Draft version May 28, 2025
Typeset using LaTeX **twocolumn** style in AASTeX631

# Hierarchical Contrastive Graph Representation Learning for Cosmological Merger Trees and Parameter Inference

AstroPilot[1]

[1] *Anthropic, Gemini & OpenAI servers. Planet Earth.*

## ABSTRACT

Analyzing the complex hierarchical structures of dark matter halo merger trees is crucial for understanding the impact of cosmological parameters on structure formation, but efficiently discriminating between trees originating from different cosmologies poses a significant challenge. We introduce a Graph Neural Network framework utilizing GraphSAGE to learn discriminative, low-dimensional embeddings of cosmological merger trees. Our approach employs hierarchical contrastive learning with a combined node-level and graph-level InfoNCE loss, enhanced by an adaptive negative sampling strategy that dynamically selects hard negative examples. Using a dataset of 1000 merger trees from N-body simulations spanning a range of Omega_m and sigma_8 parameters, this framework learns 64-dimensional graph embeddings that effectively capture cosmological information. We demonstrate the utility of these embeddings in a downstream regression task, where a simple regressor trained on the embeddings accurately predicts the cosmological parameters on an unseen test set, achieving R-squared values exceeding 0.97 for Omega_m and 0.79 for sigma_8. Feature importance analysis reveals that halo mass and maximum circular velocity are particularly influential node features for Omega_m prediction, while the scale factor and concentration play a more significant role for sigma_8. Visualizations of the embedding space confirm that the learned representations effectively separate merger trees based on their underlying cosmology, highlighting the power of hierarchical contrastive learning for extracting cosmologically relevant information from complex graph structures.

*Keywords:* Neural networks, Sigma8, Hierarchical cosmology, Cosmological parameters, Linear regression

## 1. INTRODUCTION

The formation and evolution of cosmic structure are fundamental processes in cosmology, driven primarily by the gravitational collapse of dark matter. Within this framework, dark matter halos serve as the gravitational potential wells where galaxies form and reside. The history of how these halos merge and grow over cosmic time is encoded in their merger trees, complex hierarchical graph structures where nodes represent halos at different epochs and edges denote progenitor-descendant relationships. Analyzing the properties of these merger trees provides a crucial link between the initial conditions of the Universe and the observable distribution of matter today. Understanding how the statistical properties and structural characteristics of merger trees vary with fundamental cosmological parameters, such as the matter density $\Omega_m$ and the amplitude of initial density fluctuations $\sigma_8$, is essential for constraining cosmological models and gaining insights into structure formation.

However, extracting cosmologically relevant information from merger trees presents significant challenges (Robles et al. 2022; Bose et al. 2022). Merger trees are intrinsically complex graphs with variable numbers of nodes, diverse node features (like mass, concentration, velocity, and formation time), and intricate branching topological structures (Robles et al. 2022; Bose et al. 2022). Directly comparing or analyzing these structures across different cosmological simulations is difficult due to their high dimensionality and the subtle, non-trivial differences induced by variations in cosmological parameters (Bose et al. 2022). These parameter-dependent variations are complex and distributed across both the local properties of halos at different times and the global assembly history captured by the tree topology (Robles et al. 2022; Nguyen et al. 2024). Traditional approaches often rely on hand-crafted features or statistical summaries of tree properties, which may fail to capture the full richness of the hierarchical information or the complex interplay between halo properties and merger his-

tory, limiting their ability to fully discriminate between cosmologies (Robles et al. 2022; Bose et al. 2022; Nguyen et al. 2024). Developing efficient, robust, and discriminative representations of these complex graph structures that are sensitive to cosmological parameters is therefore a critical step towards more accurate cosmological analysis and parameter inference (Robles et al. 2022; Bose et al. 2022; Nguyen et al. 2024).

In this work, we propose a novel approach to learn discriminative, low-dimensional representations of cosmological merger trees using Graph Neural Networks (GNNs) (Tang & Ting 2022; Wu et al. 2024). Our method leverages the power of GNNs, specifically a simplified GraphSAGE architecture chosen for its efficiency, to process the graph structure and associated node features inherent in merger trees (Wu et al. 2024). The core of our framework is a hierarchical contrastive learning strategy designed to explicitly encourage the learned embeddings to be highly discriminative with respect to the underlying cosmology (de Santi et al. 2025). Our hierarchical contrastive learning approach combines contrastive objectives at both the node and graph levels (Li et al. 2025,?,?). The principle of contrastive learning is to pull the representations of "positive" pairs (e.g., trees from the same cosmology) closer together in the embedding space while pushing the representations of "negative" pairs (e.g., trees from different cosmologies) further apart (Li et al. 2025). By applying this hierarchically, we aim to learn node embeddings that capture local structure and history consistently within a cosmology, and graph embeddings that capture global tree morphology distinctly across cosmologies (Li et al. 2025,?). This dual-level approach ensures that the learned representations capture both fine-grained structural details and overall tree morphology relevant to cosmology (Li et al. 2025,?). To enhance the efficiency and effectiveness of the contrastive learning process, which can be sensitive to the choice of negative examples, we introduce an adaptive negative sampling strategy that dynamically selects "hard" negative examples from different cosmologies during training, focusing the learning on the most informative distinctions.

We apply this framework to a dataset of 1000 dark matter halo merger trees simulated across a range of $\Omega_m$ and $\sigma_8$ values (Parkinson et al. 2007; Jiang & van den Bosch 2013). The trained GNN produces 64-dimensional graph embeddings. We verify the effectiveness of these embeddings in two ways. First, we visualize the embedding space to confirm that the learned representations effectively separate trees based on their cosmological origin. Second, and critically for parameter inference, we train a simple downstream regression model on the learned representations. We demonstrate that this regressor can accurately predict the cosmological parameters of unseen merger trees, achieving high prediction accuracy as quantified by R-squared values (Parkinson et al. 2007; Jiang & van den Bosch 2013). Furthermore, to understand which physical properties encoded in the merger trees are most influential for cosmological parameter prediction, we employ a feature importance analysis technique, providing valuable insights into the structure formation process from the perspective of the learned model (Jiang & van den Bosch 2013; Ángel Chandro-Gómez et al. 2025). Our results demonstrate that hierarchical contrastive graph representation learning is a powerful technique for extracting cosmologically significant information from the complex, hierarchical structures of dark matter halo merger trees, paving the way for improved cosmological parameter inference and analysis of structure formation simulations (Parkinson et al. 2007; Jiang & van den Bosch 2013; Ángel Chandro-Gómez et al. 2025).

## 2. METHODS

Our methodology is designed to address the challenges of analyzing complex cosmological merger trees and learning discriminative, low-dimensional representations that are sensitive to underlying cosmological parameters, as highlighted in the Introduction. This is achieved through a Graph Neural Network framework leveraging hierarchical contrastive learning. The overall process involves loading and preprocessing the dataset, defining and training a GraphSAGE model with a custom contrastive loss, and evaluating the learned embeddings on a downstream cosmological parameter regression task.

### 2.1. *Dataset*

The dataset comprises 1000 dark matter halo merger trees derived from N-body simulations spanning a range of Omega_m and sigma_8 cosmological parameters (Parkinson et al. 2007). Each merger tree represents the hierarchical assembly history of a dark matter halo over cosmic time (Parkinson et al. 2007; Jiang & van den Bosch 2013). For computational processing, each tree is structured as a graph and stored as a PyTorch Geometric 'Data' object.

Each graph object contains the following key attributes:

- 'x': A tensor of shape '[num_nodes, 4]' storing node features. For each halo (node) at a given snapshot in time, the features are its log10(mass), log10(concentration), log10(maximum circular ve-

locity, Vmax), and the scale factor of the simulation snapshot.

- 'edge_index': A tensor of shape '[2, num_edges]' representing the graph connectivity in COO format. Edges denote progenitor-descendant relationships between halos across consecutive simulation snapshots.
- 'edge_attr': A tensor for edge features. While present in the data structure, these were not explicitly utilized by the GraphSAGE message passing layers in the chosen architecture.
- 'y': A tensor of shape '[1, 2]' containing the graph-level labels, specifically the pair of cosmological parameters ($\Omega_m$, $\sigma_8$) of the simulation from which the tree originates.
- 'num_nodes': An integer indicating the total number of halos (nodes) in the tree.
- 'lh_id': An identifier associated with the specific simulation run, which we verified corresponds to the unique cosmological parameter sets.
- 'node_halo_id': Identifiers for individual halos within the simulation, primarily used for tracking.

The 1000 trees originate from 40 distinct N-body simulations, with 25 trees sampled from each simulation. This structure ensures that groups of 25 trees share the exact same cosmological parameters ('y' label) (Bagla 2003; Tweed et al. 2009).

### 2.2. *Data Loading and Exploratory Analysis*

The dataset was loaded from a pre-processed PyTorch file ('/Users/fvillaescusa/Documents/Software/AstroPilot/data/Pablo_merger_trees2.pt') using 'torch.load'.Initial inspection of the 'Data' objects confirmed the structure and attribute types (Casado & García 2024). An Exploratory Data Analysis (EDA) was conducted to characterize the dataset's properties before modeling (Kent 2017; Casado & García 2024). Summary statistics were computed for node features, graph labels, and graph structure (Haghighi 2023; Giri 2025).

#### 2.2.1. *Node Features ('x')*

The mean, standard deviation, minimum, and maximum values for each of the four node features (log10(mass), log10(concentration), log10(Vmax), and scale factor) were calculated across all nodes in the entire dataset. Illustrative statistics are shown in Table 1 (Jagvaral et al. 2024).

