Canopy: A Heterograph Foundation Model for Metabolic Engineering

Anonymous Authors[1]

Abstract

Designing microbial strains that produce high-value chemicals at commercially viable titers remains a central challenge in metabolic engineering. Existing computational approaches either rely on stoichiometric constraint-based models that cannot learn from experimental data, or apply tabular machine learning to hand-crafted features that discard the relational structure of biological knowledge. We present Canopy, a heterogeneous graph foundation model that integrates ten public and proprietary data sources into a unified knowledge graph (KG) of 6.9 M nodes across 13 types and 34 edge types, covering genes, proteins, metabolites, reactions, pathways, strains, and fermentation experiments. Node features are encoded through domain-specific foundation models (ESM-2 for protein sequences, MoLFormer for chemical SMILES, and PubMedBERT for biomedical text), yielding a multi-modal representation within a single graph. We pretrain a Heterogeneous Graph Transformer (HGT) augmented with SignNet positional encodings, Jumping Knowledge aggregation, and virtual nodes using four self-supervised objectives (link prediction, masked node modelling, distance prediction, and contrastive experiment clustering), balanced via learned homoscedastic uncertainty weighting. On the downstream task of fermentation titer prediction, frozen Canopy embeddings achieve $R^2 = 0.41$ with a lightweight probe, outperforming tabular baselines (best $R^2 = 0.13$) and homogeneous GNN variants.

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country. Correspondence to: Anonymous Author <anon.email@domain.com>.

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

1. Introduction

The bioeconomy depends on engineered microorganisms that convert renewable feedstocks into fuels, pharmaceuticals, and specialty chemicals. The design–build–test–learn (DBTL) cycle that underpins strain engineering is slow: a single round of genetic modification, fermentation, and analytical characterisation can take weeks to months (Opgenorth et al., 2019), and the combinatorial space of candidate modifications grows exponentially in the number of target genes. Computational tools that prioritise genetic interventions before wet-lab experiments would shorten this loop.

The dominant computational paradigm for strain design is constraint-based modelling via genome-scale metabolic models (GEMs). Flux balance analysis (FBA) and its extensions predict steady-state fluxes under stoichiometric and thermodynamic constraints, enabling in silico knockout analysis through tools such as OptKnock (Burgard et al., 2003) and StrainDesign (Schneider et al., 2022). While GEMs encode mechanistic knowledge of metabolism, they cannot incorporate experimental measurements of titer, rate, and yield; they ignore regulatory, expression-level, and environmental effects; and they treat each organism in isolation without using cross-organism transfer.

Machine learning offers a complementary approach. Previous work has applied random forests, gradient boosting, and neural networks to predict fermentation titer from features extracted from strain descriptions (Oyetunde et al., 2019; Czajka et al., 2021). These tabular approaches discard the relational structure that links genes to proteins, proteins to reactions, reactions to pathways, and pathways to production phenotypes. Graph neural networks have been applied to metabolic networks for gene essentiality (Hasibi et al., 2024) and site-of-metabolism prediction (Porokhin et al., 2023), but these operate on single-organism reaction graphs rather than on cross-organism knowledge graphs.

Meanwhile, the broader ML community has advanced heterogeneous graph transformers (Hu et al., 2020b), self-supervised graph pretraining (Hu et al., 2020a), and domain-specific foundation models for proteins (Lin et al., 2023) and small molecules (Ross et al.,

2021). Biomedical knowledge graphs such as PrimeKG (Chandak et al., 2023) have enabled GNN-based drug repurposing, but no analogous resource or foundation model exists for the metabolic engineering domain, which sits at the intersection of microbial genetics, enzyme biochemistry, and fermentation science.

We introduce Canopy, a heterogeneous graph foundation model for metabolic engineering. Our contributions are:

- A metabolic-engineering knowledge graph integrating ten data sources (MetaNetX, GO, UniRef, InterPro, NCBI Taxonomy and Genomics, KEGG, an in-house laboratory information management system (LIMS) of unpublished DBTL records, literature-mined experiments, and transcriptomics) via BioCypher (Lobentanzer et al., 2023) into a unified heterogeneous graph with 13 node types and 34 edge types.
- Multi-modal feature encoding using a schema-driven dispatch system that routes protein sequences to ESM-2 (650M), SMILES strings to MoLFormer-XL, free text to PubMedBERT, and numeric features to normalised scalars within a single graph.
- An augmented Heterogeneous Graph Transformer combining HGTConv with SignNet positional encodings, random walk structural encodings, per-type feed-forward networks (FFNs) with learnable residual scaling, Jumping Knowledge aggregation, and virtual nodes.
- Four-objective self-supervised pretraining with learned uncertainty weighting: link prediction, masked node modelling, distance prediction, and a contrastive experiment-pair loss.
- Downstream evaluation on fermentation titer prediction, showing that learned graph representations outperform tabular baselines and provide the conditioning substrate for downstream generative strain-design pipelines.

2. Related Work

Foundation models across biological scales. Domain-specific foundation models now span genomes (Nguyen et al., 2024; Brixi et al., 2025), protein sequence (Lin et al., 2023; Hayes et al., 2025) and structure (Su et al., 2024; Abramson et al., 2024), molecules (Ross et al., 2021), and single-cell transcriptomics (Cui et al., 2024; Hao et al., 2024). Each captures a single modality at a single scale; none are jointly trained over the relational structure that links genes to enzymes to metabolites to fermentation outcomes. Canopy embeds frozen ESM-2, MoLFormer, and PubMedBERT features inside a heterogeneous KG, treating these models as feature encoders within a graph that crosses scales.

Predictive and generative models for strain, pathway, and enzyme design. Kinetic GEMs such as k-ecoli457 (Khodayari & Maranas, 2016) extend FBA with parameterised rate equations fit to fluxomic data and predict product yields more accurately than purely stoichiometric models, but their parameterisation is organism-specific and does not transfer across strains. Tabular ML on hand-crafted features remains dominant for titer prediction (Oyetunde et al., 2019; Czajka et al., 2021; Radivojević et al., 2020); graph methods so far have been single-organism (Hasibi et al., 2024; Xin et al., 2024) or relied on shallow KG embeddings (Song et al., 2026; Gema et al., 2024). Diffusion- and flow-matching-based generative models are seeing rapid uptake for protein backbones and enzyme active sites (Li et al., 2026; Ahern et al., 2026), but target single molecules rather than whole-strain phenotypes. Canopy predicts titer from a learned cross-organism representation that fuses sequence, chemistry, and KG context. The same frozen embeddings can condition a downstream Bayesian-optimisation loop (Cheng et al., 2023) or generative model.

Benchmarks for ML in metabolic engineering. A useful benchmark for ML-driven metabolic engineering needs three things: experimental titer measurements, cross-organism coverage, and multi-omic context for each strain. No existing resource provides all three. SimDBTL (van Lent et al., 2023) offers consistent DBTL splits but its data are simulated rather than experimental. Therapeutics Data Commons (Huang et al., 2021) standardises prediction splits for therapeutic rather than fermentation tasks. Text-mining catalogues of over 15,000 strain-design publications (Márquez-Zavala et al., 2025) surface raw records without a held-out evaluation. Canopy's 4,791 fermentation experiments with a deterministic MD5-hashed 5x CV split address all three criteria.

