OmniJet-alpha: The first cross-task foundation model for particle physics

CCNSB Seminar

Abhiram Tilak

CND Dual Degree
IIIT Hyderabad

August 20, 2025

Table of Contents

Introduction

About the Paper

This paper was published to IOP Science (Machine Learning: Science and Technology) Journal in August 2024.

Additional Information about the paper:

- Basically a High Energy Physics adaptation of autoregressive generative pretrained transformer (GPT) model.

Introduction

About the Paper

This paper was published to IOP Science (Machine Learning: Science and Technology) Journal in August 2024.

Additional Information about the paper:

- Basically a High Energy Physics adaptation of autoregressive generative pretrained transformer (GPT) model.
- This paper was the first successful model that do both Jet Generation and Jet Tagging.

Background

Motivations

Problem with Supervised Models:

- Supervised learning typically acquire limited domain representations and focuses on a few key features for high prediction accuracy that must be learned anew for each task.

Background

Motivations

Problem with Supervised Models:

- Supervised learning typically acquire limited domain representations and focuses on a few key features for high prediction accuracy that must be learned anew for each task.
- The problem comes when we have to scale the data being fed to the model.

Background

Motivations

Problem with Supervised Models:

- Supervised learning typically acquire limited domain representations and focuses on a few key features for high prediction accuracy that must be learned anew for each task.
- The problem comes when we have to scale the data being fed to the model.
- A significant drawback of this is that the performance of ML models trained on simulations may not translate to real data, especially due to mismodeling in the former.

Background

Motivations

Problem with Supervised Models:

- Supervised learning typically acquire limited domain representations and focuses on a few key features for high prediction accuracy that must be learned anew for each task.
- The problem comes when we have to scale the data being fed to the model.
- A significant drawback of this is that the performance of ML models trained on simulations may not translate to real data, especially due to mismodeling in the former.
- There have been efforts to run supervised models on huge datasets like ParT and OmniLearn.

Background

Motivations

Problem with Supervised Models:

- Supervised learning typically acquire limited domain representations and focuses on a few key features for high prediction accuracy that must be learned anew for each task.
- The problem comes when we have to scale the data being fed to the model.
- A significant drawback of this is that the performance of ML models trained on simulations may not translate to real data, especially due to mismodeling in the former.
- There have been efforts to run supervised models on huge datasets like ParT and OmniLearn.

Therefore, current machine learning models in particle physics are task-specific and require large labeled datasets, which are often scarce especially in rare process searches.
Foundation models, like those in NLP and vision, generalize across tasks and require fewer examples for fine-tuning, making them highly attractive for particle physics applications.

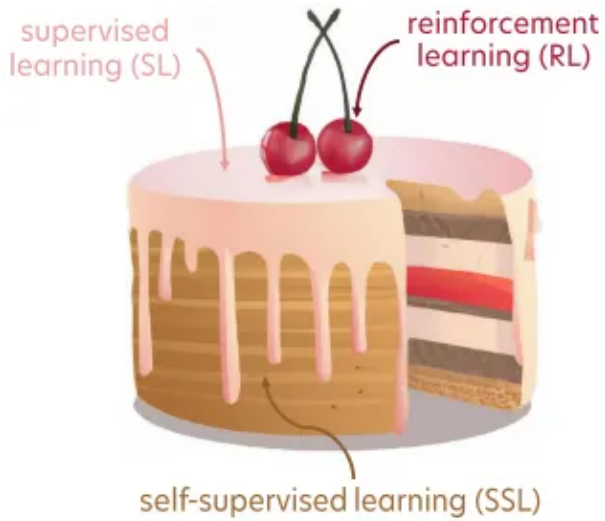
Self-Supervised Learning: A type of machine learning where models learn useful features and representations from unlabeled data
SSL aims to learn generic representations summarizing domain features that prove useful across various downstream tasks. SSL tasks can be formulated on unlabeled data.

Self-Supervised Learning: A type of machine learning where models learn useful features and representations from unlabeled data
SSL aims to learn generic representations summarizing domain features that prove useful across various downstream tasks. SSL tasks can be formulated on unlabeled data.

To understand different machine learning approaches we use the cake metaphor proposed by Yann LeCun 2023.

Self-Supervised Learning: A type of machine learning where models learn useful features and representations from unlabeled data
SSL aims to learn generic representations summarizing domain features that prove useful across various downstream tasks. SSL tasks can be formulated on unlabeled data.

