Neurosymbolic Artificial Intelligence journal:1–33, 2025

Graph Neural Network based hierarchy-aware Box Embeddings of Knowledge Graphs†

Filip Kronström FILIPKRO@CHALMERS.SE
Alexander H. Gower
Daniel Brunnsåker
Department of Computer Science and Engineering,
Chalmers University of Technology and University of Gothenburg, Sweden

Ievgeniia A. Tiukova
Department of Life Sciences, Chalmers University of Technology, Sweden
Department of Industrial Biotechnology, KTH Royal Institute of Technology, Sweden

Ross D. King
Department of Computer Science and Engineering,
Chalmers University of Technology and University of Gothenburg, Sweden
Department of Chemical Engineering and Biotechnology, University of Cambridge, United Kingdom

Abstract

We present a method for embedding knowledge graphs (KGs) using graph neural networks (GNNs) enriched with a semantic loss derived from underlying ontologies, yielding embeddings that better reflect domain knowledge. To demonstrate their utility, we predict and interpret the effects of gene deletions in the yeast *Saccharomyces cerevisiae* and learn box embeddings for KGs in the absence of a prediction task. We further show how box embeddings can serve as the basis for evaluating KG revisions.

Our yeast KG is constructed from community databases and ontology terms. Class hierarchies are encoded as low-dimensional box embeddings, which, combined with GNNs, predict cell growth for double gene knockouts, demonstrating that high-level qualitative knowledge is informative about experimental outcomes. Incorporating class hierarchy information through box embeddings improves predictive performance compared to task-specific features, and applying semantic loss further enhances this effect by aligning embeddings with ontology structure. This shows that class hierarchies from ontologies can be exploited for quantitative prediction. The model also generalises to other genetic modifications beyond those seen in training.

Additionally, we apply interpretability techniques to identify co-occurring edges important for predictions. A biological experiment validates one such finding, revealing an association between inositol utilisation and osmotic stress resistance, highlighting the model's potential to guide biological discovery.

†. This is an extended version of the paper Ontology-based box embeddings and knowledge graphs for predicting phenotypic traits in *Saccharomyces cerevisiae* (Kronström et al., 2025), presented at NeSy 2025.

1. Introduction and Related work

Embeddings of ontologies or knowledge graphs (KGs) in an n-dimensional space, $\mathbb{R}^n$, where their structure is in some way maintained, have proved useful for downstream tasks such as link and property prediction from KGs. In many fields ontologies have been carefully designed to describe how different terms in these domains relate to each other. Hierarchical information expressed as `subClassOf` relations between classes is especially prevalent. A desirable property for embeddings of KGs is that they represent, not just the information in the graph itself, but also the hierarchies defining the terms which describe the nodes. Making sure the embedding complies with our knowledge about the domain can, for example, provide additional information beneficial for prediction tasks or help generalise beyond the observed data (Gutiérrez-Basulto and Schockaert, 2018).

KG embeddings can be generated using different approaches. TransE represents links between entities as translations in a vector space (Bordes et al., 2013). The systemTransE has also been extended to model hierarchical class information in $\mathcal{EL}^{++}$ ontologies to represent '`subClassOf`' relationships as classes maintained within hyperspheres (Kulmanov et al., 2019). Vilnis et al. (2018) introduced axis-aligned hyperrectangles, or boxes, as a way of embedding entities in graphs and (Peng et al., 2022) combined it with the TransE model. Gumbel boxes have been introduced, where Gumbel distributions are used to represent box parameters, to avoid large flat regions of the loss landscape for transitive relation embeddings (Dasgupta et al., 2020). Instead of representing relations as translations of classes, graph neural networks (GNNs), e.g., GraphSAGE, can be used to aggregate features from neighbors in the graph to generate embeddings of nodes (Hamilton et al., 2017). Integrating deep learning with symbolic reasoning has, for example, been demonstrated by Xu et al. (2018), through the introduction of a semantic loss that can be combined with task-specific loss functions to penalise neural networks that violate logical constraints.

Box embeddings have been combined with GNNs, for example in recommendation systems by Liang et al. (2023) and Lin et al. (2024). However, to the best of our knowledge, they have not been used as a way of generating semantically correct KG embeddings.

KGs have successfully been used to describe heterogeneous data from various domains by combining instantiated facts with semantically meaningful concepts from ontologies. For example they have been used to model information on the internet in the Google Knowledge Graph[1] and Wikidata (Vrandečić and Krötzsch, 2014), or by Netflix to improve user recommendations (Huang et al., 2023). In the biomedical domain, KGs such as BioKG (Walsh et al., 2020) and SPOKE (Morris et al., 2023) combine information from different databases to create one large heterogeneous graph with information about, for example, genes and drugs. There are also graphs describing more narrow phenomenon such as the protein-protein associations and the drug-drug interactions in the Open Graph Benchmark (Hu et al., 2020).

A deeper understanding of cellular function and the roles of individual genes is central to biological research and critical for applications such as drug development. The yeast *Saccharomyces cerevisiae* (baker's yeast) is among the most extensively studied organisms. It has attracted research interest, not only due to its industrial applications, such as the production of beer, wine, and biofuels, but more importantly because it serves as a model

[1] `https://blog.google/products/search/introducing-knowledge-graph-things-not/`

eukaryotic organism. Through this role it helps our understanding of higher eukaryotes, such as humans and plants (Parapouli et al., 2020). Despite decades of research, our understanding of yeast biology is still incomplete: many genes remain unannotated (Wood et al., 2019), and interactions between genes can lead to complex and unexpected phenotypic outcomes (Costanzo et al., 2019). To improve our knowledge about any organism, actual experiments in the lab play a crucial role. However, given the complexity of biological systems, the number of experiments needed to fully explore even the simplest of organisms is incredibly large. Because of this, methods supporting hypothesis generation at scale are highly useful to speed up new discoveries (King et al., 2004; Brunnsåker et al., 2025).

The results of decades of research on *S. cerevisiae* are available both in literature and in more structured formats, such as databases. Saccharomyces Genome Database (SGD) aggregates curated information about *S. cerevisiae* genes. It includes Gene Ontology annotations, observed phenotypes, and information on regulatory relationships and genetic interactions (Engel et al., 2024). Information about reactions (biochemical events where a substrate is converted into a product) and pathways (a series of interconnected reactions that collectively drive cellular functions) is available in, among others, BioCyc (Karp et al., 2019).

Information in such databases is often represented and communicated using ontologies and controlled vocabularies. The Gene Ontology (GO) defines classes describing processes, functions, and components in cells (Ashburner et al., 2000). To properly represent phenotypes in SGD the Ascomycete Phenotype Ontology (APO) was developed (Costanzo et al., 2009). Phenotypes describe observed characteristics resulting from the interaction between a genotype and an environment, such as growth characteristics or resistances to environmental or chemical perturbants. Chemical compounds are specified in Chemical Entities of Biological Interest (ChEBI) (Hastings et al., 2016). The Interaction Network Ontology (INO) (Hur et al., 2015) and Molecular Interactions (MI) (Hermjakob et al., 2004) defines genetic, physical and regulatory interactions between genes and proteins. Commonly used relations between classes are introduced in the Relations Ontology (RO) (Mungall et al., 2020). The Basic Formal Ontology (BFO) is a top-level ontology developed to simplify alignment of terms from different ontologies (Arp et al., 2015).

Predicting biological properties from structured background knowledge can be done in different ways. Ma et al. (2018) encode GO-annotations together with the GO hierarchy in a neural network to predict cellular growth in *S. cerevisiae*. By predicting protein abundances using mined patterns from a Datalog knowledgebase containing facts from databases such as SGD, Brunnsåker et al. (2024) connected qualitative concepts to quantified intracellular measurements. KG embeddings have, for example, been used by Gualdi et al. (2024) to predict genes associated with diseases from a protein interaction KG.

2. Preliminaries

Box representations

Axis-aligned hyperrectangles, or "boxes" as they often are referred to, are defined as the Cartesian product of closed intervals,

$$\text{Box} = \prod_{i=1}^{n} [z_i, Z_i], \tag{1}$$

where z_i and Z_i correspond the lower and upper coordinate along dimension i. To fulfil the criteria that the upper coordinate should be greater than or equal to the lower coordinate, $Z_i \geq z_i$, boxes are often generated by applying some transformation on a latent variable. In this work, we create boxes from latent variables, θ, using the `MinDeltaBoxTensors` constructor introduced by Chheda et al. (2021), where the upper and lower box coordinates are defined as follows:

$$z_i = \theta_i^z, \qquad Z_i = z_i + \text{softplus}(\theta_i^Z) \tag{2}$$

Boxes can also be represented by their centre-point, c_i, and offset, o_i, along dimension i, found from z and Z as follows:

$$c_i = \frac{z_i + Z_i}{2}, \qquad o_i = Z_i - c_i \tag{3}$$

Semantic losses

In this work we want to embed and make predictions from KGs with rich concept hierarchies. Box embeddings, where each class is represented by a box, presents a natural interpretation of the `subClassOf` relation[2], where the class-box of a subclass is contained within its superclass. To learn box embeddings we consider two different types of loss functions for concept inclusions on the form $C \sqsubseteq D$. The first loss, here called $\mathcal{L}_{distance}$, has previously been used by, for example, Peng et al. (2022) and Jackermeier et al. (2024). Using the nomenclature presented above it is calculated by first finding the element-wise distance between the two boxes,

$$d(C_i, D_i) = |c_i^C - c_i^D| - o_i^C - o_i^D \tag{4}$$

In the loss function we use this to find the distance from the subclass being completely contained within the superclass,

$$\mathcal{L}_{distance}(C, D) = \left|\left| \Big(\max(0, d(C_i, D_i) + 2o_i^C) \Big)_{i=1}^{n} \right|\right| \tag{5}$$

To keep disjoint classes, $C \sqcap D \sqsubseteq \bot$, apart we penalise overlap of the boxes by the following loss:

$$\mathcal{L}_{distance}^{-}(C, D) = \left|\left| \Big(\max(0, -d(C_i, D_i)) \Big)_{i=1}^{n} \right|\right| \tag{6}$$

The second loss type considers the overlap between boxes and for the subsumption $C \sqsubseteq D$ it is calculated as

$$\mathcal{L}_{overlap} = -\log \left(\frac{\mathsf{Vol}(\text{Box}(C) \cap \text{Box}(D))}{\mathsf{Vol}(\text{Box}(C))} \right) \tag{7}$$

[2] The same interpretation holds for any other transitive relation as well.

For disjoint classes we instead use the following loss:

$$\mathcal{L}^{-}_{overlap} = -\log\Big(1 - \frac{\mathsf{Vol}(\mathrm{Box}(C) \cap \mathrm{Box}(D))}{\mathsf{Vol}(\mathrm{Box}(C))}\Big) \tag{8}$$

To avoid large flat regions in the loss landscape, for example when two boxes are completely disjoint, Dasgupta et al. (2020) proposes that boxes and intersections of boxes are interpreted as Gumbel random variables. They show that the volume of such boxes and intersections are determined by Bessel functions which can be reasonably approximated by softplus functions. In practice, this produces smooth intersections between boxes and ensures non-zero gradients, also in cases such as disjoint boxes.

Throughout this work we use $\mathcal{L}_{\sqsubseteq}$ and $\mathcal{L}^{-}_{\sqsubseteq}$ as placeholders for either of the inclusion and disjointness losses introduced above. To learn box embeddings the positive and negative losses are simply added together, possibly weighted differently. We present ways of doing this in more detail in Sections 3.3 and 3.4.

Regularisation losses

Patel et al. (2020) proposes to regularise the volume of boxes when training box embeddings, we use the implementation by Chheda et al. (2021) which does this by applying the L2-norm to all sides of the box,

$$R = \sum_{i}^{n} \|Z_i - z_i\|^2 \tag{9}$$

We also found that regularising too small boxes can be beneficial in some cases. This was implemented as

$$R = \sum_{i}^{n} \max\big(0, \frac{1}{\|Z_i - z_i\|} - l_0\big), \tag{10}$$

with l_0 being a threshold determining below what size the box is penalised.

