Interpreting Higgs Boson Interaction Network with Layerwise Relevance Propagation

Alex Y. Luo and Yue Xiao
University of California San Diego, La Jolla, California 92093, USA
(Dated: March 8, 2021)

ABSTRACT: While graph interaction networks achieve exceptional results in Higgs boson identification, GNN explainer methodology is still in its infancy. To introduce GNN interpretation to the particle physics domain, we apply layerwise relevance propagation (LRP) to our existing Higgs boson interaction network (HIN) to calculate relevance scores and reveal what features, nodes, and connections are most influential in prediction. We call this application HIN-LRP. The synergy between the LRP interpretation and the inherent structure of the HIN is such that HIN-LRP is able to illuminate which particles and particle features in a given jet are most significant in Higgs boson identification. The resulting interpretations are ultimately congruent with extant particle physics theory, with the model demonstrably learning the importance of concepts like the presence of muons, characteristics of secondary decay, and salient features such as impact parameter and momentum.

I. INTRODUCTION

Graph neural networks (GNN) are notoriously difficult to interpret [1 and 2], and those employed in the particle physics domain are no different. The graph interaction network has gained popularity with high energy physicists studying fundamental particles because this graph model achieves a highly competitive accuracy, while still working with relatively simple and unprocessed data [3]. However, it is often not fully understood how or why the Graph Interaction Networks make their classifications, or how these models' inner workings might relate to the physical properties of the universe.

The Higgs boson interaction network (HIN) is one such model that we seek to apply the latest research in graph explaining to. The purpose of the HIN is to determine whether or not a given jet, or spray of particles, decayed from a Higgs boson. The implementation examined in this paper is a simplified version of the HIN created in "Interaction networks for the identification of boosted $H \to \bar{b}b$ decays" [3].

Our HIN intakes one graph input and is trained solely on track/particle level features. The graph input represents a single jet instance, where each of the fully connected nodes is a particle, as in Figure 1. As an output, the HIN returns a classification: whether or not the origin of the jet decay was the elusive Higgs boson, or simply background noise. Specifically, the HIN is trained on a particular permutation of the Higgs boson decay known as $H \to \bar{b}b$, where the Higgs boson decays into b hadrons.

We seek to use layerwise relevance propagation (LRP) to explain the decision making process of the HIN. LRP can interpret even highly complex deep learning networks with a strategic application of propagation rules based on deep Taylor expansion [4].

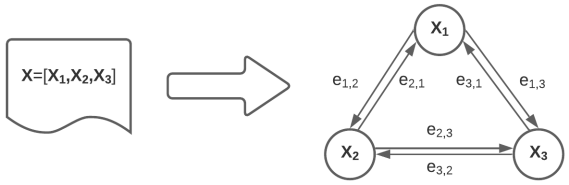

FIG. 1. Feature values X_1, X_2, etc. are regrouped under nodes such that one node is a particle, connected with directional edge weights.

Historically, LRP has been advertised as a way to find and eliminate arbitrary or extraneous features in a complex neural network. However, with respect to our HIN, we are specifically interested in how the LRP can reveal the most important nodes and edges for predictions, which would essentially represent the individual particle importance as well as the particle-particle relationships that are particularly representative of Higgs boson decay, at least to the HIN model. That is to say, it is very possible that we would see known physics phenomena reflected in the decision making of the Higgs boson interaction network.

II. RELATED WORK

GNN interpretation is a relatively new domain, largely kicked off by GNNExplainer in 2019, an interpretation methodology that approached the problem by taking a GNN and returning a salient graph substructure and its most influential node features [1]. We considered this a strong candidate for our model for a time, but were ultimately uncertain about how GNNExplainer's substructure strategy would react to a fully connected interaction network with essentially no preordained substructures. PGExplainer builds off of GNNExplainer

