Friedrich Schiller University Jena, 07743 Jena, Germany
5. Bioinformatics Group, Wageningen University and Research, Wageningen, the Netherlands

Abstract

Deciphering the metabolism of microbial species is crucial for understanding their function within complex ecosystems. Genome-scale metabolic models (GSMMs), which predict metabolic traits based on the enzymes encoded in a genome, are promising tools to study microbial ecosystems when genome sequences can be obtained. However, constructing GSMMs for uncultured bacteria is challenging, as metagenome-assembled genomes (MAGs) are typically incomplete, leading to metabolic reconstructions with numerous gaps. Existing methodologies often fill these gaps with the minimum set of reactions necessary to simulate an objective function such as growth. Here we introduce an artificial intelligence-based alternative: the Deep Neural Network Guided Imputation Of Reactomes (DNNGIOR). The DNNGIOR neural network learns weights for missing reactions in incomplete GSMMs from genomes spanning the bacterial domain. We demonstrate that reactions that occur in >30% of the training genomes can be accurately predicted (F1 score = 0.85). The weights generated by the DNNGIOR neural network improved the gap-filling of incomplete GSMMs, when assessed on a large and phylogenetically diverse testing dataset and on a small set of high-quality manually curated models. The accuracy of DNNGIOR was on average 14 times greater than the standard unweighted gap-filling for draft reconstructions and 2-9 times greater for manually curated models. DNNGIOR is available at https://github.com/MGXlab/DNNGIOR or as a pip package (https://pypi.org/project/dnngior/).

Keywords

Genome-scale metabolic models, metagenome-assembled genomes, machine learning, deep neural network, metabolic gap-filling, metabolic reconstructions, computational methods.

Introduction

Simulating microbial metabolism is an effective method to understand bacterial physiology and interactions within their communities (1–3). The functions and interactions of bacteria can be inferred from their genome sequences using genome-scale metabolic models (GSMMs) (3–6). GSMMs can be constructed either manually or automatically with tools such as RAVEN (7), ModelSEED (8), KBase (9), and CarveMe (10), which identify metabolic reactions encoded on the genome and build a metabolic network. However, if the original genome sequence is incomplete, a common occurrence with metagenome-assembled genomes (MAGs), the inferred GSMM will also be incomplete (11). Consequently, gaps in GSMMs emerge due to missing knowledge and errors introduced during sequencing (12), binning (13,14), and annotation (15). In the past, gap-filling was primarily executed through manual curation (16–18), but this method is time consuming and does not scale well for studies that include a large number of GSMMs (19–21).

Several algorithms have been developed to automate gap-filling, such as FastGapfilling (22), GlobalFit (23), CHESHIRE (24) and OptFill (25) that add the minimal set of reactions that allow a GSMM to simulate growth. The reaction sets that can gap-fill a model are not unique (26) and the organism's actual metabolism may not always align with the minimal set of reactions satisfying a user-defined objective (27). This indicates room for refining gap-filling algorithms to yield more realistic solutions. As most gap-filling algorithms allow us to weigh reactions individually according to their likelihood of being in the model (10,22,26), several attempts have been made to find weights based on genomics (28), proteomics (29), topology (24,26) or reaction type (8). Nevertheless, despite these advances, determining the optimal weights for any reaction and any model still remains challenging (4).

In this article, we introduce DNNGIOR: a python package that uses a neural network to assign weights to metabolic reactions to complete GSMMs that are built from incomplete genomes. This neural network is trained to discern patterns in the cooccurrences of reactions across the bacterial domain and to predict reactions based on incomplete reaction sets. We then used the predictions of the neural network to weight reactions in incomplete GSMMs with the goal of improving automated GSMM reconstruction. We found that the accuracy of the neural network depends on the frequency of reactions in the training data and the phylogenetic relatedness of genomes used to generate this data. Using the predictions for weighted gap-filling, we could match the results of manual curation and improve automated gap-filling.

Methods

Collection and processing of the training and testing data

We constructed the training and testing datasets using genomes collected from the BV-BRC database ((30), formerly PATRIC, accessed 26th April 2022). For training, one genome per species was selected based on sequencing quality scores using the formula: completeness - (5 * contamination). Ties for this score were resolved by selecting the genome with the highest coarse consistency. This selection resulted in a dataset of 13,359 genomes (Supplementary Table 1) that comprehensively represents the bacterial domain while reducing the risk of overfitting on well-studied species. From this training dataset we set aside the 1,659 genomes that were also the best in their genus based on the same score and used those to construct a testing dataset (Supplementary table 2). These genomes were excluded from the training dataset, ensuring that the testing dataset contained diverse bacterial genomes that were different from those in the training dataset.

For the 1,659 genomes in the testing dataset, we created a phylogenetic tree using concatenated alignment of hits to HMM profiles of 71 single-copy marker genes that was used for visualising and further investigating the performance of our approach. Phylogenetic distances in the tree represent the number of amino acid substitutions per site. From all genomes, metabolic models were constructed using CarveMe (10) and ModelSEED (8) and the set of gene-associated reactions was determined. These reaction sets were used to generate a training dataset by repeatedly deleting 30% ($\underline{n} \approx 300$) of the reactions in a given reaction set. Reactions were randomly deleted either according to a uniform probability (figure 1a) or with a bias towards lower frequency reactions, according to the equation shown in Figure 1b.

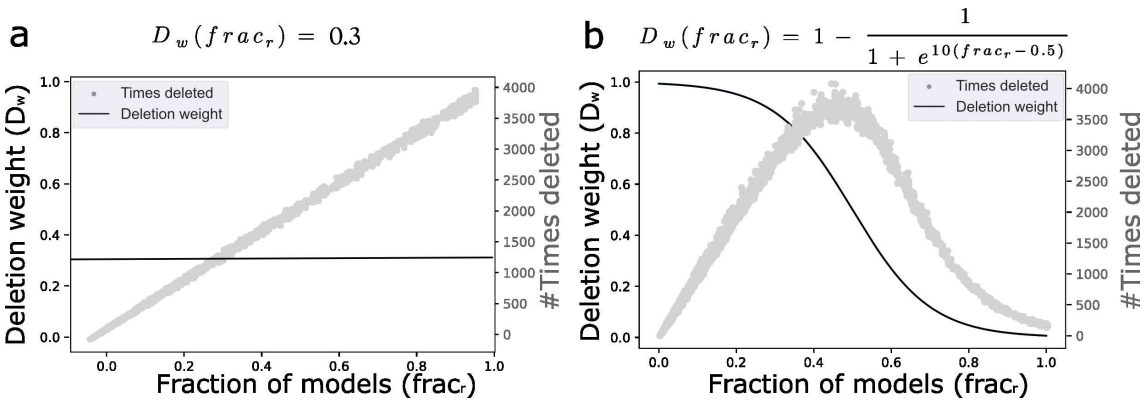

Figure 1: a) Function used for uniform deletion of reactions in the training and testing datasets. Based on these weights, the scatter plot shows how often reactions are actually deleted on average per replicate. $Frac_r$ represents the fraction of genomes in the training dataset that include a reaction. b) Function used for weighted deletion of reactions in the training and testing datasets. Based on these weights, the scatter plot shows how often reactions are actually deleted on average per replicate.

For each genome, the reaction set without reactions removed as originally predicted by the genome-based draft reconstructions was used as truth. We also generated several additional training datasets with certain phyla purposefully excluded to explore the importance of a full representation of the bacterial domain.