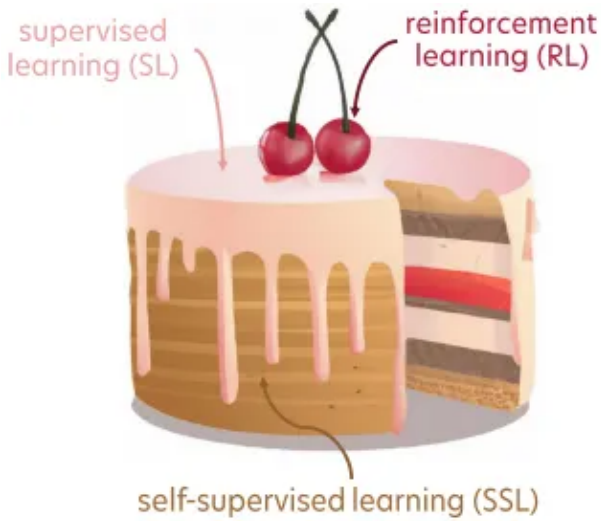
To understand different machine learning approaches we use the cake metaphor proposed by Yann LeCun 2023.



Self-Supervised Learning: A type of machine learning where models learn useful features and representations from unlabeled data
SSL aims to learn generic representations summarizing domain features that prove useful across various downstream tasks. SSL tasks can be formulated on unlabeled data.

To understand different machine learning approaches we use the cake metaphor proposed by Yann LeCun 2023.

1. (SSL) Self-supervised learning gives you information from the main content of the cake, which is orders of magnitude more in quantity than anything else.
2. (SL) Supervised learning gives you information from only the icing covering the outside of the cake
3. (RL) Reinforcement Learning trains via only the cherry on top.



Background

Motivation

Transformers have already proven flexible in physics:

- Autoregressive jet generation

Background

Motivation

Transformers have already proven flexible in physics:

- Autoregressive jet generation
- Masked jet constituent prediction (BERT-like pretraining)

Background

Motivation

Transformers have already proven flexible in physics:

- Autoregressive jet generation
- Masked jet constituent prediction (BERT-like pretraining)
- Tokenized detector image/event modeling

Background

Motivation

Transformers have already proven flexible in physics:

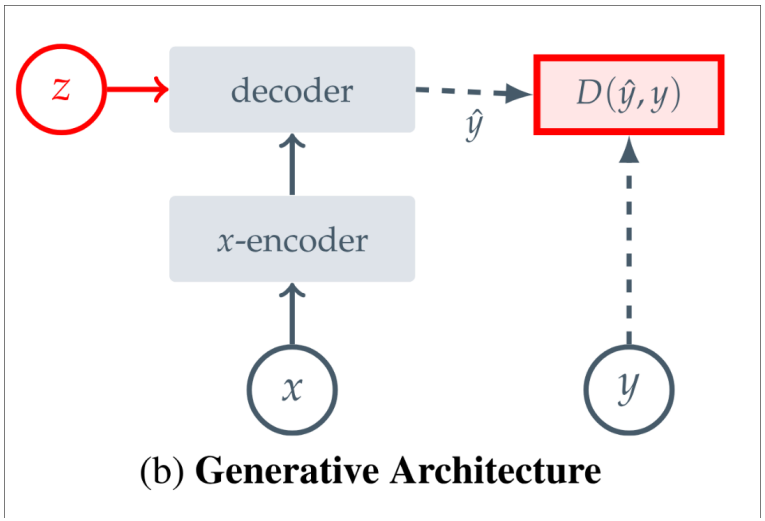
- Autoregressive jet generation
- Masked jet constituent prediction (BERT-like pretraining)
- Tokenized detector image/event modeling

OmniJet extends this trend by adapting GPT-style autoregressive modeling to continuous jet point cloud data.