3. Material and methods

3.1. Box embeddings

The losses presented in Section 2 can be used to train shallow embeddings of class hierarchies, but they can also be used for training of GNNs. Each layer $l = 1, \ldots, L$ of the GNN learns weights w_l that parametrise a function $B_l : \mathbb{R}^{n_{l-1}} \longrightarrow \mathbb{R}^{n_l}$. Treating the output of each layer in a GNN, $\theta_l = B_l(\theta_{l-1}; w_l)$, as the latent variable for a box embedding, boxes can be generated using the transformation in (2) and fed to the loss functions in (5-8). An illustration of this can be seen in Figure 1. Heterogeneous KGs, with different domains made up of classes we do not want to embed together, can have separate embeddings for each domain, which are trained using separate class hierarchies.

The approach is flexible in the sense that it is architecture agnostic and can be used either on its own for box embeddings of KGs using GNNs, or as a semantic loss together together with another loss term, for example, when training prediction models. The rationale behind this approach is that class hierarchies often contain information that is not necessarily

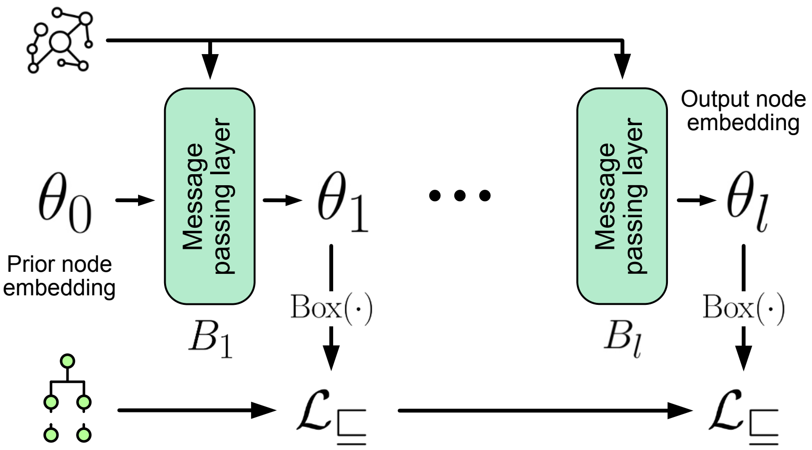


Figure 1: Illustration of how we use box embeddings to represent the class hierarchies throughout GNNs. The output of each message passing layer, which aggregates information between neighbors in the KG, is treated as a latent variable that is converted into boxes through the box transformation in (2). The boxes are trained to fulfil specified class hierarchies using the losses in (5-8), which can also be applied to the prior node embedding.

modelled in the graph edges, and can be especially useful to improve the representation of poorly connected nodes in the graph.

To prevent the embeddings from collapsing into the same boxes for all classes, negative examples can be drawn randomly and used as disjoint classes in the losses in (6) and (8).

To represent boxes and implement box-related operations, such as intersection and volume calculations, we use the `box-embeddings` Python package (v0.1.0) (Chheda et al., 2021).

3.2. Knowledge graph

We have created a heterogeneous knowledge graph describing genes in the yeast *Saccharomyces cerevisiae*, by combining facts expressed in classes and relations from multiple ontologies. The graph is specified in description logic, only using TBox statements by rewriting class assertions, $C(a)$, as $\{a\} \sqsubseteq C$ and role assertions, $r(a, b)$, as $\{a\} \sqsubseteq \exists r.\{b\}$. In this way, we get the same representation of asserted facts from databases as we have for terminological statements from ontologies, like GO or ChEBI. This simplifies the interface between KG, box embeddings, and GNN, introduced in Sections 3.1 and 3.3.

The knowledge graph is created from data in SGD, where the information is defined using terms from several different ontologies.A high level overview of the graph, showing how different node types are connected, can be seen in Figure 2a. Figure 2b shows examples of the hierarchies classes instantiating these nodes are represented in.

The GO-annotations in SGD are naturally described by classes in the Gene Ontology and relations from the OBO Relations Ontology, which are specified in the database. Phenotypes are described using terms from APO where a phenotype is represented by an '`observable`', for example '`heat sensitivity`', and possibly a '`qualifier`', for example '`increased`'. We

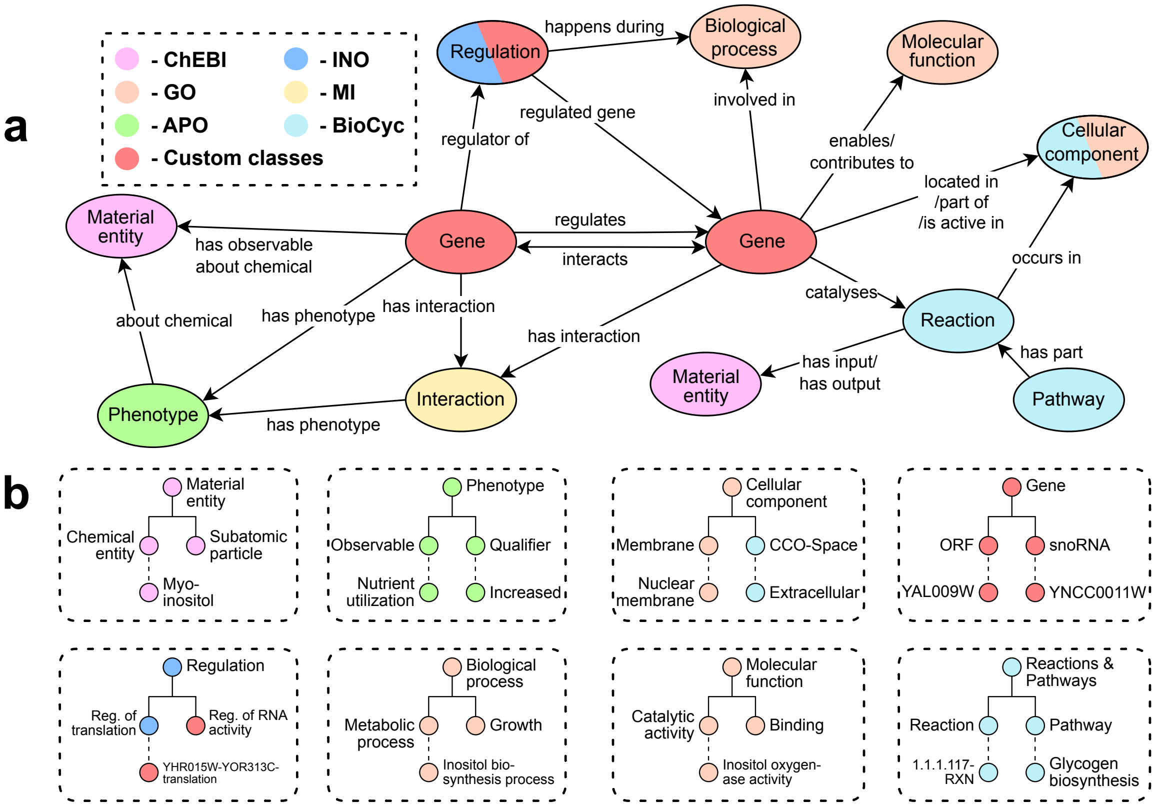


Figure 2: An overview of the different types of classes and how they are connected in the knowledge graph is shown in **a**. The color of the nodes specifies where the classes are defined. **b** shows examples from the hierarchies defining classes in the domains introduced in Section 3.3.

represent the phenotype as the subclass of the intersection of these two types of classes, and phenotypes are linked to genes using the RO relation '`has phenotype`'. Some phenotypes describe observables related to specific chemicals, in such cases the chemical class in ChEBI is linked with a custom relation, '`aboutChemical`'. To form a closer connection between genes and chemicals related to phenotypes, which proved useful for downstream tasks (see Section 3.3), a link specific to the type of observable was added between the gene and the chemical. An example of how this is implemented in description logic can be seen in (14) in Appendix A.

Gene regulation in SGD is a directed relationship between two genes that can be positive, negative, or unspecified, and of different types, for example, regulation of protein activity or expression. In some instances, a biological process from GO specifies under which conditions the regulation occurs. We introduce custom relations describing regulation type and direction, which we use to link the two genes in the graph. When a biological process is specified we also link the genes to a gene-specific subclass of the '`regulation`' class from INO, which in turn is linked to the GO-term. The description logic implementation of such a regulation can be seen in (15) in Appendix A.

Interactions between genes are represented as undirected relationships, which may also be associated with a phenotype observed alongside the interaction. Similarly to regulation this is modelled as a link between the involved genes and a gene specific subclass of either a `protein-protein interaction` from INO or a `genetic interaction` from MI, which is linked to the phenotype.

Beyond the data from SGD we have also included information about reactions and pathways from BioCyc, which uses its own controlled vocabulary. In the graph, reactions are linked to their input and output chemicals, as well as, when specified, genes they are catalysed by and locations in the cell where they take place. We link pathways to their involved reactions, as well as to the compounds that are consumed and produced.

3.3. Prediction models

To demonstrate the usefulness of our KG and how the box embedding method described in Section 3.1 can be used in practice, we train GNNs to predict phenotypic traits in *S. cerevisiae*. We use data from Costanzo et al. (2016) where cell growth is measured when pairs of genes are deleted (digenic deletions) from the genome. By comparing this growth to that of cells with no gene deletions, a fitness score can be determined, describing the impact of deleting the two genes. A subset of this data, grown under the same standard experimental conditions (30°C), is used to train our model. This results in a data set with 10,085,183 examples of deleted gene pairs and a corresponding fitness. Note that the genetic interaction relation from SGD describes a similar phenomena, often derived from the same dataset. These relations are thus removed from the graph before training to avoid data leakage.

We divide our classes in the KG into eight different domains, seen in Figure 2b, for which separate embeddings are found. These splits generally align well with the ontologies the classes are from, or disjoint branches in the same ontology. The reasoning behind this is that these domains represent non-overlapping concepts, so not much is gained by representing them in the same embedding space. Doing this also allows us to reduce the dimensionality of the embedding space and vary it depending on the number of classes in the domain. Adding reverse links to the graph to allow for message passing in both directions results in 204 different types of links. After removing infrequent (<1,000) and overlapping edges we end up with 72 different types of links, and nodes belonging to eight different domains, used for prediction.

Prior shallow node embeddings were trained using the *overlap* losses in (7) and (8), representing the classes as Gumbel boxes. Large boxes were penalised using the regularisation in (9) and negative examples are generated by drawing random classes, $\bar{p}$, that are not in the set of parents to c, i.e. not in $\{p|c \sqsubseteq^* p\}$, to better discriminate between classes. Parameters used to train the box embeddings and the dimensions of the different domains are reported in Appendix C.1.

For predicting the gene-pair fitness we use a heterogeneous GNN followed by a fully connected neural network, an overview of the architecture can be seen in Figure 3a. The GNN used is based on the max-aggregated GraphSAGE embedding algorithm introduced by Hamilton et al. (2017), which has previously shown promise in KG- and network-related prediction tasks (Ma et al., 2023; Syama et al., 2023; Vretinaris et al., 2021). In our

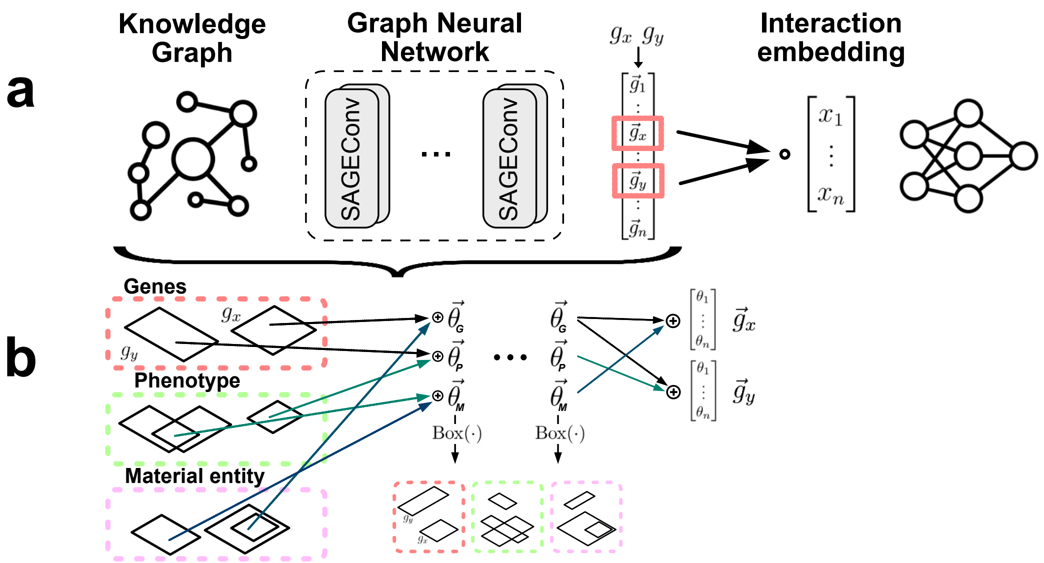


Figure 3: An overview of the system predicting the fitness when deleting pairs of genes is shown in **a**. **b** shows how classes in the different domains are represented by boxes and how information is aggregated in the GNN, as well as how the node embeddings throughout the network are interpreted as boxes using the box transformation in (2). Arrows from the boxes represent learnable SAGE modules, different for each `source domain-edge-target domain` type. The fitness is predicted from the Hadamard product of the embeddings of the two deleted genes.

heterogeneous setting, each source-edge-target type has its own SAGEConv-module, whose outputs are combined using mean aggregation to create the node embeddings from each layer. The dimensionality of the SAGE message-passing modules is adjusted based on the type of source-edge-target triple, and specifically varies with the number of edges directed toward the target domain. Domains with a high degree of incoming connectivity, such as 'Material entities' or 'Genes', are assigned higher dimensional feature spaces. The resulting class embeddings, generated by the GNN, capture aggregated neighbourhood information. By applying box-losses as introduced in Section 3.1, the training will also aim to represent the class hierarchy as box embeddings. Figure 3b illustrates how information is propagated from the initial box embeddings across different domains to the gene embeddings. To predict the fitness of a gene pair, we compute the Hadamard product of their embedding vectors and feed the result into a fully connected neural network, which outputs a real-valued prediction.