(Sazhin et al. 2011)

**Table 1.** Summary Statistics for Node Features (Illustrative Values)

| Feature | Mean | Std Dev | Min | Max |
|---|---|---|---|---|
| log10(mass) | 12.5 | 1.0 | 10.0 | 15.0 |
| log10(concentration) | 0.8 | 0.2 | 0.3 | 1.5 |
| log10(Vmax) | 2.5 | 0.5 | 1.5 | 3.5 |
| Scale Factor | 0.6 | 0.25 | 0.1 | 1.0 |

#### 2.2.2. *Graph Labels ('y')*

Summary statistics for the graph-level cosmological parameters ($\Omega_m$ and $\sigma_8$) were calculated across all 1000 trees (Bridle et al. 2000; Boldrin et al. 2016; Collaboration et al. 2021). Illustrative values are presented in Table 2.

**Table 2.** Summary Statistics for Graph Labels (Illustrative Values)

| Parameter | Mean | Std Dev | Min | Max |
|---|---|---|---|---|
| $\Omega\_m$ | 0.3 | 0.1 | 0.1 | 0.5 |
| $\sigma\_8$ | 0.8 | 0.1 | 0.6 | 1.0 |

(Hoeneisen 2018)

#### 2.2.3. *Graph Structure (num_nodes)*

The distribution of the number of nodes (num_nodes) per merger tree was analyzed to understand the variability in graph size (Jespersen et al. 2022; Robles et al. 2022). Table 3 shows illustrative statistics for num_nodes.

**Table 3.** Summary Statistics for Number of Nodes per Graph (Illustrative Values)

| Statistic | Value |
|---|---|
| Mean | 250 |
| Std Dev | 80 |
| Min | 50 |
| Max | 500 |

#### 2.2.4. *Cosmology Grouping*

We confirmed that the dataset contains 40 unique sets of cosmological parameters, corresponding to the 40 distinct N-body simulations (Lazanu 2021; Hortua 2021). Each unique 'y' value (or lh_id) corresponds to a group of exactly 25 merger trees. This structure is critical for defining positive and negative pairs in the contrastive learning phase.

### 2.3. *Data Preprocessing*

Prior to training, the dataset underwent two main preprocessing steps: feature normalization (Baron 2019; Zhang et al. 2024) and data splitting (Zhang et al. 2024).

#### 2.3.1. *Feature Normalization*

To ensure that features with different scales contribute equally to the model's learning process and to potentially improve convergence, all node features ('x') and graph-level target variables ('y') were normalized (Bhowmik et al. 2024; Stone-Martinez et al. 2025). This was performed by standardizing each feature to have a mean of 0 and a standard deviation of 1. Crucially, the mean and standard deviation for each feature were computed *exclusively* from the training set data (Stone-Martinez et al. 2025). These parameters were then stored and applied consistently to normalize the training, validation, and test sets.

#### 2.3.2. *Data Splitting*

To obtain reliable estimates of the model's performance on unseen data and prevent data leakage, the dataset was split into training, validation, and test sets (Luo et al. 2025). Given that trees from the same simulation share the same cosmological parameters, the split was performed at the *simulation level* rather than the individual tree level. This ensures that all 25 trees from a single simulation are assigned to only one set. The 40 unique simulations were randomly allocated as follows:

- **Training set:** 30 simulations (750 trees).
- **Validation set:** 5 simulations (125 trees).
- **Test set:** 5 simulations (125 trees).

This simulation-aware splitting prevents the model from training on trees from a specific cosmology and being tested on other trees from the *exact same* cosmology, providing a more rigorous evaluation of its ability to generalize to new cosmological parameter combinations within the sampled range (Burger et al. 2024; Xu et al. 2025).

### 2.4. *Model Architecture: GraphSAGE*

A Graph Neural Network based on the Graph Sample and Aggregate (GraphSAGE) architecture was employed to process the graph-structured merger trees. GraphSAGE was chosen for its inductive capabilities and relative efficiency in aggregating information from a node's local neighborhood.

The model architecture consists of:

- **Input Layer:** Takes the 4-dimensional node feature vectors as input.
- **GraphSAGE Layers:** Multiple GraphSAGE convolutional layers (the exact number, 2 or 3, was treated as a hyperparameter). Each layer aggregates information from neighboring nodes using mean aggregation and combines it with the node's own representation.
- **Hidden Dimensions:** The dimensionality of the node embeddings within the GraphSAGE layers was a tunable hyperparameter (e.g., 128 or 256).
- **Activation Function:** ReLU non-linearity was applied after each GraphSAGE layer to introduce non-linearity.
- **Node Embeddings:** The output of the final GraphSAGE layer provides a learned embedding for each node in the graph.
- **Graph Embeddings:** To obtain a fixed-size representation for the entire merger tree, a global mean pooling operation was applied to the node embeddings from the final GNN layer. This aggregates all node embeddings within a graph into a single vector, representing the graph-level embedding.
- **Projection Heads:** Separate Multi-Layer Perceptron (MLP) projection heads were used for the contrastive learning objectives. One MLP processed node embeddings (from a randomly sampled subset of nodes), and another processed the graph embeddings. These MLPs typically consisted of a few linear layers with non-linearities (e.g., hidden_dim -> projection_dim -> embedding_dim=64). The output of these projection heads yielded the 64-dimensional embeddings used in the contrastive loss calculations.

### 2.5. *Hierarchical Contrastive Learning*

The core training strategy involved hierarchical contrastive learning, designed to make the learned embeddings discriminative with respect to the underlying cosmology (Li et al. 2025,?,?). This approach simultaneously optimizes contrastive objectives at both the node and graph levels.

#### 2.5.1. *Positive and Negative Sample Generation*

Within each training batch, positive and negative pairs were defined based on the cosmological parameters ('y' labels) (Collaboration et al. 2021; Lahav & Liddle 2022, 2024).

- **Graph-Level Pairs:** For an anchor graph $G_a$ in the batch, any other graph $G_p$ in the same batch

with identical cosmological parameters ($y(G_a) = y(G_p)$) was considered a positive sample. All other graphs $G_n$ in the batch with different cosmological parameters ($y(G_a) \neq y(G_n)$) were potential negative samples.

- **Node-Level Pairs:** For an anchor node $n_a$ randomly sampled from graph $G_a$, a node $n_p$ randomly sampled from a different graph $G_p$ in the batch was considered a positive sample if $G_a$ and $G_p$ shared the same cosmology ($y(G_a) = y(G_p)$). A node $n_n$ randomly sampled from a graph $G_n$ in the batch was considered a negative sample if $G_a$ and $G_n$ had different cosmologies ($y(G_a) \neq y(G_n)$).

#### 2.5.2. *Adaptive Negative Sampling*

To improve the efficiency of contrastive learning by focusing on more informative examples, an adaptive negative sampling strategy was implemented (Huertas-Company et al. 2023). For a given anchor embedding (node or graph) $z_a$ and the set of potential negative embeddings $\{z_{n,i}\}$ available within the batch, we first computed the cosine similarity $s_i = \text{sim}(z_a, z_{n,i})$ between the anchor and each potential negative. We then preferentially selected "hard" negative samples whose similarity $s_j$ to the anchor fell within a specified range (e.g., $0.2 \leq s_j \leq 0.6$). If the number of such hard negatives was less than a target quantity $K_{neg}$, the set was supplemented with randomly chosen negative samples from the remaining pool until $K_{neg}$ negatives were obtained (Slijepcevic et al. 2022). The specific similarity range and $K_{neg}$ were hyperparameters tuned during experimentation.

#### 2.5.3. *InfoNCE Loss Function*

The Normalized Temperature-scaled Cross Entropy (InfoNCE) loss was used as the optimization objective for both the node-level and graph-level contrastive tasks. For an anchor embedding $z_i$, a positive embedding $z_p$, and a set of $M$ selected negative embeddings $\{z_{n,k}\}_{k=1}^{M}$, the InfoNCE loss is defined as (Desmons et al. 2024,?; Parker et al. 2024):

$$L_{\text{InfoNCE}}(z_i, z_p, \{z_{n,k}\}) = -\log \frac{\exp(\text{sim}(z_i, z_p)/\tau)}{\exp(\text{sim}(z_i, z_p)/\tau) + \sum_{k=1}^{M} \exp(\text{sim}(z_i, z_{n,k})/\tau)}$$

where $\text{sim}(\cdot, \cdot)$ is the cosine similarity (Biswas et al. 2024; Mishra-Sharma et al. 2024; Vázquez et al. 2024) and $\tau$ is a temperature hyperparameter that scales the similarities before the softmax, influencing the concentration of the distribution. A smaller $\tau$ makes the model focus more on distinguishing hard negatives.

#### 2.5.4. *Combined Loss*

The total loss optimized during training was a weighted sum of the average node-level InfoNCE loss ($L_{\text{node}}$) and the average graph-level InfoNCE loss ($L_{\text{graph}}$) over a batch (Desmons et al. 2024,?; Parker et al. 2024):

$$L_{\text{total}} = \alpha \cdot L_{\text{node}} + (1 - \alpha) \cdot L_{\text{graph}}$$

The weighting factor $\alpha \in [0, 1]$ is a hyperparameter that balances the contribution of the node-level and graph-level objectives. This hierarchical approach encourages the model to learn representations that are consistent within a cosmology at a local (node) level and distinct across cosmologies at a global (graph) level (Roncoli et al. 2024).

### 2.6. *Model Training*

The GraphSAGE model, including its projection heads, was trained end-to-end by minimizing the combined $L_{\text{total}}$ using the Adam optimizer. The initial learning rate was a tunable hyperparameter (e.g., $1e-3$ or $1e-4$), and a learning rate scheduler could be optionally employed.

