PROCEEDINGS OF SCIENCE

Dataset-wide Graph Neural Networks for BSM Searches at the LHC

Anna Mullin[a,*]

[a] *High Energy Physics Group, Cavendish Laboratory, JJ Thomson Avenue, Cambridge, CB3 0HE, United Kingdom*

E-mail: ajm351@cam.ac.uk

We present a new application of Graph Neural Networks (GNNs) for LHC searches that aims to improve event classification by representing entire datasets as graphs, with events as nodes and kinematically similar events connected by edges. The strategy builds from our development of graph convolutions and graph attention mechanisms, applying scalable solutions for training GNN models on large graphs with robust background validation. By merit of the search style and graph design, the GNN obtains extensive information from topological network structures such as clusters, helping to distinguish signal from background through their distinct characteristic connectivity. This work extends our previous proof of concept for dataset-wide graphs in beyond-Standard Model (BSM) searches [1], which demonstrates a promising baseline of signal-background separation. Since our recent extension to include GNNs, the strategy reveals further sensitivity improvements with a leptoquark BSM benchmark beyond a conventional Deep Neural Network (DNN) approach. The method focuses on anomaly detection with autoencoders, exploiting the new format of a dataset-wide GNN in an example unsupervised search, calculating the event-by-event anomaly score. The graph-based autoencoder achieves stronger signal-background separation than the baseline non-graph autoencoder in all tested cases.

The European Physical Society Conference on High Energy Physics (EPS-HEP2025)
7-11 July 2025
Marseille, France

*Speaker

 ISSN 1824-8039 . Published by SISSA Medialab.

PoS(EPS-HEP2025)447

1. Introduction

Measurements from particle colliders reveal hints of underlying physics processes, although the precise consequences of parton collisions are always somewhat hidden. The processes allowed by the rich set of symmetry relationships in the SM appear as detector signals that are reconstructed into a diverse set of observables characterising each collision event. GNNs are transforming areas of science research where the data contains complex relational structures and meaningful hidden connections, including collider physics, although existing LHC applications are predominantly at the sub-event level [8, 9]. Graph methods use the context from neighbouring nodes to discern discreet yet stable and reliable features about a node of interest. Graphs built from entire LHC event datasets are largely unexplored. The present work develops a new methodology for node classification, where the GNN benefits from passing the graph structure from LHC events through convolutional layers that enhance important patterns of edges.

When event datasets are plotted as nodes in a graph space of kinematic variables, their similarities and differences emerge as geometric and topological structures encoded in their edges that highlight their particle origins. Characteristic graph structures emerge from well-understood SM processes, distinct from BSM events that are likely to appear anomalous graph substructures against an SM background. By training a GNN to recognise graphs of SM events, an analysis may obtain new information from topological differences that appear only when BSM events are present in test graphs.

To approach the large phase space of unexplored BSM parameters, model-independent searches are key for efficiency. The GNNs in this project are intended to improve signal-background separation in LHC datasets where the signal model is unknown. BSM events may appear as overdensities where processes are enhanced in regions of kinematic space, or as rare outliers filling distribution tails. A model can be trained to treat either case as anomalous. Unsupervised methods can face difficulties with subtle background-like signals as they have no guidance from ground truth to learn outlying deviations. To overcome this pitfall, powerful graph-based architectures can incorporate domain knowledge, validate graph structures of the SM, check the scalability of results, apply careful benchmarking and reveal reliability through evaluation metrics in comparison with baseline non-graph anomaly detection techniques.

Autoencoders (AEs) are a competitive architecture for deterministic unsupervised learning. The convolutional AE was extended into a graph-AE (GAE) by including graphs as input [4]. In this project, an AE baseline is compared with a GAE to evaluate the potential benefit in isolating leptoquark signal events against backgrounds of single top and $t\bar{t}$ samples from the event-level graph connectivity.

2. Graph construction

A graph G is defined by a set of four matrices:

$$G = (V, E, X, W) \tag{1}$$

where V is a set of nodes that are each an n-dimensional vector in a set of quantities that define the graph space, and $E \subseteq V \times V$ is a set of edges[1]. The node attributes X are additional d-dimensional feature vectors per node of quantities not defined in the graph space, and W are edge attributes that often contain weights. For a node $v \in V$, its *neighbourhood* is the set nodes in connection.