Biological knowledge graphs and heterogeneous GNNs. Large biomedical KGs (PrimeKG, Chandak et al., 2023; Hetionet, Himmelstein et al., 2017; SPOKE, Nelson et al., 2019) and graph foundation models built on them (Huang et al., 2024; Hu et al., 2026) target human disease, not metabolic engineering. We use BioCypher (Lobentanzer et al., 2023) to integrate ten metabolic-engineering adapters and build on the Heterogeneous Graph Transformer (Hu et al., 2020b) with SignNet (Lim et al., 2022), random-walk PE (Dwivedi

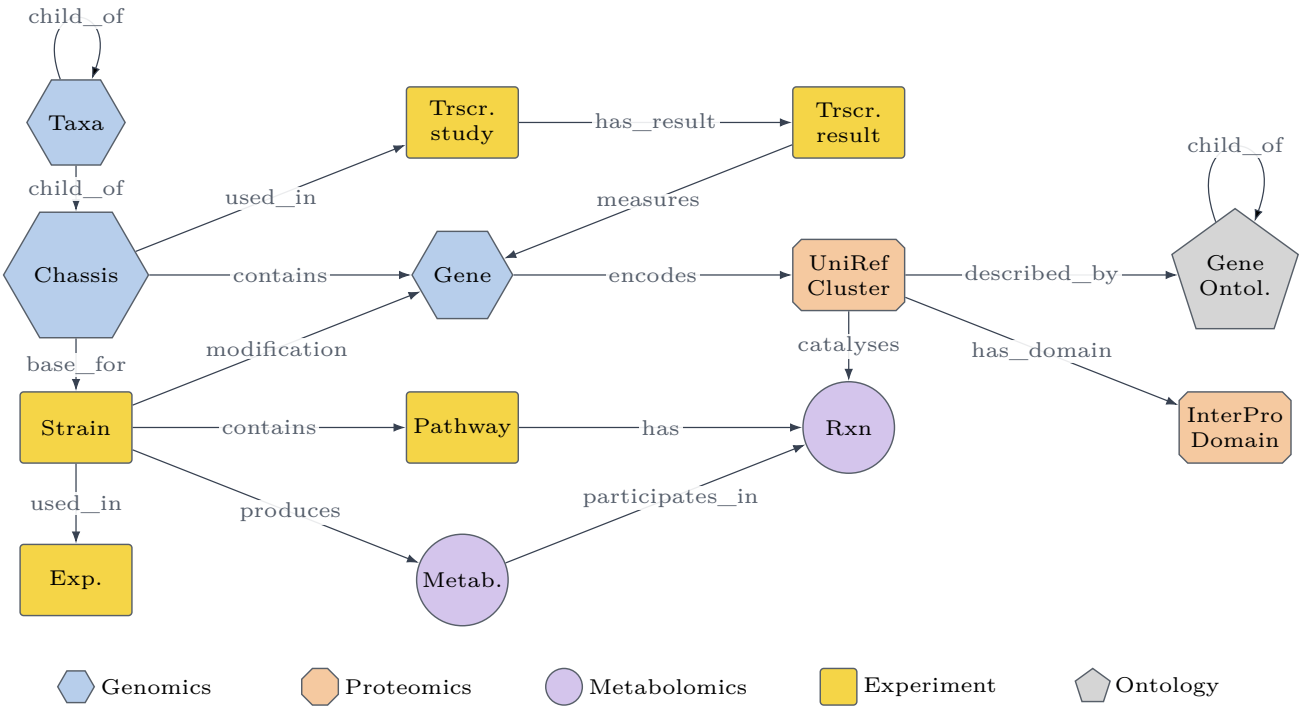

Figure 1. Canopy heterograph schema. Each node type has a distinct shape and is coloured by data domain (genomics, proteomics, metabolomics, experiment, ontology); each edge is typed by relation. Strain-to-gene modification edges include knockouts, knockins, and knockdowns.

et al., 2022), Jumping Knowledge (Xu et al., 2018), and virtual nodes (Gilmer et al., 2017); pretraining objectives draw on Hu et al. (2020a) and the graph-FM surveys of Wang et al. (2025); Mao et al. (2024). The closest cross-organism transfer is IKT4Meta (Xin et al., 2024); Canopy extends this from two-organism alignment to a 13-node-type, multi-modal KG with fermentation-scale supervision.

3. Method

3.1. Knowledge Graph Construction

Canopy's knowledge graph is constructed using BioCypher (Lobentanzer et al., 2023). We implement ten adapter modules that ingest data from complementary sources and yield nodes and edges in a unified schema.

Node types. The graph contains 13 node types spanning five biological scales: (i) molecular metabolites (InChIKey, SMILES, formula) and reactions; (ii) protein UniRef90 clusters together with their InterPro domains; (iii) genomic genes, chassis organisms, and engineered strains; (iv) functional annotations from the Gene Ontology, metabolic pathways, and NCBI taxonomy; and (v) experimental fermentation runs alongside transcriptomic experiments with per-gene expression measurements.

Edge types. Thirty-four edge types capture relationships across these scales: catalytic associations linking proteins to the reactions they catalyse, pathway membership, genetic modifications (knockouts, knockins, and overexpression edits applied to a strain), functional annotations, ontological structure, experiment tracking, and gene-expression links between transcriptomic measurements and the genes they quantify.

Property schema. Each node property carries a mandatory prefix that declares its role: sys_ for system keys used in deduplication (not exported as features), feat_ for numeric values, text_ for free text routed to the text encoder, seq_ for amino acid sequences routed to the protein encoder, and smi_ for SMILES routed to the chemical encoder. This dispatch ensures graph construction, embedding, and training share a single source of truth.

Data sources. Molecular data is drawn from MetaNetX (Moretti et al., 2021) and KEGG (Kanehisa & Goto, 2000). Protein data comes from UniRef90 (Suzek et al., 2015) with sequences from UniProt (The UniProt Consortium, 2023) and domain annotations from InterPro (Paysan-Lafosse et al., 2023). Genomic

Table 1. Knowledge graph statistics by node type. The graph contains 11.2 M relationships and 17.7 M node properties in total.

Domain	Node type	Count
Genomics	Taxon	143
	Chassis	26
	Genomic Gene	168,913
Proteomics	UniRef Cluster	150,914
	InterPro Domain	51,489
Metabolomics	Metabolite	1,495,667
	Reaction	83,796
Experiment	Strain	6,910
	Pathway	1,872
	Experiment	4,791
	Transcriptomic	4,860,266
Ontology	GO Term (BP/MF/CC)	38,739
	Total	6,863,526

data includes NCBI gene records (Brown et al., 2015) and taxonomy (Schoch et al., 2020). Functional annotations come from the Gene Ontology (Ashburner et al., 2000). Experimental data is sourced from a quality-controlled literature corpus and a proprietary experimental database; transcriptomic data provides per-gene expression linked to experimental conditions.

3.2. Multi-Modal Feature Encoding

Canopy employs three pretrained foundation models as frozen extractors. Protein sequences (seq_) are encoded by ESM-2 (Lin et al., 2023) (esm2_t33_650M_UR50D) and mean-pooled across residues. Chemical structures (smi_) are encoded by MoLFormer-XL (Ross et al., 2021). Biomedical text (text_) is encoded by S-PubMedBERT, applied to GO definitions, gene names and summaries, pathway names, and experiment descriptions. Numeric features (feat_) are z-score normalised using per-type statistics; categorical integers are one-hot encoded. All features for a node are concatenated into $\mathbf{x}_v$; per-type input projection handles dimensional heterogeneity.

3.3. Model Architecture

Canopy's encoder is a Heterogeneous Graph Transformer that processes typed nodes and edges natively, extending HGTConv (Hu et al., 2020b) with several modern components. Figures 2–4 give a schematic overview of the full architecture, including the per-type and per-relation parameterisation that distinguishes HGT from a homogeneous GNN.

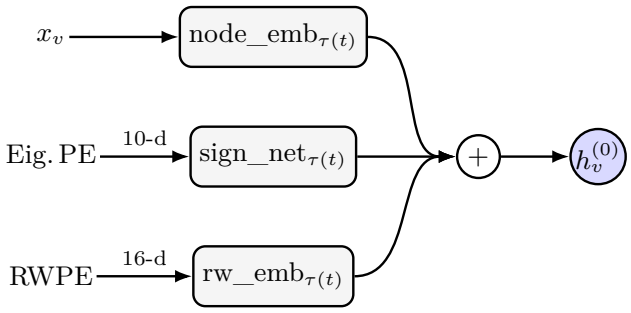

Figure 2. Input feature construction. Each node type $\tau(v)$ has its own node_emb, sign_net (consuming a 10-d Laplacian eigenvector positional encoding), and rw_emb (consuming a 16-d random-walk positional encoding); the three streams are summed to form the per-node residual stream $h_v^{(0)}$.