We train the model by minimising, using the Adam optimiser, the following loss function,

$$\mathcal{L} = \mathcal{L}_{MSE}(y, \hat{y}) + \alpha(\mathcal{L}_{\sqsubseteq} + \beta\mathcal{L}_{\sqsubseteq}^{-}) + \lambda \left\| w \right\|^2 \tag{11}$$

$\mathcal{L}_{MSE}$ denotes the mean squared errors of the fitness predictions and $\mathcal{L}_{\sqsubseteq}$ and $\mathcal{L}_{\sqsubseteq}^{-}$ are defined in (5-8). α and β are weights determining the impact of the semantic loss, measuring how well the box embeddings represents the class hierarchies. λ controls regularisation of the parameters in the network.

The models are trained and evaluated using 10-fold cross validation where the data split is based on the genes. Any gene pairs that include genes from both the training and validation sets are discarded. This ensures that no pairs involving validation-set genes are seen during training, so the learned representations of genes in the training set do not influence the predictions being evaluated.

Hyperparameters, including learning rate, regularisation (λ), the depth and width of the fully connected neural network, the depth of the GNN, and embedding dimensions throughout the GNN, are tuned using Bayesian optimization. For the embedding dimensions, values are doubled for the most common target domains and halved for the least common. Tuning is performed on a separate data split from the one evaluated in Section 4.1. This tuning is done for a model using box embeddings as prior node representations, but without semantic loss during training, the same parameters are then used for all evaluated models. The used hyperparameters are reported in Appendix C.2.

3.4. Learning GNN box embeddings without a prediction task

To demonstrate how the semantic loss can be used to train box embeddings in the absence of a prediction task, we use a simpler ontology. Using the family tree of the British royal family[3], we define terms to describe basic parental and spousal relationships, as well as place of birth, and create ABox statements for corresponding facts from the database. An overview of the included concepts and properties can be seen in Table 3 in Appendix B. Following the same methodology as Section 3.2 we rewrite role and class assertions as TBox axioms. `subClassOf` relations are used as the positive examples for $\mathcal{L}_{\sqsubseteq}$, and negative examples for $\mathcal{L}^{-}_{\sqsubseteq}$ are taken from disjoint classes, inferred from the disjointness axioms $\texttt{Person} \sqcap \texttt{Country} \sqsubseteq \bot$ and $\texttt{Man} \sqcap \texttt{Woman} \sqsubseteq \bot$, as well as for randomly drawn pairs $(\texttt{Individual}_1 \sqcap \texttt{Individual}_2 \sqsubseteq \bot)$ to distinguish between individuals in the graph.

As for the models described in Section 3.3, the GNN is constructed from SAGEConv modules for each edge type. For the purposes of demonstration, we learn embeddings in two dimensions so they can easily be visualised. We simultaneously train initial box embeddings (randomly initialised) and a GNN by minimising, again using the Adam optimiser, the loss function,

$$\mathcal{L} = \mathcal{L}_{\sqsubseteq} + \beta\mathcal{L}^{-}_{\sqsubseteq} + \lambda R, \tag{12}$$

where R is the regularisation loss from (10), penalising small boxes with an l_0 of 1, and β and λ are weights determining the impact of the negative semantic loss and regularization loss respectively. Note the absence of the mean squared error term present in the prediction task above. The regularisation term was included as disjointness tended to make boxes extremely small along one or more dimensions during training rather than move position in the space. The negative semantic loss is decomposed into

$$\mathcal{L}^{-}_{\sqsubseteq} = \mathcal{L}^{-}_{\sqsubseteq\text{data}} + \gamma\mathcal{L}^{-}_{\sqsubseteq\text{random}}, \tag{13}$$

which allows us to tune the contribution of the randomly selected disjointness axioms. For each loss type (*distance* or *overlap*) we separately tuned the hyperparameters, and the used values are reported in Table 6 in Appendix C.3.

[3] Obtained from http://kingscoronation.com/wp-includes/images/Queen_Eliz_II.ged in .GED format

3.5. Link evaluation

A potential application for the semantic losses defined above, in addition to the training of box embedding models for quantitative prediction tasks, is to rank proposed revisions to a knowledge graph based on the resultant changes to embeddings and losses. We test this by adding single edges to the graph according to the following scheme.

Say that for a given ontology we construct a graph $\mathcal{G} = (V, E)$, where each edge vertex $v \in V$ is a class in the ontology and each edge $e \in E$ represents a role assertion. Following the methodology outlined above, we use $\mathcal{G}$ as the basis for a GNN, and train box embeddings and the weights of the GNN using the semantic loss. Introducing new role assertions to the graph results in graph $\tilde{\mathcal{G}}$, and passing the prior box embeddings through the GNN using these additional edges will change the final box embeddings. We calculate the distance between the original learned box embeddings and those after the changes to the graph, giving us a measure of the change to the embeddings from the graph revision. This process is described in Algorithm 1.

Algorithm 1: Link evaluation algorithm

```
Train embedding parameters $\theta_l$ on $\mathcal{G}_{\text{train}}$
$\delta \leftarrow \emptyset$
foreach $e \in E_{test}$ do
    $\tilde{\mathcal{G}} \leftarrow \mathcal{G}_{\text{train}} \cup \{e\}$
    $B \leftarrow \text{Box}(\text{GNN}_\theta(\mathcal{G}_{\text{train}}))$
    $\tilde{B} \leftarrow \text{Box}(\text{GNN}_\theta(\tilde{\mathcal{G}}))$
    $\delta \leftarrow \delta \cup (e, \langle B, \tilde{B} \rangle)$ # distance between the generated box embeddings
end
Sort $\delta$ to get ranked revisions
```

To evaluate this proposed method, we split the edges in the graph based on the royal family tree dataset into training and test data using a 70:30 training and test split, stratified by relation type. This results in a training graph $\mathcal{G}_{\text{train}} = (V, E_{\text{train}})$ and a test graph $\mathcal{G}_{\text{test}} = (V, E_{\text{test}})$. The embeddings and GNN are trained as per Section 3.4. We run Algorithm 1, going through each edge in the test data. We also perform the same steps with randomly generated edges with source and target drawn from the same classes as the test edge, and with completely randomly drawn source and target nodes. The distance metric used is defined in (4).

4. Results

4.1. Gene deletion fitness prediction

In Table 1 we present the coefficient of determination (R^2) for different versions of the model described in Section 3.3. We evaluate a model without any information from class hierarchies, a model with the prior node embeddings in box form, and models using both prior node embeddings and the semantic loss in (11). Both the *overlap* and *distance* version of the losses are evaluated. The model not using any hierarchy information learns shallow embeddings specifically for this task to represent the nodes in the KG. For the two models with the semantic loss, we apply it to all domains except the one embedding the genes, since the class hierarchy in this domain is not considered informative. The gene hierarchy builds on a rudimentary SGD gene categorisation, offering very little information, with over 90% of genes falling into the same category. The loss is applied to all generated embeddings, including the initial embeddings, whose weights are also adjusted during training.

We also compare to a Light Gradient Boosting Machine (LightGBM) (Ke et al., 2017) on the instantiation of the phenotype information from the KG. The phenotypes describe observable characteristics of the genes and is the part of the KG we expect to be most informative for this task (further support for this is seen when considering feature importances for the GNN, mentioned in Section 4.2, which are dominated by phenotypes). The instantiation of the phenotypes is sparse with 2680 features.

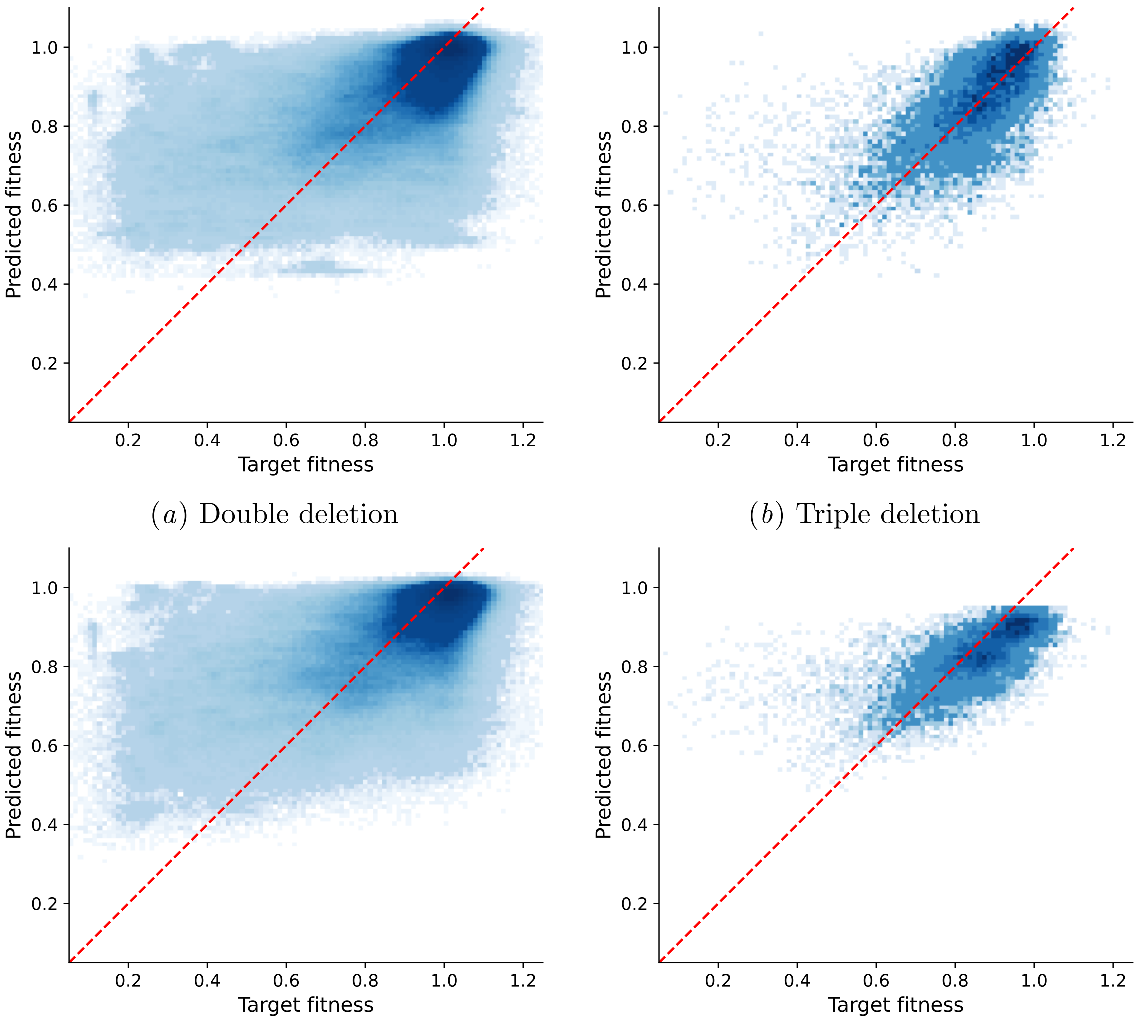
Table 1: Results from 10-fold cross-validation of digenic deletion fitness. The GNN without box embeddings learns task-specific shallow embeddings as the prior node representations. The other three GNNs uses pre-trained box embeddings and the semantic loss in (11) is applied to two of them. All GNN models share the same architecture. The instantiation model uses a sparse feature matrix with non-zero entries for phenotype annotations from the KG. Significant pairwise differences are indicated by ↑ and ↓ (p<0.05, paired t-test).