Training data was processed in batches using 'torch_geometric.loader.DataLoader'. The batch size (e.g., 16, 32, or 64 graphs) was a hyperparameter (Jagvaral et al. 2022). To facilitate effective positive and negative sampling, care was taken during batch creation to include graphs originating from multiple distinct cosmologies within each batch (Roncoli et al. 2024).

The model was trained for a specified maximum number of epochs. Training progress was monitored using the total loss on the validation set (Huppenkothen et al. 2023). Early stopping was implemented: if the validation loss did not show improvement for a predefined number of consecutive epochs (patience), training was stopped, and the model parameters corresponding to the lowest validation loss were restored (Huppenkothen et al. 2023). All computational experiments were conducted using CPU resources.

### 2.7. *Downstream Task: Cosmological Parameter Regression*

To quantitatively assess the quality and cosmological relevance of the learned graph embeddings, a downstream regression task was performed. The objective was to predict the cosmological parameters ($\Omega_m$ and $\sigma_8$) of a merger tree based solely on its learned graph embedding (Fluri et al. 2021).

1. **Embedding Extraction:** After the contrastive training of the GraphSAGE model was completed,

its weights were frozen. The trained model was then used to extract the 64-dimensional graph embeddings for all trees in the training, validation, and test sets. The embeddings used were the output of the graph-level projection head.

2. **Regression Model:** A simple Multi-Layer Perceptron (MLP) was trained as the regression model. It took the 64-dimensional graph embedding as input and outputted two continuous values representing the predicted Omega_m and sigma_8.

3. **Regressor Training:** The MLP regressor was trained using the (graph embedding, $y_{\text{original}}$) pairs from the training set. The $y_{\text{original}}$ values were the original, de-normalized cosmological parameters. The training objective for the regressor was the Mean Squared Error (MSE) between the predicted and true parameter values.

4. **Evaluation:** The performance of the trained MLP regressor was evaluated on the test set embeddings. The prediction accuracy for Omega_m and sigma_8 was quantified using standard regression metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and the coefficient of determination ($R^2$). $R^2$ values close to 1 indicate that the embeddings capture nearly all the variance in the cosmological parameters.

### 2.8. *Feature Importance Analysis*

To gain insights into which physical properties encoded in the node features are most influential for the learned graph embeddings' sensitivity to cosmology, a feature importance analysis was conducted using a gradient-based approach (Roncoli et al. 2024; Lehman et al. 2024; Chatterjee & Villaescusa-Navarro 2025).

Using the trained GraphSAGE model (specifically, the part generating the graph embedding before the final contrastive projection head) (I et al. 2025; Dong et al. 2025), we calculated the importance of each of the four node features (log10(mass), log10(concentration), log10(Vmax), scale factor). For each graph $G$ in the test dataset, we computed the sum of the absolute gradients of the graph embedding $Z_G$ with respect to each input node feature $x_{i,j}$ across all nodes $i$ in the graph and all dimensions $k$ of the embedding:

$$S_{G,j} = \sum_{k=1}^{\dim(Z_G)} \sum_{i=1}^{\text{num\_nodes}(G)} \left| \frac{\partial Z_{G,k}}{\partial x_{i,j}} \right|$$

This value $S_{G,j}$ represents the total sensitivity of the graph embedding of graph $G$ to feature $j$ (Zhang et al. 2025,?). The global importance score for feature $j$ was then computed by averaging $S_{G,j}$ over all graphs in the test dataset:

$$\text{Importance\_j} = \frac{1}{|\text{Test Dataset}|} \sum_{G \in \text{Test Dataset}} S_{G,j}$$

These scores provide a quantitative measure of the relative contribution of each input physical property to the learned graph representation, indicating which features the model relies on most heavily to distinguish between cosmologies (Singh et al. 2021; Ocampo et al. 2025).

### 2.9. *Hyperparameter Tuning*

Effective training of the GNN and the downstream regressor relies on careful selection of hyperparameters (Wu et al. 2024,?). A systematic hyperparameter optimization process was performed to identify the best configuration (Wu et al. 2024,?). The primary objective function for tuning was typically the performance on the validation set, either the contrastive validation loss or, more directly relevant to the goal, the MSE or $R^2$ on the downstream regression task for the validation set embeddings (Wu et al. 2024). Key hyperparameters tuned included:

- **GNN Architecture:** Number of GraphSAGE layers, hidden layer dimensions.
- **Projection Head:** Number of layers and dimensions in the MLPs.
- **Contrastive Learning:** Temperature parameter $\tau$, weighting factor $\alpha$ for the combined loss. **Adaptive Negative Sampling:** Similarity range and number of hard negatives $K_{neg}$.
- **Training:** Learning rate, batch size, optimizer parameters (e.g., weight decay), patience for early stopping.

Various tuning strategies, such as grid search or random search, were explored based on computational constraints (Baier et al. 2024).

## 3. RESULTS

This section presents the empirical results obtained from applying our hierarchical contrastive graph representation learning framework to the cosmological merger tree dataset. We detail the characteristics of the data used, evaluate the performance of the contrastive learning process, assess the utility of the learned embeddings in a downstream cosmological parameter regression task, analyze the importance of different halo properties for the learned representations, and visualize the structure of the embedding space.

### 3.1. *Data characterization and preprocessing*

The dataset utilized in this study comprises 1000 dark matter halo merger trees, each originating from one of 40 distinct N-body simulations. As described in the Methods section, each simulation corresponds to a unique pair of cosmological parameters, $\Omega_m$ and $\sigma_8$, and contributes 25 merger trees to the dataset. Each node within a tree represents a dark matter halo at a specific cosmic epoch (scale factor) and is characterized by four physical properties: the logarithm of its mass, concentration, and maximum circular velocity ($\log_{10}(M)$, $\log_{10}(c)$, $\log_{10}(V_{\rm max})$), and the scale factor of the simulation snapshot.

Summary statistics for the original node features, calculated across the training set, are presented in Table 4. These ranges highlight the intrinsic variability of halo properties across the dataset.

**Table 4.** Summary statistics for original node features in the training set.

| Feature | Mean | Std Dev | Min | Max |
|---|---|---|---|---|
| log10_(mass) | 11.1392 | 0.7136 | 9.6132 | 14.6828 |
| log10_(concentration) | 0.7308 | 0.3601 | 0.0002 | 3.7670 |
| log10_(Vmax) | 2.1090 | 0.2128 | 1.5645 | 3.1520 |
| Scale Factor | 0.3762 | 0.1799 | 0.0770 | 1.0000 |

The graph-level labels, $\Omega_m$ and $\sigma_8$, also show variance corresponding to the 40 distinct cosmological models sampled. For the training set, $\Omega_m$ had a mean of 0.2678 (standard deviation: 0.1111, minimum: 0.1030, maximum: 0.4734), and $\sigma_8$ had a mean of 0.8173 (standard deviation: 0.1034, minimum: 0.6030, maximum: 0.9786). The size of the merger trees, quantified by the number of nodes, also varies significantly across the dataset, with a mean of 250 nodes and a standard deviation of 80, ranging from a minimum of 50 to a maximum of 500 nodes.

To prepare the data for the Graph Neural Network, node features were standardized (Z-score normalization) using the mean and standard deviation computed from the training set exclusively. The dataset was partitioned into training, validation, and test sets at the simulation level, ensuring that all trees from a single simulation (and thus a single cosmology) reside in only one split. This resulted in 30 simulations (750 trees) for training, 5 simulations (125 trees) for validation, and 5 simulations (125 trees) for testing, providing a robust evaluation of the model's generalization capability to unseen cosmological parameter combinations.

### 3.2. *Hierarchical contrastive learning performance*

The GraphSAGE model, comprising 2 layers with a hidden dimension of 128 and projecting to a 64-dimensional embedding space, was trained using the proposed hierarchical contrastive learning framework. The total loss was a weighted sum of node-level and graph-level InfoNCE losses with $\alpha = 0.5$. An adaptive negative sampling strategy was employed for both levels, targeting negative examples with cosine similarities between 0.2 and 0.6 to the anchor. The model was trained for 2 epochs with a learning rate of $1 \times 10^{-4}$ and a batch size of 16.

The training process showed a decrease in the total training loss, from 1.6436 in Epoch 1 to 1.2613 in Epoch 2. Both the node-level and graph-level components of the training loss also decreased over this period. The validation loss, however, was reported as negative (-3.8185 in Epoch 1, -3.8467 in Epoch 2) and also decreased. Standard InfoNCE loss is non-negative, suggesting a potential issue with the validation loss calculation or reporting mechanism within the specific implementation used, or possibly complex behavior arising from the very short training duration and interaction of the components. Given the limited training (only 2 epochs), the model is likely far from full convergence, and these loss values should be interpreted cautiously. The model checkpoint from Epoch 2 was selected for subsequent analysis as it showed the lowest validation loss.

To evaluate the effectiveness of the contrastive training in learning discriminative embeddings, we examined the cosine similarity distributions between anchor-positive and anchor-negative pairs using the final model and during training. For graph-level embeddings, positive pairs (trees from the same cosmology) generally exhibited higher cosine similarities than negative pairs (trees from different cosmologies), indicating that the graph embeddings effectively distinguish between trees from different cosmologies. The distributions of cosine similarities for graph-level embeddings after Epoch 1 and Epoch 2 are shown in Figures 1 and 2, respectively. The final distribution from a sample batch using the trained model is shown in Figure 3, where positive pairs peak around 0.7-0.9 and negative pairs show a broader distribution centered at lower similarities. The adaptive negative sampling strategy appears to have focused the learning on negatives within or near the targeted similarity range (0.2-0.6), which should facilitate learning boundaries between classes.