Background

Motivations

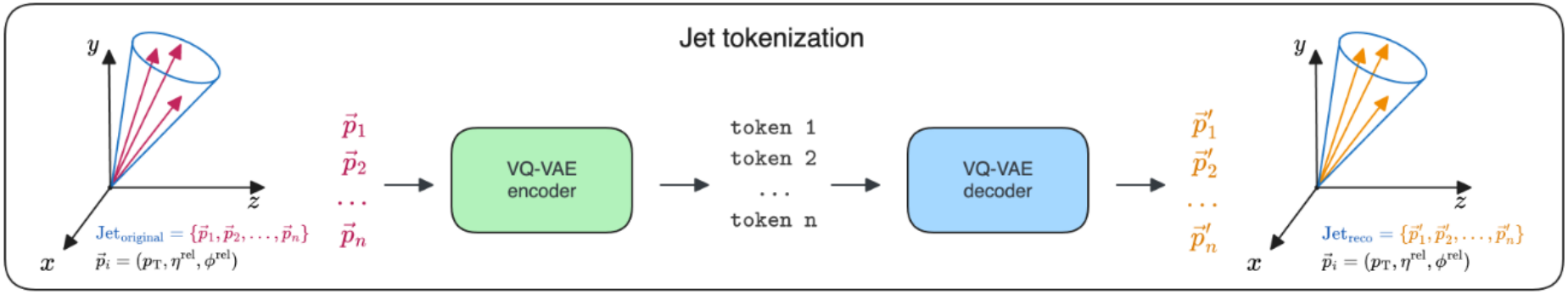
Generative Methods: This idea is at the core of self-supervised generative methods, which remove or corrupt portions of the input and learn to predict the corrupted content. In particular, mask-denoising approaches learn representations by reconstructing randomly masked patches from an input, either at the pixel or token level.
Masked pretraining tasks require less prior knowledge than view-invariance approaches and easily generalize beyond the image modality.
Only problem here is that they tend to underperform because they build only lower level semantic relationships between jet-representations.



(b) **Generative Architecture**

Architecutre

Tokenization

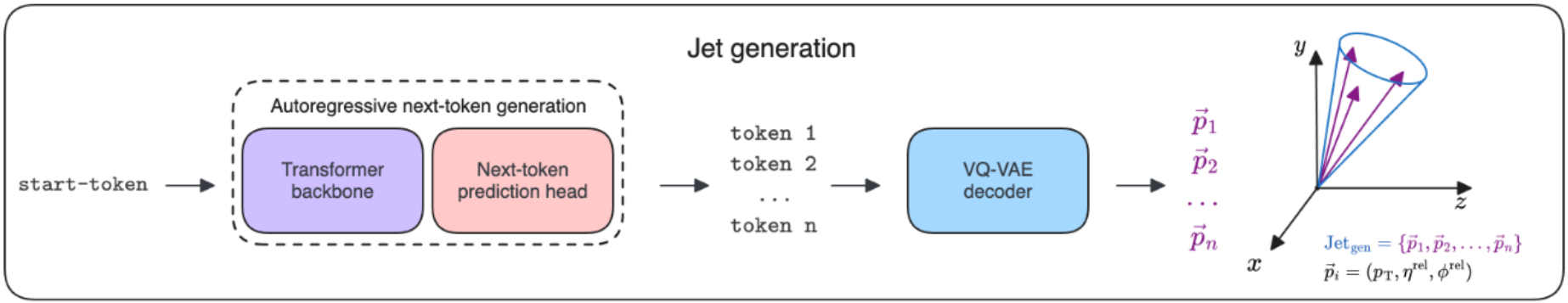


Jet constituents, represented by their $(p_T, \eta_r el, \phi_r el)$ features, are transformed into discrete tokens using different tokenization strategies:

1. Binned tokenization subdivides feature space into a fixed grid.
2. Unconditional tokenization maps each constituent independently via a VQ-VAE with an MLP.
3. Conditional tokenization uses a transformer-based VQ-VAE where token reconstruction depends on other constituents. This tokenization enables continuous jet data to be processed by transformer architectures.

Architecture

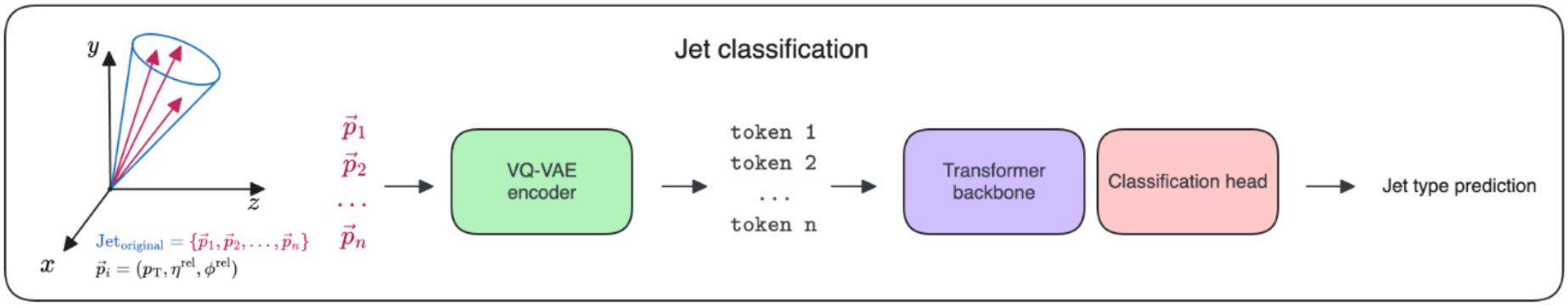
Jet Generation



Using the tokenized jets, an autoregressive GPT-style transformer backbone is trained to learn the probability distribution of tokens. Starting from a special start token, the model sequentially generates tokens until a stop token or maximum sequence length is reached. These tokens are then decoded by the VQ-VAE back into physical jet space, allowing the creation of realistic synthetic jets.

