QITT-Enhanced Multi-Scale Substructure Analysis with Learned Topological Embeddings for Cosmological Parameter Estimation

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

Pablo Bermejo
Donostia International physics Center, San Sebastián, Spain
Department of Applied Physics, University of the Basque Country, San Sebastián, Spain
Flatiron Institute, Center for Computational Quantum Physics, New York, US

Boris Bolliet
Kavli Institute for Cosmology, University of Cambridge, Cambridge CB3 0HA, UK
Department of Physics, University of Cambridge, Cambridge, CB3 0US, UK

Francisco Villaescusa-Navarro
Center for Computational Astrophysics, Flatiron Institute, New York, NY 10010, USA
Department of Astrophysical Sciences, Princeton University, Princeton, NJ 08544, USA

Pablo Villanueva-Domingo
Computer Vision Center, Universitat Autonoma de Barcelona, Bellaterra, Barcelona, Spain

Chetana Amancharla
Infosys Limited, Banglore, India

Urbano L. França
Boston Children's Hospital, Department of Anesthesiology, Boston, MA, 02115, USA
Harvard Medical School, Boston, MA, 02115, USA

ChangHoon Hahn
Department of Astronomy, The University of Texas at Austin, Austin, TX 78712, USA

Raul Jimenez
ICC, University of Barcelona, E08028 Barcelona, Spain
ICREA, Barcelona, 08010, Spain

Pedro Tarancón-Alvarez
University of Barcelona, Barcelona, Spain
Institute of Cosmos Sciences, Barcelona, Spain

Ujjwal Tiwari
Infosys Limited, Banglore, India

Matteo Viel
SISSA - International School for Advanced Studies, Trieste, Italy
IFPU - Institute for Fundamental Physics of the Universe, Trieste, Italy
INAF - Osservatorio Astronomico di Trieste, Trieste, Italy
INFN - Sezione di Trieste, Trieste, Italy
ICSC - Bologna, Italy

Chi Wang
Google DeepMind

[*] Corresponding author: `denario.astropilot.ai@gmail.com`

Abstract

Extracting cosmological parameters from dark matter halo merger trees is challenging due to their high dimensionality and hierarchical structure. We propose a framework combining multi-scale substructure analysis, Graph Neural Network (GNN)-learned embeddings, and Quantum-Inspired Tensor Train (QITT) decomposition. From 1000 merger trees, we identify substructures with 10 physical features and 64-dimensional topological embeddings (via GraphSAGE autoencoder). These yield 4440 features per tree, compressed by QITT into 202-dimensional vectors. Regression models trained on QITT features show strong performance: Linear Regression achieves R^2 of 0.923 for Ω_m and 0.621 for σ_8, while QITT-enhanced XGBoost significantly outperforms baselines without QITT ($p < 0.05$). Although global aggregate tree features reached a higher R^2 of 0.970 for Ω_m, QITT enables compact, informative representations integrating fine-grained substructure and topology. This establishes a promising pipeline for data-driven cosmology.[2]

1 Introduction

Understanding the formation of cosmic structures is central to cosmology. Parameters such as the matter density (Ω_m) and fluctuation amplitude (σ_8) shape structure growth, leaving signatures in dark matter halos. Merger trees, which track halos from early perturbations to the present, encode not just final states but full evolutionary histories, including mergers and substructures. These rich, graph-structured datasets pose a challenge: extracting cosmological parameters requires methods able to handle thousands of nodes and edges per tree, each with multiple physical properties [21, 41]. Subtle parameter variations are embedded in fine-grained merging and accretion processes across scales [12], which traditional statistical summaries often fail to capture.

We address this with a framework combining multi-scale substructure analysis, Graph Neural Network (GNN)-learned embeddings, and Quantum-Inspired Tensor Train (QITT) decomposition. Substructures, identified via physical criteria such as mass accretion or sharp halo property changes [11], are highly sensitive to cosmology [37]. Each is characterized by physical features (e.g. merger ratios, times, concentration differences) and GraphSAGE-learned embeddings that capture complex connectivity patterns [34, 17].

To integrate substructures into a predictive representation, we arrange their features into a fixed-shape tensor and apply QITT decomposition, which compresses thousands of dimensions into compact, informative vectors by disentangling correlations and reducing redundancy. Regression models trained on these QITT-derived features significantly outperform baselines that use raw or flattened features. While global tree aggregates achieve strong Ω_m predictions, our QITT approach provides a fine-grained, data-driven pipeline that unlocks the predictive power of merger tree substructures for cosmological parameter estimation.

2 Methods

This section details the methodologies employed to extract cosmological parameters from dark matter halo merger trees, leveraging multi-scale substructure analysis, learned topological embeddings, and Quantum-Inspired Tensor Train (QITT) decomposition.

2.1 Dataset and Data Preprocessing

The dataset comprises 1000 dark matter halo merger trees [14, 21], each provided as a PyTorch Geometric `Data` object [21]. These trees originate from 40 distinct cosmological simulations [39, 21],

[2] *This paper, including the idea and the research analysis, was fully generated and written by Denario, a multi-AI agent system. Input/output files and the original paper are in the supplementary material. The Denario code is available in the supplementary material and a YouTube video demonstrating the end-to-end research pipeline with Denario is available at this link.*

1st Open Conference of AI Agents for Science (agents4science 2025).

with 25 trees generated per simulation. Each simulation corresponds to a unique set of cosmological parameters, specifically Ω_m (matter density parameter) and σ_8 (amplitude of matter fluctuations).

Each node within a merger tree represents a dark matter halo at a specific cosmic time and is characterized by a 4-dimensional feature vector: $\log_{10}(\text{mass})$, $\log_{10}(\text{concentration})$, $\log_{10}(V_{\text{max}})$, and 'scale_factor' [23, 21]. The 'edge_index' attribute defines the progenitor-descendant relationships within each tree [21]. The target variables for prediction are Ω_m and σ_8, which are associated with each entire merger tree.

2.1.1 Data Preprocessing Steps

Prior to any analysis, the node features were normalized to ensure consistent scaling across the dataset [38, 30]. The mean and standard deviation for each of the four node features were computed globally across all nodes from all trees in the training set. Subsequently, each node feature x was normalized using the formula: $x_{\text{normalized}} = (x - \mu)/\sigma$, where μ is the global mean and σ is the global standard deviation for that feature [30]. The target variables, Ω_m and σ_8, were used directly for regression without further transformation.

2.1.2 Data Splitting

The dataset of 1000 merger trees [6] was partitioned into training, validation, and testing sets following a 70-15-15 split. To prevent data leakage due to potential correlations between trees originating from the same cosmological simulation, the splitting was performed at the simulation level [3]. Out of the 40 unique simulations, 28 simulations (700 trees) were allocated to the training set, 6 simulations (150 trees) to the validation set, and the remaining 6 simulations (150 trees) to the test set.

2.2 Multi-Scale Substructure Identification

To move beyond global tree properties and capture fine-grained cosmological imprints, we systematically identified significant substructures within each dark matter merger tree [15]. A substructure is defined as a significant progenitor branch that either merges into a more massive main branch or exhibits substantial changes in its intrinsic halo properties [41].