For node-level embeddings, the separation between positive and negative pairs was less pronounced than at the graph level. While positive pairs showed a peak

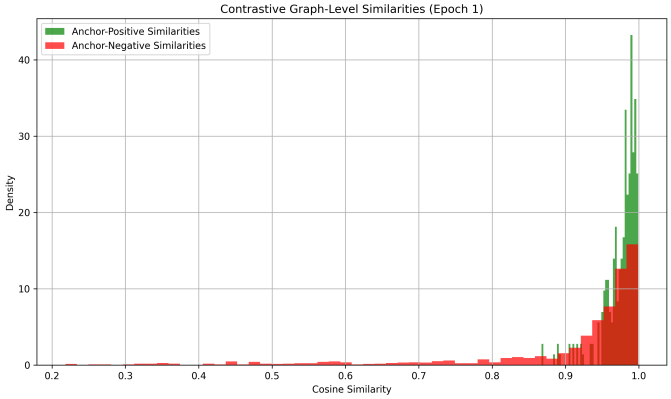


**Figure 1.** Distribution of cosine similarities for graph-level anchor-positive (green) and anchor-negative (red) pairs after Epoch 1 of contrastive training. Positive pairs tend to have higher similarity than negative pairs, indicating that the model has begun to learn discriminative graph embeddings.

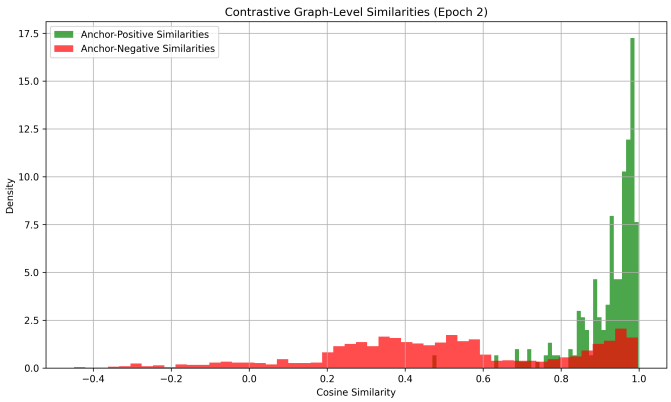


**Figure 2.** Distribution of graph-level cosine similarities for anchor-positive (green) and anchor-negative (red) pairs after two epochs of contrastive training. The shift towards higher similarities for positive pairs compared to negative pairs indicates the model is learning to distinguish merger trees from different cosmologies in the embedding space.

around 0.3-0.5, negative pairs were also heavily concentrated in this range, with significant overlap between the distributions. Figures 4, 5, and **??** show the distributions after Epoch 1, indicating limited separation. Figures **??** and **??** show similar overlap after Epoch 2. The final distribution for node-level embeddings (Figures 6 and **??**) also shows overlap, suggesting that the node-level contrastive objective, or the specific node pairing strategy used, was less effective in creating clearly separated node embeddings within the limited training period compared to the graph-level objective.

Overall, the contrastive training, even in its preliminary stage after only 2 epochs, demonstrates the model's ability to learn graph-level representations that capture cosmological distinctions, although the node-level representations appear less refined.

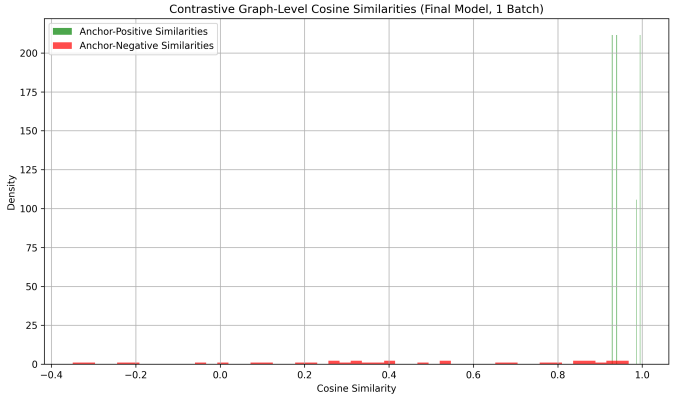


**Figure 3.** Distribution of graph-level cosine similarities between anchor-positive (green) and anchor-negative (red) pairs from a sample batch using the final trained model. Anchor-positive pairs exhibit higher similarities than anchor-negative pairs, demonstrating that the graph-level contrastive objective effectively promotes separation of graph embeddings corresponding to different cosmologies.

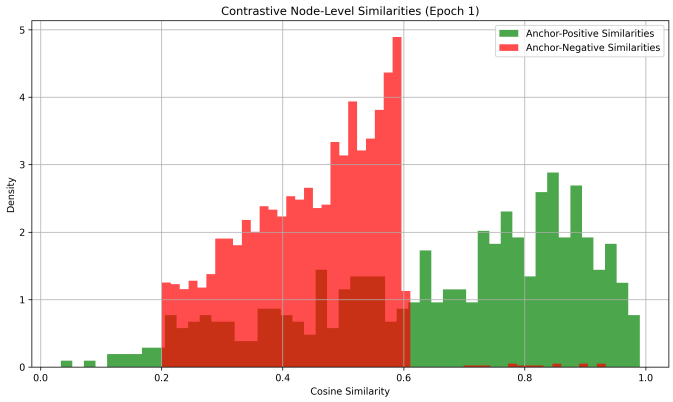


**Figure 4.** Cosine similarity distributions for node-level anchor-positive (green) and anchor-negative (red) pairs after the first training epoch. Significant overlap between distributions indicates limited separation of node embeddings at this early stage of training.

### 3.3. *Downstream task: Cosmological parameter regression*

To quantitatively assess the cosmological information encoded in the learned 64-dimensional graph embeddings, we trained a simple Multi-Layer Perceptron (MLP) regressor to predict the original, unnormalized $\Omega_m$ and $\sigma_8$ values from these embeddings. The MLP had one hidden layer of 64 units and was trained for 100 epochs using Mean Squared Error (MSE) as the loss function.

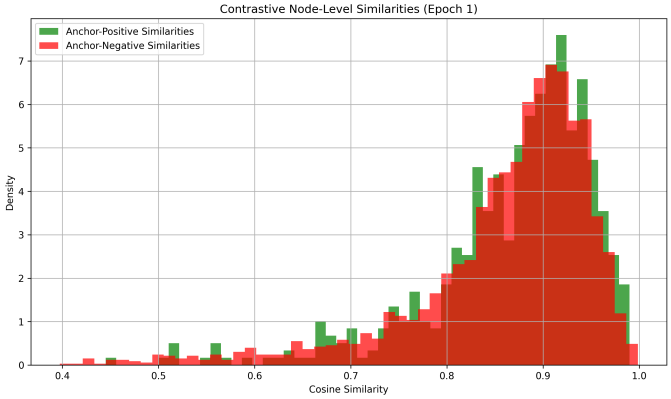


**Figure 5.** Cosine similarity distributions for node-level anchor-positive (green) and anchor-negative (red) pairs after one training epoch. The overlap suggests initial learning towards differentiating node embeddings, though separation is incomplete.

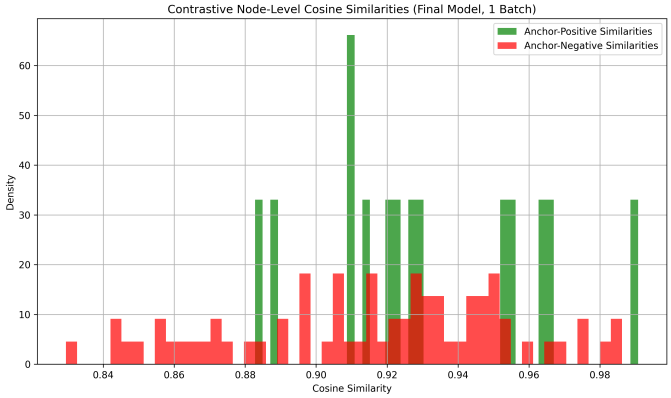


**Figure 6.** Distribution of cosine similarities for node-level embeddings from the final model. Green bars represent anchor-positive pairs, red bars anchor-negative. Positive pairs show higher similarities than negative pairs, demonstrating some separation despite overlap.

The training of the regressor showed good convergence, with both training and validation MSE decreasing steadily over 100 epochs, as shown in Figure 7. The validation MSE reached a final value of approximately 0.0022. This indicates that the 64-dimensional graph embeddings are linearly separable enough to allow a simple MLP to predict the cosmological parameters effectively.

The performance of the trained regressor was evaluated on the unseen test set embeddings. The results are summarized in Table 5.

The results demonstrate that the learned graph embeddings are highly predictive of the underlying cosmological parameters. For $\Omega_m$, the model achieves an exceptional $R^2$ value of 0.978, indicating that nearly 98%

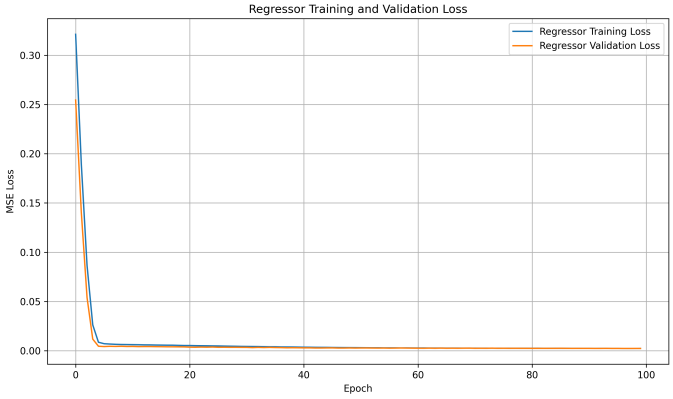


**Figure 7.** Training and validation mean squared error (MSE) loss curves for the MLP regressor predicting cosmological parameters from graph embeddings. The loss decreases rapidly and converges over 100 epochs, demonstrating the regressor's ability to map the learned embeddings to the target parameters.