Architecture

Jet Classfication



For classification, the transformer backbone is combined with a task-specific head. This can be trained from scratch or fine-tuned from a pretrained generative model. In the fine-tuning setup, the pretrained transformer provides strong initial representations, enabling better jet tagging performance, especially when training data is limited.

Architecture

Transformer Architecture

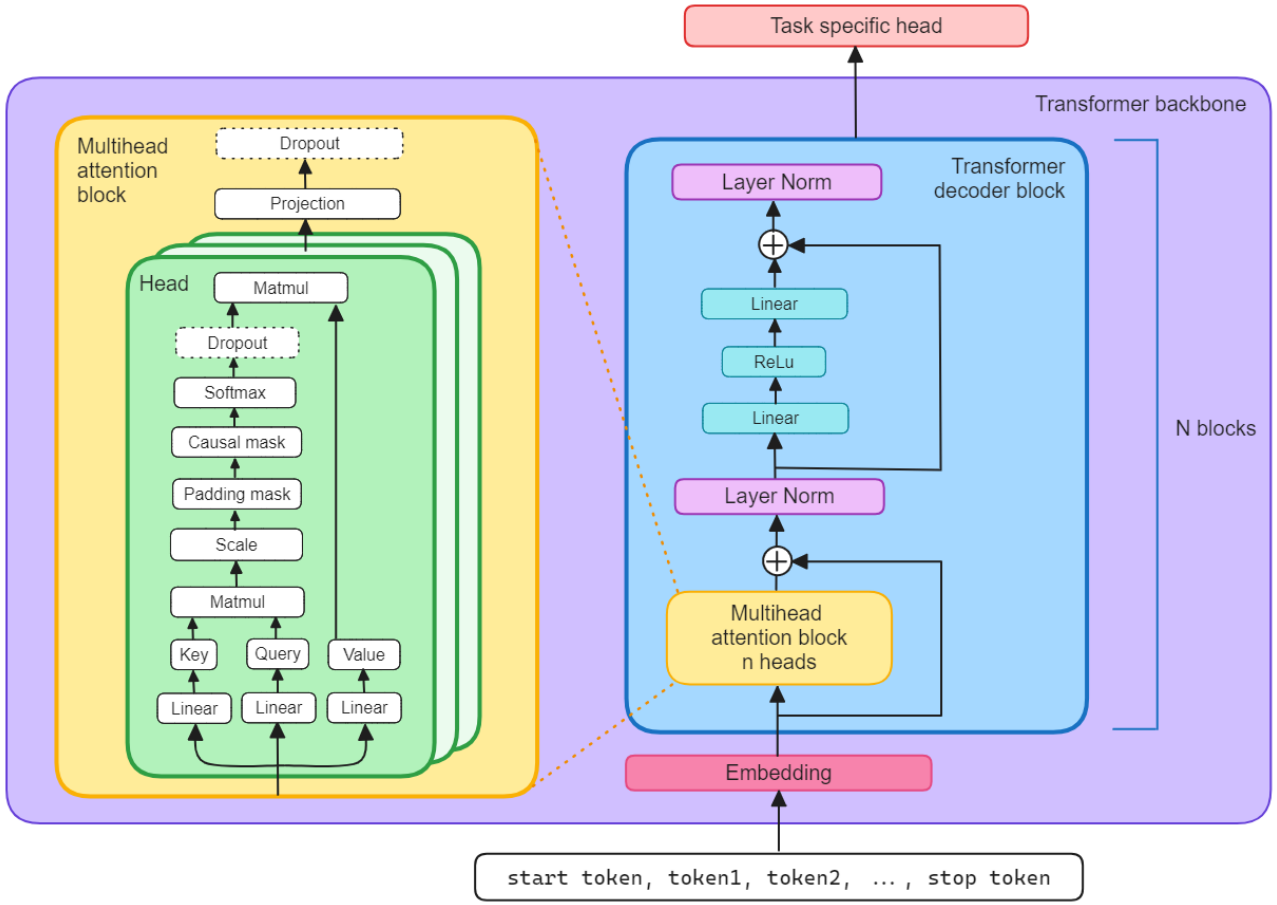


Figure: GPT-Link Transformer Encoder

Datasets and Training

Datasets:

Pretraining: JetClass: The pretraining task uses 1 Million jets, which is 1% of Jetclass dataset. It consists of 500k Top jets and 500k QCD jets.