For LHC events, a suitable graph space is the set of relevant kinematic observables that may reveal informative contrasts between types of physics processes. Within that space, many possible distances can define node similarity. These include standard distance metrics, such as the Euclidean distance:

$$d_{euc} = \sqrt{\Sigma_{i=1}^{n}(u_i - v_i)^2} \tag{2}$$

or the earth mover's distance (EMD), which measures the 'work' required to transform between two probability distributions P and Q, defined:

$$d_{\mathrm{EMD}}(P,Q) = \inf_{\gamma \in \Pi(P,Q)} \int_{\mathcal{X}\times\mathcal{X}} \|x - y\| \, d\gamma(x,y). \tag{3}$$

In the EMD, $\Pi(P,Q)$ is the set of all couplings γ, which are joint probability distributions that indicate how much 'energy' is transported between P and Q[2]. The EMD (also referred to as *energy* mover's distance) in particle physics typically visualises the metric space of jets and measures the distance between jet energy flows [3]. Euclidean distance emphasises local spatial structure and depends on absolute values of node features, whereas the EMD captures distribution shape features more generally. If two nodes fall within a user-defined radial distance or *linking length* (LL), then they share an edge. The LL is chosen to ensure that the number of edges in the graph is approximately 10% of the total number of possible edges (a ratio called *edge fraction*), as this permits significant structural information to allow signal-background separation without excess connections generating noise that overpowers graph behaviour and causes computational difficulties.The edge information is contained in an *adjacency matrix*, written:

$$A_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in E, \\ 0 & \text{otherwise} \end{cases} \tag{4}$$

where the 1 between v_i and v_j is replaced with w_{ij} if the edges carry weights W. In *directed* graphs, edges point from a 'source' node to a 'target' node, where the source node sends information (or features) that the target receives, modelling asymmetric relationships. Edge weights influence the smoothness of signals passed through the graph, and include a penalty term proportional to the inverse distance between nodes ($1/d$), emphasising local connections. Background-background edges are typically shorter distance and therefore skewed to higher edge weights. Meanwhile, node weights for simulated LHC samples are proportional to the Monte Carlo weight of the event, including factors of luminosity and cross-section.

The adjacency matrix A is normalised by a symmetric degree-based scheme that is common for ensuring numerical stability and reducing sensitivity to dominant hub nodes. The scheme normalises by the degree matrix D[3], updating A to become $A' = D^{-1/2}AD^{-1/2}$. The adjacency matrix is input to the GAE, but excluded from the AE.

[1] An edge connects two nodes if they are defined as 'related' in the user-defined graph space.

[2] The ground metric $\|x - y\|$ is the cost of moving one unit of energy from point x (in P) to point y (in Q). The optimal coupling minimises the transport cost. The distributions P and Q reside in an underlying metric space $\mathcal{X}$.

[3] The *degree* of a node is the number of edges it shares with any other nodes.

3. Anomaly detection with Graph Neural Networks

Convolutional GNNs are best suited when local neighbourhood structure is critical, and are a general-purpose method of updating node embeddings with information aggregated from both the node and its neighbourhood. Each convolution updates the features of each connected node to include information about . One can write the convolution operation with explicit matrix dimensions as $H^{(l)} = \tilde{A}H^{(l-1)}W^{(l)} + b^{(l)}$, or more explicitly:

$$H^{(l)} = \left\{ \underbrace{\begin{bmatrix} 1 & 0 & 1 & \dots \\ 0 & 1 & 1 & \dots \\ 1 & 1 & 1 & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}}_{\{[n\times n]\}} \cdot \underbrace{\begin{bmatrix} x_{11} & \dots & x_{1h'} \\ \vdots & \ddots & \vdots \\ x_{n1} & \dots & x_{nh'} \end{bmatrix}}_{[n\times h']} \right\} \cdot \underbrace{\begin{bmatrix} w_{11} & \dots & w_{1h} \\ \vdots & \ddots & \vdots \\ w_{h'1} & \dots & w_{h'h} \end{bmatrix}}_{[h'\times h]} + \underbrace{\begin{bmatrix} b_{11} & \dots & b_{1h} \\ \vdots & \ddots & \vdots \\ b_{k1} & \dots & b_{kh} \end{bmatrix}}_{[n\times h]} \tag{5}$$

where $b^{(l)}$ is a learnable bias term included in the GCN that shifts the activation function to allow the output of each node to be more than a linear combination of its inputs, a more general and flexible model for greater expressivity that also provides a baseline output value. The number of nodes in the graph is n, and the numbers of features in the current and previous layers are h and h', respectively, which give the output feature dimensions after the current and previous layers.