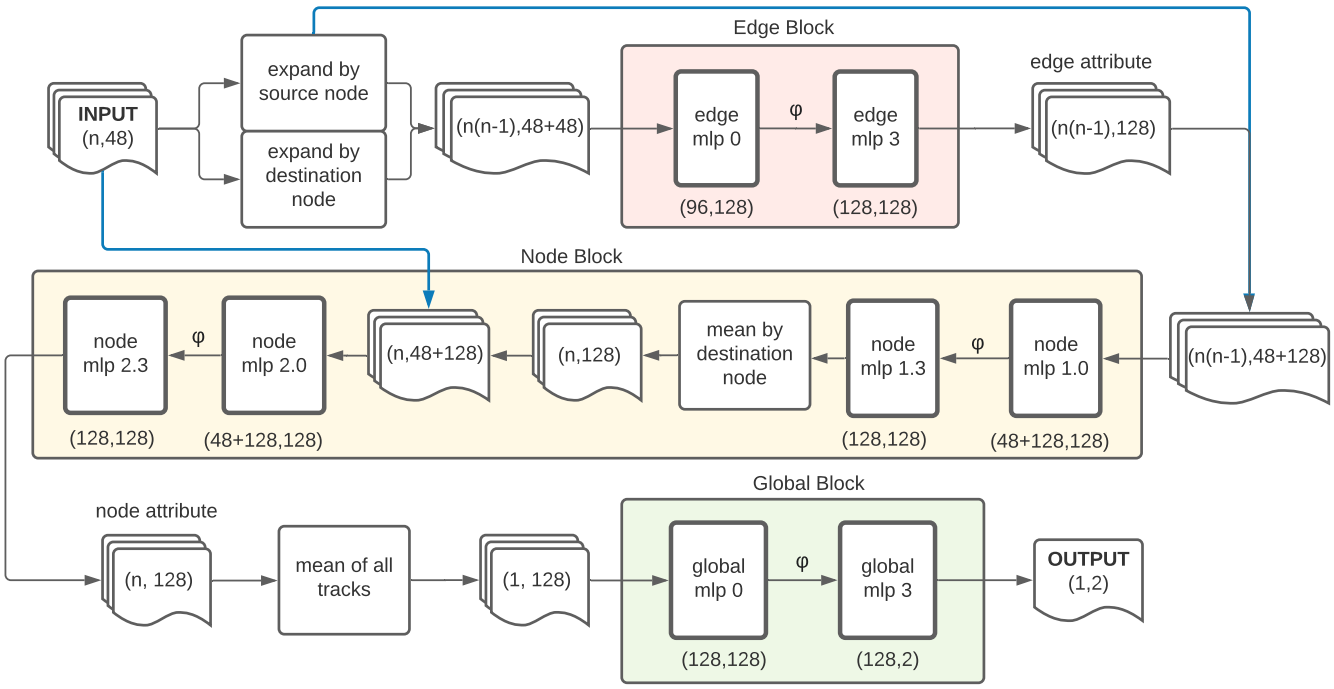

FIG. 2. HIN model architecture. Note the skip in layer indices, e.g. `edge mlp 0` to `edge mlp 3`; the skipped layers `edge mlp 1` and `edge mlp 2` are represented by the nonlinear mapping ρ, and merged with the closest linear layer preceeding them during the LRP computation. n represents the number of tracks in an arbitrary jet; ϕ encodes the nonlinear operation of a layer-wise normalization followed by ReLU. 128 is the dimension of the hidden layer abstract space. 48 is the number of particle features that can be found in a node. The blue pathways highlight the propagation relatively unique to GNNs, where an earlier input is reused multiple times through the layers.

conceptually, with an additional focus on creating a heuristic for generalized model analysis instead of instance based analysis [2]. We feel that, like GNNExplainer, this also model has merit—but ultimately, among several other emerging options for GNN interpretation, we decided to look in the direction of layerwise relevance propagation.

Layerwise relevance propagation is actually a broader technique that has existed outside of the GNN context and has successfully been used to analyze a variety of model types, particularly convolutional neural networks [4]. More recently, LRP has been applied in the context of GNNs, such as in the case of GNN-LRP, which optimizes LRP for GNNs by holistically analyzing graph pathways, or "relevant walks" [5]. LRP has also seen usage in the chemistry domain, where it is implemented on a similar "InteractionNet" GNN to ours, except instead of particle relationships it focuses on molecular structures where edges are bonds [6].

Jet tagging uses machine learning to help classify particle collision events in an efficient and automated way. For a time, the success of these models depended on training with specially crafted features, where physics domain expertise plays a substantial role in aggregating and prioritizing useful information for the neural networks. The interaction network we explore here is notable for finding success training on more fundamental level features, particularly in the case of Higgs boson classification [3]. As far as jet tagging is concerned, layerwise relevance propagation has been used on CNNs and RNNs (convolutional and recurrent, respectively) in the particle physics domain [7], but less so for GNNs. As such, our goal is to make that step by applying LRP to the Higgs boson interaction network in this paper.