Input projection. For each node type t, a learnable linear projection $\mathbf{W}_t$ maps $\mathbf{x}_v$ to a hidden representation of dimension $d = d_h \cdot H$, with d_h the per-head dimension and H the number of heads.

Positional encodings. We augment node features with two complementary signals. The k smallest nontrivial Laplacian eigenvectors are computed via randomised SVD and processed through SignNet (Lim et al., 2022):

$$\mathrm{SignNet}(\mathbf{e}) = \mathrm{MLP}(\mathbf{e}) + \mathrm{MLP}(-\mathbf{e}). \tag{1}$$

Random walk structural encodings of length ℓ are computed per subgraph sample and projected per-type. Both encodings are added element-wise to the projected features.

Transformer blocks. The model stacks L blocks: per-type pre-norm; HGTConv with type-dependent multi-head attention; GELU; a residual connection scaled by a learnable ScaleLayer initialised at 0.1; a per-type FFN with expansion r and dropout; and a second scaled residual. The small initial scale stabilises training of deep networks by letting the residual stream dominate early. On top of the stack, we apply Jumping Knowledge (Xu et al., 2018) with max-pooling across all L layers per type. A virtual node type is added with bidirectional edges to all other nodes, reducing effective graph diameter and providing a shortcut path between otherwise disconnected components. The final representations are layer-normalised.

3.4. Multi-Task Self-Supervised Pretraining

We pretrain Canopy with four complementary objectives.

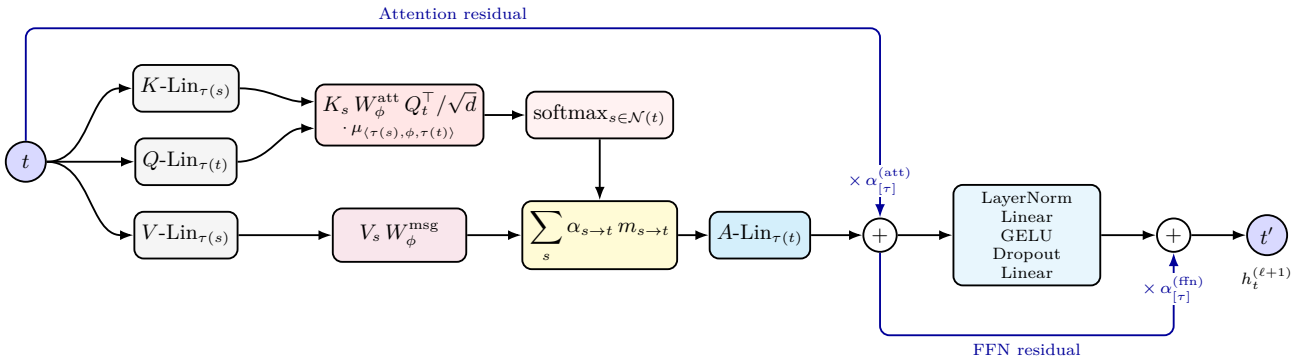

Figure 3. One HGT layer. The dst-side query $Q_{\tau(t)}$ is type-specific; keys $K_{\tau(s)}$ and values $V_{\tau(s)}$ are projected per source type, while attention logits and messages are scaled by per-relation matrices $W_\phi^{\text{att}}, W_\phi^{\text{msg}}$ and a learnable relation prior $\mu_{\langle\tau(s),\phi,\tau(t)\rangle}$. Attention is softmaxed over the destination's neighbourhood and aggregated; a per-type output projection A-Lin$_{\tau(t)}$ feeds an FFN sub-block. Both attention and FFN use learnable scaled residual connections (blue).

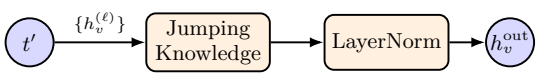

Figure 4. Readout. After L stacked HGT layers, Jumping-Knowledge max-pools the per-layer representations $\{h_v^{(\ell)}\}$ (combined with a learnable input-residual scale $\alpha_{[\tau]}^{(\text{in})}$) and a final per-type LayerNorm produces the encoder output h_v^{out}.

Hold-out integrity. The same hash-keyed Experiment-node split used at probe time (Section 4.1) gates every pretraining objective that touches an Experiment node: edges incident to any val/test Experiment are excluded from the link-prediction label set, masked-node-modelling ignores held-out Experiment rows when sampling its mask, and the contrastive Experiment-clustering loss filters held-out Experiments out of its pair pool before sampling. Held-out Experiment nodes remain in the message-passing graph so probe-time message passing matches the pretrain-time graph the model was conditioned on; only supervision signal involving their identity, features, or neighbour structure is removed.

Link prediction. Thirty percent of edges are deterministically reserved as supervision labels via an MD5 hash of edge endpoints, ensuring consistent splits across sampling runs (subject to the hold-out filter above). A dot-product predictor scores edges, $\hat{y}_{uv} = \sigma(\mathbf{z}_u^\top \mathbf{z}_v)$, trained with binary cross-entropy. For each edge type we draw negatives at a 1:1 ratio with positives by uniformly sampling type-constrained (src, dst) pairs from the corresponding node pools within the sampled subgraph and rejecting pairs that are already edges.

Masked node modelling. We mask 15% of node features per batch. Per-type linear decoders reconstruct masked features under MSE, which acts as a regulariser against representation collapse.

Distance prediction. For each subgraph, 200 random node pairs are sampled and their shortest-path distances computed via BFS on the undirected graph before virtual-node edges are added (capped at 5 hops); virtual nodes would otherwise collapse the diameter to two and trivialise the target. A distance head (the Hadamard product of source and target embeddings followed by a two-layer MLP) regresses these distances under MSE, supervising pairwise embedding distances against the graph metric.

Contrastive experiment-pair loss. Experiment node embeddings are contrasted within each batch with a target similarity inversely proportional to graph distance, $s_{ij} = 1 - d_{ij}/d_{\max}$, trained with temperature-scaled cosine similarity (τ=0.07).

Loss combination. The four losses are combined via homoscedastic uncertainty weighting (Kendall et al., 2018). Each task has a learnable log-variance $\log \sigma_i$, and the total loss is

$$\mathcal{L} = \sum_i \frac{\mathcal{L}_i}{2\sigma_i^2} + \log \sigma_i. \tag{2}$$

$\log \sigma_i$ is clamped at -4.0 to prevent collapse; tasks with noisier supervision automatically receive lower weight.

3.5. Scalable Subgraph Sampling

Training on the full graph is infeasible; Canopy uses a Neo4j-backed streaming sampler that constructs

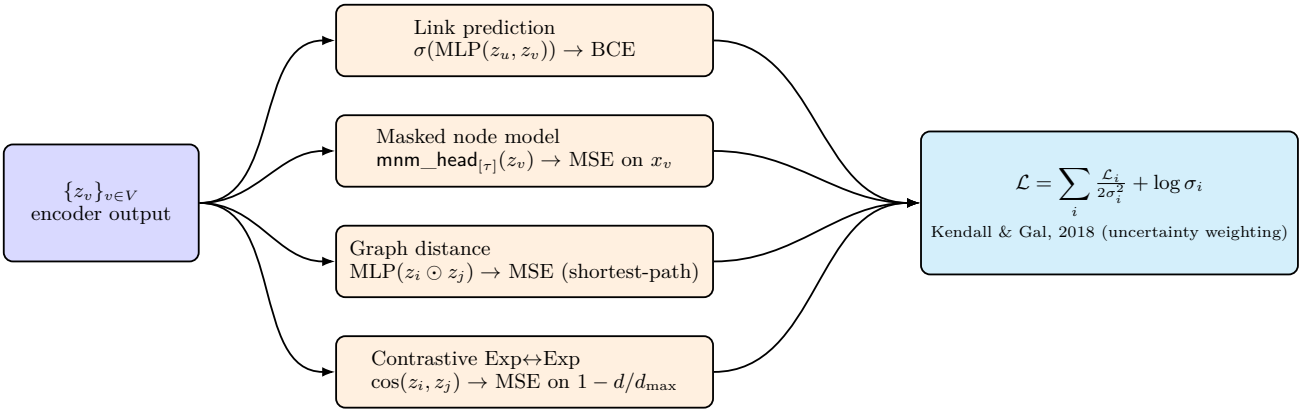

Figure 5. Pretraining objective. Four parallel heads consume the encoder output $\{z_v\}_{v \in V}$: link prediction (BCE), per-type masked node modelling (MSE), graph-distance regression (MSE), and an Experiment↔Experiment contrastive loss (MSE on cosine similarity). Their weighted sum, using Kendall et al. (2018) uncertainty weighting, forms the training objective.

mini-batch subgraphs in parallel worker processes and writes each to disk as a PyTorch Geometric (Fey & Lenssen, 2019) HeteroData object. The sampler supports three strategies (simple BFS, batched random walks, and multi-anchor); we use multi-anchor by default. Seeds are split 50/50 between yield-bearing Experiment nodes (so every probe target appears in some subgraph) and inverse-degree-weighted random nodes (so peripheral nodes are not crowded out by hubs).