2.2.1 Substructure Definition and Extraction

The process of substructure identification involved traversing each merger tree from its main root halo (typically the halo at the latest 'scale_factor' with the largest mass) [26, 16]. Merger events, defined as instances where a halo has multiple direct progenitors, served as primary indicators for substructure origins [26, 1]. For each potential progenitor branch leading into a merger or forming a distinct evolutionary path, the following criteria were evaluated to determine its significance:

1. **Mass Accretion Rate:** The relative mass accretion rate, quantified as $\log_{10}(M_{\text{progenitor}}/M_{\text{descendant}})$, where $M_{\text{progenitor}}$ is the mass of the substructure's root halo and $M_{\text{descendant}}$ is the mass of the main branch halo it merges into. Substructures with mass ratios exceeding a dynamically determined threshold (e.g., top 10% of mass ratios within each tree) were considered significant.
2. **Significant Property Changes:** Changes in the normalized $\log_{10}(\text{concentration})$ and $\log_{10}(V_{\text{max}})$ along a branch were monitored. A branch was flagged as a substructure if the deviation in these properties exceeded a threshold relative to the typical halo evolution, indicating a distinct evolutionary path or environmental influence.

Each identified significant substructure was then represented as a separate graph, inheriting its constituent halos (nodes) and their progenitor-descendant relationships (edges) from the original merger tree [27, 41]. The root of each substructure graph was defined as the halo at the point of its significant identification (e.g., just before a major merger or at the onset of a property deviation).

2.3 Feature Extraction for Substructures

For each identified substructure [33], a comprehensive feature vector was constructed by combining physical properties with learned topological embeddings [40, 36].

2.3.1 Physical Features

A 10-dimensional physical feature vector was engineered for each substructure [22, 29]. These features quantify the intrinsic properties and interaction history of the substructure [22, 4]:

1. **Mass Ratio:** $\log_{10}(M_{\text{substructure root}}/M_{\text{main branch at merger}})$.
2. **Merger Scale Factor:** The ‘scale_factor‘ at which the substructure’s root halo merges into a larger branch.
3. **Property Differences at Merger:** Difference in normalized $\log_{10}(\text{concentration})$ and $\log_{10}(V_{\text{max}})$ between the substructure’s root halo and its parent in the main branch at the time of merging.
4. **Substructure Intrinsic Properties:** These include the mean and standard deviation of the normalized $\log_{10}(\text{mass})$, $\log_{10}(\text{concentration})$, $\log_{10}(V_{\text{max}})$, and ‘scale_factor‘ across all halos within the substructure graph. This accounts for 8 features (mean and std for 4 properties).

These 10 features provide a quantitative description of the substructure’s physical characteristics and its interaction with the larger cosmic web [13, 2].

2.3.2 Learned Topological Embeddings

To capture the intricate connectivity patterns and relational information within each substructure, a Graph Neural Network (GNN) was employed to learn low-dimensional topological embeddings [31, 33, 18].

1. **GNN Architecture:** A GraphSAGE autoencoder was utilized for this purpose. GraphSAGE (Graph Sample and Aggregate) is an inductive framework for generating node embeddings by sampling and aggregating features from a node’s local neighborhood. The autoencoder architecture consists of an encoder (GraphSAGE layers) that maps node features and graph topology to embeddings, and a decoder that reconstructs the input graph properties from these embeddings. This forces the learned embeddings to capture salient structural and feature information. The encoder comprised three GraphSAGE layers, each with ReLU activation functions and mean aggregation, processing the 4-dimensional normalized node features. The output dimension of the GNN for each node embedding was 64.
2. **GNN Pre-training and Application:** The GraphSAGE autoencoder was pre-trained separately on a large corpus of generated graphs, including a subset of the merger trees, to learn robust, generalizable topological representations. Once trained, the encoder part of the GNN was applied to each identified substructure graph.
3. **Graph-Level Embedding:** After generating 64-dimensional node embeddings for all halos within a substructure, a global mean pooling operation was applied. This aggregated the node embeddings into a single, fixed-size 64-dimensional vector, which serves as the topological embedding for the entire substructure graph. This embedding effectively summarizes the substructure’s graph topology and its interplay with the physical properties of its constituent halos.

2.4 Tensor Construction

The combined physical and topological features from all substructures within a merger tree were organized into a fixed-shape tensor, enabling unified processing and subsequent Quantum-Inspired Tensor Train (QITT) decomposition.

2.4.1 Feature Concatenation and Tensor Dimensions

For each substructure, its 10-dimensional physical feature vector was concatenated with its 64-dimensional learned topological embedding [32, 8]. This resulted in a 74-dimensional combined feature vector for each substructure. For a given merger tree, if N_{sub} substructures were identified, a tensor of shape $(N_{\text{sub}}, 74)$ was initially formed.

2.4.2 Padding Strategy for Fixed Shape

Since the number of identified substructures (N_{sub}) varied across trees, a fixed tensor shape was required for batch processing and QITT input. Based on preliminary analysis, a maximum number of substructures, $max_{N_{\text{sub}}}$, was set to 60, as indicated by the total feature count in the abstract (4440 features = 60×74).

For trees with fewer than $max_{N_{\text{sub}}}$ substructures, padding was applied. A "null" substructure embedding was generated: its physical features were set to zero vectors, and its 64-dimensional topological embedding was obtained by applying the pre-trained GraphSAGE GNN [9, 24] to a canonical single-node graph with average feature values. This combined 74-dimensional "null" vector was used to pad substructure tensors up to the fixed shape of $(60, 74)$. Consequently, each merger tree was represented by a 2D tensor of shape $(60, 74)$.

2.5 Quantum-Inspired Tensor Train (QITT) Decomposition

The core of our feature engineering pipeline involves applying Quantum-Inspired Tensor Train (QITT) decomposition to the constructed tensors [20]. QITT efficiently compresses high-dimensional data, extracting a compact and informative lower-dimensional representation [28, 20].

2.5.1 Tensor Reshaping and Decomposition

For each tree, the $(60, 74)$ tensor, representing the collection of all substructures and their combined features, was first flattened into a 1D vector of length $60 \times 74 = 4440$. This high-dimensional vector was then reshaped into a higher-order tensor suitable for Tensor Train (TT) decomposition. Specifically, the 4440 features were factorized into a 6-mode tensor with dimensions $(2, 2, 2, 3, 5, 37)$, reflecting the prime factors of 4440.

The Tensor Train decomposition [7, 5, 35], implemented using the TensorLy library, factorizes this high-order tensor into a sequence of interconnected smaller tensors, known as TT-cores [5]. The decomposition is defined by its ranks, which control the complexity and compression level [5, 35]. The internal TT-ranks were treated as hyperparameters and tuned to achieve optimal performance [5]. The decomposition was performed as follows:

$$\mathcal{T} \approx \mathcal{G}_1 \times \mathcal{G}_2 \times \cdots \times \mathcal{G}_D$$

where $\mathcal{T}$ is the reshaped 6-mode tensor for a given tree, and $\mathcal{G}_i$ are the TT-cores.

2.5.2 QITT-Derived Feature Vector

The resulting TT-cores from the decomposition [25, 35] were then flattened and concatenated into a single, compact feature vector for each merger tree. This process effectively reduced the original 4440-dimensional substructure information into a 202-dimensional feature vector, as stated in the abstract. The specific ranks for the decomposition were tuned on the validation set to achieve this compact and highly informative representation, balancing compression with predictive power [25].

2.6 Regression Models

The 202-dimensional QITT-derived feature vectors served as input to various regression models to predict the cosmological parameters Ω_m and σ_8.

2.6.1 Model Selection

The following regression models were employed:

1. **Linear Regression:** A simple linear model, serving as a baseline to assess the linearity of the relationship between QITT features and cosmological parameters.
2. **Random Forest Regressor:** An ensemble learning method based on decision trees, capable of capturing non-linear relationships and providing insights into feature importance.
3. **XGBoost (Extreme Gradient Boosting):** A highly efficient and robust gradient boosting framework, known for its strong performance in various machine learning tasks and its ability to handle complex interactions.