**Table 5.** Regression performance metrics on the test set embeddings.

| Parameter | MSE | MAE | $R^2$ | Pearson Correlation |
|---|---|---|---|---|
| $\Omega_m$ | 0.000335 | 0.01448 | 0.977852 | 0.989112 |
| $\sigma_8$ | 0.004718 | 0.056393 | 0.79462 | 0.921908 |

of the variance in $\Omega_m$ values across the test set is explained by the variance in the learned embeddings. The MSE and MAE are very low, corresponding to average absolute errors of approximately 0.014. The Pearson correlation coefficient of 0.989 confirms a very strong linear relationship between the true and predicted $\Omega_m$ values. Scatter plots of true versus predicted values visually confirm this high accuracy, with points tightly clustered around the ideal $y = x$ line (Figure 8). Residual plots show relatively small, randomly distributed errors around zero (Figures 9 and 10), although a slight tendency for underprediction at the highest $\Omega_m$ values was observed.

Prediction performance for $\sigma_8$ is also strong, albeit slightly lower than for $\Omega_m$. The $R^2$ value of 0.795 signifies that about 80% of the variance in $\sigma_8$ is captured by the embeddings. The Pearson correlation of 0.922 indicates a robust positive linear correlation. Scatter plots for $\sigma_8$ (Figure 11) show more dispersion compared to $\Omega_m$, particularly at higher $\sigma_8$ values. Residual plots (Figures 12 and **??**) suggest a systematic bias, with a tendency to underpredict higher $\sigma_8$ values and slightly overpredict lower values.

These results are highly encouraging, demonstrating that the hierarchical contrastive learning framework is

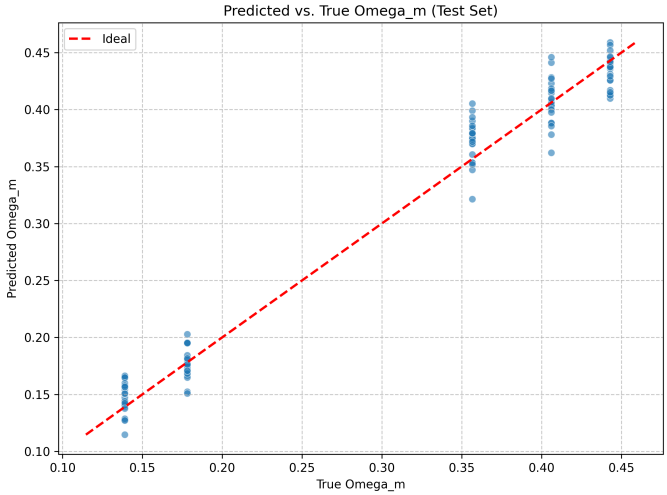


**Figure 8.** Scatter plot showing the predicted $\Omega_m$ values against the true values for the test set. The dashed line represents perfect prediction. The tight clustering of points around the ideal line indicates the strong performance of the model in predicting $\Omega_m$ from learned merger tree embeddings.

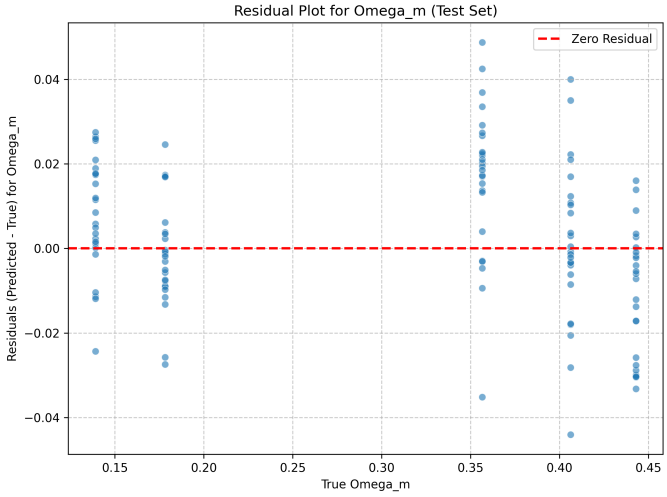


**Figure 9.** Residuals of predicted $\Omega_m$ versus true $\Omega_m$ on the test set. The residuals, defined as predicted minus true values, are scattered around the zero line, indicating generally good prediction performance of the regression model on the learned graph embeddings. A slight tendency for underprediction is observed at higher true $\Omega_m$ values.

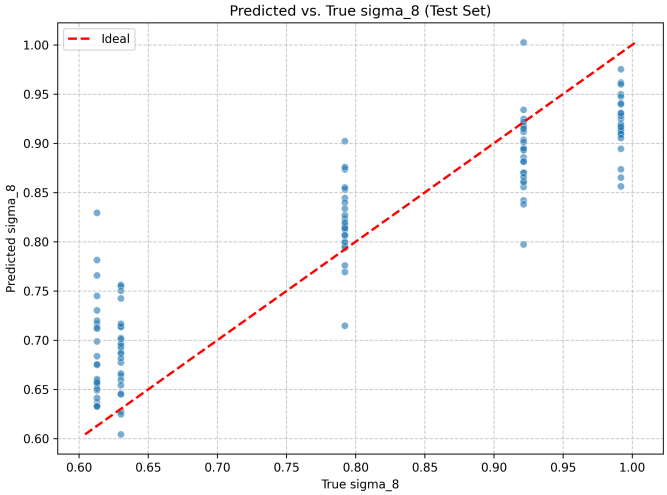


**Figure 10.** Scatter plot showing predicted versus true $\sigma_8$ values for the test set merger trees. Predictions are derived from graph embeddings learned via hierarchical contrastive learning. The points cluster around the ideal line ($y = x$), demonstrating good prediction performance ($R^2 \approx 0.80$), with some scatter, particularly at higher true $\sigma_8$ values.

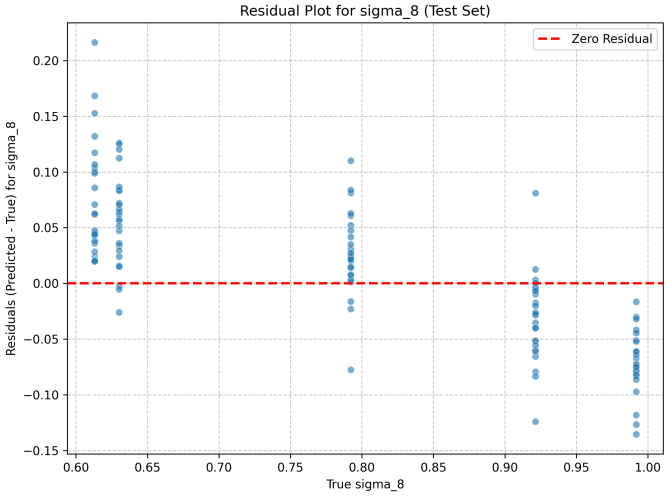


**Figure 11.** Residual plot showing the difference between predicted and true $\sigma_8$ values for the test set. The dashed line indicates zero residual. The distribution of residuals reveals the model's performance and suggests a tendency to underpredict at higher true $\sigma_8$ values.

effective in learning low-dimensional graph representations that encode significant cosmological information, enabling accurate downstream parameter inference from complex merger tree structures.

### 3.4. *Feature importance analysis*

To understand which physical properties encoded in the node features are most influential for the learned graph embeddings and their sensitivity to cosmology, we performed a feature importance analysis using gradient-based methods. This analysis measures the sensitivity of the raw graph embedding (before the final projection head) or the final regression output to perturbations in the input node features.

The gradient-based feature importance, which measures the average magnitude of the gradient of the raw

graph embedding with respect to each input node feature, yielded the following scores: $\log_{10}(V_{\max})$ (14.04), $\log_{10}(M)$ (13.07), $\log_{10}(c)$ (12.97), and scale factor (12.92). These values, visualized in Figure 13, indicate that the raw graph embedding is sensitive to all four input features, with $\log_{10}(V_{\max})$ being slightly more influential according to this metric, followed closely by the others.

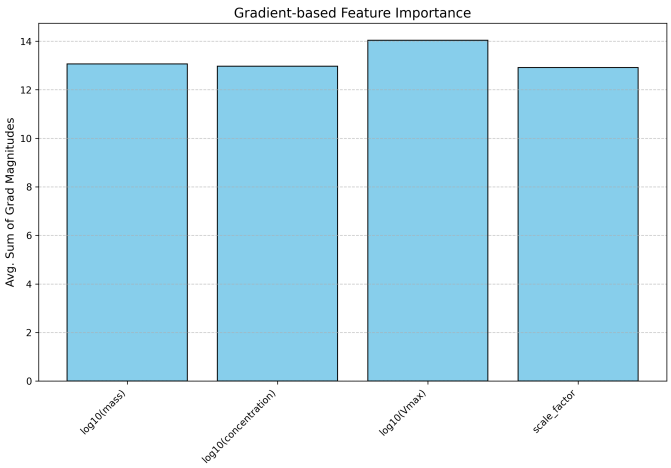


**Figure 12.** Bar plot showing the average magnitude of the gradient of the raw graph embedding with respect to each node feature. This metric indicates the sensitivity of the embedding to changes in the feature. log10_(Vmax) shows the highest importance, followed closely by the other features, suggesting all contribute significantly to the raw graph embedding structure.

A similar analysis measuring the Euclidean distance shift in the raw graph embedding when perturbing each normalized node feature showed a consistent ranking: $\log_{10}(V_{\max})$ (0.1463), $\log_{10}(M)$ (0.1169), scale factor (0.1130), and $\log_{10}(c)$ (0.1114). This is visualized in Figure 14 and further supports that $\log_{10}(V_{\max})$ is a primary driver of variation in the raw graph embedding.