	Description	Mean R^2	SD	a	b	c	d	e
a	Instantiations + LightGBM	0.211	0.022	-	↓	↓	↓	↓
b	GNN without box embeddings	0.329	0.043	↑	-	↓	↓	↓
c	GNN with prior box embeddings	0.360	0.043	↑	↑	-		↓
d	GNN with prior box embeddings + $\mathcal{L}_{overlap}$	0.368	0.038	↑	↑		-	
e	GNN with prior box embeddings + $\mathcal{L}_{distance}$	**0.374**	0.042	↑	↑	↑		-

From the results it is clear that the GNN generates gene embeddings which can be used for predicting this fitness to a reasonable degree, given the amount of noise typically present in biological measurements (Li et al., 2021). Using the box embeddings to represent classes rather than learning them from scratch results in a significant (p<0.05, paired t-test) improvement. Enforcing the hierarchical class structure through the semantic loss throughout the model improves the results further, the model trained with the *distance*-loss performs significantly (p<0.05, paired t-test) better than the models not using the semantic loss. The instantiated phenotype information seems to be somewhat useful for prediction, but is not as informative as the full KG.

The parity plots for the predictions are shown in Figure 4(a) and 4(c). From this we can see that most double gene deletions do not have major impact on the fitness. We can also see a clear shrinkage effect where the model mispredicts extreme values, especially deletions with low fitness are overestimated. Comparing the predictions from the model trained with the semantic loss we can see that they in general are rather similar, but that the semantic loss model seems to have fewer large underestimations.

Figure 5 show the semantic losses in the different domains, introduced in Figure 2b, for the best performing model, using $\mathcal{L}_{distance}$. A similar pattern is observed for all domains where both the positive loss for the first layer (the pretrained box embeddings) and the negative loss for the second layer is low and fairly constant. The positive losses for the second layers decreases across all domains throughout training, while the negative loss for the first layer does not change much and is substantially higher than the others. This suggests that,



(*a*) Double deletion

(*b*) Triple deletion

(*c*) Double deletion with semantic loss

(*d*) Triple deletion with semantic loss

Figure 4: Parity plots for double, (*a*) and (*c*), and triple, (*b*) and (*d*), gene deletions. (*a*) and (*b*) shows the parity plot for the model using box embeddings as prior node representations only, while (*c*) and (*d*) shows the predictions from a model also trained with the *distance*-based semantic losses, $\mathcal{L}_{distance}$. For the double deletion, the predictions from all validation sets in the cross validation are shown.

even though they result in a significant improvement in prediction performance, the initial box embeddings has a lot of overlap between classes. On the other hand, the embeddings generated by the GNN discriminates very well between classes, already at the first epoch and the hierarchical structure is learnt throughout training.

To evaluate our model on a slightly modified version of the original task, we use data from Kuzmin et al. (2018), who performed a study similar to the one used for training our models, but focused on trigenic deletion fitness. This dataset comprises a total of 15,095 triple deletion datapoints. For this we use one model trained on the full dataset

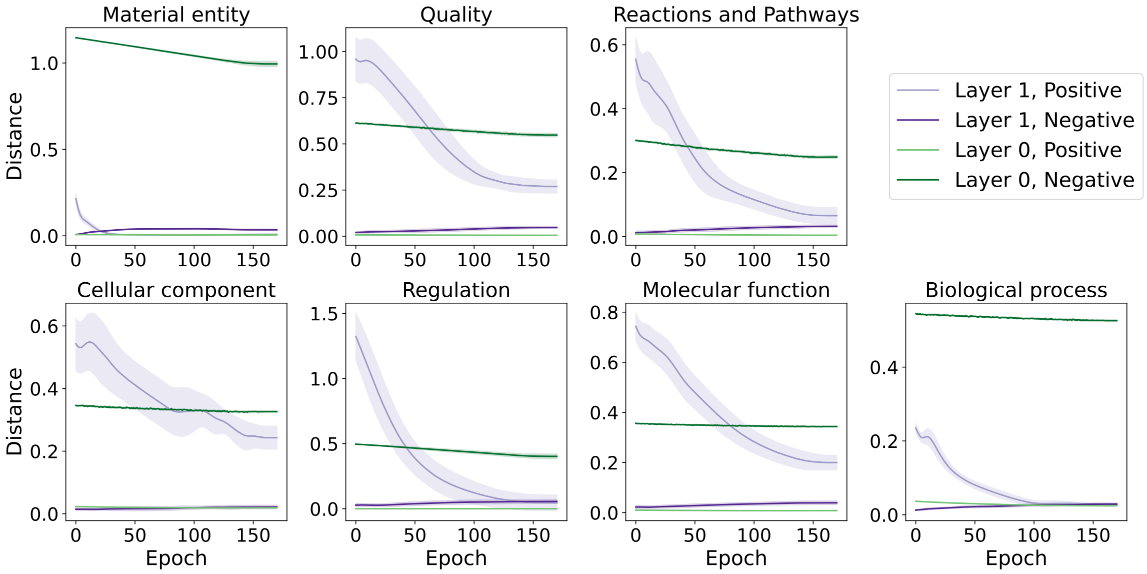
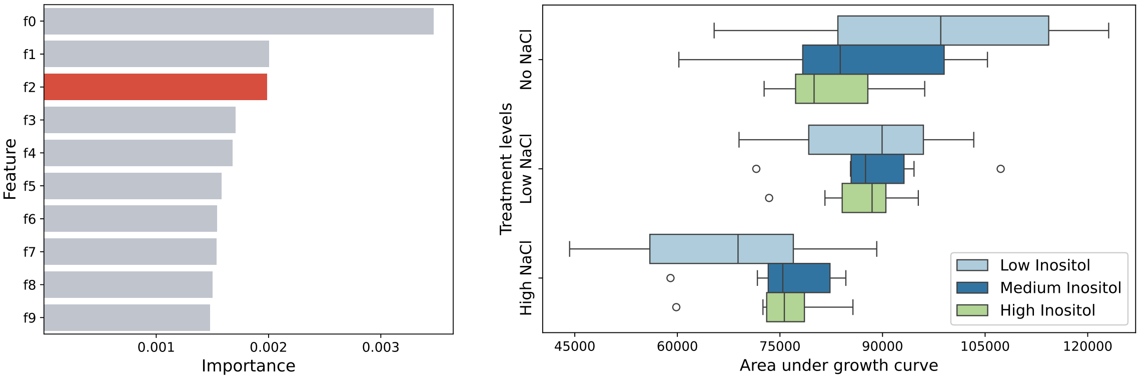


Figure 5: Average $\mathcal{L}_{distance}$ and $\mathcal{L}^{-}_{distance}$ losses per class for the different domains in the KG, during training of the best performing model in Table 1. The line is the average loss across the 10 folds and the shaded area shows ± one standard deviation. Note that this loss is not applied to the gene-domain, since its class hierarchy is not deemed to be informative.

from Costanzo et al. (2016), but instead perform the Hadamard product between the three involved genes. Notably we achieve an R^2 of 0.380 for a model using box embeddings as prior node representations, and 0.415 for a model using the same prior node embeddings, but trained with the *distance*-based semantic loss. These values are slightly higher than the average performance observed in the cross-validation of digenic deletions. The parity plots, seen in Figure 4(*b*) and 4(*d*), shows promise in generalising to a new task. Again, the prediction patterns for the two models look similar, but the model trained with the semantic loss does not predict as high fitness. An important note on this experiment is that, unlike the double deletion experiment, the individual genes making up the triple deletions are now seen as parts of double deletion examples in training.

4.2. Model interpretation and experimental evaluation

Since our predictions stem from a KG where each link holds domain-relevant meaning, we explore patterns among important edges. We apply the *input* × *gradient* method (Shrikumar et al., 2017) via Captum (Kokhlikyan et al., 2020) to attribute edge importance for predictions by a model using prior box embeddings (model *c* in Table 1). By multiplying the individual importance scores for the involved genes we get a measure of the importance of co-occurring edges. Summing these values for all predictions gives a global importance of such edge-pairs. This can be interpreted as the impact of the interaction between the two traits on the fitness.



(a) Feature importances

(b) AUC growth boxplots

Figure 6: An overview of the selection and results of the experiment we performed. (a) shows the highest ranked importances of edge-pairs and the pair selected for the experiment, nutrient utilisation of inositol and stress resistance to NaCl, is highlighted in red. *f0* and *f1*, which have a higher assigned weight, are discarded due to safety and lab constraints as it involves the chemical bleomycin. (b) Box plot showing the distribution of AUC for all of the experimental conditions tested. Inositol supplementation significantly impacts growth dynamics in high doses ($p < 0.05$). NaCl stress changes the impact of inositol in a dose dependent manner, suggesting an interactive effect ($p < 0.05$).

To identify patterns corresponding to viable experiments for standard lab setups we filtered for edges related to nutrient utilisation phenotypes. A more detailed description of the filtering process can be found in Appendix D.1. The top ten most important edge pairs are shown in Figure 6(a) and detailed in Appendix D.2. The highest-weighted, safely testable pair was selected and highlighted in red in Figure 6(a), linking one of the involved genes to inositol (vitamin B8) utilisation and the other to NaCl stress resistance, suggesting a potential interaction between these traits.

To experimentally test this hypothesis, a perturbation experiment was performed in an automated laboratory cell (Williams et al., 2015), in which inositol and NaCl was supplied in a range of concentrations, details about the experimental design and cultivation methods can be found in Appendix D.3. An $\Delta ino1$ mutant (INOsitol requiring) was used for all subsequent experiments, as it is unable to synthesise inositol on its own, ensuring that any intracellular accumulation was acquired only through transport from the media. The growth dynamics of the cells in the different experimental conditions were summarised with the area under curve (AUC) of the growth curves, providing a single-valued measure of the biomass accumulation over the course of the experiment. The full growth dynamics can be seen in Figure 9 in Appendix D.4 and summarising boxplots are shown in Figure 6(b). Statistical testing for interaction effects was done with a Gaussian generalised linear model (GLM), further details can be found in Appendix D.4.

These empirical results, seen in Figure 6(b) and Table 2, indicate a significant interaction between inositol supplementation and induced NaCl stress, verifying that the proposed edge-interactions are consistent with experimental data. Specifically, supplementing with inositol

Table 2: Estimated parameters from the GLM examining the effects of myo-Inositol-supplementation and NaCl treatment on growth dynamics. The table presents coefficient estimates, p-values and confidence intervals for the main effects and interaction terms. Significant interactions indicate that the effect of myo-inositol supplementation changes depending on treatment levels. The two highlighted rows indicate the significant interaction effect.