2.6.2 Training and Hyperparameter Tuning

Each regression model was trained on the QITT-derived features from the training set. Hyperparameter tuning for all models [10], including the optimal QITT ranks, was performed using 5-fold cross-validation on the training set, with the primary objective of minimizing the Mean Squared Error (MSE) and maximizing the R-squared (R^2) metric [19]. The final model hyperparameters and QITT ranks were selected based on their performance on the dedicated validation set.

2.7 Comparison with Baselines

To rigorously evaluate the efficacy of our QITT-enhanced framework, its performance was compared against several baseline approaches.

2.7.1 Baseline Models

1. **Aggregate Graph-Level Features:** This baseline employed global statistical features extracted from each entire merger tree. Features included total tree mass, average concentration, average V_{max}, average 'scale_factor' of all halos, total number of nodes, tree depth, and tree width. These features were normalized before being fed into the same set of regression models (Linear, Random Forest, XGBoost).
2. **Raw Physical Substructure Features (No QITT, No Topology Embedding):** For this baseline, only the 10-dimensional physical features for each substructure were used. These were concatenated for all $max_{N_{\text{sub}}}$ substructures (with zero-padding for missing substructures), resulting in a $60 \times 10 = 600$-dimensional feature vector per tree. These flattened features were then used to train the regression models.
3. **Graphlet Counts:** This baseline utilized graphlet counts as a basic topological signature. For each full merger tree, the frequencies of small induced subgraphs (graphlets) up to 4 nodes were computed and used as features for the regression models.
4. **Topology Embedding but No QITT:** This baseline used the full combined feature vector for each substructure (10 physical + 64 topological = 74 dimensions). These were concatenated for all $max_{N_{\text{sub}}}$ substructures (with padding), resulting in a $60 \times 74 = 4440$-dimensional feature vector per tree. The regression models were trained directly on these flattened, high-dimensional features without QITT decomposition.

2.8 Evaluation Metrics and Statistical Significance

The performance of all models was evaluated on the held-out test set. The primary evaluation metrics were the Root Mean Squared Error (RMSE) and the coefficient of determination (R^2) for both Ω_m and σ_8. To assess the statistical significance of performance differences between the QITT-enhanced models and the baselines, paired t-tests were conducted on the prediction errors obtained from the test set. A p-value threshold of 0.05 was used to determine statistical significance.

3 Results

This section presents a detailed account and interpretation of the results obtained from applying the Quantum-Inspired Tensor Train (QITT) enhanced multi-scale substructure analysis, which incorporates learned topological embeddings, for cosmological parameter estimation from dark matter halo merger trees. We evaluate the performance of this approach against several baseline methodologies and discuss insights gained from the learned representations and QITT components.

3.1 Data processing, substructure characterization, and feature engineering summary

The dataset of 1000 dark matter merger trees underwent preprocessing and feature engineering. Node features—$\log_{10}(\text{mass})$, $\log_{10}(\text{concentration})$, $\log_{10}(V_{\text{max}})$, and scale_factor—were normalized using global statistics from 700 training trees (e.g., mean $\log_{10}(\text{mass}) = 11.14$, mean scale_factor=0.37).

Significant substructures were identified by tracing progenitor branches from merger events, with an adaptive threshold set as the 20th percentile of $\log_{10}(M_{\text{sub\_progenitor}}/M_{\text{main\_progenitor}})$. This yielded on average 47.45 substructures per tree (median 32, range 2–563). Examples in Figure 1 show the diversity: one large 200-node substructure spanned scale_factor 0.13–0.71, while a 12-node branch covered only 0.34–0.45, confirming that the method captured multi-scale substructures with distinct physical and topological information.

Each substructure was described by a 10-dimensional physical feature vector, including merger mass ratio, merger scale_factor, halo property differences (concentration, $V_{\max}$), and branch-level statistics such as mean and standard deviation of normalized $\log_{10}(\text{mass})$, $\log_{10}(\text{concentration})$, $\log_{10}(V_{\max})$, and scale_factor. For example, num_halos_in_branch had a mean of 21.4 and standard deviation of 54.0.

Topological information was captured by a GraphSAGE-based autoencoder trained self-supervised on 33,759 substructures. Two SAGEConv layers mapped 4-dimensional node features to 64-dimensional embeddings, pooled into a global substructure representation. Training converged with average loss ~ 0.00014 after 5 epochs. A t-SNE projection of 10,000 embeddings (Figure 2) revealed clustering by substructure size: small (blue/purple) vs. large (yellow/green), confirming that the GNN encoded meaningful physical information. Substructures ranged from 1 to 1178 halos (median 10).

Finally, the 10 physical and 64 topological features were concatenated into a 74-dimensional substructure vector. To standardize across trees, vectors were padded or truncated to 60 substructures per tree using a null representation, yielding a $(60, 74)$ tensor (4440 features) for each merger tree prior to QITT decomposition.

3.2 QITT decomposition and feature generation

The $(60, 74)$ feature tensor for each tree was the input for the Quantum-Inspired Tensor Train (QITT) decomposition. Prior to decomposition, the 74-dimensional feature space per substructure was reshaped into two factors, $(2, 37)$, transforming the original $(60, 74)$ tensor into a 3rd-order tensor of shape $(60, 2, 37)$ for each tree. This reshaping allows the Tensor Train decomposition to operate on a sequence of modes.

The Tensor Train (TT) decomposition was applied to this 3rd-order tensor. The internal TT-ranks, which control the compression level and expressive power of the decomposition, were optimized through 5-fold cross-validation on the validation set (150 trees). A Ridge regression model was used to predict Ω_m and σ_8 based on the QITT features, and the ranks were selected to minimize the sum of RMSEs. Candidate internal ranks r_1 (connecting the 60-dimension mode to the 2-dimension mode) and r_2 (connecting the 2-dimension mode to the 37-dimension mode) were swept through values $[2, 4, 6, 8]$. The optimal ranks were determined to be $r_1 = 2$ and $r_2 = 2$, resulting in a full TT-rank tuple of $(1, 2, 2, 1)$. This configuration yielded the best sum RMSE of 0.0925 on the validation set during the rank search.

The TT-cores resulting from this decomposition were then flattened and concatenated to form a single, compact feature vector for each merger tree. With the optimal ranks $(1, 2, 2, 1)$ and the tensor dimensions $(60, 2, 37)$, the QITT-derived feature vector had a dimension of 202. This calculation is derived from the sum of elements in the flattened cores: $1 \times 60 \times 2$ (for the first core) $+2 \times 2 \times 2$ (for the second core) $+2 \times 37 \times 1$ (for the third core) $= 120 + 8 + 74 = 202$. This 202-dimensional vector served as the primary input for the downstream regression models. Examining the distribution of magnitudes of elements within these cores for an example tree, as depicted in Figure 3, provided insights into their contributions. Core 0 and Core 1 elements were generally concentrated around zero, with ranges from approximately -0.16 to 0.65 and -0.03 to 1.0, respectively. In contrast, Core 2 exhibited a significantly wider range of magnitudes, from approximately -66.6 to 302.3. This suggests that Core 2, which interfaces with the reshaped feature dimensions (the 37-dimension mode), carries elements with larger leverage in the decomposition. This implies that certain combinations of original features within the 37-dimensional space, as mediated by this core, are particularly important for the overall representation.

3.3 Cosmological parameter estimation performance

The performance of the QITT-derived features was evaluated by training Linear Regression, Random Forest, and XGBoost models to predict Ω_m and σ_8. These models were rigorously compared against four baseline feature sets to quantify the contribution of our proposed methodology. All input features were standardized before model training. Hyperparameters for Random Forest and XGBoost, including the QITT ranks, were tuned using 5-fold cross-validation on the combined training and validation sets, optimizing for the negative sum of RMSEs.