Hyper-parameterization and loss function of the neural network

The neural network was built using Tensorflow (31) v2.0.0. The optimizer used for training was the Adam optimizer (32) with the following parameters: lr = 0.005, beta1 = 0.9, beta2=0.999, epsilon = 1.0e-8, decay = 0.01. Hyper-parameterisation was performed in 100 trials using the Optuna package (33) v2.0 and resulted in the following hyper-parameters: nnodes = 256, batch size = 50, nlayers = 3, dropout = 0.1, epochs = 10.

For the loss function we used a customised version of the binary cross-entropy function (34) (CE, Equation 1).

Equation 1: Loss function based on binary cross-entropy (34) to calculate the difference between the neural network output ($\vec{O}$) and the correct reactions ($\vec{T}$).

$$m\overrightarrow{CE} = (\vec{1} - \vec{I}) \circ \left[(1 - b0)\left\{\vec{T} \circ log\ (\vec{O})\right\} - b0\left\{(\vec{1} - \vec{T}) \circ\ log\ (1 - \vec{O})\right\}\right]$$

CE is calculated as the log-loss of the difference between what the network predicts ($\vec{O}$) and the truth ($\vec{T}$). We introduced two adaptations to increase the performance based on our training data. First, as for a given reaction set only ~960 of the possible 2,457 reactions are present, we introduced a scaling factor ($b_0 = 0.3$) that allowed us to scale the loss of the two classes (absent and present). We multiplied b_0 by the loss for the absent class and $1 - b_0$ by the loss for the present class. Second, we added a masking vector (1-T) that allowed us to exclude the loss associated with predictions for reactions that were already known to be part of the genome, i.e. those given as input to the neural network. We multiplied the loss of both classes element-wise b$\vec{1} - \vec{I}$ where $\vec{I}$ is the vector of reactions given as input. This adaptation ensures that the neural network learns to complete the reaction set and does not simply repeat the input.

Gap-filling algorithm and database

After predicting weights for all reactions, we used those weights to guide a half interval search for the minimal set of reactions that simultaneously has a high probability and generates biomass flux that is greater than zero. The half interval gap-filling algorithm was adapted from (22). Briefly, this algorithm iteratively minimises the following objective function with linear programming conditional on flux through the biomass reaction (f_b):

Equation 2: Objective function of the half interval gap-filling algorithm that optimises gap-filling of an incomplete metabolic network based on weights and flux of reactions in a network .

$$\sum_{r \in m} c_r f_r \mid f_b > 0$$

This sums over all reactions (r) in the candidate set (M), with f_r is the flux through reaction r and c_r being a user defined cost used to implement the different weighting schemes during gap-filling. The reaction database from where reactions were selected was downloaded from the BiGG website (http://BiGG.ucsd.edu, Supplementary Table 3) for the BiGG and CarveMe models and from the ModelSEED website (https://modelseed.org, Supplementary Table 4) for the ModelSEED models.

Weighting schemes for guiding the gap-filling algorithm

We created two control weighting schemes, which together with the scheme without weights and the NN-weights gives a total of four weighting schemes: 1. No weights, 2. Naive binary weights, 3. Frequency-based weights and 4. NN-based weights. For no weights all reactions in the database (n = 43,775) are given the default cost ($c_r = 50$). For the naive binary weighting scheme, a lower fixed cost ($c_r = 1$) is given to reactions that are present at least once in the training dataset (R_{train}, n = 2457) and the default cost to all other reactions in the database. For the frequency weighting scheme, a lower cost is given to a reaction if it is present in a higher fraction of genomes in the training dataset (Equation 3). For the NN weighting scheme a lower cost is given to reactions that have a higher prediction score. Because the half interval gap-filling algorithm tries to minimise the product of a reaction's cost and flux, reactions that are given lower costs are given more weight in the final solution.

Equation 3: Weighting schemes to guide the half-interval gap-filling algorithm and serve as a control for the neural network weights.

$$W_{None} = 50$$
$$W_{Naive\ binary} = \begin{cases} 50, r \notin R_{train} \\ 1, r \in R_{train} \end{cases}$$
$$W_{Frequency} = (1 - frac_r)$$
$$W_{NN} = (1 - p_{NN})$$

We compared the four different weighting schemes by gap-filling the models that were constructed based on the 1,659 genomes from the testing dataset from which 30% of reactions ($\underline{n} \approx 300$) was deleted at random. After gap-filling, we counted which removed reactions were re-added correctly (TPs) or not (FNs), and which reactions were falsely added (FPs). These were used to calculate a F1-score for the different weighting schemes.

Curated genome-scale metabolic models

We selected six high-quality manually curated models based on literature (iML1515 (16), iJB785 (35), iYL1228 (36), iND750 (37), iYO844 (38), and iYS854 (39)) from the BiGG database (40). We deleted in 10x replicate 30% of reactions from these models and gap-filled them with the same weighting schemes as for gap-filling the draft-models: No weights, Naive binary weights, Frequency weights and NN-weights. For these models the neural network trained on CarveMe models was used as the reaction identifiers matched those from the curated models. To illustrate which reactions are most likely to be found with the different weighting schemes we also deleted in 500x replicate 30% of reactions from the *E coli* model (iML1515) and constructed an Escher map (41) of the central metabolism.

Results and Discussion

With DNNGIOR we improve the automated reconstruction of metabolic models from incomplete genomes and MAGs. Based on the occurrence of 2,457 metabolic reactions in over 13 thousand different bacterial species, we trained a deep neural network that predicts missing reactions from incomplete genomes. A schematic overview of our approach is depicted in Figure 2.

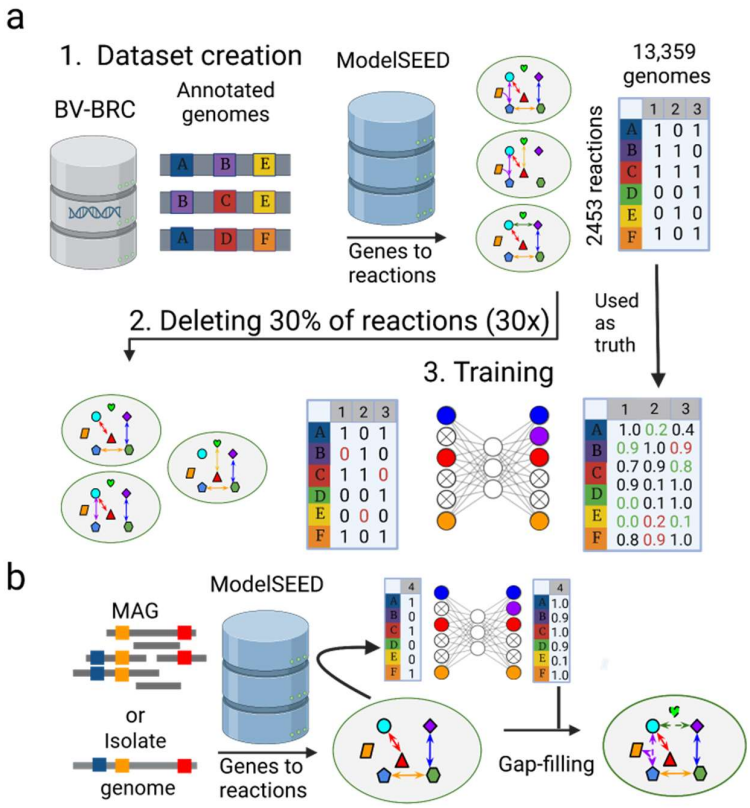

Figure 2: Schematic overview of training and testing the DNNGIOR neural network and weighted gap-filling. a) Construction of the training dataset and training of the neural network. Step 1. Genomes were collected from the BV-BRC (30) database, genomes were annotated and metabolic networks constructed creating an incidence matrix of reactions in different genomes. Step 2. We deleted repeatedly (30x) at random 30% of reactions. Step 3. The network is trained to predict the reactions that were removed during artificial deletion and not predict the reactions that are not part of the original draft model. All reactions given as input to the network are ignored when calculating the loss from the predictions. b) Steps to apply for constructing and gap-filling metabolic models. An automatically constructed model from a metagenome assembled genome (MAG) or isolated genome will be filled with the half interval gap-filling algorithm guided by the predictions generated by the Neural Network. This figure was created using BioRender (42).