For each Experiment seed batch the multi-anchor strategy resolves four BFS anchors: the seed Experiment; its Strain (via uses_strain); parent Taxon plus capped sibling strains (via belongs_to, capped at 10 per species); and adjacent MetabolicPathways (via has_pathway from the Strain). It also resolves four directed metapath anchors that walk fixed causal chains the undirected BFS undercovers: Experiment→Metabolite (target compound), Pathway→Reaction→UnirefCluster (pathway enzymes), Strain→GenomicGene→UnirefCluster (genetic edits to encoded proteins), and the reverse-direction chain Metabolite→Reaction→UnirefCluster→GenomicGene (target enzymes). Each anchor is allocated a fraction of a global node budget $N_{\max}$ (default 1,000); unused budget from inactive anchors is proportionally redistributed. Ontological edges (subclass_of, part_of, regulates) and transcriptomic measurement edges are excluded from BFS expansion to prevent hub-dominated subgraphs but are included in the final subgraph if both endpoints are present. The multi-anchor configuration rebalances the GenomicGene fraction from 78% to 15% and increases Experiment-node coverage roughly 13× relative to a naive 5-hop BFS. The 30% link-supervision split is computed deterministically via MD5 hashing of (src, rel, dst), so an edge belongs to the same set regardless of which subgraph contains it; edges incident to held-out Experiment nodes are kept in the message-passing graph but excluded from supervision. Continuous scalar features are z-score normalised per node type, and random-walk positional encodings are added to each subgraph at materialisation.

3.6. Downstream Tasks

After pretraining, we train lightweight probes on frozen experiment-node embeddings: a linear probe and a two-layer MLP probe for titer regression. We report R^2, RMSE, and Spearman ρ for regression, and AUROC and F1 for binary classification (above/below median titer). This protocol isolates representation quality from probe capacity.

4. Experiments

4.1. Setup

Knowledge graph statistics. Canopy's graph comprises 6.9 M nodes across 13 types and 11.2 M edges across 34 typed relations (Table 1, Section 3.1). At training time we additionally materialise reverse edges, self-loops, and virtual-node edges, bringing the per-batch typed-tuple count above 100.

Sampling. We sample 10,000 subgraphs with the multi-anchor expansion (k=3, $N_{\max}$=1000) of Section 3.5. Pretraining uses the deterministic MD5 edge-supervision split described in Section 3.5; the downstream titer probe uses a 5x CV Experiment-node split (Experiment IDs hashed and bucketed) so test exper-

iments are unseen during both pretrain and probe fitting.

Model configurations. We evaluate three scales:

Table 2. Model configurations used in this paper.

Config	Hidden	Heads	Layers	FFN	Params
Demo	64	4	6	4	∼80M
500M	128	4	6	4	∼0.5B
3B	256	8	12	4	∼3B

Training. AdamW (β_1=0.9, β_2=0.999) with cosine annealing and linear warmup; bfloat16 mixed precision; FSDP for distributed training across Intel Data Center GPU Max accelerators (Dawn). Gradient clipping at ℓ_2 norm 1.0. Hyperparameters tuned via a two-tier Optuna (Akiba et al., 2019) sweep: Tier 1 (1,000 trials at 500M, ∼500 XPU-hours, search over $h \in \{64, 128, 192, 256\}$, $L \in \{4, 6, 8\}$); Tier 2 (200 trials at 3B, ∼800 XPU-hours, search narrowed from Tier 1).

Baselines. Ridge, MLP (two hidden layers, 128→64), and XGBoost (500 trees, depth 6) trained directly on the 429-dim raw experiment-condition feature vector (the same input that feeds the Experiment node in the graph), using the identical MD5-hashed train/test split as the Canopy probe. We further compare against GraphSAGE (homogeneous SAGEConv) and HGT (vanilla) without SignNet, JK, or virtual nodes.

4.2. Main Results

Table 3. Titer prediction on the held-out test split (n=410 unique experiments). Tabular baselines and the Canopy probe share the same MD5-hashed split.

Method	R^2 ↑	AUROC↑
Ridge (raw conditions)	0.054	0.758
XGBoost (raw conditions)	0.065	0.779
MLP (raw conditions)	0.126	0.754
GraphSAGE	0.334	0.719
HGT (vanilla)	0.308	0.711
Canopy (Demo)	0.301	0.703
Canopy (500M)	0.380	0.805
Canopy (3B)	0.413	0.820

4.3. Ablations

Unless stated otherwise, ablations are run at the 500M scale defined in Section 4.1 with a 5k-sample budget; the runs in Table 3 use the full 10k-sample budget. Absolute R^2 values in the ablations therefore sit below the headline numbers and should be read as relative comparisons.

We ablate the four self-supervised tasks (Table 4), the three architectural augmentations (SignNet PE, Jumping Knowledge, and virtual nodes; Table 5), and network depth (Table 6). For depth, we compare the default L=6 architecture against an iso-parameter shallow variant (L=2, h=224) at 500M scale. The shallow model reaches a higher peak probe R^2, runs ∼2× faster per epoch, and, unlike the deeper variant, does not exhibit probe degradation through 20 epochs. We attribute this to a lower oversmoothing burden once HGTConv attention has enough per-head capacity (d_h=56); Jumping Knowledge alone is insufficient at L=6 on this graph. We retain the deeper default in the headline configuration: the gap is small (ΔR^2=0.011) and the deeper topology matches the 3B configuration from which we report headline numbers. Mitigations beyond JK and scaled residuals (DropEdge, GraphNorm, or deeper feature fusion) remain unexplored and are a natural next ablation.

Table 4. Pretraining-objective ablation (probe R^2, 500M scale, 5k samples).

Link	MNM	Dist	Contrast	R^2 ↑ / Δpp
✓	✓	✓	✓	0.359
	✓	✓	✓	−1.6
✓		✓	✓	−0.6
✓	✓		✓	−4.7
✓	✓	✓		−7.6

Task weighting. The four pretraining losses operate on very different scales: BCE for link prediction, MSE for masked-node and distance, and contrastive cosine for experiment pairs. Under flat (equal) weighting the highest-scale loss dominates the gradient and the probe collapses to R^2=−9.18, worse than the constant-mean predictor (Table 7). Replacing flat weights with the homoscedastic uncertainty scheme of Kendall et al. (2018) recovers R^2=0.359. The learned $\log \sigma_i$ values converge to per-task scales that balance gradient contributions automatically and adapt as the relative difficulty of each task shifts during training, removing the manual sweep over fixed per-task weights that would otherwise be needed.

5. Discussion

Each of the four pretraining objectives contributes (Table 4), with no single task dominating. Learned uncertainty weighting (Table 7) eliminates manual loss balancing and adapts as training progresses.

The 3B model improves R^2 from 0.380 to 0.413 over

Table 5. Architectural ablation (500M, 5k samples; probe R^2).