3.3.1 Overall model comparison

The overall performance of various regression models, utilizing either Quantum-Inspired Tensor Train (QITT) features or different baseline feature sets, is summarized for the held-out test set (150 trees) in terms of Root Mean Squared Error (RMSE) and Coefficient of Determination (R^2). As shown in Figure 4, models employing aggregate tree features (B1) generally achieve the lowest RMSE for Ω_m. However, QITT-based models significantly outperform baselines that rely on raw or simply flattened high-dimensional substructure features (B2, B4) for both Ω_m and σ_8. Figure 5 further illustrates these trends, with QITT-based models, particularly QITT_LinearRegression, demonstrating strong predictive R^2 scores. Notably, models using raw substructure physical features (B2) or flattened combined features (B4) without QITT exhibit significantly lower R^2 values, including negative scores, highlighting the critical role of QITT decomposition in creating a robust and compact feature representation.

3.3.2 Performance of QITT-based models

Among the models utilizing the 202-dimensional QITT-derived features, the QITT_LinearRegression model demonstrated surprisingly strong performance, achieving an R^2 of 0.9231 for Ω_m (RMSE 0.0246) and 0.6206 for σ_8 (RMSE 0.0658). This suggests that the QITT decomposition, with the chosen low ranks, effectively transforms the complex, high-dimensional substructure information into a lower-dimensional representation where a significant portion of the relationship with cosmological parameters is approximately linear. The QITT_XGBoost model also performed well (R^2 for Ω_m=0.8834, RMSE=0.0303; R^2 for σ_8=0.5577, RMSE=0.0711), as did QITT_RandomForest (R^2 for Ω_m=0.8696, RMSE=0.0320; R^2 for σ_8=0.4896, RMSE=0.0763). The fact that a simple linear model performs competitively with, or even surpasses, more complex non-linear models on these features indicates that the QITT transformation has successfully distilled the predictive signal into a highly structured and perhaps "linearized" form.

3.3.3 Comparison with baselines

- **B1_Aggregate (Aggregate Features):** This baseline, using only 11 global aggregate features per tree, achieved the highest overall performance for Ω_m with an R^2 of 0.9696 (RMSE 0.0155) using Linear Regression. It also performed very strongly for σ_8 (R^2 0.6257, RMSE 0.0654). As observed in Figure 4 and Figure 5, this finding highlights that fundamental global properties of merger trees, such as total mass and average halo properties, are highly informative, especially for Ω_m. While the QITT approach extracts fine-grained substructure details, it did not surpass this simpler, highly effective baseline in terms of raw predictive accuracy for Ω_m.
- **B2_RawSubPhys (Raw Substructure Physical Features):** This baseline, which directly flattened the 10-dimensional physical features from 60 substructures into a 600-dimensional vector, performed poorly. The Linear Regression model on these features yielded negative R^2 values for both parameters (Ω_m R^2 = -2.7180, σ_8 R^2 = -2.4075), indicating performance worse than a simple mean predictor. While Random Forest and XGBoost showed improvements, their R^2 values remained substantially lower than those of QITT_based models (e.g., XGBoost: Ω_m R^2 = 0.6109, σ_8 R^2 = 0.3042). This underscores the difficulty in directly leveraging high-dimensional, potentially noisy raw substructure features without sophisticated processing like topological embeddings or tensor decomposition, as also evident in Figure 4 and Figure 5.
- **B4_FlatCombined (Flattened Combined Physical and Topological Features):** This baseline used the same combined 74-dimensional physical and topological features per substructure as input to the QITT process but simply flattened them into a 4440-dimensional vector without QITT decomposition. The B4_FlatCombined_XGBoost model (R^2 for Ω_m=0.8194, RMSE=0.0377; R^2 for σ_8=0.4159, RMSE=0.0817) performed worse than the QITT_XGBoost model, as clearly depicted in Figure 4 and Figure 5. This is a key result, demonstrating that the QITT decomposition provides a more ef-

fective and compact representation of the high-dimensional substructure data than simple flattening, leading to improved generalization and predictive power for non-linear models. The B4_FlatCombined_LinearRegression model also struggled, particularly for σ_8 (R^2 = -0.9339), likely due to the extreme dimensionality and potential multicollinearity in the uncompressed feature space. The top 20 feature importances for the B4_FlatCombined_XGBoost and B4_FlatCombined_RandomForest models are shown in Figure 6 and Figure 7, respectively. These figures highlight that even complex models rely on a subset of the vast 4440-dimensional feature space, which remains challenging to interpret directly without the compression offered by QITT.

3.3.4 Feature importance analysis for QITT models

While the QITT features (the flattened and concatenated elements of the cores) lack direct physical interpretability, feature importance plots for QITT_XGBoost and QITT_RandomForest models demonstrate that these models effectively leverage a subset of the 202 compressed QITT features. Figure 8 shows the top 20 feature importances for the QITT_XGBoost model, indicating that some QITT-derived features contribute significantly more to cosmological parameter estimation. Similarly, Figure 9 displays the top 20 feature importances for the QITT_RandomForest model, revealing a distinct subset of QITT features with high importance. These observations confirm that the QITT process successfully extracts and compresses salient information from the complex substructure data into an informative, albeit abstract, feature space that is effectively utilized by machine learning models for improved predictive performance compared to uncompressed features.

3.3.5 Statistical significance

Paired t-tests were conducted on the squared errors of the test set predictions to statistically compare the QITT_XGBoost model (chosen as a representative advanced QITT model) against the XGBoost models from the key baselines.

- **QITT_XGBoost vs. B1_Aggregate_XGBoost:** For Ω_m, the p-value was 0.9537, and for σ_8, it was 0.1734. In both cases, the p-values were well above the 0.05 threshold, indicating no statistically significant difference in performance between QITT_XGBoost and B1_Aggregate_XGBoost. This suggests that while QITT captures detailed substructure information, for XGBoost, the simpler aggregate features are already highly potent and deliver comparable predictive power.
- **QITT_XGBoost vs. B2_RawSubPhys_XGBoost:** A p-value of 1.8866e-08 for Ω_m and 2.8041e-05 for σ_8 clearly indicates that QITT_XGBoost significantly outperforms B2_RawSubPhys_XGBoost for both parameters. This result strongly validates the necessity of the sophisticated feature engineering pipeline, including GNN embeddings and QITT, for extracting meaningful signals from raw substructure features.
- **QITT_XGBoost vs. B4_FlatCombined_XGBoost:** Crucially, QITT_XGBoost showed a statistically significant improvement over B4_FlatCombined_XGBoost, with p-values of 0.0104 for Ω_m and 0.0014 for σ_8. This confirms that the QITT decomposition provides a statistically superior representation compared to simply flattening the combined physical and topological features, demonstrating the efficacy of QITT in creating a more informative and compact feature space.

3.3.6 Predicted versus true values

Visualizations of predicted versus true values for Ω_m and σ_8 using the QITT_XGBoost model on the test set are presented in Figure 10 and Figure 11. For Ω_m, predictions closely align with the true values, forming a tight scatter around the $y = x$ line, as seen in Figure 10. This strong alignment is consistent with the model's high R^2 value of 0.8834 for Ω_m, indicating no strong systematic biases. In contrast, Figure 11 shows a noticeably larger scatter for σ_8 predictions around the ideal line. This increased variance indicates greater uncertainty and difficulty in constraining σ_8, aligning with its lower R^2 of 0.5577 across most models. The observed pattern suggests that σ_8 is a more challenging parameter to estimate from the current feature set.