III. HIGGS BOSON INTERACTION NETWORK

The Higgs boson interaction network, or HIN, is programmed using PyTorch Geometric, which is a streamlined package specifically meant for GNN implementation [8]. PyTorch Geometric simplifies both the creation of the particle-particle interaction graphs and the training of the model itself. Jet entries in the data are comprised of track level features, tracks being the reconstructed pathways and measurements for a given particle. When the track features are adapted for the GNN, each jet level entry becomes a particle-particle

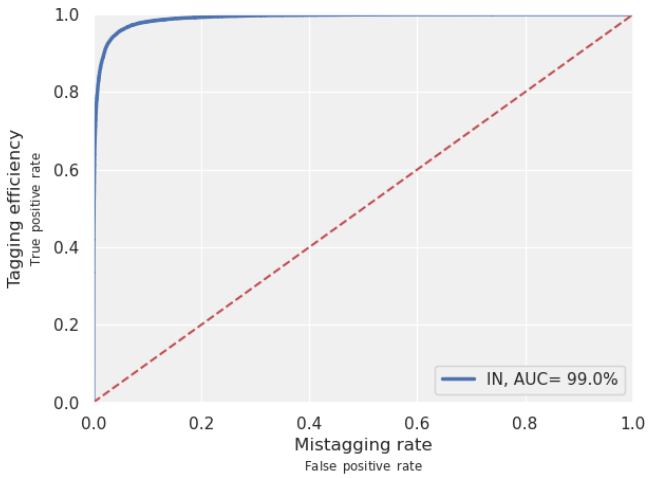

FIG. 3. ROC AUC for the Higgs boson interaction network. The dash line in red serves as a baseline reference.

interaction graph representing the relationships of every particle to all other particles in the graph bidirectionally. Every track gets its own node, and for each track, the features are regrouped under the corresponding node. Broadly, graph models are desirable for jet representation because they reflect the absence of an inherit ordering in the jets.

After processing the data into graphs, we use PyTorch Geometric to train the GNN by passing the data through three function blocks: and edge block, node block, and global block. As in Figure 2, the model's forward propagation adjusts corresponding edge weights, node weights, and global weights in each block in accordance with encoded transformation sequences: concatenations, linear transformations, batch normalizations, and ReLU activations. These transformation functions constitute the pathways that LRP will backpropagate across in order to acquire relevancy scores, which we will elaborate upon in the next section.

As seen in Figure 3, the interaction network performs exceptionally well, with a 99.0% AUC. This is what we would hope to see from what has experimentally yielded some of the best performance for Higgs boson identification thus far [3]. However, we still need layerwise relevance propagation to uncover how exactly the HIN is accomplishing such invaluable performance.

See Appendix A for additional context for the model training.

IV. LAYERWISE RELEVANCE PROPAGATION

LRP essentially redistributes relevance scores backward, starting from the model output, passing through the layers, all the way to the input. Each layer's relevance score is propagated from the layers closest to output, through the hidden connections, to the current layer, and at each juncture such the sum of the relevance score is kept approximately the same. The foundation of LRP is built off of deep taylor expansion, taking gradients at each layer to deduce activation with respect to the following layer. Because LRP traverses the entire model, we can calculate relevancy for every edge, every node, every node feature, and more.

A. Conservation Law

The flow of the relevance scores as it is backpropagated is analogous to the flow of water in a river: the total amount is conserved as it flows through the forks. This is described as a conservation property for LRP [4]. As such, the partial relevance scores ultimately attributed to the raw input from different paths should be directly summed up to approximate the actual prediction score of the output.

There is a distinction to acknowledge in applying LRP to GNNs, such as our Interaction Network, as opposed to other deep learning frameworks, like CNNs. Other models have a more straightforward layer by layer propagation, whereas the HIN re-propagates certain layers, particularly the input and the input source node transformation, reapplying those values in multiple instances across the block layers (see the blue highlighted pathways in Figure 2). This can be thought of as a weight sharing, and it affects the consideration of the conservation property. Namely, LRP backpropagation reaches the input from multiple pathways, and each occurrence must be considered in the conservation calculations.

Figure 4 depicts an example of one backpropagation step in LRP. For an arbitrary node in Layer j, call it v_j, it receives the relevance scores from all the nodes that connects to it from the layer k that follows it. Hidden layer k, like other layers, draws from the output of an adjacent layer in activation from layer j, but unlike other layers, also pulls from the raw input. Thus, when propagating the relevance score through the model backwards, the relevance score R_k is split into two parts, R'_k and R_{src}, among which R'_k flows into layer j and the other layers beneath it, and R_{src} is attributed directly to the input.