- This model only uses three kinematic features, relative pseudorapidity (η^{rel}) and azimuthal angle (ϕ^{rel}).

Datasets and Training

Datasets:

Pretraining: JetClass: The pretraining task uses 1 Million jets, which is 1% of Jetclass dataset. It consists of 500k Top jets and 500k QCD jets.

- This model only uses three kinematic features, relative pseudorapidity (η^{rel}) and azimuthal angle (ϕ^{rel}).
- Also removes low energy constituents using, ($|\eta^{rel}| < 0.8$ and $|\phi^{rel}| < 0.8$).

Datasets and Training

Datasets:

Pretraining: JetClass: The pretraining task uses 1 Million jets, which is 1% of Jetclass dataset. It consists of 500k Top jets and 500k QCD jets.

- This model only uses three kinematic features, relative pseudorapidity (η^{rel}) and azimuthal angle (ϕ^{rel}).
- Also removes low energy constituents using, ($|\eta^{rel}| < 0.8$ and $|\phi^{rel}| < 0.8$).

Finetuning: Uses top-jet ($t \to bqq'$) and quark-jet (q/g) classes from the JetClass, for binary classification.

Evaluation

Metrics:

- Token Quality: Multi-class accuracy on original constituents vs. reconstructed constituents

Other Evaluations and Visualizations:

Evaluation

Metrics:

- Token Quality: Multi-class accuracy on original constituents vs. reconstructed constituents
- Prediction Accuracy on Binary Classification between top and quark jets (number of correct classifications vs wrong)

Other Evaluations and Visualizations:

Evaluation

Metrics:

- Token Quality: Multi-class accuracy on original constituents vs. reconstructed constituents
- Prediction Accuracy on Binary Classification between top and quark jets (number of correct classifications vs wrong)
- AUC-ROC metric: Area under the ROC curve

Other Evaluations and Visualizations:

Evaluation

Metrics:

- Token Quality: Multi-class accuracy on original constituents vs. reconstructed constituents
- Prediction Accuracy on Binary Classification between top and quark jets (number of correct classifications vs wrong)
- AUC-ROC metric: Area under the ROC curve

Other Evaluations and Visualizations:

- Visualizations for the token quality in different tokenization settings

Evaluation

Metrics:

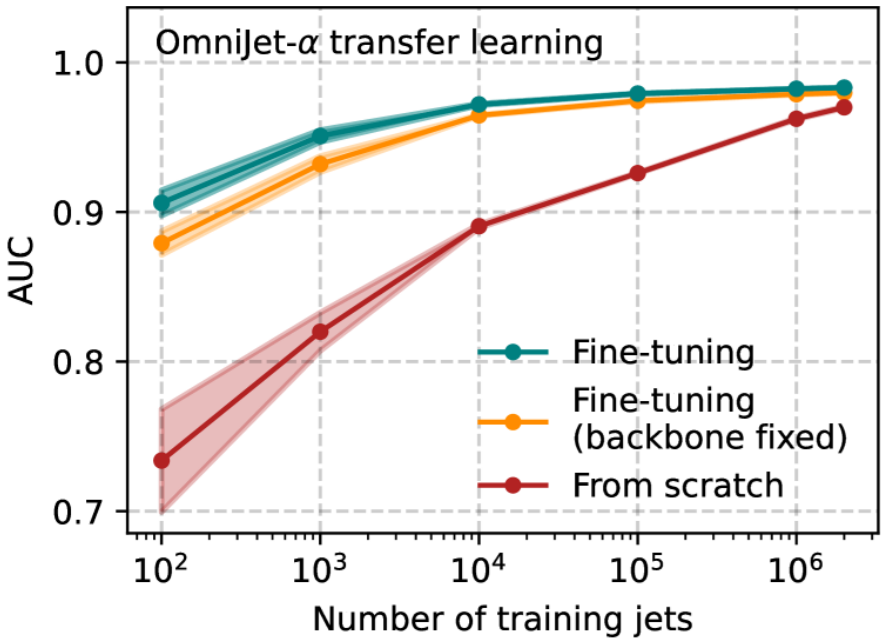
- Token Quality: Multi-class accuracy on original constituents vs. reconstructed constituents
- Prediction Accuracy on Binary Classification between top and quark jets (number of correct classifications vs wrong)
- AUC-ROC metric: Area under the ROC curve