Reaction frequency is an important factor for prediction accuracy

We identified two important factors for the neural network to make accurate predictions: (1) the frequency of reaction across all bacteria and (2) the phylogenetic distance between the organisms in the testing and training dataset. Frequent reactions have higher recall and precision than rare reactions, with core reactions that are present in >90% of all bacteria having a recall of 0.96 (sd = 0.007) and precision of 0.86 (sd = 0.039) (Figure 3a). The neural network has more opportunities to learn reactions and their associated patterns than when a reaction is only present in a few genomes.

Specifically, true positives (TPs) increase linearly with frequency while true negatives (TN) decrease linearly (Figure 3e). Since the total number of times a reaction is deleted also increases with frequency, there are more opportunities to correctly predict that a reaction is present or absent. Moreover, the false positives (FPs) and false negatives (FNs) decrease for more frequent reactions (Figure 3c and d). This indicates that the neural network can accurately predict whether a reaction should be present or absent from the model if they are sufficiently represented in the training set.

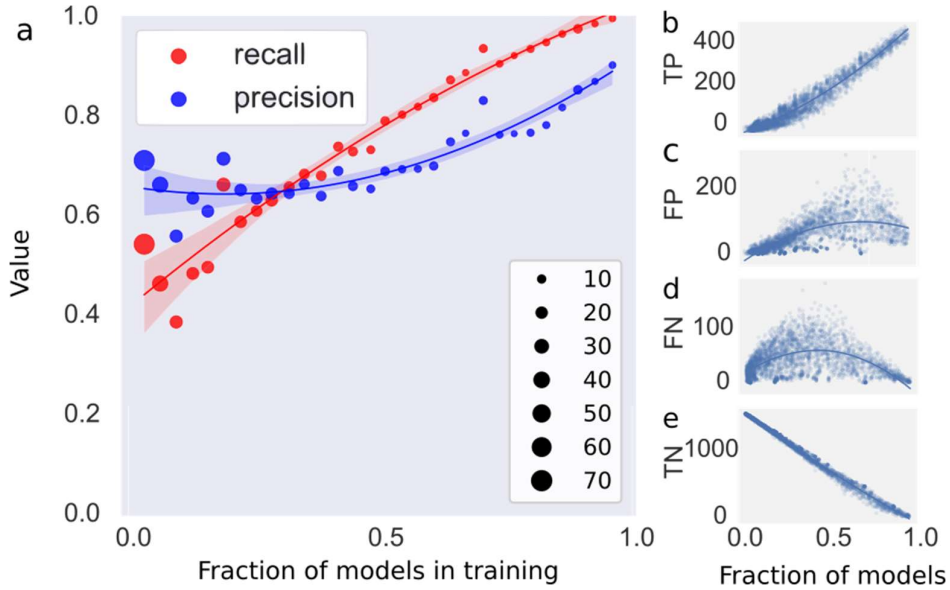

Figure 3: Prediction accuracy increases with reaction frequency a) Relationship between the recall (red) and precision (blue) of predictions as a function of the reaction frequency. Reactions were binned in ranges of 50 reactions, dot size corresponds to bin size, on the y axis precision and recall as a fraction. The shaded region shows the 95% confidence interval of the regression. b-e) Regression plots of True Positives (TP, b), False Positives (FP, c), False Negatives (FN, d) and True Negatives (TN, e) as function of the fraction of genomes a reaction is present in. Trend lines were estimated using a Polynomial Regression Model.

Potential biases in missing reactions

As we have seen above, the frequently occurring reactions can be more accurately predicted. However, the most frequent reactions might not be the ones that are most likely to be missing from incomplete models in practice. There are several biases in binning and annotation that affect the distribution of missing reactions in real data. For instance, accessory genes are more likely to be missing from MAGs because the binning of mobile genetic elements is less effective than the rest of the genome (14). Additionally, large genomes with many accessory genes are more difficult to annotate than smaller genomes with mostly core genes (15). As accessory genes are by definition rarer than core genes (43), this leads to a bias towards rare genes being missing from MAGs. Another important bias is that taxa with less-researched members are more difficult to annotate accurately using homology-based tools (15,44). Finally, reactions can be perceived to be rarer because they are more often missing from the data.

These biases could be a concern if we want to use DNNGIOR to gap-fill models based on larger genomes, MAGs, or from the less researched parts of the bacterial domain as reactions missing from those models might likely be rare. Therefore, we tested the DNNGIOR neural network on data that contained a deliberate deletion bias, where rarer reactions were deleted more often (see Figure 1 in the Methods). In this case, the F1-score decreased 36% as the network overestimated common reactions and underestimated rare reactions (Figure S1). In contrast, if the network was trained on data with the same bias towards deleting rarer reactions, this effect was reduced (from 36 to 18%), and the network became better at predicting rare reactions (Figure S1). This means that it is possible to train the network to account for biases that could potentially be present in the data. However, this and the bias is difficult to quantify, the DNNGIOR package uses a neural network trained on random deletion by default.

Short phylogenetic distances and complete representation improve prediction

The second factor that we identified as important for the prediction accuracy of missing metabolic reactions is the phylogenetic distances between the species in the training data. Predictions were more accurate for species that had close relatives in the training data than for species that were more distantly related (Figure 4). When we made predictions for every species in the test dataset, we found that the F1-score of the predictions correlates with the distance to the closest neighbour (Pearson r^2 = 0.261, p = $1.53*10^{-40}$). Clearly, species that have close relatives in the training data are easier to predict than species that are phylogenetically unique (Figure 4c).

The correlation between F1-score and interspecies distance illustrates the importance of a good and complete representation of the bacterial domain as this would reduce the average distance between query species and those in the training data. We confirmed this hypothesis by excluding some phyla from the training data, which resulted in lower F1-scores compared to when all phyla were included (Figure 4d). We also confirmed that the reverse is true, i.e. training on only reaction sets from one phylum increases the performance of that phylum (Figure S2). Using DNNGIOR specialised neural networks can be trained exclusively on genomes from a certain phylum or biome. This sacrifices general applicability, but it may be expected to yield a (moderate) increase in performance. However, as biome annotations are often subjective and may be both redundant and ambiguous (45), we opted to train a single model and present DNNGIOR as a universal tool that can be directly applied to genomes and MAGs from all phyla and derived from any biome.