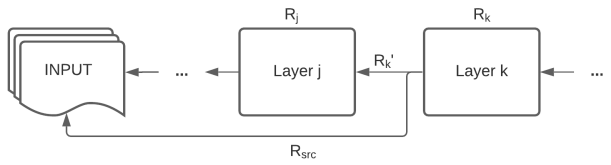

FIG. 4. Relevance propagation from Layer k backwards into Layer j and input. In the forward pass k sources from both Layer j and the input, so the relevance propagation would work accordingly. Intuitively, $\mathbf{R_k} = \mathbf{R'_k} + \mathbf{R_{src}}$

B. LRP-ϵ

LRP methodology provides several propagation rules that are outlined in "Layer-Wise Relevance Propagation: An Overview" [4]. In the case of the HIN, we apply a variant of the LRP-ϵ rule uniformly on the layers in our model, layers which can be roughly divided into two types: simple linear layers, and normalized rectified linear layers. For example, in our LRP propagation, `edge mlp 3` is one of the simple linear layers; on the other hand, `edge mlp 0` is joined with the nonlinear operation ϕ (batch normalization followed by ReLU activation) into a normalized rectified linear layer.

Denote the relevance score of node v_i in layer j as $(\mathbf{R_j})_i$, which can be computed as a proportion of the relevance score from the following layer, layer k. As shown, the amount of contribution can be conveniently computed by a forward pass. The LRP-ϵ rule for a given layer is as follows:

$$(\mathbf{R_j})_i = \sum_{u,\exists(i,u)} \frac{(\mathbf{a_j w_j^T})_{iu}(\mathbf{R_k})_u}{\epsilon' + \sum_u (\mathbf{a_j w_j^T})_{iu} + \mathbf{b_j}}$$
$$\text{where } \epsilon' = \epsilon \cdot \text{sign}(\sum_u (\mathbf{a_j w_j^T})_{iu} + \mathbf{b_j})$$

Here $\mathbf{a_j}$ stands for the input at layer j, $\mathbf{w_j}$ and $\mathbf{b_j}$ respectively stand for the weight and bias at layer j. ϵ is introduced in the denominator with an appropriate sign to absorb some of the relevance score as well as to prevent division by zero, in accordance with the epsilon rule. Bold typeface indicates when the variables are tensors instead of scalars.

The proportion calculated in the LRP-ϵ rule follows a "gradient×input" convention to find out how each part of the input contributes to the layer activation, by computing by the product of the layer input and the partial derivative of the layer output with respect to the layer input. The partial derivative of the layer output can be viewed as "the rate of contribution to the activation" and thus the product can be viewed as the particular amount of contribution of the part of input towards the layer activation. For an arbitrary linear layer j, represented as a linear function ρ of layer input $\mathbf{a_j}$,

$$\begin{aligned} \rho(\mathbf{a_j}) &= \mathbf{a_j w_j^T} + \mathbf{b_j} \\ \frac{\partial}{\partial \mathbf{a_j}} \rho(\mathbf{a_j}) &= \frac{\partial}{\partial \mathbf{a_j}} (\mathbf{a_j w_j^T} + \mathbf{b_j}) \\ &= \mathbf{w_j^T} \\ \Rightarrow \mathbf{a_j} \cdot \frac{\partial}{\partial \mathbf{a_j}} \rho(\mathbf{a_j}) &= \mathbf{a_j} \cdot \mathbf{w_j^T} \end{aligned}$$

C. Node and Edge Relevance

Following the paths illustrated in Figure 2 backwards, we can obtain several partial relevance scores from layers `node mlp 2.0`, `node mlp 1.0`, and `edge mlp 0`, which we denote $\mathbf{R'_{input}}, \mathbf{R'_{src}}, \mathbf{R_{src,dest}}$ respectively. While the partial relevance score $\mathbf{R'_{input}}$ from `node mlp 2.0` is directly attributed to the raw input, the scores attained from `node mlp 1.0` and `edge mlp 0` correspond to the relevance of the edges. Thus, the edge relevance score can be obtained by aggregating $\mathbf{R'_{src}}$ and $\mathbf{R_{src,dest}}$. To quantify the relative relevancy of each edge in a particular jet graph to the classification, we introduce the edge significance, $\mathbf{S_{edge}}$, computed as follows:

$$\text{Let } \mathbf{R_{edge}} = \mathbf{R_{src,dest}} + [\mathbf{R'_{src}}, \mathbf{I}_{n\cdot(n-1)}],$$
$$\forall i \in [0, n\cdot(n-1)], \quad (\mathbf{S_{edge}})_i = \frac{||(\mathbf{R_{edge}})_i||_F}{\sum_j (\mathbf{S_{edge}})_j}.$$

Here $\mathbf{R_{edge}}$ is a matrix with dimension $(n(n-1), 48+48)$, in which the first 48 columns are source node features of the directed edge and the last 48 columns are destination node features. The idea of $\mathbf{S_{edge}}$ is to measure the importance of the edge by the amount of information flow through it in the decision making process.