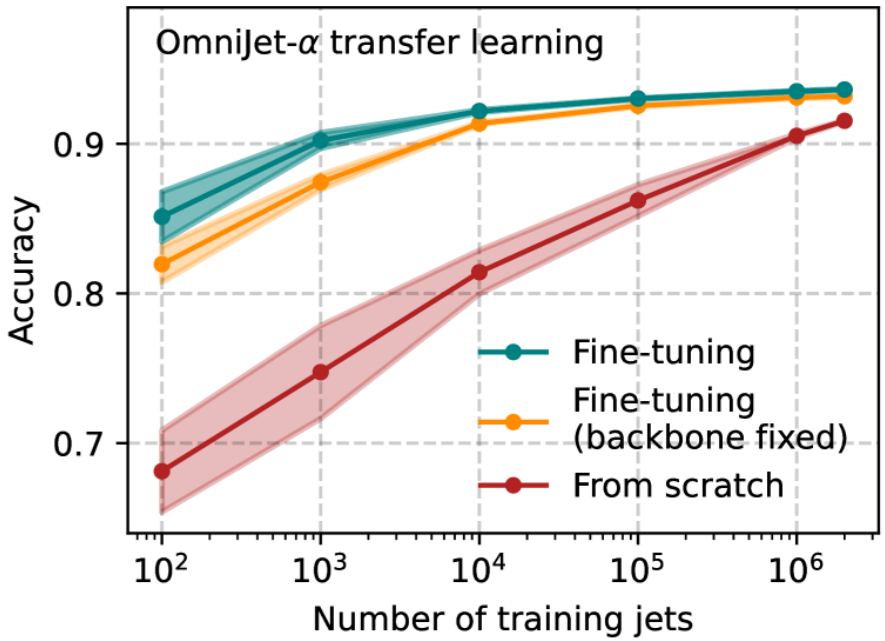
Other Evaluations and Visualizations:

- Visualizations for the token quality in different tokenization settings
- Reconstruction plots for different parameters