Variant	$R^2 \uparrow$ / Δpp
Full Canopy	0.359
− SignNet PE	−2.7
− JK aggregation	−3.5
− Virtual nodes	−10.3
− All three	−11.1

Table 6. Depth ablation (iso-parameter, 500M scale, 5k samples).

Variant	$R^2 \uparrow$ / Δpp	AUROC / Δpp
L=2, h=224 (shallow)	0.3703	0.7881
L=6, h=128 (deep)	−1.0	−1.1

the 500M model, a modest gain relative to the 6× parameter increase. With 4,791 total literature-mined fermentation records, the regime is likely data-bound rather than capacity-bound at this scale; substantiating scaling claims will require more experimental data, particularly across the long tail of compounds and organisms.

Limitations. First, experimental data is sparse relative to the KG, with uneven distribution across compounds and organisms. Second, the model is predictive rather than mechanistic; attention analysis offers some interpretability but does not replace mechanistic modelling. Third, our baselines do not include a graph-free pooled-encoder control, e.g., concatenating frozen ESM-2 and MoLFormer embeddings of a strain's genes and target compound and applying an MLP; such a probe would isolate the contribution of graph structure from the underlying foundation-model encoders and is a planned ablation. Fourth, release scope is constrained for this submission: the 4,791-experiment literature-mined benchmark and its split files will be released in a forthcoming publication, while the in-house LIMS records and trained model weights are not released with this workshop paper, with an archival release of the full pipeline planned to accompany a later journal submission.

Future work. Several extensions are already underway. On the representation side, we are scaling node features with larger and more sophisticated embedding models, ingesting orders of magnitude more protein sequences, and adding predicted protein structures as a complementary modality. On the data side, we are expanding the KG with DNA parts (promoters, RBSs, terminators, CDS variants) and metagenomic sequences from environmental and engineered communities, broadening coverage well beyond the current cultured-organism backbone. On the application side, the same frozen embeddings are being extended from titer prediction to chassis selection and de novo pathway design, both of which reuse Canopy's heterogeneous representation but condition on different query node types. Substantiating the "foundation model" framing also requires breadth in downstream evaluation; beyond titer prediction, we plan probes on chassis selection, gene essentiality, reaction prediction, and cross-organism transfer (e.g., pretraining on E. coli experiments and evaluating on yeast) to test whether the same frozen embeddings transfer across distinct metabolic-engineering tasks. Finally, we are pairing the learned representation with generative strain-design pipelines: a flow-matching model conditioned on Canopy embeddings proposes candidate multi-gene interventions, which are then scored by the frozen titer probe inside a Bayesian-optimisation loop, turning the predictive oracle into a closed-loop generative design system.

Table 7. Task-weighting comparison (500M, 5k samples).

Weighting	$R^2 \uparrow$
Flat (equal)	-9.18
Learned (Kendall)	0.359

Broader impact. Accelerating strain design supports a shift from petrochemical to fermentative production of chemicals, fuels, and therapeutics.

6. Conclusion

Frozen Canopy embeddings outperform tabular and homogeneous graph baselines on titer prediction, providing a cross-organism representation that can be reused as the conditioning substrate for downstream generative strain-design pipelines.

Impact Statement

This paper presents work whose goal is to advance the field of machine learning for biological design. There are many potential societal consequences of our work, including positive impact through greener bioproduction.

References

Abramson, J., Adler, J., Dunger, J., Evans, R., Green, T., Pritzel, A., Ronneberger, O., Willmore, L., Ballard, A. J., Bambrick, J., Bodenstein, S. W., Evans,

D. A., Hung, C.-C., O'Neill, M., Reiman, D., Tunyasuvunakool, K., Wu, Z., Žemgulytė, A., Arvaniti, E., Beattie, C., Bertolli, O., Bridgland, A., Cherepanov, A., Congreve, M., Cowen-Rivers, A. I., Cowie, A., Figurnov, M., Fuchs, F. B., Gladman, H., Jain, R., Khan, Y. A., Low, C. M. R., Perlin, K., Potapenko, A., Savy, P., Singh, S., Stecula, A., Thillaisundaram, A., Tong, C., Yakneen, S., Zhong, E. D., Zielinski, M., Žídek, A., Bapst, V., Kohli, P., Jaderberg, M., Hassabis, D., and Jumper, J. M. Accurate structure prediction of biomolecular interactions with AlphaFold 3. Nature, 630(8016):493–500, June 2024. ISSN 1476-4687. doi: 10.1038/s41586-024-07487-w. URL https://www.nature.com/articles/s41586-024-07487-w.

Ahern, W., Yim, J., Tischer, D., Salike, S., Woodbury, S. M., Kim, D., Kalvet, I., Kipnis, Y., Coventry, B., Altae-Tran, H. R., Bauer, M. S., Barzilay, R., Jaakkola, T. S., Krishna, R., and Baker, D. Atom-level enzyme active site scaffolding using RFdiffusion2. Nature Methods, 23 (1):96–105, January 2026. ISSN 1548-7105. doi: 10.1038/s41592-025-02975-x. URL https://www.nature.com/articles/s41592-025-02975-x.

Akiba, T., Sano, S., Yanase, T., Ohta, T., and Koyama, M. Optuna: A Next-generation Hyperparameter Optimization Framework. In Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '19, pp. 2623–2631, New York, NY, USA, July 2019. Association for Computing Machinery. ISBN 978-1-4503-6201-6. doi: 10.1145/3292500.3330701. URL https://dl.acm.org/doi/10.1145/3292500.3330701.

Ashburner, M., Ball, C. A., Blake, J. A., Botstein, D., Butler, H., Cherry, J. M., Davis, A. P., Dolinski, K., Dwight, S. S., Eppig, J. T., Harris, M. A., Hill, D. P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J. C., Richardson, J. E., Ringwald, M., Rubin, G. M., and Sherlock, G. Gene Ontology: tool for the unification of biology. Nature Genetics, 25(1):25–29, May 2000. ISSN 1546-1718. doi: 10.1038/75556. URL https://www.nature.com/articles/ng0500_25.

Brixi, G., Durrant, M. G., Ku, J., Poli, M., Brockman, G., Chang, D., Gonzalez, G. A., King, S. H., Li, D. B., Merchant, A. T., Naghipourfar, M., Nguyen, E., Ricci-Tam, C., Romero, D. W., Sun, G., Taghibakshi, A., Vorontsov, A., Yang, B., Deng, M., Gorton, L., Nguyen, N., Wang, N. K., Adams, E., Baccus, S. A., Dillmann, S., Ermon, S., Guo, D., Ilango, R., Janik, K., Lu, A. X., Mehta, R., Mofrad, M. R. K., Ng, M. Y., Pannu, J., Ré, C., Schmok, J. C., John, J. S., Sullivan, J., Zhu, K., Zynda, G., Balsam, D., Collison, P., Costa, A. B., Hernandez-Boussard, T., Ho, E., Liu, M.-Y., McGrath, T., Powell, K., Burke, D. P., Goodarzi, H., Hsu, P. D., and Hie, B. L. Genome modeling and design across all domains of life with Evo 2, February 2025. URL https://www.biorxiv.org/content/10.1101/2025.02.18.638918v1. Pages: 2025.02.18.638918 Section: New Results.

Brown, G. R., Hem, V., Katz, K. S., Ovetsky, M., Wallin, C., Ermolaeva, O., Tolstoy, I., Tatusova, T., Pruitt, K. D., Maglott, D. R., and Murphy, T. D. Gene: a gene-centered information resource at NCBI. Nucleic Acids Research, 43(Database issue):D36–42, January 2015. ISSN 1362-4962. doi: 10.1093/nar/gku1055.

Burgard, A. P., Pharkya, P., and Maranas, C. D. Optknock: A bilevel programming framework for identifying gene knockout strategies for microbial strain optimization. Biotechnology and Bioengineering, 84(6):647–657, 2003. ISSN 1097-0290. doi: 10.1002/bit.10803. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/bit.10803.

Chandak, P., Huang, K., and Zitnik, M. Building a knowledge graph to enable precision medicine. Scientific Data, 10(1):67, February 2023. ISSN 2052-4463. doi: 10.1038/s41597-023-01960-3. URL https://www.nature.com/articles/s41597-023-01960-3.