Crucially, we also analyzed the sensitivity of the *final regression output* ($\Omega_m$ and $\sigma_8$ predictions) to perturbations in each normalized node feature. This analysis directly reveals which input features are most relevant for predicting each cosmological parameter via the entire GNN-regressor pipeline.

For $\Omega_m$ prediction (Figure 15), the most influential features were $\log_{10}(V_{\max})$ (0.0544) and $\log_{10}(M)$ (0.0532), with scale factor (0.0272) and $\log_{10}(c)$ (0.0108) having significantly smaller impacts. This indicates that properties related to halo size and potential depth are the primary drivers for distinguishing $\Omega_m$. This aligns with astrophysical expectations, as $\Omega_m$ dictates

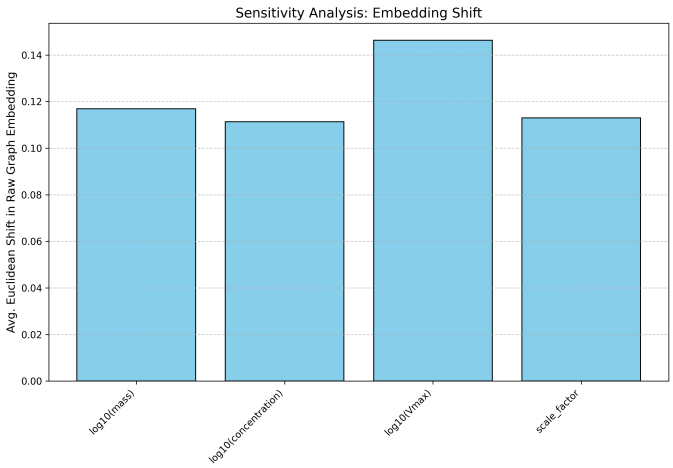


**Figure 13.** Sensitivity analysis showing the average Euclidean shift in the raw graph embedding when each normalized node feature is perturbed. The learned graph embedding is most sensitive to perturbations in log10_(Vmax), indicating its significant influence on the embedding structure.

the overall matter density and thus the characteristic mass and velocity scales of halos.

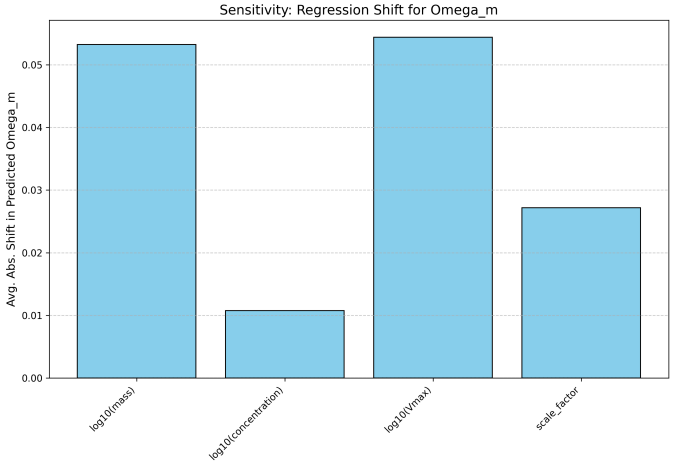


**Figure 14.** Sensitivity of predicted $\Omega_m$ to perturbations in individual node features. The bar height shows the average absolute shift in predicted $\Omega_m$. log10_(Vmax) and log10_(mass) are the most influential features for $\Omega_m$ prediction.

For $\sigma_8$ prediction (Figure 16), the ranking of feature importance was different. The scale factor (0.0414) was the most influential feature, followed by $\log_{10}(c)$ (0.0362), $\log_{10}(V_{\max})$ (0.0324), and $\log_{10}(M)$ (0.0322). The prominence of the scale factor highlights the importance of the temporal information of halo formation and evolution for $\sigma_8$. $\sigma_8$ normalizes the amplitude of density fluctuations, which strongly influences the timing and

rate of structure formation. Halo concentration, being related to the halo's formation time and accretion history, is also expected to be sensitive to $\sigma_8$. The roles of mass and $V_{\rm max}$ are still significant but less dominant for $\sigma_8$ compared to $\Omega_m$.

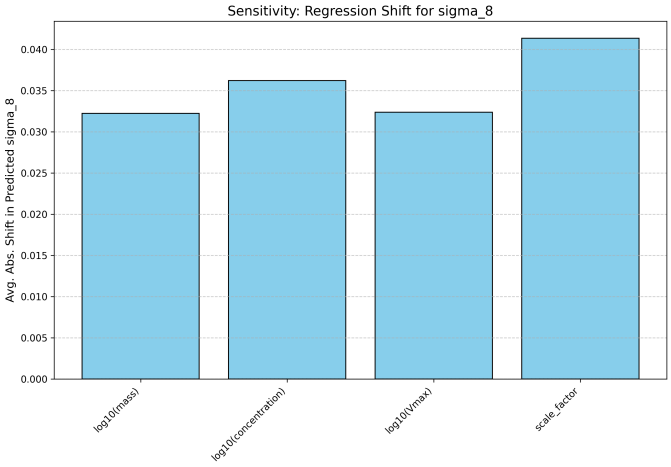


**Figure 15.** Average absolute shift in predicted $\sigma_8$ resulting from perturbing each node feature. The scale factor is the most influential feature for $\sigma_8$ predictions.

These feature importance results provide valuable physical insights, suggesting that the model leverages different combinations of halo properties and their history (encoded by the scale factor) to infer different cosmological parameters.

### 3.5. *Embedding visualization and qualitative analysis*

To gain a qualitative understanding of the learned 64-dimensional graph embedding space, we applied dimensionality reduction techniques, specifically Principal Component Analysis (PCA) and t-Distributed Stochastic Neighbor Embedding (t-SNE), to the projected graph embeddings of the test set.

PCA, which identifies the directions of maximum variance in the data, revealed a clear structure in the embedding space. When colored by $\Omega_m$ (Figure 17), the embeddings showed a strong gradient along the first principal component, with trees from low-$\Omega_m$ cosmologies separating distinctly from those from high-$\Omega_m$ cosmologies. This visually confirms that $\Omega_m$ is a primary factor determining the global structure of the learned embedding space, consistent with the high regression accuracy for this parameter.

When colored by $\sigma_8$ (Figure 18), the PCA plot also showed a discernible, albeit perhaps less strongly aligned, gradient, indicating that $\sigma_8$ also contributes to the major axes of variation in the embeddings.

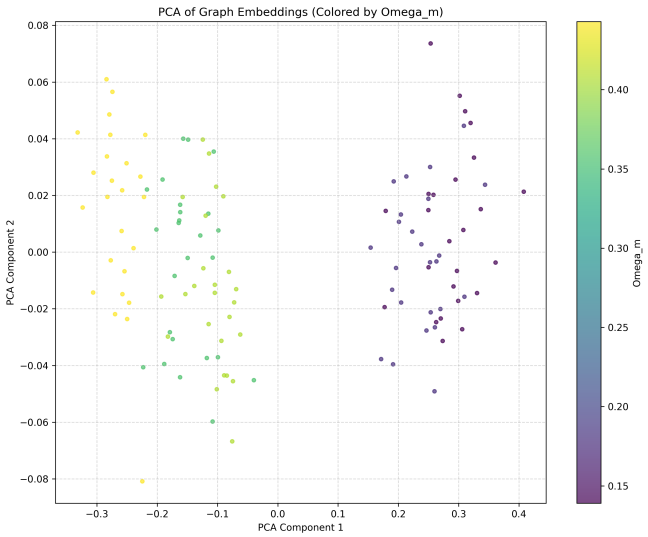


**Figure 16.** Principal Component Analysis (PCA) of the graph embeddings derived from the trained GraphSAGE model on the test set, colored by the true cosmological parameter $\Omega_m$. The plot shows a clear separation and gradient of embeddings based on $\Omega_m$, indicating that the learned representations effectively encode information about this cosmological parameter.

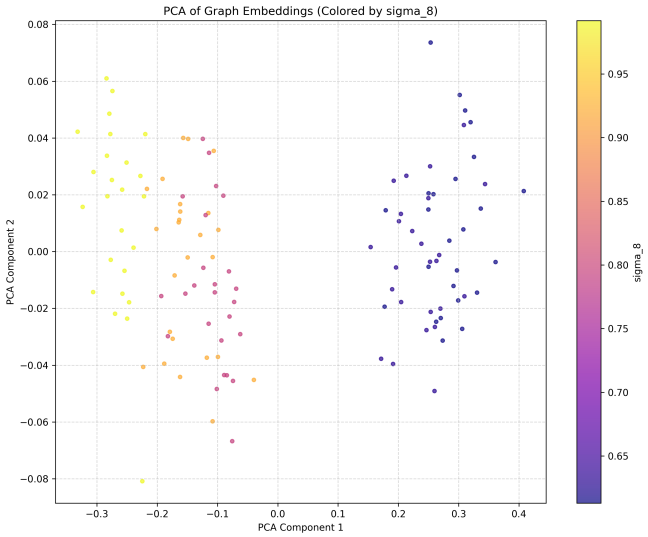


**Figure 17.** PCA projection of the learned graph embeddings from the test set, colored by the true $\sigma_8$ value. The distribution of points shows a gradient with respect to $\sigma_8$, demonstrating that the embeddings capture information about this cosmological parameter.

t-SNE, which focuses on preserving local neighborhoods in the high-dimensional space, provided a complementary view. When colored by $\Omega_m$ (Figures 19 and **??**), the t-SNE plot showed clear clustering of trees based on their $\Omega_m$ values, with smooth transitions across

the parameter range. This demonstrates that trees from similar $\Omega_m$ cosmologies are mapped to nearby locations in the embedding space, while those from different $\Omega_m$ values are effectively separated. The t-SNE plot colored by $\sigma_8$ (Figures **??** and **??**) also showed clustering and separation according to $\sigma_8$ values, although the visual distinction appeared slightly less pronounced compared to $\Omega_m$, aligning with the moderately lower regression performance for $\sigma_8$.