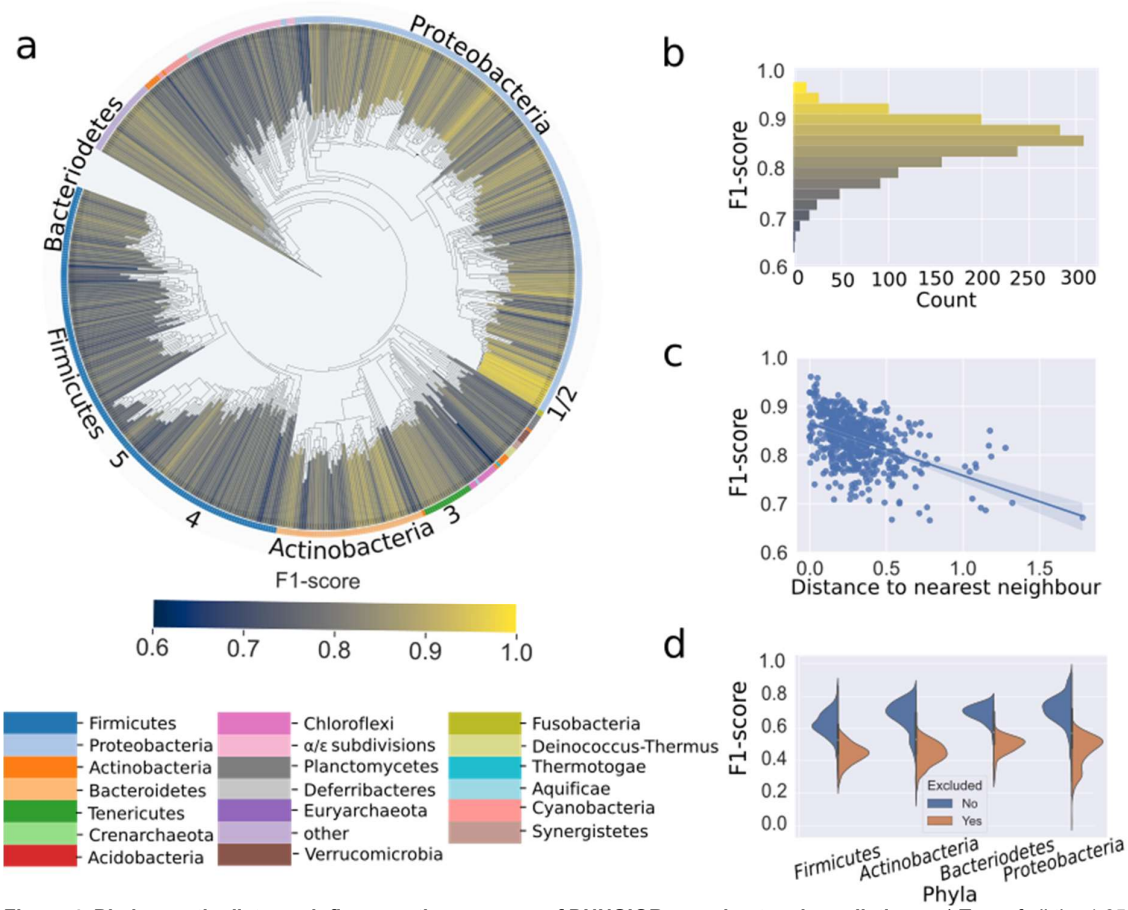

Figure 4: Phylogenetic distance influences the accuracy of DNNGIOR neural network predictions. a) Tree of all the 1,659 genera that are part of the testing dataset. Branch colour represents the F1-score, the colour of the outer ring corresponds to the phyla ordered by size, all phyla with less than 5 species are combined in other, the four largest phyla are annotated. The five species with high-quality curated metabolic models used during the gap-fill analysis are also marked: 1. *E. coli*, 2. *K. pneumoniae*, 3. *S. elongatus*, 4. *Bacillus subtilis*, and 5. *S. aureus*. The tree is based on a concatenated alignment of hits to HMM profiles of 71 single-copy marker genes. Phylogenetic distances in the tree represent the number of amino acid substitutions per site. b) Histogram of the F1-scores coloured with the same colormap as the Tree. c) Scatterplot of the F1-score versus the distance to the nearest neighbour expressed in the Jaccard distance, trendline (Pearson r^2 = -0.51, p = $1.53*10^{-40}$). d) Split violin plots of F1-scores of the predictions for neural networks when different phyla are included (left) or excluded (right) from the training data. Four different neural networks were trained excluding the four largest phyla and the F1-score calculated (in blue), in orange are the F1-scores for the phylum for a neural network trained on the full dataset.

Neural network weights improve gap-filling of draft models

Next, we determined the effectiveness of using NN-predicted weights to guide gap-filling, compared to other alternatives. For this, we gap-filled the models from the test dataset using the half interval gap-filling algorithm (see Methods), where weights may be assigned to individual reactions and the algorithm finds a metabolic network that is capable of generating a biomass flux, while minimising the weights that are added overall. We assessed whether the artificially removed reactions could be recovered when reactions were weighted based on four different weighting schemes: 1. No weights, 2. Naive binary weights, 3. Frequency weights, and 4. NN-weights (see Methods). To be clear, the process of gap-filling results in a functional model that can produce biomass, while the training set consists of draft models that generally cannot produce biomass and contain exclusively reactions derived from genome annotations, from which a fraction was deleted (see Figure 2). We found that the NN-weights significantly improved the accuracy of automated gap-filling compared to other weighting schemes, i.e. they allowed a larger fraction of the deleted reactions to be recovered in the model. By using the NN-weights, the F1-score increased 13.98 times compared to gap-filling with no weights, 1.97 times compared to the naive binary weights and 1.09 times compared to weights based on frequency alone (Figure 5A). When one considers the relative ranking of the F1-scores between the weighting schemes, the NN-weights always outranked using no weights or naive binary weights and outranked the frequency-based weights 80% of the time (Figure 5B).

As we have seen above, when assessing the accuracy of the predictions of the DNNGIOR neural network directly, the frequency of a reaction is also important for gap-filling. Although this may be expected, the reaction frequency has often been neglected when gap-filling strategies are developed. These strategies often focus on flux or network topology, giving the same cost to all reactions. This leads to addition of a minimum number of reactions that are necessary for growth, agnostic to all external information. Here, we found that simply weighing reactions by their frequencies in the bacterial domain already significantly improves the accuracy of gap-filling (Figure 5). The DNNGIOR neural network scores further improve gap-filling, suggesting that additional information has been learned from the co-occurrence patterns.