Cheng, Y., Bi, X., Xu, Y., Liu, Y., Li, J., Du, G., Lv, X., and Liu, L. Machine learning for metabolic pathway optimization: A review. Computational and Structural Biotechnology Journal, 21: 2381–2393, January 2023. ISSN 2001-0370. doi: 10.1016/j.csbj.2023.03.045. URL https://www.csbj.org/article/S2001-0370(23)00144-7/fulltext.

Cui, H., Wang, C., Maan, H., Pang, K., Luo, F., Duan, N., and Wang, B. scGPT: toward building a foundation model for single-cell multi-omics using generative AI. Nature Methods, 21 (8):1470–1480, August 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02201-0. URL https://www.nature.com/articles/s41592-024-02201-0.

Czajka, J. J., Oyetunde, T., and Tang, Y. J. Integrated knowledge mining, genome-scale modeling, and machine learning for predicting Yarrowia lipolytica bioproduction. Metabolic Engineering, 67:227–236, September 2021. ISSN 1096-7176. doi: 10.1016/j.ymben.2021.07.003. URL https://www.sciencedirect.com/science/article/pii/S1096717621001130.

Dwivedi, V. P., Luu, A. T., Laurent, T., Bengio, Y., and Bresson, X. Graph Neural Networks with Learnable Structural and Positional Representations, February 2022. URL http://arxiv.org/abs/2110.07875. arXiv:2110.07875 [cs].

Fey, M. and Lenssen, J. E. Fast Graph Representation Learning with PyTorch Geometric, April 2019. URL http://arxiv.org/abs/1903.02428. arXiv:1903.02428 [cs].

Gema, A. P., Grabarczyk, D., De Wulf, W., Borole, P., Alfaro, J. A., Minervini, P., Vergari, A., and Rajan, A. Knowledge graph embeddings in the biomedical domain: are they useful? A look at link prediction, rule learning, and downstream polypharmacy tasks. Bioinformatics Advances, 4 (1):vbae097, January 2024. ISSN 2635-0041. doi: 10.1093/bioadv/vbae097. URL https://doi.org/10.1093/bioadv/vbae097.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., and Dahl, G. E. Neural Message Passing for Quantum Chemistry. In Proceedings of the 34th International Conference on Machine Learning, pp. 1263–1272. PMLR, July 2017. URL https://proceedings.mlr.press/v70/gilmer17a.html.

Hao, M., Gong, J., Zeng, X., Liu, C., Guo, Y., Cheng, X., Wang, T., Ma, J., Zhang, X., and Song, L. Large-scale foundation model on single-cell transcriptomics. Nature Methods, 21(8):1481–1491, August 2024. ISSN 1548-7105. doi: 10.1038/s41592-024-02305-7. URL https://www.nature.com/articles/s41592-024-02305-7.

Hasibi, R., Michoel, T., and Oyarzún, D. A. Integration of graph neural networks and genome-scale metabolic models for predicting gene essentiality. npj Systems Biology and Applications, 10(1):24, March 2024. ISSN 2056-7189. doi: 10.1038/s41540-024-00348-2. URL https://www.nature.com/articles/s41540-024-00348-2.

Hayes, T., Rao, R., Akin, H., Sofroniew, N. J., Oktay, D., Lin, Z., Verkuil, R., Tran, V. Q., Deaton, J., Wiggert, M., Badkundri, R., Shafkat, I., Gong, J., Derry, A., Molina, R. S., Thomas, N., Khan, Y. A., Mishra, C., Kim, C., Bartie, L. J., Nemeth, M., Hsu, P. D., Sercu, T., Candido, S., and Rives, A. Simulating 500 million years of evolution with a language model. Science, 387(6736):850–858, February 2025. doi: 10.1126/science.ads0018. URL https://www.science.org/doi/10.1126/science.ads0018.

Himmelstein, D. S., Lizee, A., Hessler, C., Brueggeman, L., Chen, S. L., Hadley, D., Green, A., Khankhanian, P., and Baranzini, S. E. Systematic integration of biomedical knowledge prioritizes drugs for repurposing. eLife, 6:e26726, September 2017. ISSN 2050-084X. doi: 10.7554/eLife.26726. URL https://doi.org/10.7554/eLife.26726.

Hu, E. Y., Oleshko, S., Firmani, S., Cheng, H., Zhu, Z., Ulmer, M., Arnold, M., Colomé-Tatché, M., Tang, J., Xhonneux, S., and Marsico, A. Enhancing link prediction in biomedical knowledge graphs with BioPathNet. Nature Biomedical Engineering, pp. 1–23, January 2026. ISSN 2157-846X. doi: 10.1038/s41551-025-01598-z. URL https://www.nature.com/articles/s41551-025-01598-z.

Hu, W., Liu, B., Gomes, J., Zitnik, M., Liang, P., Pande, V., and Leskovec, J. Strategies for Pre-training Graph Neural Networks, February 2020a. URL http://arxiv.org/abs/1905.12265. arXiv:1905.12265 [cs].

Hu, Z., Dong, Y., Wang, K., and Sun, Y. Heterogeneous graph transformer. CoRR, abs/2003.01332, 2020b. URL https://arxiv.org/abs/2003.01332.

Huang, K., Fu, T., Gao, W., Zhao, Y., Roohani, Y. H., Leskovec, J., Coley, C. W., Xiao, C., Sun, J., and Zitnik, M. Therapeutics Data Commons: Machine Learning Datasets and Tasks for Drug Discovery and Development. June 2021. URL https://openreview.net/forum?id=8nvgnORnoWr.

Huang, K., Chandak, P., Wang, Q., Havaldar, S., Vaid, A., Leskovec, J., Nadkarni, G. N., Glicksberg, B. S., Gehlenborg, N., and Zitnik, M. A foundation model for clinician-centered drug repurposing. Nature Medicine, 30(12):3601–3613, December 2024. ISSN 1546-170X. doi: 10.1038/s41591-024-03233-x. URL https://www.nature.com/articles/s41591-024-03233-x.

Kanehisa, M. and Goto, S. KEGG: Kyoto Encyclopedia of Genes and Genomes. Nucleic Acids Research, 28(1):27–30, January 2000. ISSN 0305-1048. doi: 10.1093/nar/28.1.27. URL https://doi.org/10.1093/nar/28.1.27.

Kendall, A., Gal, Y., and Cipolla, R. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics, April 2018. URL http://arxiv.org/abs/1705.07115. arXiv:1705.07115 [cs].

Khodayari, A. and Maranas, C. D. A genome-scale Escherichia coli kinetic metabolic model k-ecoli457 satisfying flux data for multiple mutant strains. Nature Communications, 7(1):13806, 2016. doi: 10.1038/ncomms13806.

Li, Z., Zeng, Z., Lin, X., Fang, F., Qu, Y., Xu, Z., Liu, Z., Ning, X., Wei, T., Liu, G., Tong, H., and He, J. Flow matching meets biology and life science: a survey. npj Artificial Intelligence, 2(1):17, January 2026. ISSN 3005-1460. doi: 10.1038/s44387-025-00066-y. URL https://www.nature.com/articles/s44387-025-00066-y.

Lim, D., Robinson, J., Zhao, L., Smidt, T., Sra, S., Maron, H., and Jegelka, S. Sign and Basis Invariant Networks for Spectral Graph Representation Learning, September 2022. URL http://arxiv.org/abs/2202.13013. arXiv:2202.13013 [cs].

Lin, Z., Akin, H., Rao, R., Hie, B., Zhu, Z., Lu, W., Smetanin, N., Verkuil, R., Kabeli, O., Shmueli, Y., dos Santos Costa, A., Fazel-Zarandi, M., Sercu, T., Candido, S., and Rives, A. Evolutionary-scale prediction of atomic-level protein structure with a language model. Science, 379(6637):1123–1130, March 2023. doi: 10.1126/science.ade2574. URL https://www.science.org/doi/10.1126/science.ade2574.