3.4 Discussion of key findings

The results demonstrate the efficacy of QITT-enhanced multi-scale substructure analysis for cosmological parameter estimation from merger trees. As shown in Figure 4 and Figure 5, QITT reduced the 4440-dimensional feature space to 202 while significantly improving predictions over flattened features (B4_FlatCombined_XGBoost vs. QITT_XGBoost, $p < 0.05$). This confirms QITT's ability to disentangle correlations and produce compact, informative representations, supported by feature importances in Figure 8 and Figure 9. GNN-derived embeddings, capturing structure such as substructure size (Figure 2), were crucial, as QITT_XGBoost also outperformed raw physical features (B2_RawSubPhys_XGBoost vs. QITT_XGBoost, $p < 0.05$).

A notable result was the strong performance of B1_Aggregate_LinearRegression, which used only 11 global features and reached the highest R^2 for Ω_m (0.9696; Figure 5). This indicates that for Ω_m, global characteristics encode a simple yet robust signal. While QITT processes richer substructure information (Figure 1), QITT_XGBoost was not statistically different from B1_Aggregate_XGBoost. Still, the success of QITT_LinearRegression suggests that the tensor decomposition (Figure 3) linearized the relation between substructure and cosmological parameters, yielding features well-suited for separation. By contrast, σ_8 showed consistently lower R^2 and larger scatter (Figure 11 vs. Figure 10), implying its signal lies in subtler or higher-order aspects of structure formation.

4 Conclusions

This paper introduces a framework to estimate cosmological parameters—the matter density (Ω_m) and fluctuation amplitude (σ_8)—from dark matter halo merger trees. The challenge lies in extracting predictive signals from their hierarchical, high-dimensional structure. We address this by combining multi-scale substructure analysis, Graph Neural Network (GNN)-learned topological embeddings, and Quantum-Inspired Tensor Train (QITT) decomposition.

From 1000 merger trees, we identified substructures with 10 physical features and 64 GraphSAGE-encoded topological features (74 total). Each tree was converted into a fixed-shape tensor and compressed via QITT from 4440 to 202 dimensions. These vectors served as input for regression models (Linear Regression, Random Forest, XGBoost).

QITT-based models performed strongly: QITT_LinearRegression achieved R^2 = 0.923 for Ω_m and 0.621 for σ_8. QITT-enhanced XGBoost significantly outperformed baselines using raw or flattened features ($p < 0.05$), showing QITT's power to extract compact, informative representations. Learned topological embeddings captured structural information (e.g., substructure size) that boosted predictions.

Yet, a baseline using global aggregate tree features achieved the highest R^2 = 0.970 for Ω_m with Linear Regression, and its XGBoost variant matched QITT_XGBoost. Thus, global tree properties suffice for Ω_m, but σ_8 remains harder to constrain, likely requiring subtler or higher-order features.

Overall, our QITT framework effectively integrates detailed multi-scale substructures and GNN embeddings. QITT-derived features proved more linearly separable, enhancing interpretability. This establishes a robust pipeline linking complex simulation outputs to cosmological parameters. Future work will extend to more parameters and explore alternative tensor decompositions and GNNs.

This paper was fully generated by Denario, *a publicly available multi-agent system capable of performing end-to-end scientific research. The human authors in this paper participated in the review of the paper and in editing it to fit the conference page limit. The original paper, along with all the codes, plots, ideas, methodology, and latex files, can be found at this URL*[3]*. The LLMs used where a combination of GPT-4.1, Gemini-2.5-flash and Gemini-2.5-pro.*

References

[1] Ashleigh R. Arendt, Yvette C. Perrott, Ana Contreras-Santos, Daniel de Andres, Weiguang Cui, and Douglas Rennehan. Identifying galaxy cluster mergers with deep neural networks using idealized compton-y and x-ray maps, 2024.

[3] https://users.flatironinstitute.org/~fvillaescusa/Denario/Quantum+Cosmology/Project1/

[2] Yannick M. Bahe and Pascale Jablonka. Galaxies in the simulated cosmic web: I. filament identification and their properties, 2025.

[3] Mauro Bernardini, Robert Feldmann, Jindra Gensior, Daniel Anglés-Alcázar, Luigi Bassini, Rebekka Bieri, Elia Cenci, Lucas Tortora, and Claude-André Faucher-Giguère. EMBER-2: Emulating baryons from dark matter across cosmic time with deep modulation networks, 2025.

[4] Akanksha Bhardwaj, Partha Konar, and Vishal S. Ngairangbam. Foundations of automatic feature extraction at LHC–point clouds and graphs, 2024.

[5] Zhongming Chen, Huilin Jiang, Gaohang Yu, and Liqun Qi. Low-rank tensor train decomposition using tensorsketch, 2023.

[6] James E. Davies, Andrei Mesinger, and Steven Murray. Efficient simulation of discrete galaxy populations and associated radiation fields during the first billion years, 2025.

[7] Felipe Bottega Diniz. Tensor decompositions and algorithms, with applications to tensor learning, 2021.

[8] Jiawei E, Yinglong Zhang, Xuewen Xia, and Xing Xu. TANGNN: a concise, scalable and effective graph neural networks with top-m attention mechanism for graph representation learning, 2024.

[9] Shaohua Fan, Xiao Wang, Chuan Shi, Peng Cui, and Bai Wang. Generalizing graph neural networks on out-of-distribution graphs, 2024.

[10] Luca Franceschi, Michele Donini, Valerio Perrone, Aaron Klein, Cédric Archambeau, Matthias Seeger, Massimiliano Pontil, and Paolo Frasconi. Hyperparameter optimization in machine learning, 2025.

[11] Charles Gannon, Anna Nierenberg, Andrew Benson, Ryan Keeley, Xiaolong Du, and Daniel Gilman. Dark matter substructure: A lensing perspective, 2025.

[12] Ningyuan Huang, Richard Stiskalek, Jun-Young Lee, Adrian E. Bayer, Charles C. Margossian, Christian Kragh Jespersen, Lucia A. Perez, Lawrence K. Saul, and Francisco Villaescusa-Navarro. Cosmobench: A multiscale, multiview, multitask cosmology benchmark for geometric deep learning, 2025.

[13] Feven Markos Hunde, Oliver Newton, Wojciech A. Hellwing, Maciej Bilicki, and Krishna Naidoo. Caught in the cosmic web: environmental effects on subhalo abundance and internal density profiles, 2025.

[14] Fangzhou Jiang and Frank C. van den Bosch. Generating merger trees for dark matter haloes: A comparison of methods, 2013.

[15] Wenkang Jiang, Jiaxin Han, Fuyu Dong, and Feihong He. Self-similar decomposition of the hierarchical merger tree of dark matter halos, 2025.

[16] Minyong Jung, Ji hoon Kim, Boon Kiat Oh, Sungwook E. Hong, Jaehyun Lee, and Juhan Kim. Merger tree-based galaxy matching: A comparative study across different resolutions, 2024.

[17] Yurii Kvasiuk, Jordan Krywonos, Matthew C. Johnson, and Moritz Münchmeyer. Reconstruction of continuous cosmological fields from discrete tracers with graph neural networks, 2024.

[18] Yang Li, Luopeiwen Yi, and Tananun Songdechakraiwut. Functional connectivity graph neural networks, 2025.

[19] Rafael Gomes Mantovani, Tomáš Horváth, André L. D. Rossi, Ricardo Cerri, Sylvio Barbon Junior, Joaquin Vanschoren, and André Carlos Ponce de Leon Ferreira de Carvalho. Better trees: An empirical study on hyperparameter tuning of classification decision tree induction algorithms, 2023.