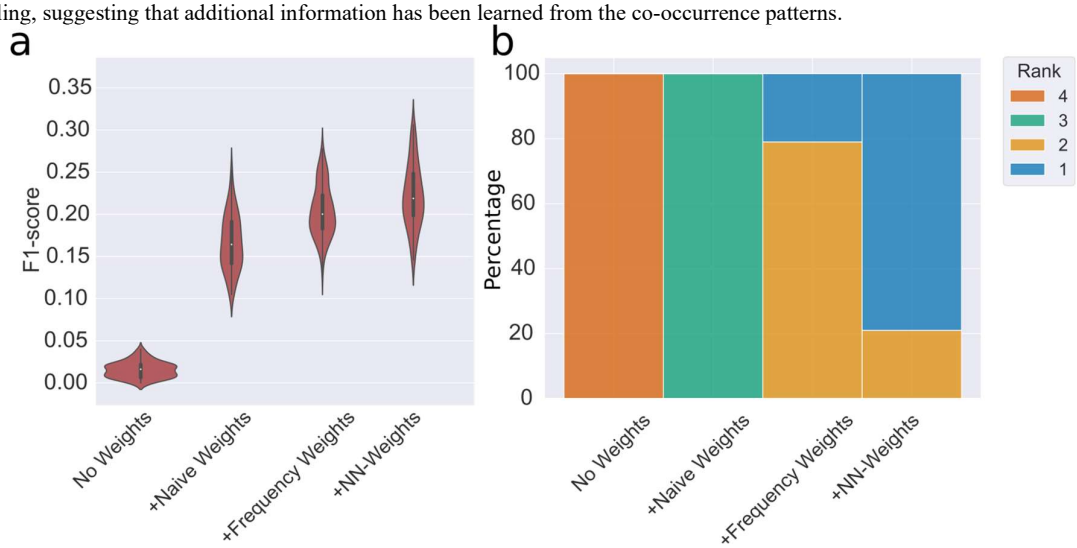

Figure 5: Weighted gap-filling of draft models. a) Violin plot of F1-scores of the gap-filling of draft models with 30% of reactions deleted. For naive binary weights all reactions that are present in the training set were given the same weights. For Freq-Based weights the frequency of the reaction was used according to Equation 4. For NN-weights the prediction scores generated by the DNNGIOR neural network were used. All groups were significantly different (Supplementary Table 8). b) Distribution of the ranks of the F1-scores for different weights with four being the worst ranking and one being the best.

When observing these results, we note that the F1-scores were lower than might be expected based on the prediction accuracy (Figure 3b). This consistent trend can be attributed to the fact that the draft models were already incomplete before the additional reactions were removed, because they were based on genome annotations alone. Thus, some reactions were necessarily added by the half interval gap-filling algorithm to enable biomass production that were counted as false positives here since they were not present in the genome annotation. Furthermore, the objective of the half interval gap-filling algorithm was not to find back reactions but rather to find a set of reactions that allows biomass production, while minimising their overall weights. As many of the annotated reactions that were removed were not strictly required for biomass production they were not added back, leading to false negatives. Although the F1-scores were thus systematically reduced, we can still interpret the trends in performance of the four different weighting schemes in Figure 5.

Overall, reactions that were assigned a high probability by the DNNGIOR neural network are likely to be present in the metabolic network, whereas low probability reactions are likely to be absent. To further incentivise the inclusion of high probability reactions while still satisfying the biomass production objective, we provided these reactions with negative, rather than zero costs (see Equation 2). These reactions were thus even more stimulated to be included, and indeed this nearly doubled (x1.95) the F1-score compared to using only positive weights (Figure S3a). However, this approach also led to an increase in false positives for reactions that were absent but were still assigned high probabilities by DNNGIOR (Figure S3b) (46).

Neural network-weights improve gap-filling of curated models

Next, we assessed the performance of the four weighting schemes using six high-quality manually curated models. As above, we artificially removed reactions from these models and tested how these reactions were weighted by the DNNGIOR neural network (see Methods). As shown in Figure 6, the NN-weights outperformed the other weights for all six models. *Saccharomyces cerivisiae* (iND750) performed the worst (mean = 0.11, sd = 0.05) of the six models since it's a eukaryote and only distantly related to the prokaryotic reference genomes that comprised the training dataset. The best performance (mean = 0.22, sd = 0.08) was found for *Escherichia coli* (iML1515), a species from a well-studied family with many reference genomes.

The fact that the NN-weights performed better than the frequency-based weights indicates that the neural network learned more than the reaction frequency alone. To gain an intuition into what additional information the network could have learned, we built Escher maps of the citric acid cycle of *E. coli* and coloured reactions by their mean recall over 500 iterations where we randomly deleted 30% of the reactions each time (Figure 7). The NN-weights produced the highest recall (Figure 7d), followed by the frequency weights (Figure 7c). We found that recall partially correlated with the reaction frequency in the training data for both the frequency weight (Pearson $r^2 = 0.5$, $p = 1.06*10^{-116}$) and NN-weights (Pearson $r^2 = 0.25$, $p = 1.22*10^{-26}$, Supplementary Figure S4b) but that the NN-weights also found some rare reactions that were more specific to *E. coli,* indicating that the network also learned co-occurrence patterns. The rest of the central metabolism of *E. coli* showed a similar pattern (Supplementary Figures S5 and S6).

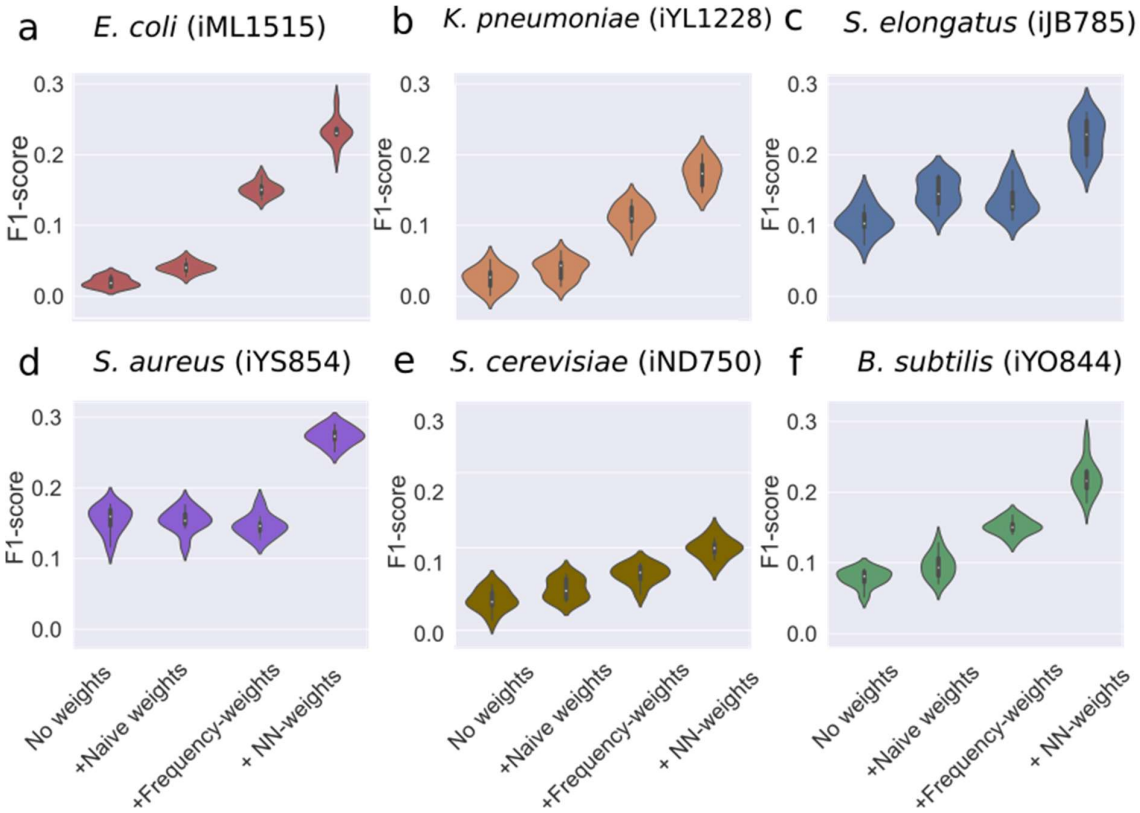

Figure 6: Weighted gap-filling of six curated metabolic models. Violin plots of F1-scores of the gap-filling of curated models, where we randomly deleted 30% of reactions 500 times. These reduced models were gap-filled using four different weighting schemes. In the No weights scheme, all reactions were given a weight of 1. For the naive binary weights, all reactions that are present in any of the genomes of the training set were given a weight of 1, and all other reactions a weight of 0. For the Frequency-based weights the frequency of the reaction in the training dataset was used as weight, ranging from 0 (reaction absent) to 1 (reaction present in all genomes). For NN-weights, the prediction scores generated by the DNNGIOR neural network were used. a) *E. coli* iML1515, b) *K. pneumoniae* iYL1228, c) *S. elongatus* iJB785, d) *S. aureus* iYS854, e) *S. cerevisiae* iND750, and f) *B. subtilis* iYO844.