Lobentanzer, S., Aloy, P., Baumbach, J., Bohar, B., Charoentong, P., Danhauser, K., Doğan, T., Dreo, J., Dunham, I., Fernandez-Torras, A., Gyori, B. M., Hartung, M., Hoyt, C. T., Klein, C., Korcsmaros, T., Maier, A., Mann, M., Ochoa, D., Pareja-Lorente, E., Popp, F., Preusse, M., Probul, N., Schwikowski, B., Sen, B., Strauss, M. T., Turei, D., Ulusoy, E., Wodke, J. A. H., and Saez-Rodriguez, J. Democratising Knowledge Representation with BioCypher, January 2023. URL http://arxiv.org/abs/2212.13543. arXiv:2212.13543 [q-bio].

Mao, H., Chen, Z., Tang, W., Zhao, J., Ma, Y., Zhao, T., Shah, N., Galkin, M., and Tang, J. Position: Graph Foundation Models are Already Here, May 2024. URL http://arxiv.org/abs/2402.02216. arXiv:2402.02216 [cs].

Moretti, S., Tran, V. D. T., Mehl, F., Ibberson, M., and Pagni, M. Metanetx/mnxref: unified namespace for metabolites and biochemical reactions in the context of metabolic models. Nucleic Acids Research, 49(D1):D570–D574, 01 2021. ISSN 0305-1048. doi: 10.1093/nar/gkaa992. URL https://doi.org/10.1093/nar/gkaa992.

Márquez-Zavala, E., Bartolomeu, F. D., and Machado, D. A database of over 15.000 strain design publications reveals a conserved set of metabolic engineering targets across microbial hosts and products, December 2025. URL https://www.biorxiv.org/content/10.64898/2025.12.15.694291v1. ISSN: 2692-8205 Pages: 2025.12.15.694291 Section: New Results.

Nelson, C. A., Butte, A. J., and Baranzini, S. E. Integrating biomedical research and electronic health records to create knowledge-based biologically meaningful machine-readable embeddings. Nature Communications, 10(1):3045, July 2019. ISSN 2041-1723. doi: 10.1038/s41467-019-11069-0. URL https://www.nature.com/articles/s41467-019-11069-0.

Nguyen, E., Poli, M., Durrant, M. G., Kang, B., Katrekar, D., Li, D. B., Bartie, L. J., Thomas, A. W., King, S. H., Brixi, G., Sullivan, J., Ng, M. Y., Lewis, A., Lou, A., Ermon, S., Baccus, S. A., Hernandez-Boussard, T., Ré, C., Hsu, P. D., and Hie, B. L. Sequence modeling and design from molecular to genome scale with evo. Science, 386(6723):eado9336, 2024. doi: 10.1126/science.ado9336. URL https://www.science.org/doi/abs/10.1126/science.ado9336.

Opgenorth, P., Costello, Z., Okada, T., Goyal, G., Chen, Y., Gin, J., Benites, V., de Raad, M., Northen, T. R., Deng, K., Deutsch, S., Baidoo, E. E. K., Petzold, C. J., Hillson, N. J., Garcia Martin, H., and Beller, H. R. Lessons from two design–build–test–learn cycles of dodecanol production in escherichia coli aided by machine learning. ACS Synthetic Biology, 8(6):1337–1351, 2019. doi: 10.1021/acssynbio.9b00020. PMID: 31072100.

Oyetunde, T., Liu, D., Martin, H. G., and Tang, Y. J. Machine learning framework for assessment of microbial factory performance. PLOS ONE, 14(1):e0210558, January 2019. ISSN 1932-6203. doi: 10.1371/journal.pone.0210558. URL https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0210558.

Paysan-Lafosse, T., Blum, M., Chuguransky, S., Grego, T., Pinto, B. L., Salazar, G., Bileschi, M., Bork, P., Bridge, A., Colwell, L., Gough, J., Haft, D., Letunić, I., Marchler-Bauer, A., Mi, H., Natale, D., Orengo, C., Pandurangan, A., Rivoire, C., Sigrist, C. J. A., Sillitoe, I., Thanki, N., Thomas, P. D., Tosatto, S. C. E., Wu, C., and Bateman, A. InterPro in 2022. Nucleic Acids Research, 51(D1):D418–D427, January 2023. ISSN 0305-1048. doi: 10.1093/nar/gkac993. URL https://doi.org/10.1093/nar/gkac993.

Porokhin, V., Liu, L.-P., and Hassoun, S. Using graph neural networks for site-of-metabolism prediction and its applications to ranking promiscuous enzymatic products. Bioinformatics, 39(3):btad089, March 2023. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad089. URL https://doi.org/10.1093/bioinformatics/btad089.

Radivojević, T., Costello, Z., Workman, K., and Garcia Martin, H. A machine learning Automated Recommendation Tool for synthetic biology. Nature Communications, 11(1):4879, September 2020. ISSN 2041-1723. doi: 10.1038/s41467-020-18008-4. URL https://www.nature.com/articles/s41467-020-18008-4.

Ross, J., Belgodere, B., Chenthamarakshan, V., Padhi, I., Mroueh, Y., and Das, P. Large-Scale Chemical Language Representations Capture Molecular Structure and Properties, June 2021. URL https://arxiv.org/abs/2106.09553v3.

Schneider, P., Bekiaris, P. S., von Kamp, A., and Klamt, S. StrainDesign: a comprehensive Python package for computational design of metabolic networks. Bioinformatics, 38(21):4981–4983, November 2022. ISSN 1367-4811. doi: 10.1093/bioinformatics/btac632. URL https://doi.org/10.1093/bioinformatics/btac632.

Schoch, C. L., Ciufo, S., Domrachev, M., Hotton, C. L., Kannan, S., Khovanskaya, R., Leipe, D., Mcveigh, R., O'Neill, K., Robbertse, B., Sharma, S., Soussov, V., Sullivan, J. P., Sun, L., Turner, S., and Karsch-Mizrachi, I. NCBI Taxonomy: a comprehensive update on curation, resources and tools. Database, 2020:baaa062, January 2020. ISSN 1758-0463. doi: 10.1093/database/baaa062. URL https://doi.org/10.1093/database/baaa062.

Song, T., Yin, L., Han, Z., and Xu, Z. Improving Enzyme Prediction with Chemical Reaction Equations by Hypergraph-Enhanced Knowledge Graph Embeddings, January 2026. URL http://arxiv.org/abs/2601.05330. arXiv:2601.05330 [cs].

Su, J., Han, C., Zhou, Y., Shan, J., Zhou, X., and Yuan, F. SaProt: Protein Language Modeling with Structure-aware Vocabulary, April 2024. URL https://www.biorxiv.org/content/10.1101/2023.10.01.560349v5. Pages: 2023.10.01.560349 Section: New Results.

Suzek, B. E., Wang, Y., Huang, H., McGarvey, P. B., Wu, C. H., and the UniProt Consortium. UniRef clusters: a comprehensive and scalable alternative for improving sequence similarity searches. Bioinformatics, 31(6):926–932, March 2015. ISSN 1367-4803. doi: 10.1093/bioinformatics/btu739. URL https://doi.org/10.1093/bioinformatics/btu739.

The UniProt Consortium. UniProt: the Universal Protein Knowledgebase in 2023. Nucleic Acids Research, 51(D1):D523–D531, January 2023. ISSN 0305-1048. doi: 10.1093/nar/gkac1052. URL https://doi.org/10.1093/nar/gkac1052.

van Lent, P., Schmitz, J., and Abeel, T. Simulated Design–Build–Test–Learn Cycles for Consistent Comparison of Machine Learning Methods in Metabolic Engineering. ACS Synthetic Biology, 12(9):2588–2599, August 2023. ISSN 2161-5063. doi: 10.1021/acssynbio.3c00186. URL https://pmc.ncbi.nlm.nih.gov/articles/PMC10510747/.