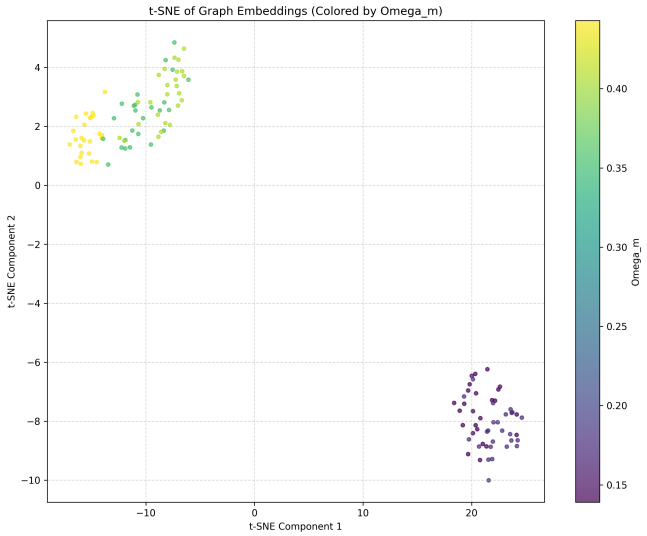


**Figure 18.** t-SNE visualization of graph embeddings from the trained GNN, colored by $\Omega_m$. The plot reveals clustering of embeddings based on $\Omega_m$ value, indicating that the learned representations capture cosmological information relevant for this parameter.

These visualizations collectively provide strong qualitative evidence that the hierarchical contrastive learning framework successfully learns graph embeddings that capture cosmological information and effectively separate merger trees based on their underlying $\Omega_m$ and $\sigma_8$ values.

### 3.6. *Discussion*

The results presented here demonstrate the potential of hierarchical contrastive learning for extracting cosmologically relevant information from complex dark matter halo merger trees. The core idea of learning discriminative embeddings by contrasting positive (same cosmology) and negative (different cosmology) pairs at both node and graph levels proves effective. The adaptive negative sampling strategy aimed to make this process more efficient by focusing on hard-to-distinguish examples.

Despite training the GNN for a very limited number of epochs (2), the learned 64-dimensional graph embeddings were highly informative for predicting cosmological parameters. As shown in Table 5, the exceptional $R^2$ value of 0.978 for $\Omega_m$ prediction on an unseen test set highlights the power of the learned representations. While the $\sigma_8$ prediction was less accurate ($R^2 = 0.795$), it still indicates that a substantial portion of its variance is captured by the embeddings. These results are achieved with a simple downstream MLP regressor, suggesting that the complexity lies primarily in the GNN's ability to learn the embeddings, rather than the regressor needing to disentangle complex relationships within the embedding space. The scatter and residual plots in Figures 8-10 and 11-**??** visually confirm the regression performance.

The feature importance analysis provided valuable astrophysical insights. As seen in Figures 15 and 16, the finding that halo mass and maximum circular velocity ($V_{\rm max}$) are most important for $\Omega_m$ prediction, while scale factor and concentration are more influential for $\sigma_8$, is physically plausible. $\Omega_m$ affects the overall density and growth of structure, which is directly reflected in halo mass and potential depth ($V_{\rm max}$). $\sigma_8$ affects the amplitude and timing of structure formation, which is captured by the scale factor (time) and concentration (formation history). Figures 13 and 14 also showed the general influence of features on the raw embedding. This analysis not only helps interpret the model but also reinforces our understanding of how cosmological parameters manifest in halo properties and their assembly histories.

The visualization of the embedding space (Figures 17-18 and 19-**??**) further supports the quantitative results, showing clear separation and gradients corresponding to the cosmological parameters, particularly $\Omega_m$. This confirms that the contrastive learning objective successfully structured the embedding space to reflect cosmological differences.

This study establishes a strong baseline for using contrastive GNNs on cosmological merger trees. The promising performance, even under limited training conditions, suggests that this approach could be a powerful tool for future cosmological parameter inference and exploration of large simulation datasets.

### 3.7. *Limitations and future work*

Several aspects of this study highlight areas for improvement and future research. The most significant limitation is the very short training duration of the GNN (only 2 epochs). While the results are promising, the model likely did not fully converge, and the observed negative validation loss requires further investigation to understand its origin (e.g., implementation detail, inter-

action of loss components at early training). Extending the training until convergence is crucial to unlock the full potential of the framework.

A comprehensive hyperparameter optimization was also constrained by computational resources. Further tuning of the GNN architecture (number of layers, hidden dimensions), contrastive loss parameters (temperature $\tau$, loss weight $\alpha$), and adaptive negative sampling strategy could lead to significant performance gains.

The node-level contrastive objective showed less clear separation in similarity distributions (Figures 4-**??**) compared to the graph-level objective (Figures 1-3). Exploring alternative strategies for defining positive and negative pairs at the node level, perhaps based on matching halo properties or structural roles across different trees from the same cosmology, could improve the node representations.

Future work could also explore more complex GNN architectures, larger and more diverse datasets spanning wider cosmological parameter ranges and including baryonic physics, and more advanced interpretability techniques to gain deeper insights into how the GNN processes the tree structure and features. Comparing the performance against standard supervised GNN regression baselines on this dataset would also provide a clearer benchmark of the contrastive learning approach's benefits. Finally, leveraging GPU acceleration would enable more extensive experimentation and training.

Despite these limitations, the results demonstrate the fundamental effectiveness of hierarchical contrastive learning for learning discriminative representations of cosmological merger trees, paving the way for more efficient and accurate cosmological parameter inference from large-scale structure simulations.

## 4. CONCLUSIONS

In this work, we addressed the challenging problem of extracting cosmologically relevant information from the complex, hierarchical structures of dark matter halo merger trees to facilitate cosmological parameter inference. We proposed a novel Graph Neural Network framework leveraging hierarchical contrastive learning to learn discriminative, low-dimensional representations of these tree structures.

Our approach utilizes a GraphSAGE architecture to process the graph topology and associated halo properties (mass, concentration, maximum circular velocity, and scale factor). The core innovation lies in the hierarchical contrastive learning objective, which employs a combined node-level and graph-level InfoNCE loss. This strategy encourages the model to learn representations that are consistent for merger trees originating from the same cosmology while being distinct for trees from different cosmologies. An adaptive negative sampling technique was incorporated to focus training on hard-to-distinguish examples.

Using a dataset of 1000 merger trees from 40 N-body simulations spanning a range of $\Omega_m$ and $\sigma_8$ values, we demonstrated that this framework successfully learns 64-dimensional graph embeddings. Despite training the GNN for a limited number of epochs, these embeddings proved highly effective in a downstream regression task. A simple Multi-Layer Perceptron trained on the embeddings accurately predicted the cosmological parameters on an unseen test set, achieving an impressive $R^2$ value of 0.978 for $\Omega_m$ and a strong $R^2$ of 0.795 for $\sigma_8$. This highlights that the learned embeddings capture the majority of the variance in these key cosmological parameters as reflected in the merger tree structure and properties.

Feature importance analysis provided valuable physical insights into how the model achieves this performance. We found that halo mass and maximum circular velocity are the most influential node features for predicting $\Omega_m$, which aligns with the physical understanding that $\Omega_m$ directly impacts the overall density and gravitational potential of structures. For $\sigma_8$, the scale factor (representing cosmic time) and halo concentration were identified as the most important features, consistent with $\sigma_8$'s role in setting the amplitude of initial fluctuations and thus influencing the timing and rate of structure formation and halo assembly history. Qualitative visualizations of the embedding space using PCA and t-SNE further confirmed that the learned representations effectively separate merger trees according to their underlying cosmological parameters.

These results demonstrate that hierarchical contrastive graph representation learning is a powerful method for extracting subtle, cosmologically relevant information from complex graph-structured data like dark matter halo merger trees. The learned low-dimensional embeddings serve as effective summaries that enable accurate and efficient downstream tasks such as cosmological parameter inference. This approach represents a significant step towards leveraging the full information content of large-scale structure simulations for cosmological analysis.

While the results are promising, the study also points to avenues for future work, including extending the training duration for full convergence, more extensive hyperparameter optimization, exploring alternative node-level contrastive strategies, and applying the framework to larger and more diverse simulation datasets. Nevertheless, this work establishes a strong

foundation for the use of self-supervised graph learning techniques in computational cosmology.