	Coefficient ($\times 10^3$)	Confidence interval ($\times 10^3$)	p-value
Intercept	97.29	[88.20, 106]	0.000
Medium inositol	-11.84	[-24.7, 1.02]	0.071
High inositol	-14.56	[-27.9, -1.24]	**0.032**
Low NaCl	-9.50	[-22.4, 3.37]	0.148
High NaCl	-30.61	[-43.5, -17.8]	**0.000**
Medium inositol × Low NaCl	13.11	[-5.40, 31.6]	0.165
High inositol × Low NaCl	13.38	[-5.45, 32.2]	0.164
Medium inositol × High NaCl	20.92	[2.41, 39.4]	**0.027**
High inositol × High NaCl	22.64	[3.40, 41.9]	**0.021**

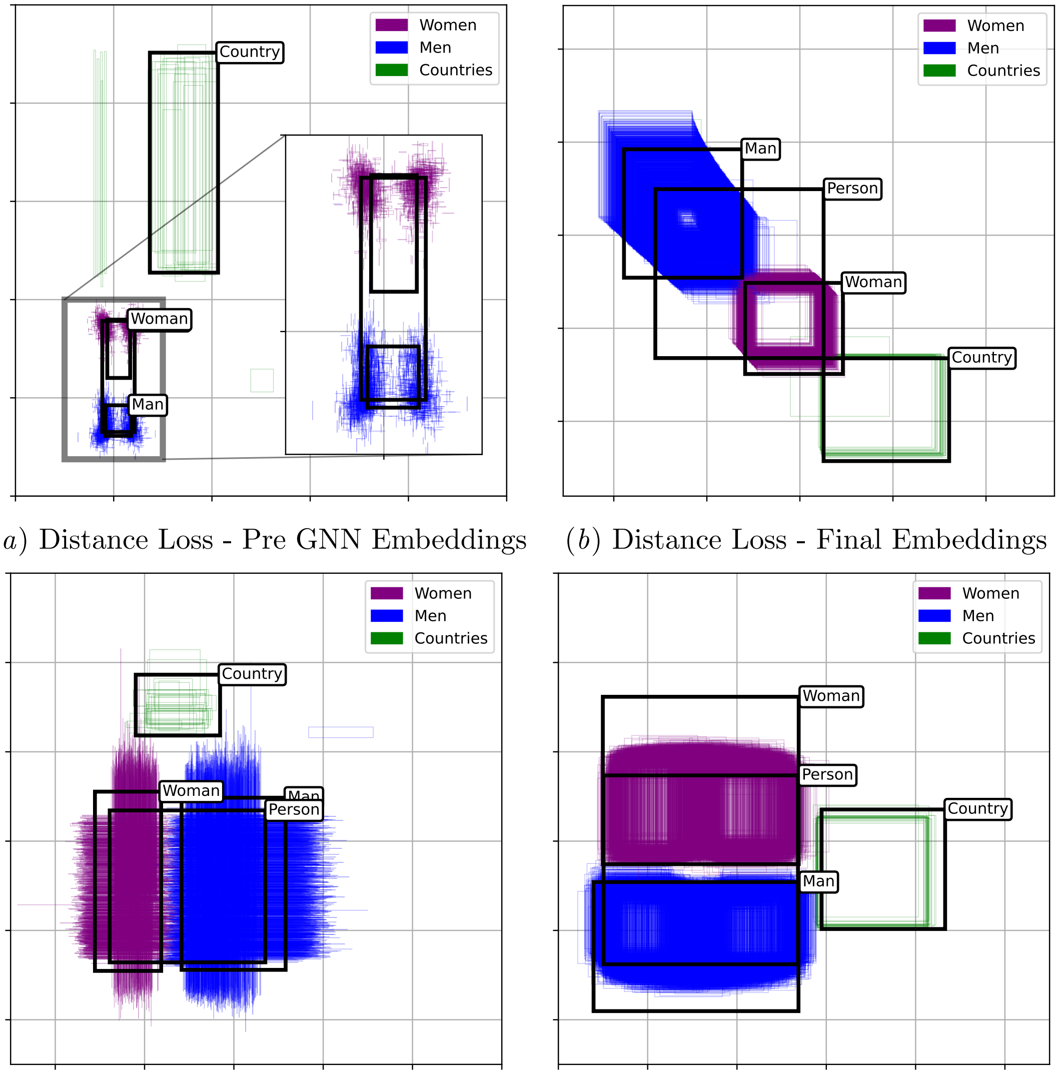
rescued cells from NaCl-induced stress, indicating that inositol availability enhances their ability to withstand salt stress. Inositol has previously been implicated in biosynthesis and integrity of cell membranes (Culbertson and Henry, 1975). Since NaCl can disrupt osmotic balance, enhanced membrane stability is likely to have a protective effect for the cells.

4.3. Demonstration of royal family box embeddings

For the knowledge graph constructed from the royal family tree dataset, after the hyperparameter search for *distance*-loss and *overlap*-loss, we constructed final box embeddings in two dimensions using a GNN with one message passing layer. In Figure 7 we plot the box embeddings for each loss, before input into the GNN and then the final embeddings. We see clearly that the GNN is performing a transformation of the prior embeddings. Furthermore, both losses result in learned embeddings that capture semantic concepts from the KG, in particular that `Man` and `Woman` are disjoint, yet both under the `Person` class, and that each of these is disjoint from `Country`. Both losses also responded to the randomly drawn negative loss contributions ($\mathcal{L}^{-}_{\sqsubseteq \text{random}}$) in attempting to separate individuals, though this is more evident with the *overlap*-loss.

Comparing Figures 7(a) and 7(b), we see that prior to being passed through the GNN, the boxes corresponding to different `Woman` and `Man` individuals are clustered around the corners of the superclass boxes. An intuitive explanation for this phenomenon is that they are being "pulled" into the box for their gender, but once they are in the box there is no longer any signal affecting their position. The loss from randomly drawn disjointness axioms keeps them somewhat separated. Having passed through the GNN, the boxes take on dramatically different shapes and relative positions. The semantic loss of these embeddings is low, but individuals of the same superclass have now very similar embeddings. By contrast, comparing Figures 7(a) and 7(b), we see that the shape of the boxes is long and thin, minimising the volume of each intersection (in this case the Bessel volume, see Section 2 and

Dasgupta et al. (2020) for details). And the final embeddings do not differ from the prior embeddings to the same extent as with distance loss. The overall structure of the parent classes is broadly the same, albeit rotated. With both losses, the box volume increased after passing through the GNN, and the difference between height and width decreased.



(*a*) Distance Loss - Pre GNN Embeddings (*b*) Distance Loss - Final Embeddings

(*c*) Overlap Loss - Pre GNN Embeddings (*d*) Overlap Loss - Final Embeddings

Figure 7: Learned box embeddings in two dimensions for the royal family tree dataset. 7(*a*) and 7(*b*) box embeddings prior to input into GNN; 7(*b*) and 7(*d*) show final embeddings for distance and overlap loss respectively.

4.4. Evaluation of graph revisions

The median distances and loss changes for individual edge revisions to the graph were small, and these measures overall had very large variance. However there were signs in these data which suggest that using these values could be used as a tool to rank candidate revisions to a knowledge base. For the `birthPlace` relation, the distance of graphs constructed from the test data was significantly smaller (p <0.05, student's t-test) than both completely randomly drawn data and constrained randomly drawn data. However, for the relations between `Person` entities, there was common pattern. The graphs constructed from the test data had a lower distance to the original embeddings when compared to the completely randomly drawn edges. However when constraining the random draw to appropriate classes, in this case either `Man` or `Woman`, the distances were smaller still than the test data. This effect can be seen in Figure 8.

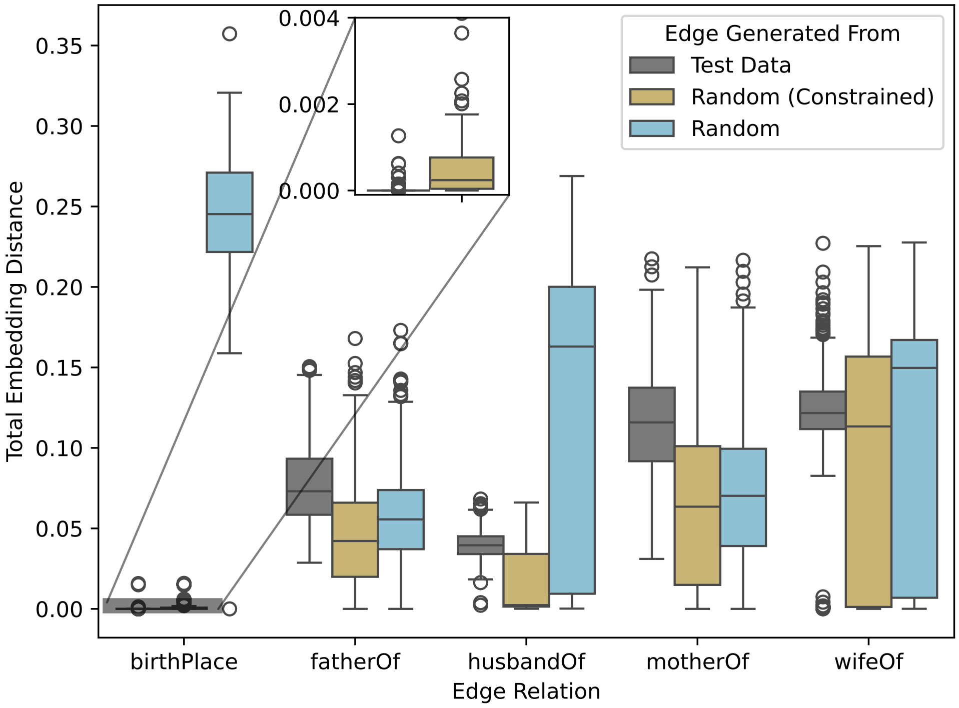


Figure 8: Distribution of distances of box embeddings generated from revised graphs $\tilde{\mathcal{G}}$ to the original embeddings learned from $\mathcal{G}$, shown by relation type. (The method for calculating these differences is described in 3.5). Completely randomly drawn edges have a higher mean difference than the test edges (truth data), though for all edges the lowest mean difference was for constrained randomly drawn edges.

5. Discussion

In this work we have presented a method generating KG embeddings using GNNs, taking hierarchical class information as well as graph structure into account. We do this by introducing a semantic loss term to the training acting on box transformations of the node embeddings. We have seen that it can be used on its own to generate KG embeddings adhering to subsumptions defined in ontologies, but more importantly this method shows promise when used together with another, task-specific prediction loss.

We observed this effect when predicting digenic deletion fitness from a KG describing *S. cerevisiae* genes which we constructed. While the predictive R^2 of 0.374 may seem low, biological data is inherently noisy, and even replicating experiments is challenging (Roper et al., 2022). Moreover, our model predicts quantitative outcomes from high-level qualitative information. What is more interesting is how the prediction performance was improved by introducing more hierarchical information to the models. One explanation we envision for the improved performance is that enforcing the class hierarchies has a regularising effect, while also providing semantic grounding for the modelled concepts. The improved performance could also suggests that the ontologies used, for example ChEBI and GO, are, at least somewhat, good models of the domains.

KG embeddings that, at least to a large extent, adhere to their underlying ontologies can potentially be used for several tasks, even if they were trained with a particular problem in mind. Our trigenic gene deletion experiments are one example of applying the model slightly outside its original domain. The increased performance in this task compared to the digenic deletion is likely, at least partly, due to the individual genes involved no longer being unseen during training. The fitness will depend heavily on the traits of the individual genes, which will be better represented for genes in the training data. The embeddings could potentially be applied to a broader range of tasks, such as GO annotation of genes, which is typically addressed by integrating multiple knowledge sources (Merino et al., 2022).

We have not utilised any sequence information for our fitness predictions, despite it being the most informative data about genes and fully available for *S. cerevisiae*. Representing the initial gene embeddings as some encoding of their sequence would provide richer and more meaningful gene embeddings and most likely result in better predictions. However, our current setup will put more emphasis on using the information in the KG as the basis of the predictions. In this way this helps demonstrate both the knowledge in the KG and the usefulness of our embedding method.

The capability of making predictions from qualitative facts enabled interpretability techniques to guide experiment selection, underscoring the value of structured data representation and computational methods in accelerating research. Our edge filtering for viable experiments introduces biases regarding the type of hypotheses generated. Leveraging large language models could be one approach to automatically refine this selection and reveal overlooked experiments.

We suggested two different loss functions for learning box embeddings. The first is based on the volume of overlap between boxes, rewarding overlap for subclasses, and penalising overlaps in the case of disjointness. The second was based on the distance from the boxes fulfilling the subsumption axioms. Studying the embeddings of the family tree in Figure 7, the embedding learnt through the *overlap* loss using Gumbel boxes seems to have better captured the semantics of the ontology. Boxes for `Man` and `Woman` are aligned along one dimension and the box for `Country` is placed orthogonally to this along the other dimension. The embedding learnt through the *distance*-based loss instead places the boxes for each class along a diagonal, which here means in the embedding space `Woman` is more similar to `Country` than `Man` is, which is not faithful to the semantics of the ontology. Another property of the embeddings learned in this example is the variation in position among instances, which is better for the *overlap*-loss. Figure 7(*b*) also shows some variation, primarily among instances of `Man`, but the training was less stable, putting too high weight on the negative examples.