The unique design of node and edge weights in this method aims to accurately represent LHC datasets in the GAE. Background validation indicates that large node weights allow a subset of nodes to represent a sample with larger event size, permitting Monte Carlo simulations to produce graphs that behave similarly to graphs of real LHC datasets [1]. In addition, background-only graphs possess consistent graph characteristics independently of the presence of connected signal events, leading to reliable understanding of SM-only behaviour.

4. Setup

Events are generated in this proof-of-principle study using a signal model where pair-produced vector leptoquarks with mass 1 TeV each decay into a top quark and a neutrino, and the top quark decays leptonically, simulated with the `MadGraph` "VectorLQ" FeynRules UFO from the LQ Toolbox [2]. The two dominant backgrounds are $t\bar{t}$ and single top production are also simulated in `MadGraph`. A set of preselections is applied to the events prior to graph-building to increase signal efficiency, including the requirement that the missing transverse energy $E_{\mathrm{T}}^{\mathrm{miss}} > 200$ GeV.

The GAE and AE are trained on the same set of background-only events to compare their performance in three unsupervised learning scenarios where signal is injected into test sets in proportions of 1%, 10% and 20% of the weighted background yield. The test background sample is orthogonal to the training set. The model architectures are constructed from the Python Graph Outlier Detection (`PyGOD`) [5] and Python Outlier Detection (`PyOD`) [6] packages. The Python library `PyTorch Geometric` [7] implementation of the 'GCNConv' model is the default backbone of the GAE model in `PyGOD`, adapted through source code in this project into an edge- and node-weighted version. The adapted version performs the standard convolution function described in Equation 5.

Hyperparameters are selected for training the GAE and AE models consistently with the purpose of comparing them fairly, ensuring common learning rate, activation function and loss function, amongst other settings that are defined in their respective libraries[4] [5, 6]. Another commonality is setting the *contamination percentage* to zero in both cases, indicating that neither model expects a known ratio of signal-to-background events prior to evaluation, and is fully unsupervised.

5. Results

Initial comparisons of the GAE with the baseline AE indicate stronger performance from the graph-based method. We consider the case where leptoquark signal is injected with a number of events that is 20% of the SM background yield, and graphs are constructed in an underlying space of five typical kinematic variables with distances calculated using the EMD. The Z-score more than doubles for a lower-bound cut on the anomaly score distribution in the graph case compared with the baseline AE, as shown in Figures 1 and 2. Current tests indicate that searches for BSM physics at the LHC may benefit from obtaining greater information about the rich relations between sectors of particles and their interactions through graph structures.

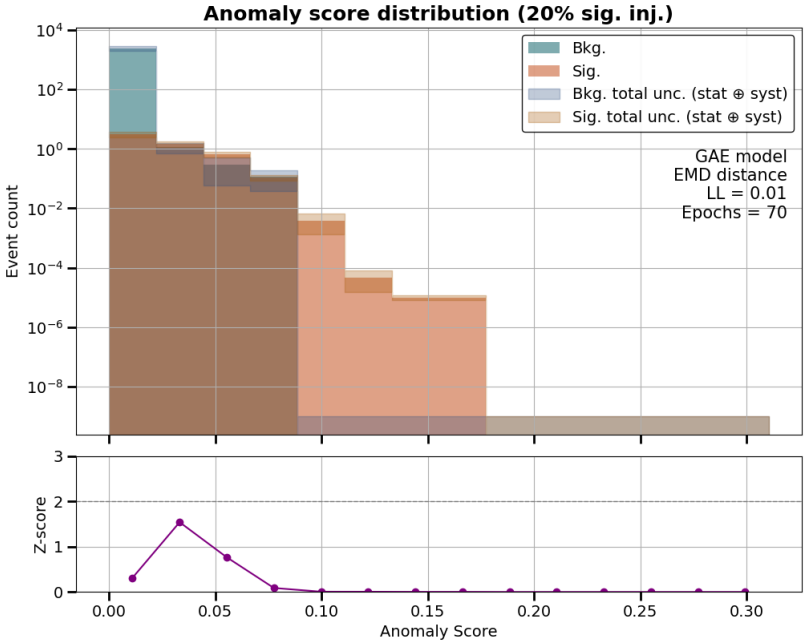

Figure 1: (top) Distribution of anomaly score for leptoquark signal and SM background in graphs using EMD. The test sample combines 20% signal injection evaluated over 70 epochs using the `PyGOD` GAE trained on 20,000 background-only events. (bottom) Z-score distribution for a lower-bound cut.