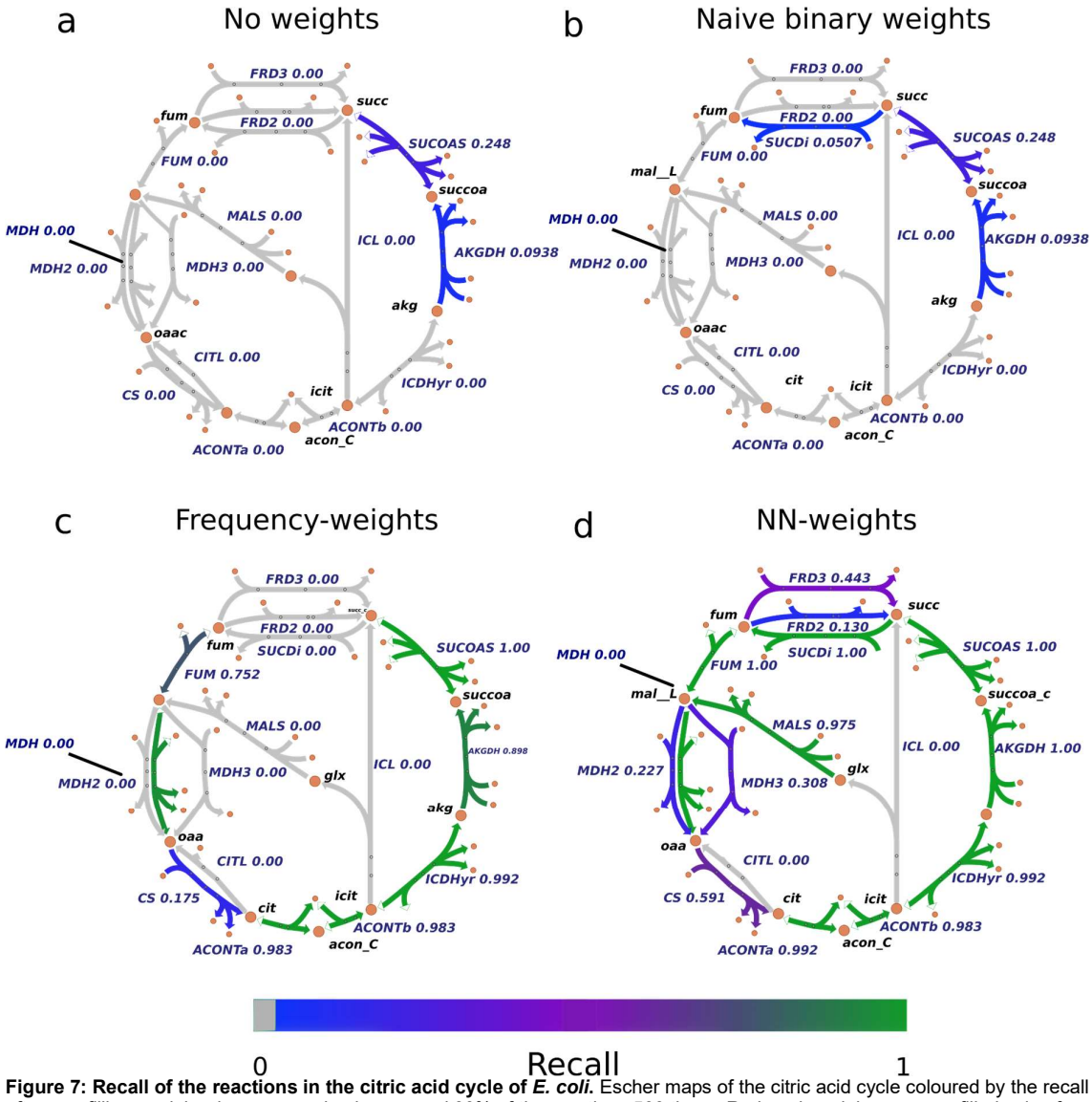

Figure 7: Recall of the reactions in the citric acid cycle of *E. coli*. Escher maps of the citric acid cycle coloured by the recall after gap-filling models where we randomly removed 30% of the reactions 500 times. Reduced models were gap-filled using four different weighting schemes: a) no weights b) naive binary weights c) frequency weights and d) NN-weights (see Figure 5 caption and Methods for details). IDs for secondary metabolites were omitted, the full metabolic network maps can be found in (Supplementary Figures S5 and S6).

Future potential of the neural network

The DNNGIOR neural network-based reaction weights derived herein already significantly improved the gap-filling of GSMMs for a wide variety of bacteria. This success derives from the network learning aspects including the frequency and co-occurrence relationships between reactions. Additional features outside the scope of the current study, such as reaction fluxes, pathway annotation or environmental factors that were previously shown to be useful for gap-filling (23–25,43–45) may also be included, which could improve the prediction accuracy of the weights and subsequent gap-filling even further. A further promising avenue is using the network structure of the input data, including phylogenetic and metabolic networks. Graph convolution takes networks as input and may be a powerful tool to address such complex problems (28,46). A promising approach using hypergraph link prediction based on the stoichiometric matrix has recently been suggested as an alternative method for gap-filling metabolic networks, showing the potential of including the graph topology into the neural network (28). Combining this with the broader taxonomic signal found in our more diverse dataset could, in the future, lead to improved performance over a broad range of organisms.

Extensions to increase applicability

Currently, efforts are being made to reconcile the different reaction databases into MetaNetX (50). Once finished, a new neural network could be created that would use the reconciled database and that would be universally applicable. In this paper we focused on ModelSEED models, but we have also created a version trained on CarveMe models showing similar results (Supplementary Figure S8). This shows that it is possible to create neural networks for weighing the reactions in models from different sources. The neural network gives a prediction score for all reactions, not just for the missing ones. Currently, most of these predictions are ignored as we only aimed to fill in the missing reactions and retain the ones that were based on genome annotation, i.e. that have genetic evidence. However, mistakes during metagenome binning (13,14) and annotation do not only result in missing reactions (incomplete MAGs) but also reactions that are falsely attributed to a genome (contaminated MAGs) and thus spurious reactions in the corresponding metabolic models (51). Combining evidence about the predicted taxonomic affiliation of metagenomic contigs (52) with the weights predicted by the DNNGIOR neural network could potentially help identify such erroneous reactions which could then be removed during model curation.

Final remarks

We developed DNNGIOR, a neural network that predicts which metabolic reactions are present in a given bacterial strain, based on incomplete information. As the neural network learns about these reactions from known bacteria, it is particularly effective for scoring reactions that are relatively common, and for organisms that are relatively closely related to those present in the training dataset. Advanced users can tailor the training dataset for their specific needs, e.g. training models for certain biomes or taxa. The predicted weights can be utilised during gap-filling to improve the accuracy and overall quality of the reconstructed metabolic models. Increasingly, genome-scale metabolic models are being used to interpret microbial metabolic traits, growth, or environmental associations. DNNGIOR should be a valuable tool to enhance the potential of these models.