Interestingly, when class hierarchies were enforced in the gene deletion fitness predictions, the *distance*-based loss yielded better predictive performance. One potential explanation for this finding is that the *distance*-based loss may be particularly well suited as a semantic loss to complement a task-specific loss. By introducing a semantic loss component to our total loss, we of course want to capture how faithfully a given embedding adheres to the semantics of the source ontology. But to have smooth training, a desirable feature of a semantic loss measure is that its gradients are informative when the constraints are not fulfilled. This is exactly what $\mathcal{L}_{distance}$ does. To obtain useful gradients with $\mathcal{L}_{overlap}$, one can for example use Gumbel boxes as in Dasgupta et al. (2020), but a result of the introduced smoothing is that losses can remain nonzero even when the semantic constraints are fulfilled. With another loss term primarily guiding the training, in this case $\mathcal{L}_{MSE}$, the issue of $\mathcal{L}_{distance}$ not discriminating between classes that we observed with the family tree dataset is not as pressing, as the primary loss will likely also push towards being able to discriminate.

Our proposed method for evaluating link revisions to a KG can be seen as an interesting application and direction for future research. Evaluating only the distance in the generated embeddings is, in this setting, not enough to discriminate between true and random edges. It could possibly work better for a more heterogeneous graph, with a richer class hierarchy. A measure of distance, combined with the semantic losses, could represent a measure of surprise. In a scientific discovery context, surprise can be used as part of creating and evaluating hypotheses, opening up opportunities for box embeddings to be used in this context.

6. Conclusion

In this work we have presented a method generating KG embeddings using GNNs, taking hierarchical class information as well as graph structure into account. We show that enforcing the class hierarchies as semantic losses throughout the model can help predictive performance while also producing internal representations which better correspond to our knowledge of the domain. This is demonstrated on a KG we have created from publicly available data about the yeast *S. cerevisiae*. Based on this KG we can, not only predict biological measurements, but also use interpretability tools to form a hypothesis about phenotype interactions. One such hypothesis was tested and supported by performing a biological experiment, uncovering an association between inositol utilisation and NaCl stress. This illustrates how models with semantic grounding can help in scientific discovery.

The code and data for this project are available at `https://github.com/filipkro/kg-box-emb`.

Acknowledgments

The computations were enabled by resources provided by the National Academic Infrastructure for Supercomputing in Sweden (NAISS), partially funded by the Swedish Research

Council through grant agreement no. 2022-06725. This work was supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Alice Wallenberg Foundation, the UK Engineering and Physical Sciences Research Council (EPSRC) [EP/R022925/2, EP/W004801/1 and EP/X032418/1], the Chalmers AI Research Centre, and Swedish Research Council Formas [2020-01690].

References

Robert Arp, Barry Smith, and Andrew D. Spear. *Building Ontologies with Basic Formal Ontology*. The MIT Press, 2015. ISBN 978-0-262-52781-1.

Michael Ashburner, Catherine A. Ball, Judith A. Blake, David Botstein, Heather Butler, J. Michael Cherry, Allan P. Davis, Kara Dolinski, Selina S. Dwight, Janan T. Eppig, Midori A. Harris, David P. Hill, Laurie Issel-Tarver, Andrew Kasarskis, Suzanna Lewis, John C. Matese, Joel E. Richardson, Martin Ringwald, Gerald M. Rubin, and Gavin Sherlock. Gene ontology: tool for the unification of biology. *Nature Genetics*, 25(1): 25–29, 2000. ISSN 1546-1718. doi: 10.1038/75556. URL https://www.nature.com/articles/ng0500_25. Number: 1 Publisher: Nature Publishing Group.

Antoine Bordes, Nicolas Usunier, Alberto Garcia-Durán, Jason Weston, and Oksana Yakhnenko. Translating embeddings for modeling multi-relational data. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, volume 2 of *NIPS'13*, pages 2787–2795. Curran Associates Inc., 2013.

Daniel Brunnsåker, Filip Kronström, Ievgeniia A Tiukova, and Ross D King. Interpreting protein abundance in saccharomyces cerevisiae through relational learning. *Bioinformatics*, 40(2):btae050, 2024. ISSN 1367-4811. doi: 10.1093/bioinformatics/btae050. URL https://doi.org/10.1093/bioinformatics/btae050.

Daniel Brunnsåker, Alexander H. Gower, Prajakta Naval, Erik Y. Bjurström, Filip Kronström, Ievgeniia A. Tiukova, and Ross D. King. Agentic AI integrated with scientific knowledge: Laboratory validation in systems biology, 2025. URL https://www.biorxiv.org/content/10.1101/2025.06.24.661378v3. ISSN: 2692-8205 Pages: 2025.06.24.661378 Section: New Results.

Tejas Chheda, Purujit Goyal, Trang Tran, Dhruvesh Patel, Michael Boratko, Shib Sankar Dasgupta, and Andrew McCallum. Box embeddings: An open-source library for representation learning using geometric structures, 2021. URL http://arxiv.org/abs/2109.04997.

Maria C. Costanzo, Marek S. Skrzypek, Robert Nash, Edith Wong, Gail Binkley, Stacia R. Engel, Benjamin Hitz, Eurie L. Hong, J. Michael Cherry, and the Saccharomyces Genome Database Project. New mutant phenotype data curation system in the saccharomyces genome database. *Database: The Journal of Biological Databases and Curation*, 2009: bap001, 2009. ISSN 1758-0463. doi: 10.1093/database/bap001.

Michael Costanzo, Benjamin VanderSluis, Elizabeth N. Koch, Anastasia Baryshnikova, Carles Pons, Guihong Tan, Wen Wang, Matej Usaj, Julia Hanchard, Susan D. Lee, Vicent Pelechano, Erin B. Styles, Maximilian Billmann, Jolanda van Leeuwen, Nydia van Dyk, Zhen-Yuan Lin, Elena Kuzmin, Justin Nelson, Jeff S. Piotrowski, Tharan Srikumar, Sondra Bahr, Yiqun Chen, Raamesh Deshpande, Christoph F. Kurat, Sheena C. Li, Zhijian Li, Mojca Mattiazzi Usaj, Hiroki Okada, Natasha Pascoe, Bryan-Joseph San Luis, Sara Sharifpoor, Emira Shuteriqi, Scott W. Simpkins, Jamie Snider, Harsha Garadi Suresh, Yizhao Tan, Hongwei Zhu, Noel Malod-Dognin, Vuk Janjic, Natasa Przulj, Olga G. Troyanskaya, Igor Stagljar, Tian Xia, Yoshikazu Ohya, Anne-Claude Gingras, Brian Raught, Michael Boutros, Lars M. Steinmetz, Claire L. Moore, Adam P. Rosebrock, Amy A. Caudy, Chad L. Myers, Brenda Andrews, and Charles Boone. A global genetic interaction network maps a wiring diagram of cellular function. *Science*, 353(6306):aaf1420, 2016. doi: 10.1126/science.aaf1420. URL https://www.science.org/doi/10.1126/science.aaf1420. Publisher: American Association for the Advancement of Science.

Michael Costanzo, Elena Kuzmin, Jolanda van Leeuwen, Barbara Mair, Jason Moffat, Charles Boone, and Brenda Andrews. Global genetic networks and the genotype-to-phenotype relationship. *Cell*, 177(1):85–100, 2019. ISSN 0092-8674. doi: 10.1016/j.cell.2019.01.033. URL https://www.sciencedirect.com/science/article/pii/S0092867419300960.

Michael R Culbertson and Susan A Henry. INOSITOL-REQUIRING MUTANTS OF SACCHAROMYCES CEREVISIAE. *Genetics*, 80(1):23–40, 1975. ISSN 1943-2631. doi: 10.1093/genetics/80.1.23. URL https://doi.org/10.1093/genetics/80.1.23.

Shib Sankar Dasgupta, Michael Boratko, Dongxu Zhang, Luke Vilnis, Xiang Lorraine Li, and Andrew McCallum. Improving local identifiability in probabilistic box embeddings. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, pages 182–192. Curran Associates Inc., 2020. ISBN 978-1-7138-2954-6.

Stacia R. Engel, Suzi Aleksander, Robert S. Nash, Edith D. Wong, Shuai Weng, Stuart R. Miyasato, Gavin Sherlock, and J. Michael Cherry. Saccharomyces genome database: Advances in genome annotation, expanded biochemical pathways, and other key enhancements. *Genetics*, page iyae185, 2024. ISSN 1943-2631. doi: 10.1093/genetics/iyae185.

María Andreína Francisco Rodríguez, Jordi Carreras Puigvert, and Ola Spjuth. Designing microplate layouts using artificial intelligence. *Artificial Intelligence in the Life Sciences*, 3:100073, 2023. ISSN 2667-3185. doi: 10.1016/j.ailsci.2023.100073. URL https://www.sciencedirect.com/science/article/pii/S266731852300017X.

Francesco Gualdi, Baldomero Oliva, and Janet Piñero. Predicting gene disease associations with knowledge graph embeddings for diseases with curtailed information. *NAR Genomics and Bioinformatics*, 6(2):lqae049, 2024. ISSN 2631-9268. doi: 10.1093/nargab/lqae049. URL https://doi.org/10.1093/nargab/lqae049.

Víctor Gutiérrez-Basulto and S. Schockaert. From knowledge graph embedding to ontology embedding? An analysis of the compatibility between vector space representations and rules. In *International Conference on Principles of Knowledge Representation and Reasoning*, 2018.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, pages 1025–1035. Curran Associates Inc., 2017. ISBN 978-1-5108-6096-4.

Janna Hastings, Gareth Owen, Adriano Dekker, Marcus Ennis, Namrata Kale, Venkatesh Muthukrishnan, Steve Turner, Neil Swainston, Pedro Mendes, and Christoph Steinbeck. ChEBI in 2016: Improved services and an expanding collection of metabolites. *Nucleic acids research*, 44:D1214–9, 2016. ISSN 1362-4962. doi: 10.1093/nar/gkv1031. URL https://europepmc.org/articles/PMC4702775.

Henning Hermjakob, Luisa Montecchi-Palazzi, Gary Bader, Jérôme Wojcik, Lukasz Salwinski, Arnaud Ceol, Susan Moore, Sandra Orchard, Ugis Sarkans, Christian von Mering, Bernd Roechert, Sylvain Poux, Eva Jung, Henning Mersch, Paul Kersey, Michael Lappe, Yixue Li, Rong Zeng, Debashis Rana, Macha Nikolski, Holger Husi, Christine Brun, K. Shanker, Seth G. N. Grant, Chris Sander, Peer Bork, Weimin Zhu, Akhilesh Pandey, Alvis Brazma, Bernard Jacq, Marc Vidal, David Sherman, Pierre Legrain, Gianni Cesareni, Ioannis Xenarios, David Eisenberg, Boris Steipe, Chris Hogue, and Rolf Apweiler. The HUPO PSI's molecular interaction format–a community standard for the representation of protein interaction data. *Nature Biotechnology*, 22(2):177–183, 2004. ISSN 1087-0156. doi: 10.1038/nbt926.

Weihua Hu, Matthias Fey, Marinka Zitnik, Yuxiao Dong, Hongyu Ren, Bowen Liu, Michele Catasta, and Jure Leskovec. Open graph benchmark: datasets for machine learning on graphs. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, pages 22118–22133. Curran Associates Inc., 2020. ISBN 978-1-7138-2954-6.

Zijie Huang, Baolin Li, Hafez Asgharzadeh, Anne Cocos, Lingyi Liu, Evan Cox, Colby Wise, and Sudarshan Lamkhede. Synergistic signals: Exploiting co-engagement and semantic links via graph neural networks, 2023.

Junguk Hur, Arzucan Özgür, Zuoshuang Xiang, and Yongqun He. Development and application of an interaction network ontology for literature mining of vaccine-associated gene-gene interactions. *Journal of Biomedical Semantics*, 6(1):2, 2015. ISSN 2041-1480. doi: 10.1186/2041-1480-6-2. URL https://doi.org/10.1186/2041-1480-6-2.

Mathias Jackermeier, Jiaoyan Chen, and Ian Horrocks. Dual box embeddings for the description logic EL++. In *Proceedings of the ACM Web Conference 2024*, WWW '24, pages 2250–2258. Association for Computing Machinery, 2024. ISBN 979-8-4007-0171-9. doi: 10.1145/3589334.3645648. URL https://dl.acm.org/doi/10.1145/3589334.3645648.