References

[1] A. Mullin, S. Nicholls, H. Pacey, M. Parker, M. White and S. Williams, JHEP 02 (2021), 160 doi:10.1007/JHEP02(2021)160 [arXiv:1912.10625 [hep-ph]].

[4] Consequently, the models are are not individually optimised, suggesting that strong potential exists to improve performance.

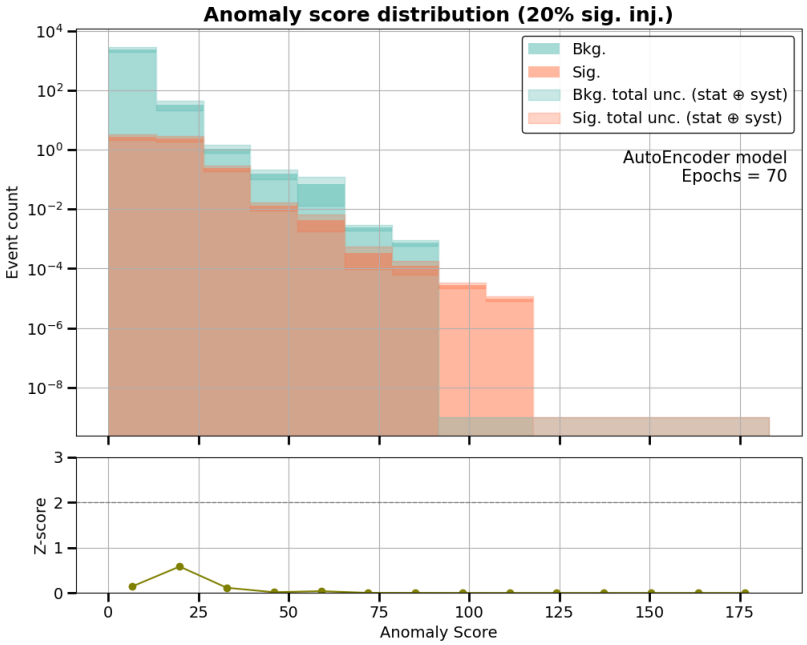

Figure 2: (top) Distribution of anomaly score for leptoquark signal and SM background in graphs using EMD. The test sample combines 20% signal injection evaluated over 70 epochs using the`PyOD` baseline AE trained on 20,000 background-only events. (bottom) Z-score distribution for a lower-bound cut.

[2] I. Doršner and A. Greljo, "Leptoquark toolbox for precision collider studies," JHEP 05 (2018), 126 doi:10.1007/JHEP05(2018)126 [arXiv:1801.07641 [hep-ph]].

[3] P. T. Komiske, R. Mastandrea, E. M. Metodiev, P. Naik and J. Thaler, "Exploring the space of jets with CMS open data," Phys. Rev. D 101 (2020) no.3, 034009 doi:10.1103/PhysRevD.101.034009.

[4] T. N. Kipf and M. Welling, "Variational Graph Auto-Encoders," ICLR Workshop (2017), [arXiv:1611.07308 [stat.ML]].

[5] K. Liu, Y. Dou, X. Ding, X. Hu, R. Zhang, H. Peng, L. Sun and P. S. Yu, "PyGOD: A Python Library for Graph Outlier Detection," J. Mach. Learn. Res. 25 (2024) no.141, 1–9.

[6] Y. Zhao, Z. Nasrullah and Z. Li, "PyOD: A Python Toolbox for Scalable Outlier Detection," J. Mach. Learn. Res. 20 (2019) no.96, 1–7

[7] M. Fey and J. E. Lenssen, "Fast Graph Representation Learning with PyTorch Geometric," ICLR Workshop on Representation Learning on Graphs and Manifolds (2019), [arXiv:1903.02428 [cs.LG]].

[8] ATLAS Collaboration, "Transforming jet flavour tagging at ATLAS," Nature Communications (2025), arXiv:2505.19689 [hep-ex].

[9] H. Qu and L. Gouskos, "ParticleNet: Jet Tagging via Particle Clouds," Phys. Rev. D 101 (2020) 056019, arXiv:1902.08570 [hep-ph].