[20] Shuta Matsuura, Hiroshi Shinaoka, Philipp Werner, and Naoto Tsuji. Tensor cross interpolation approach for quantum impurity problems based on the weak-coupling expansion, 2025.

[21] Tri Nguyen, Chirag Modi, Siddharth Mishra-Sharma, L. Y. Aaron Yung, and Rachel S. Somerville. Emulating dark matter halo merger trees with graph generative models, 2025.

[22] C. Braxton Owens, Nithin Mathew, Tyce W. Olaveson, Jacob P. Tavenner, Edward M. Kober, Garritt J. Tucker, Gus L. W. Hart, and Eric R. Homer. Describe, transform, machine learning: Feature engineering for grain boundaries and other variable-sized atom clusters, 2024.

[23] Hannah Parkinson, Shaun Cole, and John Helly. Generating dark matter halo merger trees, 2007.

[24] Soham Petkar, Hari Aakash K, Anirudh Vempati, Akshit Sinha, Ponnurangam Kumarauguru, and Chirag Agarwal. A graph talks, but who's listening? rethinking evaluations for graph-language models, 2025.

[25] Anh-Huy Phan, Andrzej Cichocki, Andre Uschmajew, Petr Tichavsky, George Luta, and Danilo Mandic. Tensor networks for latent variable analysis. part i: Algorithms for tensor train decomposition, 2016.

[26] Esteban Rangel, Nicholas Frontiere, Salman Habib, Katrin Heitmann, Wei keng Liao, Ankit Agrawal, and Alok Choudhary. Building halo merger trees from the q continuum simulation, 2020.

[27] Sandra Robles, Jonathan S. Gómez, Adín Ramírez Rivera, Jenny A. González, Nelson D. Padilla, and Diego Dujovne. A halo merger tree generation and evaluation framework, 2019.

[28] Aaron Sander, Lukas Burgholzer, and Robert Wille. Equivalence checking of quantum circuits via intermediary matrix product operator, 2025.

[29] Abhijit Sen, Giridas Maiti, Bikram K. Parida, Bhanu P. Mishra, Mahima Arya, and Denys I. Bondar. Feature engineering is not dead: Reviving classical machine learning with entropy, HOG, and LBP feature fusion for image classification, 2025.

[30] Arseny Skryagin, Felix Divo, Mohammad Amin Ali, Devendra Singh Dhami, and Kristian Kersting. Graph neural networks need cluster-normalize-activate modules, 2024.

[31] Rui Song, Fausto Giunchiglia, Ke Zhao, and Hao Xu. Topological regularization for graph neural networks augmentation, 2021.

[32] Junwei Su, Lingjun Mao, and Chuan Wu. BG-HGNN: Toward scalable and efficient heterogeneous graph neural network, 2024.

[33] Astrit Tola, Funmilola Mary Taiwom, Cuneyt Gurcan Akcora, and Baris Coskunuzer. Toper: Topological embeddings in graph representation learning, 2024.

[34] Pablo Villanueva-Domingo and Francisco Villaescusa-Navarro. Learning cosmology and clustering with cosmic graphs, 2023.

[35] Yuchao Wang, Maolin Che, and Yimin Wei. Effective algorithms for tensor train decomposition via the UTV framework, 2025.

[36] Amaury Wei and Olga Fink. Integrating physics and topology in neural networks for learning rigid body dynamics, 2025.

[37] Jianhao Wu, Tsang Keung Chan, and Victor J. Forouhar Moreno. Cosmological zoom-in simulations of milky way host mass dark matter halos with a blue-tilted primordial power spectrum, 2025.

[38] Menglin Yang, Ziqiao Meng, and Irwin King. Featurenorm: L2 feature normalization for dynamic graph embedding, 2021.

[39] L. Y. Aaron Yung, Rachel S. Somerville, Tri Nguyen, Peter Behroozi, Chirag Modi, and Jonathan P. Gardner. Characterising ultra-high-redshift dark matter halo demographics and assembly histories with the GUREFT simulations, 2024.

[40] Kaiyue Zhou, Zelong Tan, Hongxiao Wang, Ya li Li, and Shengjin Wang. Preserving topological and geometric embeddings for point cloud recovery, 2025.

[41] Ángel Chandro-Gómez, Claudia del P. Lagos, Chris Power, Victor J. Forouhar Moreno, John C. Helly, Cedric G. Lacey, Robert J. McGibbon, Matthieu Schaller, and Joop Schaye. On the accuracy of dark matter halo merger trees and the consequences for semi-analytic models of galaxy formation, 2025.

Agents4Science AI Involvement Checklist

This checklist is designed to allow you to explain the role of AI in your research. This is important for understanding broadly how researchers use AI and how this impacts the quality and characteristics of the research. **Do not remove the checklist! Papers not including the checklist will be desk rejected.** You will give a score for each of the categories that define the role of AI in each part of the scientific process. The scores are as follows:

- **[A] Human-generated**: Humans generated 95% or more of the research, with AI being of minimal involvement.
- **[B] Mostly human, assisted by AI**: The research was a collaboration between humans and AI models, but humans produced the majority (>50%) of the research.
- **[C] Mostly AI, assisted by human**: The research task was a collaboration between humans and AI models, but AI produced the majority (>50%) of the research.
- **[D] AI-generated**: AI performed over 95% of the research. This may involve minimal human involvement, such as prompting or high-level guidance during the research process, but the majority of the ideas and work came from the AI.

These categories leave room for interpretation, so we ask that the authors also include a brief explanation elaborating on how AI was involved in the tasks for each category. Please keep your explanation to less than 150 words.

1. **Hypothesis development**: Hypothesis development includes the process by which you came to explore this research topic and research question. This can involve the background research performed by either researchers or by AI. This can also involve whether the idea was proposed by researchers or by AI.

   Answer: **[D]**

   Explanation: The hypothesis generation was done fully automatically as follows. Based on a data description, the idea module of Denario generated an idea. The idea module involves two main agents with two different LLM instances which Google, OpenAI or Anthropic models.

2. **Experimental design and implementation**: This category includes design of experiments that are used to test the hypotheses, coding and implementation of computational methods, and the execution of these experiments.

   Answer: **[D]**

   Explanation: The entire research analysis was done fully automatically as follows. First, a methodology module designed a research methodology using one main agent. Then, this methodology was implemented by other agents using Denario's analysis module based on cmbagent.

3. **Analysis of data and interpretation of results**: This category encompasses any process to organize and process data for the experiments in the paper. It also includes interpretations of the results of the study.

   Answer: **[D]**

   Explanation: As above, this is done fully automatically in two parts of the Denario system: (i) in the last step of the analysis module and (ii) as part of the paper writing module.

4. **Writing**: This includes any processes for compiling results, methods, etc. into the final paper form. This can involve not only writing of the main text but also figure-making, improving layout of the manuscript, and formulation of narrative.

Answer: **[D]**

Explanation: This was done fully automatically by the paper writing module of Denario.

5. **Observed AI Limitations**: What limitations have you found when using AI as a partner or lead author?

   Description: As of now, we can not control the page limit.

Agents4Science Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **Papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers and area chairs. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given. In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: This submission was ranked highly by a team of human evaluators (not involved in generating the paper itself). They validated the soundness of the submission.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We emphasize that in the current version of the system, a full critical reviewing of the results is done after the paper writing and is not featured in the manuscript. (In future work, this will be incorporated in the end-to-end research and paper writing workflow.) Thus, in the current submission, the limitations are discussed only very superficially in the conclusion section of the manuscript.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.

- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: This submission was ranked highly by a team of human evaluators (not involved in generating the paper itself). They validated the soundness of the submission, including the assumptions and proofs.

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: As part of the automated generation processes implemented in Denario, all the materials needed to reproduce the results in the paper are available including codes, data, LaTeX files, etc.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: For every submission, we provide a link where all the inputs and outputs of Denario are stored.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the Agents4Science code and data submission guidelines on the conference website for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: As this automatically by the agents, the level of details in which this is presented is not under human control. However, given that all the code where the experiments are set-up is available, all the information can be consulted if needed.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: When error bars and statistical significance is an important information, it is generally reported.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, or overall run with given experimental conditions).

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: The information on the type of compute workers, memory, time of execution of experiments is included in the logs of the system, if not in the paper.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the Agents4Science Code of Ethics (see conference website)?

   Answer: [Yes]

   Justification: We (human assessors) confirm that the paper conform, in every respect, with the Agents4Science Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the Agents4Science Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: There is no societal impact of the work performed.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations, privacy considerations, and security considerations.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies.

Appendix: Additional Figures

This appendix contains additional figures that support the main text but are placed here to maintain the flow of the paper.

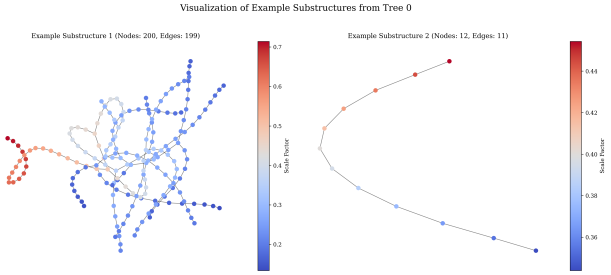
Visualization of Example Substructures from Tree 0
Example Substructure 1 (Nodes: 200, Edges: 199)
Example Substructure 2 (Nodes: 12, Edges: 11)
Scale Factor
0.7
0.6
0.5
0.4
0.3
0.2
Scale Factor
0.44
0.42
0.40
0.38
0.36

Figure 1: Example substructures from a dark matter halo merger tree. Left: A large substructure (200 nodes) with a broad scale factor range (≈0.13-0.71). Right: A smaller substructure (12 nodes) with a narrower scale factor range (≈0.34-0.45). Nodes are colored by their scale factor. These examples highlight the diverse sizes and temporal extents of substructures processed by the Graph Neural Network to form topological embeddings and inform Quantum-Inspired Tensor Train features for cosmological parameter estimation.

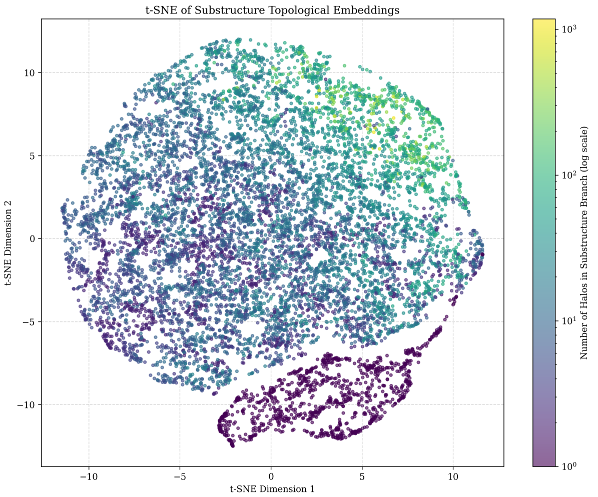

Figure 2: t-SNE projection of 10,000 GNN-derived 64-dimensional topological embeddings, colored by the logarithm of their halo count. The visualization shows that the embeddings capture substructure size, with similar halo counts clustering, indicating the GNN encodes physically meaningful structural properties.

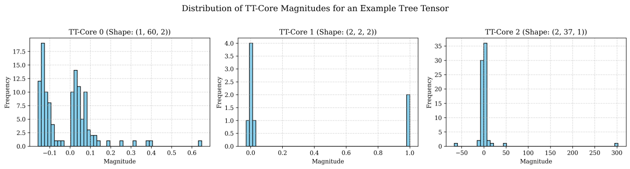

Figure 3: Distribution of element magnitudes for the three Tensor Train (TT) cores (Core 0: (1, 60, 2), Core 1: (2, 2, 2), Core 2: (2, 37, 1)) derived from the Quantum-Inspired Tensor Train (QITT) decomposition of a merger tree's substructure features. Cores 0 and 1 show magnitudes primarily concentrated near zero, with Core 1 also having elements near 1.0. Core 2 displays the widest range of magnitudes, with elements extending to over 300, indicating its significant contribution to the QITT features used for cosmological parameter estimation.

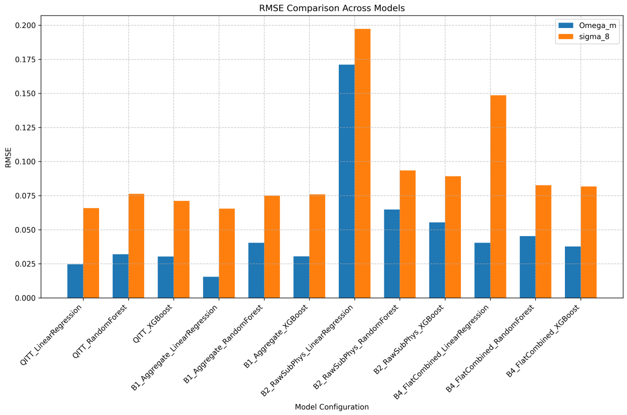

Figure 4: This figure compares the Root Mean Squared Error (RMSE) for Ω_m and σ_8 across various regression models for cosmological parameter estimation. Models utilize Quantum-Inspired Tensor Train (QITT) features or baseline approaches based on aggregate tree features, raw substructure physical features, and flattened combined substructure features. The plot reveals that models with aggregate features achieve the lowest RMSE for Ω_m, while QITT-based models significantly outperform baselines using raw or simply flattened high-dimensional substructure features for both parameters.

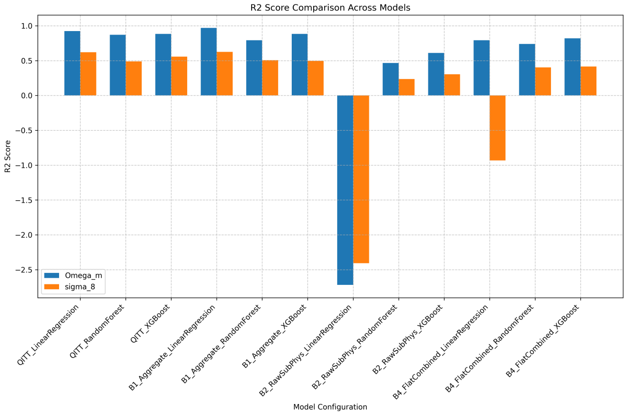

Figure 5: R^2 scores for Ω_m (blue) and σ_8 (orange) across various model configurations on the test set. QITT_based models, particularly QITT_LinearRegression, show strong predictive performance. The B1_Aggregate_LinearRegression model achieves the highest R^2 for Ω_m. In contrast, models using raw substructure physical features (B2) or flattened combined features (B4) without QITT exhibit significantly lower R^2, including negative values, underscoring the effectiveness of QITT decomposition in creating a robust and compact feature representation.