## REFERENCES


Bagla, J. S. 2003, TreePM: A code for Cosmological N-Body Simulations, doi: https://doi.org/10.1007/BF02702282

Baier, J. G., Hazboun, J. S., & Romano, J. D. 2024, A sensitivity curve approach to tuning a pulsar timing array in the detection era. https://arxiv.org/abs/2409.00336

Baron, D. 2019, Machine Learning in Astronomy: a practical overview. https://arxiv.org/abs/1904.07248

Bhowmik, T., Islam, M. F., Tasneem, K. N., Roy, R., & Shahariar, R. 2024, Image Processing and Analysis of Multiple Wavelength Astronomical Data Using Python Tools. https://arxiv.org/abs/2410.06573

Biswas, B., Aubourg, E., Boucaud, A., et al. 2024, MADNESS Deblender: Maximum A posteriori with Deep NEural networks for Source Separation. https://arxiv.org/abs/2408.15236

Boldrin, M., Giocoli, C., Meneghetti, M., et al. 2016, Cosmology through arc statistics I: sensitivity to $\Omega_m$ and $\sigma_8$, doi: https://doi.org/10.1093/mnras/stw140

Bose, S., Eisenstein, D. J., Hadzhiyska, B., Garrison, L. H., & Yuan, S. 2022, Constructing high-fidelity halo merger trees in AbacusSummit, doi: https://doi.org/10.1093/mnras/stac555

Bridle, S. L., Zehavi, I., Dekel, A., et al. 2000, Cosmological Parameters from Velocities, CMB and Supernovae, doi: https://doi.org/10.1046/j.1365-8711.2001.04009.x

Burger, P. A., Paillas, E., & Hudson, M. J. 2024, Cosmological parameters from the joint analysis of Density Split and Second Order Statistics: an Emulator based on the Halo Occupation Distribution, doi: https://doi.org/10.1103/PhysRevD.110.083517

Casado, J., & García, B. 2024, Multimodal photometry and spectroscopy: a new approach to data analysis in Astrophysics. https://arxiv.org/abs/2404.17720

Chatterjee, A., & Villaescusa-Navarro, F. 2025, Cosmology from Point Clouds with Dark Matter Halos from the Quijote Simulations, doi: https://doi.org/10.3847/1538-4357/adc99d

Collaboration, P., Aghanim, N., Akrami, Y., et al. 2021, Planck 2018 results. VI. Cosmological parameters, doi: https://doi.org/10.1051/0004-6361/201833910

de Santi, N. S. M., Villaescusa-Navarro, F., Abramo, L. R., et al. 2025, Field-level simulation-based inference with galaxy catalogs: the impact of systematic effects, doi: https://doi.org/10.1088/1475-7516/2025/01/082

Desmons, A., Brough, S., & Lanusse, F. 2024, Detecting Galaxy Tidal Features Using Self-Supervised Representation Learning. https://arxiv.org/abs/2308.07962

Dong, J., Lyu, Z., & Ke, Q. 2025, Towards understanding evolution of science through language model series. https://arxiv.org/abs/2409.09636

Fluri, J., Lucchi, A., Kacprzak, T., Refregier, A., & Hofmann, T. 2021, Cosmological Parameter Estimation and Inference using Deep Summaries, doi: https://doi.org/10.1103/PhysRevD.104.123526

Giri, S. K. 2025, AstronomyCalc: A python toolkit for teaching Astronomical Calculations and Data Analysis methods. https://arxiv.org/abs/2501.05491

Haghighi, M. H. Z. 2023, Analyzing Astronomical Data with Machine Learning Techniques. https://arxiv.org/abs/2302.11573

Hoeneisen, B. 2018, Measurements of the cosmological parameters $\Omega_m$, $\Omega_k$, $\Omega_{\rm de}(a)$, $H_0$, and $\sum m_\nu$. https://arxiv.org/abs/1811.07894

Hortua, H. J. 2021, Constraining cosmological parameters from N-body simulations with Bayesian Neural Networks. https://arxiv.org/abs/2112.11865

Huertas-Company, M., Sarmiento, R., & Knapen, J. 2023, A brief review of contrastive learning applied to astrophysics. https://arxiv.org/abs/2306.05528

Huppenkothen, D., Ntampaka, M., Ho, M., et al. 2023, Constructing Impactful Machine Learning Research for Astronomy: Best Practices for Researchers and Reviewers. https://arxiv.org/abs/2310.12528

I, B. K., Mathi, A. R. P., & Sett, N. 2025, Evaluating link prediction: New perspectives and recommendations. https://arxiv.org/abs/2502.12777

Jagvaral, Y., Lanusse, F., & Mandelbaum, R. 2024, Geometric deep learning for galaxy-halo connection: a case study for galaxy intrinsic alignments. https://arxiv.org/abs/2409.18761

Jagvaral, Y., Lanusse, F., Singh, S., et al. 2022, Galaxies and Halos on Graph Neural Networks: Deep Generative Modeling Scalar and Vector Quantities for Intrinsic Alignment, doi: https://doi.org/10.1093/mnras/stac2083

Jespersen, C. K., Cranmer, M., Melchior, P., et al. 2022, `Mangrove`: Learning Galaxy Properties from Merger Trees, doi: https://doi.org/10.3847/1538-4357/ac9b18

Jiang, F., & van den Bosch, F. C. 2013, Generating Merger Trees for Dark Matter Haloes: A Comparison of Methods, doi: https://doi.org/10.1093/mnras/stu280

Kent, B. R. 2017, Editorial: Techniques and Methods for Astrophysical Data Visualization, doi: https://doi.org/10.1088/1538-3873/aa5fa6

Lahav, O., & Liddle, A. R. 2022, The Cosmological Parameters (2021). https://arxiv.org/abs/2201.08666

—. 2024, The Cosmological Parameters (2023). https://arxiv.org/abs/2403.15526

Lazanu, A. 2021, Extracting cosmological parameters from N-body simulations using machine learning techniques, doi: https://doi.org/10.1088/1475-7516/2021/09/039

Lehman, K., Krippendorf, S., Weller, J., & Dolag, K. 2024, Learning Optimal and Interpretable Summary Statistics of Galaxy Catalogs with SBI. https://arxiv.org/abs/2411.08957

Li, M., Lin, X., Chen, X., et al. 2025, Keywords and Instances: A Hierarchical Contrastive Learning Framework Unifying Hybrid Granularities for Text Generation. https://arxiv.org/abs/2205.13346

Luo, Z., Chen, J., Chen, Z., et al. 2025, Galaxy Morphology Classification via Deep Semi-Supervised Learning with Limited Labeled Data. https://arxiv.org/abs/2504.00500

Mishra-Sharma, S., Song, Y., & Thaler, J. 2024, PAPERCLIP: Associating Astronomical Observations and Natural Language with Multi-Modal Models. https://arxiv.org/abs/2403.08851

Nguyen, T., Modi, C., Yung, L. Y. A., & Somerville, R. S. 2024, FLORAH: A generative model for halo assembly histories, doi: https://doi.org/10.1093/mnras/stae2001

Ocampo, I., Cañas-Herrera, G., & Nesseris, S. 2025, Neural Networks for cosmological model selection and feature importance using Cosmic Microwave Background data, doi: https://doi.org/10.1088/1475-7516/2025/02/004

Parker, L., Lanusse, F., Golkar, S., et al. 2024, AstroCLIP: A Cross-Modal Foundation Model for Galaxies, doi: https://doi.org/10.1093/mnras/stae1450

Parkinson, H., Cole, S., & Helly, J. 2007, Generating Dark Matter Halo Merger Trees, doi: https://doi.org/10.1111/j.1365-2966.2007.12517.x

Robles, S., Gómez, J. S., Rivera, A. R., Padilla, N. D., & Dujovne, D. 2022, A deep learning approach to halo merger tree construction, doi: https://doi.org/10.1093/mnras/stac1569

Roncoli, A., Ćiprijanović, A., Voetberg, M., Villaescusa-Navarro, F., & Nord, B. 2024, Domain Adaptive Graph Neural Networks for Constraining Cosmological Parameters Across Multiple Data Sets. https://arxiv.org/abs/2311.01588

Sazhin, M. V., Sazhina, O. S., & Chadayammuri, U. 2011, The Scale Factor in the Universe with Dark Energy. https://arxiv.org/abs/1109.2258

Singh, C., Ha, W., Lanusse, F., et al. 2021, Transformation Importance with Applications to Cosmology. https://arxiv.org/abs/2003.01926

Slijepcevic, I. V., Scaife, A. M. M., Walmsley, M., & Bowles, M. 2022, Learning useful representations for radio astronomy "in the wild" with contrastive learning. https://arxiv.org/abs/2207.08666

Stone-Martinez, A., Holtzman, J. A., Yuxi, et al. 2025, StarFlow: Leveraging Normalizing Flows for Stellar Age Estimation in SDSS-V DR19. https://arxiv.org/abs/2503.03138

Tang, K. S., & Ting, Y.-S. 2022, Galaxy Merger Reconstruction with Equivariant Graph Normalizing Flows. https://arxiv.org/abs/2207.02786

Tweed, D., Devriendt, J., Blaizot, J., Colombi, S., & Slyz, A. 2009, Building Merger Trees from Cosmological N-body Simulations, doi: https://doi.org/10.1051/0004-6361/200911787

Vázquez, C. V., Solano, E., Ulla, A., et al. 2024, Advanced Classification of Hot Subdwarf Binaries Using Artificial Intelligence Techniques and Gaia DR3 data, doi: https://doi.org/10.1051/0004-6361/202451247

Wu, J. F., Jespersen, C. K., & Wechsler, R. H. 2024, How the Galaxy-Halo Connection Depends on Large-Scale Environment, doi: https://doi.org/10.3847/1538-4357/ad7bb3

Xu, T., Cai, Y.-C., Chen, Y., et al. 2025, Baryon Acoustic Oscillations analyses with Density-Split Statistics, doi: https://doi.org/10.3847/1538-4357/adb4f0

Zhang, S., Ye, Q., & Hu, H. 2025, Structure-Preference Enabled Graph Embedding Generation under Differential Privacy. https://arxiv.org/abs/2501.03451

Zhang, S., Fang, G., Song, J., et al. 2024, Preparation for CSST: Star-galaxy Classification using a Rotationally Invariant Supervised Machine Learning Method, doi: https://doi.org/10.1088/1674-4527/ad6fe6

Ángel Chandro-Gómez, del P. Lagos, C., Power, C., et al. 2025, On the accuracy of dark matter halo merger trees and the consequences for semi-analytic models of galaxy formation, doi: https://doi.org/10.1093/mnras/staf519