Results

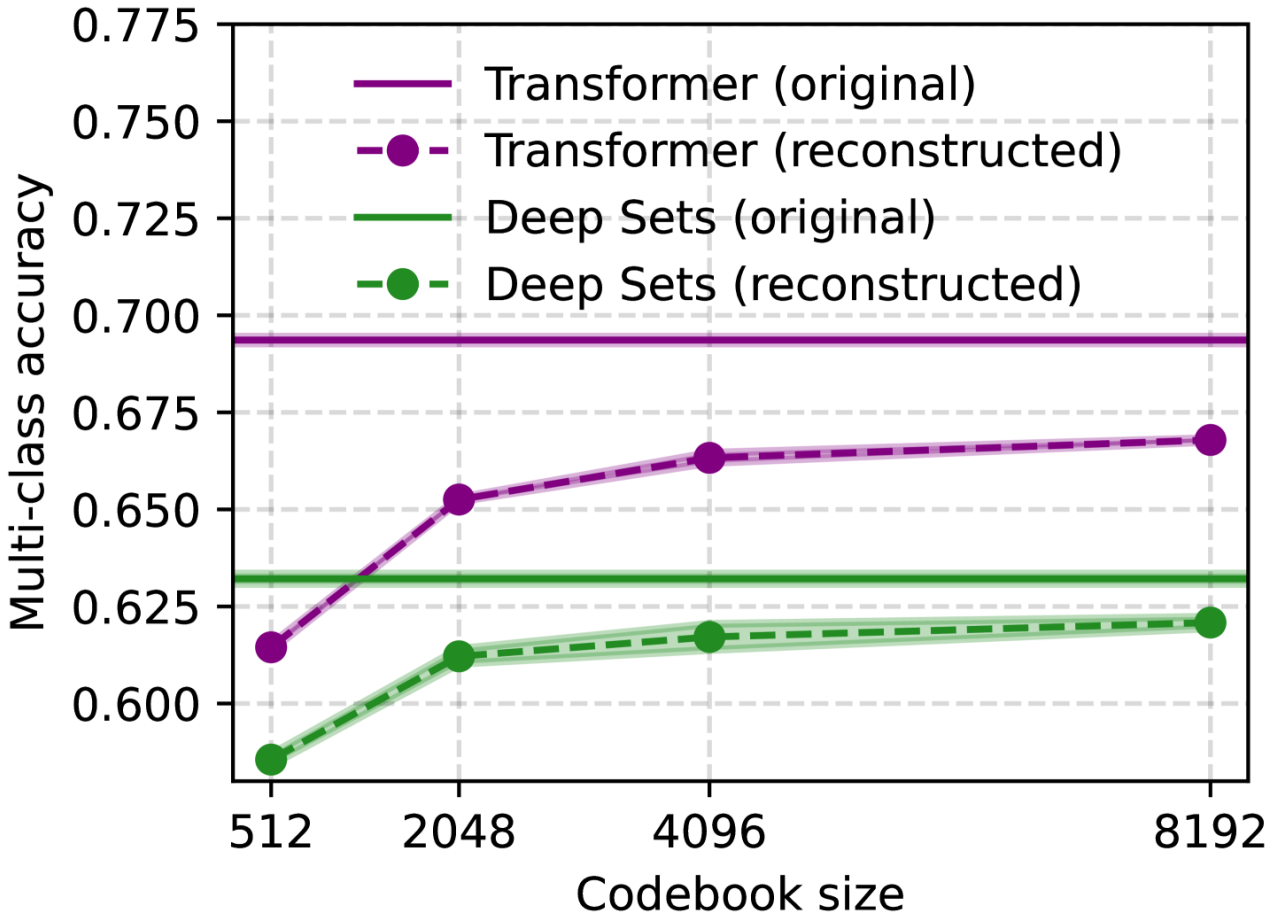
Classification Accuracy adn AUC-ROOC





Results

Multi-Class Classification on Reconstructed tokens



Visualizations

Token Reconstructions in eta-phi plane

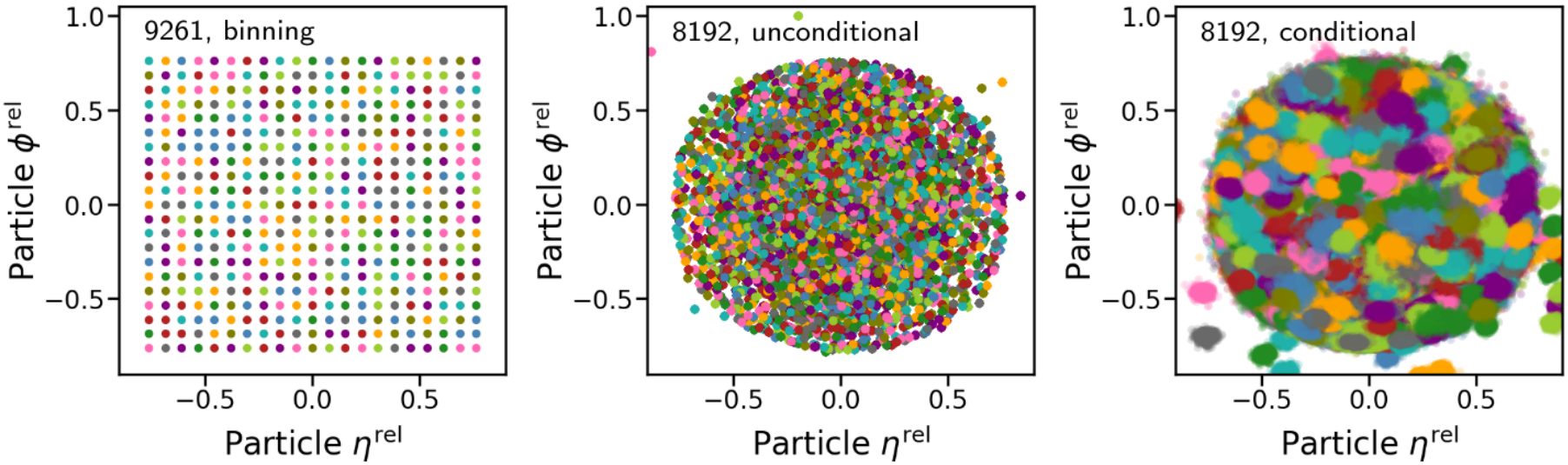


Figure: Randomly sampled 50 tokens, Each token is reconstructed 500 times, each color corresponds to a particular token.