Peter D. Karp, Richard Billington, Ron Caspi, Carol A. Fulcher, Mario Latendresse, Anamika Kothari, Ingrid M. Keseler, Markus Krummenacker, Peter E. Midford, Quang Ong, Wai Kit Ong, Suzanne M. Paley, and Pallavi Subhraveti. The BioCyc collection of microbial genomes and metabolic pathways. *Briefings in Bioinformatics*, 20(4):1085–1093, 2019. ISSN 1477-4054. doi: 10.1093/bib/bbx085.

Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A highly efficient gradient boosting decision tree. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://papers.nips.cc/paper_files/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html.

Ross D. King, Kenneth E. Whelan, Ffion M. Jones, Philip G. K. Reiser, Christopher H. Bryant, Stephen H. Muggleton, Douglas B. Kell, and Stephen G. Oliver. Functional genomic hypothesis generation and experimentation by a robot scientist. *Nature*, 427 (6971):247–252, 2004. ISSN 1476-4687. doi: 10.1038/nature02236. URL https://www.nature.com/articles/nature02236. Publisher: Nature Publishing Group.

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. Captum: A unified and generic model interpretability library for pytorch, 2020.

Filip Kronström, Daniel Brunnsåker, Ievgeniia A. Tiukova, and Ross D. King. Ontology-based box embeddings and knowledge graphs for predicting phenotypic traits in saccharomyces cerevisiae. In *19th International Conference on Neurosymbolic Learning and Reasoning*, 2025.

Maxat Kulmanov, Wang Liu-Wei, Yuan Yan, and Robert Hoehndorf. EL embeddings: Geometric construction of models for the description logic EL++. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 6103–6109. International Joint Conferences on Artificial Intelligence Organization, 2019. ISBN 978-0-9992411-4-1. doi: 10.24963/ijcai.2019/845. URL https://www.ijcai.org/proceedings/2019/845.

Elena Kuzmin, Benjamin VanderSluis, Wen Wang, Guihong Tan, Raamesh Deshpande, Yiqun Chen, Matej Usaj, Attila Balint, Mojca Mattiazzi Usaj, Jolanda van Leeuwen, Elizabeth N. Koch, Carles Pons, Andrius J. Dagilis, Michael Pryszlak, Zi Yang Wang, Julia Hanchard, Margot Riggi, Kaicong Xu, Hamed Heydari, Bryan-Joseph San Luis, Ermira Shuteriqi, Hongwei Zhu, Nydia Van Dyk, Sara Sharifpoor, Michael Costanzo, Robbie Loewith, Amy Caudy, Daniel Bolnick, Grant W. Brown, Brenda J. Andrews, Charles Boone, and Chad L. Myers. Systematic analysis of complex genetic interactions. *Science*, 360(6386):eaao1729, 2018. doi: 10.1126/science.aao1729. URL https://www.science.org/doi/10.1126/science.aao1729. Publisher: American Association for the Advancement of Science.

Gang Li, Jan Zrimec, Boyang Ji, Jun Geng, Johan Larsbrink, Aleksej Zelezniak, Jens Nielsen, and Martin KM Engqvist. Performance of regression models as a function of experiment noise. *Bioinformatics and Biology Insights*, 15:11779322211020315, 2021. ISSN 1177-9322. doi: 10.1177/11779322211020315. URL https://doi.org/10.1177/11779322211020315. Publisher: SAGE Publications Ltd STM.

Tingting Liang, Yuanqing Zhang, Qianhui Di, Congying Xia, Youhuizi Li, and Yuyu Yin. Contrastive box embedding for collaborative reasoning. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '23, pages 38–47. Association for Computing Machinery, 2023. ISBN 978-1-4503-9408-6. doi: 10.1145/3539618.3591654.

Fake Lin, Ziwei Zhao, Xi Zhu, Da Zhang, Shitian Shen, Xueying Li, Tong Xu, Suojuan Zhang, and Enhong Chen. When box meets graph neural network in tag-aware recommendation. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pages 1770–1780. Association for Computing Machinery, 2024. ISBN 979-8-4007-0490-1. doi: 10.1145/3637528.3671973.

Chunyu Ma, Zhihan Zhou, Han Liu, and David Koslicki. KGML-xDTD: a knowledge graph–based machine learning framework for drug treatment prediction and mechanism description. *GigaScience*, 12:giad057, 2023. ISSN 2047-217X. doi: 10.1093/gigascience/giad057. URL https://doi.org/10.1093/gigascience/giad057.

Jianzhu Ma, Michael Ku Yu, Samson Fong, Keiichiro Ono, Eric Sage, Barry Demchak, Roded Sharan, and Trey Ideker. Using deep learning to model the hierarchical structure and function of a cell. *Nature Methods*, 15(4):290–298, 2018. ISSN 1548-7105. doi: 10.1038/nmeth.4627. URL https://www.nature.com/articles/nmeth.4627. Publisher: Nature Publishing Group.

Gabriela A Merino, Rabie Saidi, Diego H Milone, Georgina Stegmayer, and Maria J Martin. Hierarchical deep learning for predicting GO annotations by integrating protein knowledge. *Bioinformatics*, 38(19):4488–4496, 2022. ISSN 1367-4803. doi: 10.1093/bioinformatics/btac536. URL https://doi.org/10.1093/bioinformatics/btac536.

John H Morris, Karthik Soman, Rabia E Akbas, Xiaoyuan Zhou, Brett Smith, Elaine C Meng, Conrad C Huang, Gabriel Cerono, Gundolf Schenk, Angela Rizk-Jackson, Adil Harroud, Lauren Sanders, Sylvain V Costes, Krish Bharat, Arjun Chakraborty, Alexander R Pico, Taline Mardirossian, Michael Keiser, Alice Tang, Josef Hardi, Yongmei Shi, Mark Musen, Sharat Israni, Sui Huang, Peter W Rose, Charlotte A Nelson, and Sergio E Baranzini. The scalable precision medicine open knowledge engine (SPOKE): a massive knowledge graph of biomedical information. *Bioinformatics*, 39(2):btad080, 2023. ISSN 1367-4811. doi: 10.1093/bioinformatics/btad080. URL https://doi.org/10.1093/bioinformatics/btad080.

Chris Mungall, David Osumi-Sutherland, James A. Overton, Jim Balhoff, Clare72, pgaudet, Matthew Brush, Nico Matentzoglu, Vasundra Touré, Damion Dooley, Michael Sinclair, Anthony Bretaudeau, Scott Cain, Melissa Haendel, diatomsRcool, Jen Hammock, Marie-Angélique Laporte, Mark Jensen, and Martin Larralde. oborel/obo-relations: 2020-07-21, 2020. URL https://zenodo.org/records/3955125.

Maria Parapouli, Anastasios Vasileiadis, Amalia-Sofia Afendra, and Efstathios Hatziloukas. Saccharomyces cerevisiae and its industrial applications. *AIMS Microbiology*, 6(1):1–31, 2020. ISSN 2471-1888. doi: 10.3934/microbiol.2020001. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC7099199/.

Dhruvesh Patel, Shib Sankar Dasgupta, Michael Boratko, Xiang Li, Luke Vilnis, and Andrew McCallum. Representing joint hierarchies with box embeddings. In *Automated Knowledge Base Construction (AKBC)*, 2020. doi: 10.24432/C5KS37.

Xi Peng, Zhenwei Tang, Maxat Kulmanov, Kexin Niu, and Robert Hoehndorf. Description logic EL++ embeddings with intersectional closure, 2022. URL http://arxiv.org/abs/2202.14018.

Katherine Roper, A. Abdel-Rehim, Sonya Hubbard, Martin Carpenter, Andrey Rzhetsky, Larisa Soldatova, and Ross D. King. Testing the reproducibility and robustness of the cancer biology literature by robot. *Journal of The Royal Society Interface*, 19(189): 20210821, 2022. doi: 10.1098/rsif.2021.0821. URL https://royalsocietypublishing.org/doi/10.1098/rsif.2021.0821. Publisher: Royal Society.

Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences, 2017. URL http://arxiv.org/abs/1605.01713.

K. Syama, J. Angel Arul Jothi, and Namita Khanna. Automatic disease prediction from human gut metagenomic data using boosting GraphSAGE. *BMC Bioinformatics*, 24(1): 126, 2023. ISSN 1471-2105. doi: 10.1186/s12859-023-05251-x. URL https://doi.org/10.1186/s12859-023-05251-x.

Cornelis Verduyn, Erik Postma, W. Alexander Scheffers, and Johannes P. Van Dijken. Effect of benzoic acid on metabolic fluxes in yeasts: A continuous-culture study on the regulation of respiration and alcoholic fermentation. *Yeast*, 8(7):501–517, 1992. ISSN 1097-0061. doi: 10.1002/yea.320080703. URL https://onlinelibrary.wiley.com/doi/abs/10.1002/yea.320080703. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/yea.320080703.

Luke Vilnis, Xiang Li, Shikhar Murty, and Andrew McCallum. Probabilistic embedding of knowledge graphs with box lattice measures. In Iryna Gurevych and Yusuke Miyao, editors, *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 263–272. Association for Computational Linguistics, 2018. doi: 10.18653/v1/P18-1025.

Denny Vrandečić and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85, 2014. ISSN 0001-0782. doi: 10.1145/2629489.

Alina Vretinaris, Chuan Lei, Vasilis Efthymiou, Xiao Qin, and Fatma Özcan. Medical entity disambiguation using graph neural networks. In *Proceedings of the 2021 International Conference on Management of Data*, SIGMOD '21, pages 2310–2318. Association for Computing Machinery, 2021. ISBN 978-1-4503-8343-1. doi: 10.1145/3448016.3457328. URL https://dl.acm.org/doi/10.1145/3448016.3457328.

Brian Walsh, Sameh K. Mohamed, and Vít Nováček. BioKG: A knowledge graph for relational learning on biological data. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*, CIKM '20, pages 3173–3180. Association for Computing Machinery, 2020. ISBN 978-1-4503-6859-9. doi: 10.1145/3340531.3412776. URL https://dl.acm.org/doi/10.1145/3340531.3412776.

Kevin Williams, Elizabeth Bilsland, Andrew Sparkes, Wayne Aubrey, Michael Young, Larisa N. Soldatova, Kurt De Grave, Jan Ramon, Michaela de Clare, Worachart Sirawaraporn, Stephen G. Oliver, and Ross D. King. Cheaper faster drug development validated by the repositioning of drugs against neglected tropical diseases. *Journal of The Royal Society Interface*, 12(104):20141289, 2015. doi: 10.1098/rsif.2014.1289. URL https://royalsocietypublishing.org/doi/10.1098/rsif.2014.1289. Publisher: Royal Society.

Valerie Wood, Antonia Lock, Midori A. Harris, Kim Rutherford, Jürg Bähler, and Stephen G. Oliver. Hidden in plain sight: what remains to be discovered in the eukaryotic proteome? *Open Biology*, 9(2):180241, 2019. ISSN 2046-2441. doi: 10.1098/rsob.180241. URL https://www.ncbi.nlm.nih.gov/pmc/articles/PMC6395881/.

Jingyi Xu, Zilu Zhang, Tal Friedman, Yitao Liang, and Guy Van den Broeck. A semantic loss function for deep learning with symbolic knowledge. In Jennifer Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 5502–5511. PMLR, 10–15 Jul 2018.