Wang, Z., Liu, Z., Ma, T., Li, J., Zhang, Z., Fu, X., Li, Y., Yuan, Z., Song, W., Ma, Y., Zeng, Q., Chen, X., Zhao, J., Li, J., Jiang, M., Lio, P., Chawla, N., Zhang, C., and Ye, Y. Graph Foundation Models: A Comprehensive Survey, May 2025. URL http://arxiv.org/abs/2505.15116. arXiv:2505.15116 [cs].

Xin, K., Wang, Q., Chen, J., Yu, P., Zhao, H., and Ji, H. Gene-Metabolite Association Prediction with Interactive Knowledge Transfer Enhanced Graph for Metabolite Production, October 2024. URL http://arxiv.org/abs/2410.18475. arXiv:2410.18475 [cs].

Xu, K., Li, C., Tian, Y., Sonobe, T., Kawarabayashi, K.-i., and Jegelka, S. Representation Learning on Graphs with Jumping Knowledge Networks. In Proceedings of the 35th International Conference on Machine Learning, pp. 5453–5462. PMLR, July 2018. URL https://proceedings.mlr.press/v80/xu18c.html.

A. Knowledge graph construction details

Canopy's graph is assembled with BioCypher (Lobentanzer et al., 2023), which separates what the graph contains from how it is collected. A single declarative schema (YAML, keyed by Biolink-aligned node and edge types) fixes the allowed types, their identifier namespaces, and their property contracts. Ten adapter modules each emit typed node and edge streams against that schema; BioCypher rejects labels that are not declared, collapses sub-classes onto the declared parent type, and writes neo4j-admin-importable CSVs. Adapters can therefore be added, swapped, or version-pinned without touching graph code, and the same schema drives both build-time validation and downstream sampling. The ten adapters span public reference resources (sequence, structure, pathway, ontology), a literature-mined fermentation corpus, and one in-house LIMS adapter contributing unpublished DBTL records (Table 8).

Table 8. BioCypher adapters used to construct Canopy's KG.

Adapter	Source	Nodes / edges produced
Genomic	UniProt REST + curated chassis list	Chassis, GenomicGene; ENCODES (Gene→UnirefCluster), HAS_GENE (Chassis→Gene).
UniRef	UniRef90 .tsv, MetaNetX cross-refs	UnirefCluster; CATALYZED_BY (Reaction→UnirefCluster), HAS_GO_TERM.
InterPro	protein2ipr.dat.gz (bulk) or UniProt batch API	InterProDomain; HAS_DOMAIN (UnirefCluster→Domain).
Gene Ontology	GO OBO release (obonet)	GOTerm (BP/MF/CC); ontology hierarchy edges (subclass_of, part_of, regulates).
MetaNetX	MNXref (Moretti et al., 2021)	Metabolite, Reaction; HAS_PARTICIPANT, HAS_PRODUCT.
Taxonomy	NCBI Taxonomy	Taxon; BELONGS_TO (Strain→Taxon).
Transcriptomic	Public RNA-seq compendia	Transcriptomic measurement nodes; MEASURED_BY_TRANSCRIPTOMIC, DERIVED_FROM_TRANSCRIPTOMIC, USES_STRAIN_TRANSCRIPTOMIC edges.
Experimental scrape	Literature-mined fermentation records	Experiment, Strain, MetabolicPathway; USES_STRAIN, HAS_PATHWAY, TARGETS_COMPOUND, MEASURES_COMPOUND, HAS_KNOCKOUT, HAS_KNOCKIN, HAS_OVEREXPRESSION.
Pathway enrichment	KEGG REST	enriches MetabolicPathway nodes with KEGG metadata; HAS_STEP (Pathway→Reaction).
LIMS (in-house)	Internal experiment registry	supplements Experiment / Strain with unpublished DBTL records and the edit-edge types above.

Schema harmonisation. BioCypher's ontology layer maps every adapter's emitted labels to Biolink superclasses. Only types declared in the project schema are retained at build time, and sub-class collapse prevents label fragmentation (e.g. "BiologicalProcess" and "MolecularFunction" are both rolled up under GOTerm). Edges referencing undeclared node types are dropped before import and logged.

Identifier normalisation and deduplication. Cross-source identifiers follow a fixed precedence: MetaNetX MNX IDs for metabolites and reactions, UniRef90 cluster IDs for proteins, NCBI gene IDs for genes, NCBI Taxonomy IDs for taxa, GO IDs for ontology terms, and InterPro IDs for domains. Organism names in the literature-mined corpus are normalised through a curated synonym table that maps frequently-confused taxonomic labels (e.g. Pichia pastoris → Komagataella phaffii, Clostridium thermocellum → Acetivibrio thermocellus) to the canonical chassis list. Duplicate metabolite and reaction nodes from MetaNetX cross-references are merged on MNX ID; Uniref→Reaction and Uniref→GOTerm mappings are resolved against the MetaNetX curated reaction model and the GO release pinned at build time. Edges with malformed or missing endpoints are dropped at adapter time and logged.

Build pipeline. Adapters write CSVs to a staging directory, BioCypher emits the matching neo4j-admin headers and edge files, and the resulting graph is loaded into Neo4j 5 via neo4j-admin import. The final graph contains 6.86 M nodes and 11.2 M schema-typed edges across 13 node types and 34 typed relations (Table 1).

B. Hyperparameter sweep details

Hyperparameters are selected with a staged Optuna sweep that narrows the search as model scale increases (Section 4.1). Tier 1 explores broadly at the 500M scale on a single XPU; Tier 2 refines around the Tier 1 optimum at the 3B operating point under FSDP across four XPUs, optionally with gradient checkpointing. All trials optimise the held-out titer probe R^2 at its best epoch, and both studies use Optuna's TPESampler with median pruning.

Tables 9 and 10 list the Tier 1 and Tier 2 search spaces in full. Trials whose hidden_channels is not divisible by heads are pruned at sample time.

Table 9. Tier 1 search space (500M scale, single XPU). 1,000 trials, ∼500 XPU-hours total compute budget, 80 epochs per trial, probe evaluated every 5 epochs.

Parameter	Range / set	Sampling
Architecture		
hidden_channels	$\{64, 128, 192, 256\}$	categorical
num_layers	$[2, 8]$	integer
heads	$[2, 8]$	integer
ffn_expansion	$\{2, 4\}$	categorical
dropout	$[0.05, 0.40]$	float
Optimisation		
lr	$[10^{-4}, 5{\times}10^{-3}]$	log-uniform
weight_decay	$[10^{-4}, 10^{-1}]$	log-uniform
warmup_epochs	$[1, 10]$	integer
cosine_end_offset	$[1, 30]$	integer
batch_size	$\{8, 16, 32, 64, 128, 256\}$	categorical
accumulate_grad_batches	$\{1, 2, 4\}$	categorical
Probe geometry		
probe_hidden_dim	$\{32, 64, 128, 256\}$	categorical
probe_num_layers	$[1, 4]$	integer
probe_dropout	$[0.0, 0.3]$	float

Table 10. Tier 2 search space (3B scale, FSDP/4). 200 trials, 30 epochs per trial, probe evaluated every 5 epochs. Initial trials warm-started from the top-10 Tier-1 configurations.

Parameter	Range / set	Sampling
Architecture		
hidden_channels	$\{192, 256, 384\}$	categorical
num_layers	$[8, 12]$	integer
heads	$[4, 8]$	integer
ffn_expansion	$\{2, 4\}$	categorical
dropout	$[0.05, 0.30]$	float
Optimisation		
lr	$[5{\times}10^{-5}, 3{\times}10^{-3}]$	log-uniform
weight_decay	$[10^{-4}, 10^{-1}]$	log-uniform
warmup_epochs	$[2, 10]$	integer
cosine_end_offset	$[1, 15]$	integer
batch_size	$\{32, 64, 128\}$	categorical
accumulate_grad_batches	$\{1, 2, 4, 8\}$	categorical

The Tier-1 winner combines L=6, h=128 (matching the 500M row of Table 2) with bs=256, lr=$2{\times}10^{-3}$, four warmup epochs, a cosine_end_offset of 6, and all four pretraining losses active. This configuration is used for the 500M headline run and seeded the Tier-2 search at 3B scale.