Visualizations

Constituent Level reconstructions

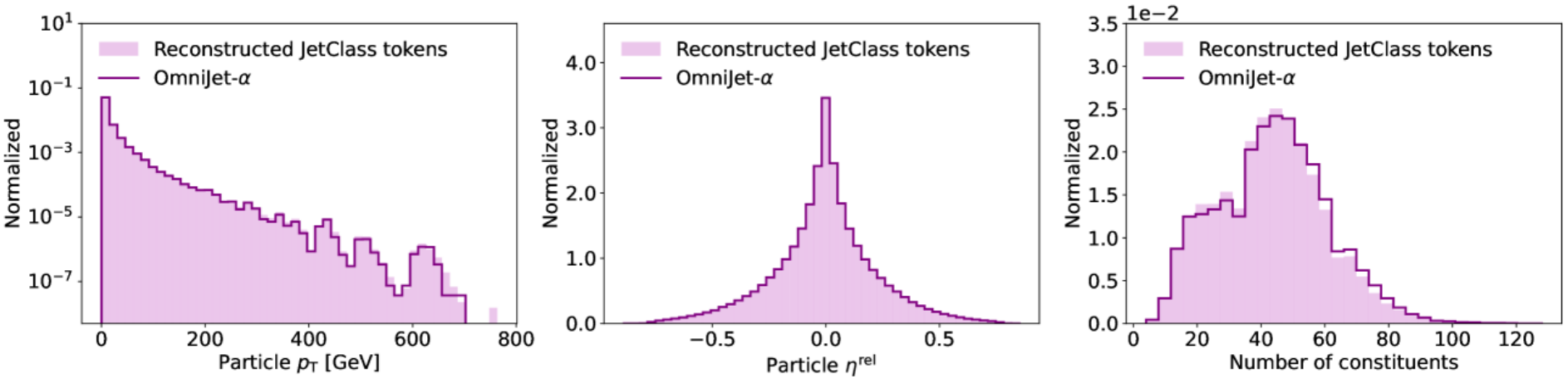


Figure: Normalized reconstructions of 3 different particle-level features

Visualizations

Jet Level reconstructions

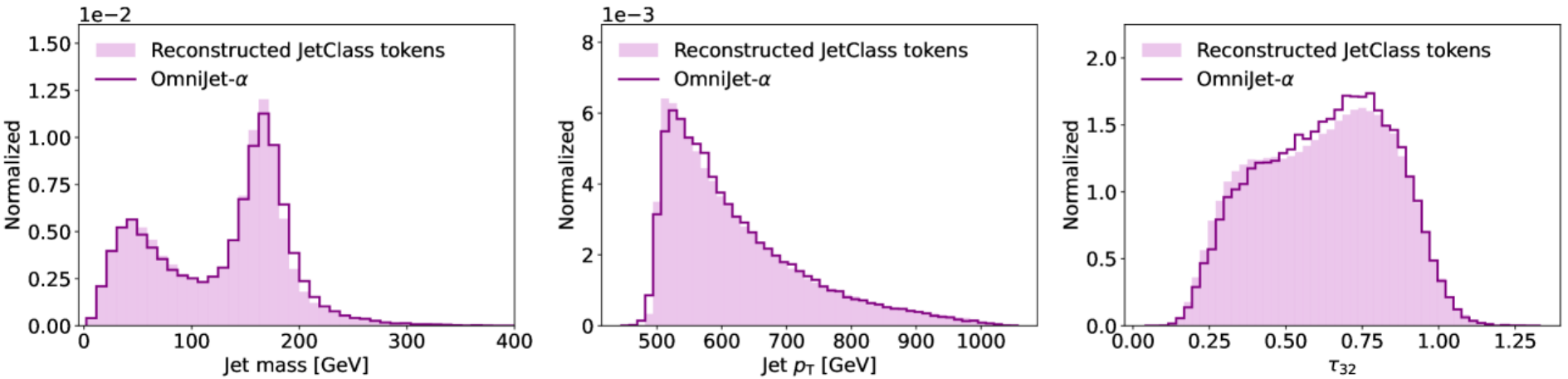


Figure: Normalized reconstructions of 3 jet level features

Conclusion

- Conditional tokenization with large codebooks (8192) preserves jet information better than other approaches, improving observable resolution.

Conclusion

- Conditional tokenization with large codebooks (8192) preserves jet information better than other approaches, improving observable resolution.
- OmniJet-α successfully generates realistic jets, matching global kinematics and substructure.

Conclusion

- Conditional tokenization with large codebooks (8192) preserves jet information better than other approaches, improving observable resolution.
- OmniJet-α successfully generates realistic jets, matching global kinematics and substructure.
- Pretraining on generation transfers well to classification, giving strong gains in low-data regimes.

Conclusion

- Conditional tokenization with large codebooks (8192) preserves jet information better than other approaches, improving observable resolution.
- OmniJet-α successfully generates realistic jets, matching global kinematics and substructure.
- Pretraining on generation transfers well to classification, giving strong gains in low-data regimes.
- While not yet state-of-the-art, the model shows clear potential and future improvements can further enhance performance.

Thank You