Acknowledgements & funding

The authors would like to extend their gratitude to the TBB group for moral support and to J.K. van Amerongen for technical support. Funding support for this work came from the UU-NIOZ project “Turning the tide” (NZ4543.26) awarded to B.E. Dutilh and A.F. Haas, the European Research Council (ERC) Consolidator grant 865694: DiversiPHI, the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under Germany’s Excellence Strategy – EXC 2051 – Project-ID 390713860, and the Alexander von Humboldt Foundation in the context of an Alexander von Humboldt-Professorship founded by German Federal Ministry of Education and Research.

Author Contributions

Writing – Original Draft, M.D.B.; Writing Review & Editing,C.M., A.F.H., D.G., B.E.D.;Visualisation, M.D.B;Software, M.D.B., H.Z., D.G.; Funding Acquisition, A.F.H. and B.E.D.; Supervision, C.M., A.F.H., D.G., B.E.D.

References

(1) Melkonian C, Zorrilla F, Kjærbølling I, Blasche S, Machado D, Junge M, et al. Inter-species metabolic interactions in cheese flavour formatio 2023. https://doi.org/10.21203/rs.3.rs-2574132/v1.

(2) Kodera SM, Das P, Gilbert JA, Lutz HL. Conceptual strategies for characterizing interactions in microbial communities. IScience 2022;25:103775. https://doi.org/10.1016/j.isci.2022.103775.

(3) García-Jiménez B, Torres-Bacete J, Nogales J. Metabolic modelling approaches for describing and engineering microbial communities. Comput Struct Biotechnol J 2020;19:226–46. https://doi.org/10.1016/j.csbj.2020.12.003.

(4) Gu C, Kim GB, Kim WJ, Kim HU, Lee SY. Current status and applications of genome-scale metabolic models. Genome Biology 2019;20:121. https://doi.org/10.1186/s13059-019-1730-3.

(5) Fang X, Lloyd CJ, Palsson BO. Reconstructing organisms in silico: genome-scale models and their emerging applications. Nat Rev Microbiol 2020;18:731–43. https://doi.org/10.1038/s41579-020-00440-4.

(6) Passi A, Tibocha-Bonilla JD, Kumar M, Tec-Campos D, Zengler K, Zuniga C. Genome-Scale Metabolic Modeling Enables In-Depth Understanding of Big Data. Metabolites 2021;12:14. https://doi.org/10.3390/metabo12010014.

(7) Wang H, Marcišauskas S, Sánchez BJ, Domenzain I, Hermansson D, Agren R, et al. RAVEN 2.0: A versatile toolbox for metabolic network reconstruction and a case study on Streptomyces coelicolor. PLOS Computational Biology 2018;14:e1006541. https://doi.org/10.1371/journal.pcbi.1006541.

(8) Henry CS, DeJongh M, Best AA, Frybarger PM, Linsay B, Stevens RL. High-throughput generation, optimization and analysis of genome-scale metabolic models. Nat Biotechnol 2010;28:977–82. https://doi.org/10.1038/nbt.1672.

(9) Arkin AP, Cottingham RW, Henry CS, Harris NL, Stevens RL, Maslov S, et al. KBase: The United States Department of Energy Systems Biology Knowledgebase. Nat Biotechnol 2018;36:566–9. https://doi.org/10.1038/nbt.4163.

(10) Machado D, Andrejev S, Tramontano M, Patil KR. Fast automated reconstruction of genome-scale metabolic models for microbial species and communities. Nucleic Acids Research 2018;46:7542–53. https://doi.org/10.1093/nar/gky537.

(11) Bowers RM, Kyrpides NC, Stepanauskas R, Harmon-Smith M, Doud D, Reddy TBK, et al. Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. Nat Biotechnol 2017;35:725–31. https://doi.org/10.1038/nbt.3893.

(12) Meziti A, Rodriguez-R LM, Hatt JK, Peña-Gonzalez A, Levy K, Konstantinidis KT. The Reliability of Metagenome-Assembled Genomes (MAGs) in Representing Natural Populations: Insights from Comparing MAGs against Isolate Genomes Derived from the Same Fecal Sample. Applied and Environmental Microbiology 2021;87:e02593-20. https://doi.org/10.1128/AEM.02593-20.

(13) Nelson WC, Tully BJ, Mobberley JM. Biases in genome reconstruction from metagenomic data. PeerJ 2020;8:e10119. https://doi.org/10.7717/peerj.10119.

(14) Maguire F, Jia B, Gray KL, Lau WYV, Beiko RG, Brinkman FSL. Metagenome-assembled genome binning methods with short reads disproportionately fail for plasmids and genomic Islands. Microbial Genomics 2020;6:e000436. https://doi.org/10.1099/mgen.0.000436.

(15) Lobb B, Tremblay BJ-M, Moreno-Hagelsieb G, Doxey AC. An assessment of genome annotation coverage across the bacterial tree of life. Microb Genom 2020;6:e000341. https://doi.org/10.1099/mgen.0.000341.

(16) Monk JM, Lloyd CJ, Brunk E, Mih N, Sastry A, King Z, et al. iML1515, a knowledgebase that computes Escherichia coli traits. Nat Biotechnol 2017;35:904–8. https://doi.org/10.1038/nbt.3956.

(17) Aminian-Dehkordi J, Mousavi SM, Jafari A, Mijakovic I, Marashi S-A. Manually curated genome-scale reconstruction of the metabolic network of Bacillus megaterium DSM319. Sci Rep 2019;9:18762. https://doi.org/10.1038/s41598-019-55041-w.

(18) Thiele I, Palsson BØ. A protocol for generating a high-quality genome-scale metabolic reconstruction. Nat Protoc 2010;5:93–121. https://doi.org/10.1038/nprot.2009.203.

(19) Magnúsdóttir S, Heinken A, Kutt L, Ravcheev DA, Bauer E, Noronha A, et al. Generation of genome-scale metabolic reconstructions for 773 members of the human gut microbiota. Nat Biotechnol 2017;35:81–9. https://doi.org/10.1038/nbt.3703.

(20) Garza DR, van Verk MC, Huynen MA, Dutilh BE. Towards predicting the environmental metabolome from metagenomics with a mechanistic model. Nat Microbiol 2018;3:456–60. https://doi.org/10.1038/s41564-018-0124-8.

(21) Zorrilla F, Buric F, Patil KR, Zelezniak A. metaGEM: reconstruction of genome scale metabolic models directly from metagenomes. Nucleic Acids Research 2021;49:e126. https://doi.org/10.1093/nar/gkab815.

(22) Latendresse M. Efficiently gap-filling reaction networks. BMC Bioinformatics 2014;15:225. https://doi.org/10.1186/1471-2105-15-225.

(23) Hartleb D, Jarre F, Lercher MJ. Improved Metabolic Models for E. coli and Mycoplasma genitalium from GlobalFit, an Algorithm That Simultaneously Matches Growth and Non-Growth Data Sets. PLOS Computational Biology 2016;12:e1005036. https://doi.org/10.1371/journal.pcbi.1005036.

(24) Chen C, Liao C, Liu Y-Y. Teasing out Missing Reactions in Genome-scale Metabolic Networks through Deep Learning 2022:2022.06.27.497720. https://doi.org/10.1101/2022.06.27.497720.