Appendix A. Examples of description logic in KG

A description logic example of how a phenotype with a qualifier and chemical are specified in the KG. This example is about decreased (APO_0000003) utilisation of carbon source (APO_0000096) of lactate (CHEBI_16004), observed for the gene YBL030C.

$$\begin{aligned}
&\texttt{APO\_0000098-APO\_0000003-CHEBI\_16004} \sqsubseteq \texttt{APO\_0000098} \sqcap \texttt{APO\_0000003}\\
&\texttt{APO\_0000098-APO\_0000003-CHEBI\_16004} \sqsubseteq \exists \texttt{aboutChemical.CHEBI\_16004}\\
&\texttt{YBL030C} \sqsubseteq \exists \texttt{RO\_0002200.APO\_0000098-APO\_0000003-CHEBI\_16004}\\
&\texttt{YBL030C} \sqsubseteq \exists \texttt{hasChemNutrientUtilization\_Decreased.CHEBI\_16004}.
\end{aligned} \tag{14}$$

A description logic example of how a gene (YCR073C) is positively regulating the protein activity (INO_0000104) of another gene (YLR113W). This regulation happens during (RO_0002092) cellular response to heat (GO_0034605).

$$\begin{aligned}
&\texttt{YCR073C-YLR113W-protein\_activity-positive} \sqsubseteq \texttt{INO\_0000104}\\
&\texttt{YCR073C} \sqsubseteq \exists \texttt{positive\_regulator\_of.YCR073C-YLR113W-protein\_activity-positive}\\
&\texttt{positive\_regulator\_of.YCR073C-YLR113W-protein\_activity-positive}\\
&\qquad\qquad \sqsubseteq \exists \texttt{regulated\_gene.YLR113W}\\
&\texttt{positive\_regulator\_of.YCR073C-YLR113W-protein\_activity-positive}\\
&\qquad\qquad \sqsubseteq \exists \texttt{RO\_0002092.GO\_0034605}\\
&\texttt{YCR073C} \sqsubseteq \exists \texttt{positively\_regulating.YLR113W}.
\end{aligned} \tag{15}$$

Appendix B. Royal family KG

Table 3: Terms used in family tree demonstration

Term	Type	Superclass/-property	Domain	Range
Person	owl:Class	–	–	–
Man	owl:Class	Person	–	–
Woman	owl:Class	Person	–	–
Country	owl:Class	–	–	–
parentOf	owl:objectProperty	–	Person	Person
fatherOf	owl:objectProperty	parentOf	Man	Person
motherOf	owl:objectProperty	parentOf	Woman	Person
childOf	owl:objectProperty	–	Person	Person
birthPlace	owl:objectProperty	–	Person	Country
spouseOf	owl:objectProperty	–	Person	Person
husbandOf	owl:objectProperty	spouseOf	Man	Person
wifeOf	owl:objectProperty	spouseOf	Woman	Person

Appendix C. Hyperparameters

C.1. Box embedding parameters

Table 4: Parameters used for the box embeddings of the different domains

Domain	Dimensions	Epochs	Lr	Regularisation	Gumbel temperature	Neg. ex. ratio
Material entity	10	1,000	1e-2	1e-3	0.25	2.0
Genes	8	600	1e-2	1e-3	0.25	4.0
Regulations	5	500	1e-2	1e-3	0.25	2.0
Molecular functions	5	500	1e-2	1e-3	0.25	2.0
Biological processes	5	500	1e-2	1e-3	0.25	2.0
Phenotypes	4	500	1e-2	1e-3	0.25	2.0
Reactions & Pathways	4	500	1e-2	1e-3	0.25	2.0
Cellular components	4	500	1e-2	1e-3	0.25	2.0

C.2. Prediction model hyperparameters

The best performing model was trained for 500 epochs, with a learning rate of 1e-5, and L2 regularisation weight of 0.1. The depth of the GNN was 2 and the embedding dimensions for the domains are listed in Table 5 and are the same throughout the GNN. The fully connected neural network predicting the interaction from the embeddings is of depth 3 with 64, 8, and 1 neurons respectively. For models trained with the semantic loss we used $\alpha = 0.1$ and $\beta = 0.2$.

Table 5: Embedding dimensions for the different domains throughout the GNN.

Embedding dimensions	32	64	128
Domains	Cellular components Molecular functions Reactions Regulations	Biological processes Phenotypes	Material entities Genes

C.3. Family tree box embeddings

Table 6: The embedding models, trained with both $\mathcal{L}_{distance}$ and $\mathcal{L}_{overlap}$ used the same hyperparameters. The learning rate was after each epoch multiplied with $(1 -$ Lr decay) and the regularisation used is presented in (10), penalising small boxes, with $l_0 = 1$. λ, β, and γ refer to weights in the losses in (12) and (13).

Epochs	Initial lr	Lr decay	Regular-isation, λ	Negative weight, β	Negative weight, γ
500	5e-1	1e-3	1e-2	5e-1	1.0

Appendix D. Model-driven experiment

D.1. Edge filtering

Table 7: We filter for co-occurring edge pairs in which at least one edge connects a gene to a node that is a subclass of one of the following APO classes, related to nutrient utilisation.

APO Class	Description
APO_0000096	General nutrient utilisation
APO_0000097	Auxotrophy
APO_0000099	Utilisation of nitrogen source
APO_0000100	Nutrient uptake
APO_0000125	Utilisation of phosphorous source
APO_0000219	Utilisation of sulfur source

Table 8: We also allow edge pairs where at least one of the edges links a gene to a chemical through any of the following relations.

```
hasChemNutrientUtilization
hasChemNutrientUtilization_Increased
hasChemNutrientUtilization_Decreased
```

D.2. Top edge pairs

Table 9: The 10 edge pairs with the highest importance weight after filtering for the criteria specified in Appendix D.1. The edge pair selected for the experiment is highlighted. Ch.Nutr.Util. is short for hasChemNutrientUtilization, Ch.Nutr.Util.Dec. is short for hasChemNutrientUtilization_Decreased, and Ch.StressRes. is short for hasChemStressResistance.

Importance	Relation1	Class1	Relation2	Class2
0.003471	Ch.Nutr.Util.	CHEBI_17268	Ch.StressRes.	CHEBI_22907
0.002002	Ch.StressRes.	CHEBI_22907	has_phenotype	APO_0000099- APO_0000245- CHEBI_14321
0.001985	Ch.Nutr.Util.	CHEBI_17268	Ch.StressRes.	CHEBI_26710
0.001705	Ch.Nutr.Util.Dec.	CHEBI_23414	Ch.StressRes.	CHEBI_22907
0.001679	hasChemCellMorph	CHEBI_26710	has_phenotype	APO_0000099- APO_0000245- CHEBI_14321
0.001580	Ch.Nutr.Util.	CHEBI_17268	Ch.StressRes.	CHEBI_50145
0.001541	Ch.Nutr.Util.Dec.	CHEBI_77995	Ch.StressRes.	CHEBI_49470
0.001537	Ch.StressRes.	CHEBI_22907	has_phenotype	APO_0000099- APO_0000245- CHEBI_26271
0.001500	Ch.Nutr.Util.	CHEBI_17268	has_phenotype	APO_0000059- APO_0000002- CHEBI_26710
0.001477	Ch.Nutr.Util.Dec.	CHEBI_16236	Ch.StressRes.	CHEBI_22907

D.3. Cultivation method

The Δ*ino1* deletion mutant was taken from the EUROSCARF deletion collection, with the strain background being BY4741, genotype: MATa, *his3*Δ1, *leu2*Δ0, *met15*Δ0, *ura3*Δ0 (Y01272).

The Δ*ino1* mutant was pre-cultured overnight in minimally buffered delft media containing the following: 5g/L (NH4)2SO4, 3g/L KH2PO4, 0.5g/L MGSO4. 7H2O, and 1mL/L trace metal and vitamin solutions as described by Verduyn et al. (1992), 25 mg/L myo-inositol and 2% glucose (w/v) in 30°C, and 220rpm. The pre-culture was adjusted to 0.5 OD600, and robotically dispensed with a 1:20 dilution into a 96-well microculture plate using a Hamilton Microlab Star liquid handling robot. A negative control was also included to assess the baseline growth of the Δ*ino1* mutant without any supplementation of myo-inositol. Additionally, myo-inositol-free media with 0.25% (w/v) glucose, myo-inositol (Sigma aldrich 57570-100G), Sodium chloride (Merck 1064041000) and MilliQ-water was robotically dispensed, resulting in a total volume of 250μL and the concentrations defined in Table 10.

Table 10: The concentrations of inositol and NaCl used for the experiment.

Inositol	NaCl
0.00 mMolar	0.0 Molar
0.01 mMolar	0.0 Molar
0.01 mMolar	0.3 Molar
0.01 mMolar	0.6 Molar
0.05 mMolar	0.0 Molar
0.05 mMolar	0.3 Molar
0.05 mMolar	0.6 Molar
0.25 mMolar	0.0 Molar
0.25 mMolar	0.3 Molar
0.25 mMolar	0.6 Molar

A robust plate layout was generated with PLAID (Francisco Rodríguez et al., 2023). The processed plate was cultivated in the automated laboratory cell Eve. The plate was transferred from an automated incubator (30°C) to a Teleshaker Magnetic Shaking System, where it was shaken for 30s at 800 rpm, divided evenly between clockwise and counter-clockwise double-orbital shaking. After shaking, the plate was transferred to a BMG Polarstar plate reader, where it underwent optical density measurements at 600 nM (the temperature in the plate reader was kept at a constant 30°C). After measuring, the plate was returned to the incubator. The protocol was automatically repeated every 20 min for up to 24 h.

D.4. Growth data processing and statistical testing

Outliers in the growth curves (measured through optical density at 600nm) were identified and filtered using the interquartile range (IQR), where any data points outside the range of [Q1-1.5 IQR, Q3+1.5 IQR] were excluded from the dataset. The filtered curves were then subsequently smoothed using a rolling mean of window size 3. The resulting averaged growth curves can be seen in Figure 9. Area under curve was calculated using `numpy.trapz` (`v1.26.4`). To assess the effects of inositol and NaCl on AUC, a generalised linear model was employed (`statsmodels v0.14.4`). The model was fitted using a Gaussian family distribution. Choice α-value was set at 0.05. We modelled all factors as categorical to avoid imposing any assumptions on linearity. The model is specified as follows:

$$\text{AUC} \backsim C(Inositol) \times C(NaCl). \tag{16}$$

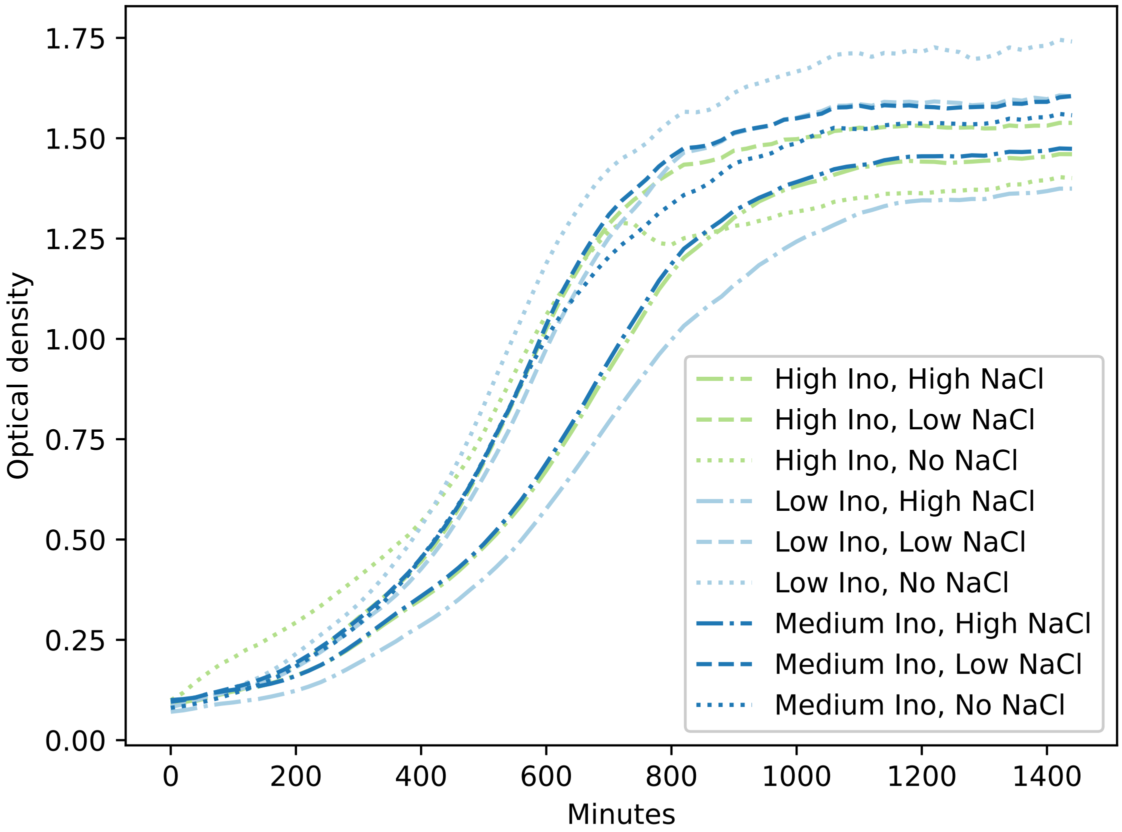


Figure 9: Growth curves showing the mean optical densities of the 6-8 repetitions for the different experimental groups. Optical density (at 600nM) is a unitless measurement typically used as an indirect measure of cell density and biomass.