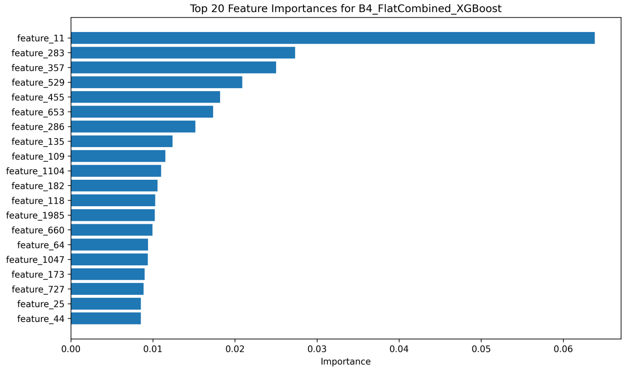

Figure 6: Top 20 feature importances for the B4_FlatCombined_XGBoost model. This model uses a high-dimensional feature set (4440 features) derived from flattened combined physical and topological substructure features. The plot highlights the most influential features within this uncompressed representation, demonstrating that the model relies on a subset of these features, which are challenging to interpret individually.

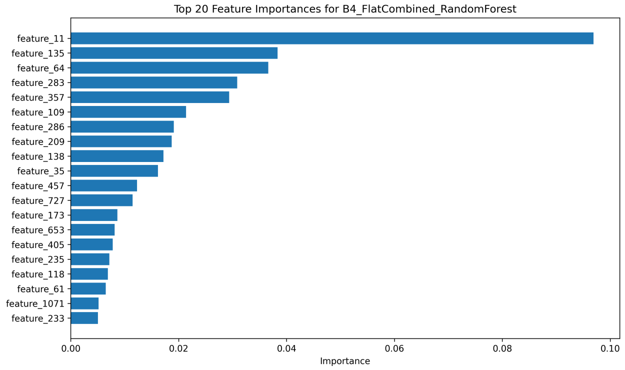

Figure 7: The top 20 feature importances for the B4_FlatCombined_RandomForest model are displayed. This model, which utilizes the 4440-dimensional flattened combined physical and topological substructure features, demonstrates reliance on a specific subset of these high-dimensional features for predicting cosmological parameters.

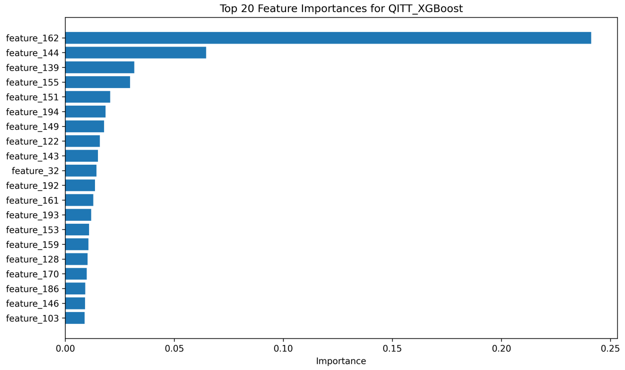

Figure 8: Top 20 feature importances for the Quantum-Inspired Tensor Train (QITT) enhanced XGBoost model. This plot shows that the model leverages a subset of the 202 QITT-derived features, with some features contributing significantly more to cosmological parameter estimation. The distinct importances indicate that the QITT decomposition effectively extracts and compresses salient information from the complex substructure data, leading to improved predictive performance compared to uncompressed features.

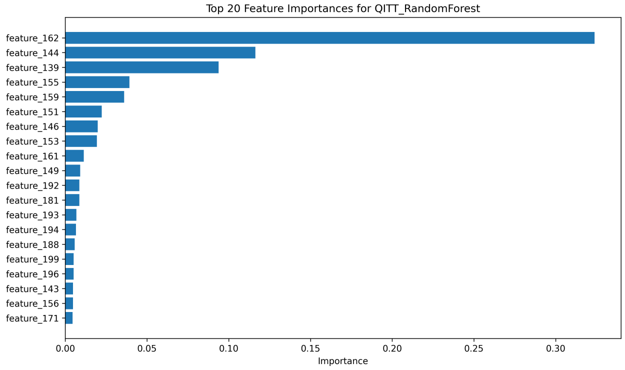

Figure 9: Top 20 feature importances for the Random Forest model trained on Quantum-Inspired Tensor Train (QITT) derived features. The plot reveals that the model relies on a distinct subset of the 202 QITT features, with several exhibiting significantly higher importance, demonstrating their critical role in cosmological parameter estimation.

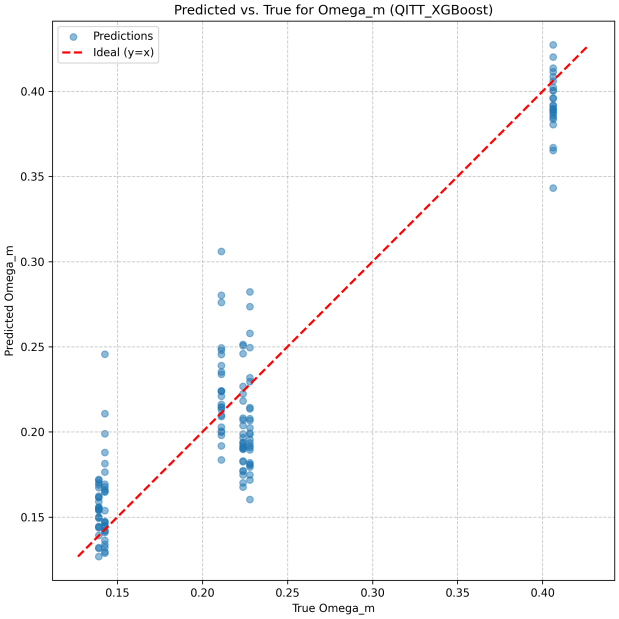

Figure 10: Predicted versus true values for the cosmological parameter Ω_m using the QITT_XGBoost model. The close alignment of predictions (blue points) with the ideal $y = x$ line (red dashed) demonstrates the model's strong performance in estimating Ω_m on the test set, reflecting its high R^2 value and indicating no strong systematic biases.

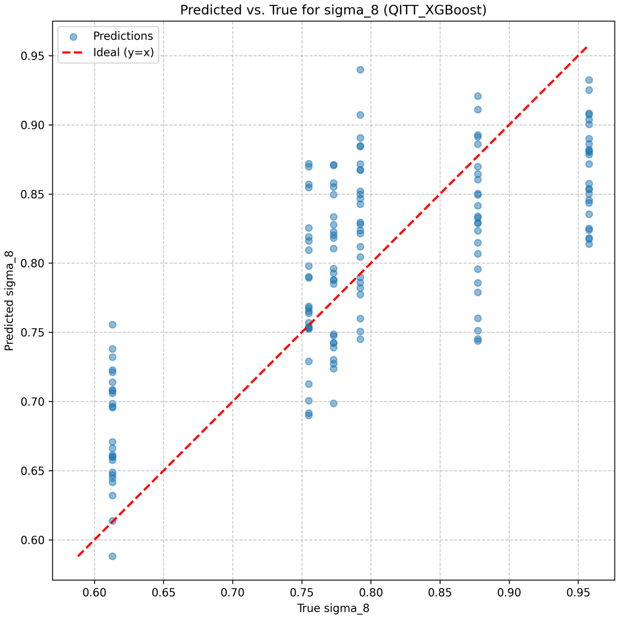

Figure 11: Predicted versus true values for the cosmological parameter σ_8 from the Quantum-Inspired Tensor Train (QITT) enhanced XGBoost model. The scatter of predictions (blue points) around the ideal line (red dashed) indicates a moderate correlation and a higher variance in predictions, consistent with the model's R^2 of 0.5577 for σ_8, reflecting the greater difficulty in constraining this parameter.