(25) Schroeder WL, Saha R. OptFill: A Tool for Infeasible Cycle-Free Gapfilling of Stoichiometric Metabolic Models. IScience 2019;23:100783. https://doi.org/10.1016/j.isci.2019.100783.

(26) Prigent S, Frioux C, Dittami SM, Thiele S, Larhlimi A, Collet G, et al. Meneco, a Topology-Based Gap-Filling Tool Applicable to Degraded Genome-Wide Metabolic Networks. PLOS Computational Biology 2017;13:e1005276. https://doi.org/10.1371/journal.pcbi.1005276.

(27) Ong WK, Midford PE, Karp PD. Taxonomic weighting improves the accuracy of a gap-filling algorithm for metabolic models. Bioinformatics 2020;36:1823–30. https://doi.org/10.1093/bioinformatics/btz813.

(28) King B, Farrah T, Richards MA, Mundy M, Simeonidis E, Price ND. ProbAnnoWeb and ProbAnnoPy: probabilistic annotation and gap-filling of metabolic reconstructions. Bioinformatics 2018;34:1594–6. https://doi.org/10.1093/bioinformatics/btx796.

(29) Krumholz EW, Libourel IGL. Thermodynamic Constraints Improve Metabolic Networks. Biophysical Journal 2017;113:679–89. https://doi.org/10.1016/j.bpj.2017.06.018.

(30) Wattam AR, Davis JJ, Assaf R, Boisvert S, Brettin T, Bun C, et al. Improvements to PATRIC, the all-bacterial Bioinformatics Database and Analysis Resource Center. Nucleic Acids Res 2017;45:D535–42. https://doi.org/10.1093/nar/gkw1017.

(31) Abadi M, Agarwal A, Barham P, Brevdo E, Chen Z, Citro C, et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems 2015:19.

(32) Kingma DP, Ba J. Adam: A Method for Stochastic Optimization. ArXiv:14126980 [Cs] 2017.

(33) Akiba T, Sano S, Yanase T, Ohta T, Koyama M. Optuna: A Next-generation Hyperparameter Optimization Framework. Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, Anchorage, AK, USA: Association for Computing Machinery; 2019, p. 2623–31. https://doi.org/10.1145/3292500.3330701.

(34) Godoy D. Understanding binary cross-entropy / log loss: a visual explanation. Medium 2019. https://towardsdatascience.com/understanding-binary-cross-entropy-log-loss-a-visual-explanation-a3ac6025181a (accessed March 31, 2020).

(35) Broddrick JT, Rubin BE, Welkie DG, Du N, Mih N, Diamond S, et al. Unique attributes of cyanobacterial metabolism revealed by improved genome-scale metabolic modeling and essential gene analysis. Proc Natl Acad Sci U S A 2016;113:E8344–53. https://doi.org/10.1073/pnas.1613446113.

(36) Liao Y-C, Huang T-W, Chen F-C, Charusanti P, Hong JSJ, Chang H-Y, et al. An experimentally validated genome-scale metabolic reconstruction of Klebsiella pneumoniae MGH 78578, iYL1228. J Bacteriol 2011;193:1710–7. https://doi.org/10.1128/JB.01218-10.

(37) Bro C, Regenberg B, Förster J, Nielsen J. In silico aided metabolic engineering of Saccharomyces cerevisiae for improved bioethanol production. Metabolic Engineering 2006;8:102–11. https://doi.org/10.1016/j.ymben.2005.09.007.

(38) Vikromvarasiri N, Shirai T, Kondo A. Metabolic engineering design to enhance (R,R)-2,3-butanediol production from glycerol in Bacillus subtilis based on flux balance analysis. Microb Cell Fact 2021;20:196. https://doi.org/10.1186/s12934-021-01688-y.

(39) Seif Y, Monk JM, Mih N, Tsunemoto H, Poudel S, Zuniga C, et al. A computational knowledge-base elucidates the response of Staphylococcus aureus to different media types. PLoS Comput Biol 2019;15:e1006644. https://doi.org/10.1371/journal.pcbi.1006644.

(40) King ZA, Lu J, Dräger A, Miller P, Federowicz S, Lerman JA, et al. BiGG Models: A platform for integrating, standardizing and sharing genome-scale models. Nucleic Acids Res 2016;44:D515–22. https://doi.org/10.1093/nar/gkv1049.

(41) King ZA, Dräger A, Ebrahim A, Sonnenschein N, Lewis NE, Palsson BO. Escher: A Web Application for Building, Sharing, and Embedding Data-Rich Visualizations of Biological Pathways. PLOS Computational Biology 2015;11:e1004321. https://doi.org/10.1371/journal.pcbi.1004321.

(42) How do I cite BioRender? | BioRender Help Center n.d. https://help.biorender.com/en/articles/3619405-how-do-i-cite-biorender (accessed July 1, 2023).

(43) Tettelin H, Masignani V, Cieslewicz MJ, Donati C, Medini D, Ward NL, et al. Genome analysis of multiple pathogenic isolates of Streptococcus agalactiae: Implications for the microbial "pan-genome." Proc Natl Acad Sci U S A 2005;102:13950–5. https://doi.org/10.1073/pnas.0506758102.

(44) Ejigu GF, Jung J. Review on the Computational Genome Annotation of Sequences Obtained by Next-Generation Sequencing. Biology 2020;9:295. https://doi.org/10.3390/biology9090295.

(45) von Meijenfeldt FAB, Hogeweg P, Dutilh BE. A social niche breadth score reveals niche range strategies of generalists and specialists. Nat Ecol Evol 2023;7:768–81. https://doi.org/10.1038/s41559-023-02027-7.

(46) Biggs MB, Papin JA. Managing uncertainty in metabolic network structure and improving predictions using EnsembleFBA. PLOS Computational Biology 2017;13:e1005413. https://doi.org/10.1371/journal.pcbi.1005413.

(47) Borenstein E, Kupiec M, Feldman MW, Ruppin E. Large-scale reconstruction and phylogenetic analysis of metabolic environments. Proceedings of the National Academy of Sciences 2008;105:14482–7. https://doi.org/10.1073/pnas.0806162105.

(48) Giannari D, Ho CH, Mahadevan R. A gap-filling algorithm for prediction of metabolic interactions in microbial communities. PLOS Computational Biology 2021;17:e1009060. https://doi.org/10.1371/journal.pcbi.1009060.

(49) Vitkin E, Shlomi T. MIRAGE: a functional genomics-based approach for metabolic network model reconstruction and its application to cyanobacteria networks. Genome Biol 2012;13:R111. https://doi.org/10.1186/gb-2012-13-11-r111.

(50) Moretti S, Tran VDT, Mehl F, Ibberson M, Pagni M. MetaNetX/MNXref: unified namespace for metabolites and biochemical reactions in the context of metabolic models. Nucleic Acids Research 2021;49:D570–4. https://doi.org/10.1093/nar/gkaa992.

(51) Bernstein DB, Sulheim S, Almaas E, Segrè D. Addressing uncertainty in genome-scale metabolic model reconstruction and analysis. Genome Biol 2021;22:64. https://doi.org/10.1186/s13059-021-02289-z.

(52) von Meijenfeldt FAB, Arkhipova K, Cambuy DD, Coutinho FH, Dutilh BE. Robust taxonomic classification of uncharted microbial sequences and bins with CAT and BAT. Genome Biology 2019;20:217. https://doi.org/10.1186/s13059-019-1817-x.