The node relevance map $\mathbf{R_{node}}$ can be computed by joining the edge relevance scores with the partial input relevance score. To attribute relevance score for each node feature, the edge relevance scores are aggregated by taking the mean with respect to their corresponding nodes:

$$\begin{aligned} \mathbf{R_{node}} = \mathbf{R'_{input}} \\ &+ \texttt{scatter\_mean}(\mathbf{R'_{src}}, \texttt{src}) \\ &+ \texttt{scatter\_mean}(\mathbf{R_{src,dest}}[:, :48], \texttt{src}) \\ &+ \texttt{scatter\_mean}(\mathbf{R_{src,dest}}[:, 48:], \texttt{dest}) \end{aligned}$$

Here `src` and `dest` are the node indices of the source and destination node of the directed edge.

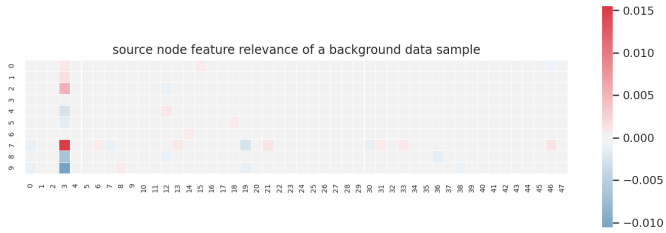

FIG. 5. $\mathbf{R}_{\text{node}}$ heat map of IN trained on dummy input. Note that the relevance scores are concentrated in the 4th column, which corresponds to $\texttt{feature}_3$

D. LRP Dummy Model

To verify the validity and to evaluate the primary output, $\mathbf{R}_{\text{node}}$, we designed the LRP dummy model. The dummy model has the exact same architecture as our HIN, but is trained on synthesized data instead.

The synthesized dummy data mimics the actual data, with 48 features and a fixed number of 10 tracks per jet. The entries of all features except feature 0 and 3 are all meaningless Gaussian noise with mean set to 0 and standard deviation set to 1. Feature 0 and 3 are designed to represent meaningful measurements, taking up values in $\{0, 10\}$ and $\{-20, -10, 0, 10\}$ respectively. The label y of each jet is computed as following,

$$y = \text{sign}(\frac{1}{10}\sum_{i=0}^{9} \texttt{feature}_0 + 2 \times \texttt{feature}_3)$$

Notice that even though $\texttt{feature}_0$ is involved in the formula, it is completely dominated by the value of $\texttt{feature}_3$. Therefore, we would reasonably expect to see that $\mathbf{R}_{\text{node}}$ should have large magnitudes at column 3 only. Figure 5 shows that we are able to capture the contribution of $\texttt{feature}_3$ to the prediction, with a reasonably small amount of noise.

V. RESULTS

The biggest question of this project is whether or not the application of layerwise relevance propagation would bear fruit when applied to the Higgs boson interaction network. The specific goal is that on a case by case examination of inputs, we can understand the physics concepts that the HIN is determining to be most valuable. We explore validity of layerwise relevance propagation's interpretation of the Higgs boson interaction network in HIN-LRP.

A. HIN-LRP

Since LRP is applied on an instance level, it gives us GNN analysis on a jet by jet basis. So for our HIN, we get a relevance map that corresponds directly to the shape and values of a given jet input. And thanks to the organization of the jet graph input, the HIN-LRP interpretation of a jet essentially enumerates exactly which particles and aspects of those particles are most important to whether it is a Higgs boson signal.

We present the interpretation result of each jet as a pair visualization of with a heat map and a 3D network plot. The heat map plots a saliency matrix such that every individual input relevance score is clearly laid out. The tracks are left to right ordered by increasing momentum (`track_pt`). The more intense the color in a cell the more relevant that entry of the input is to the Higgs probability output. Notably, positive and negative activation is not responsible for corresponding positive and negative Higgs signal labeling, all magnitudes are significant. The 3D plot is a combination of physical space on the xy plane with polar-esque values of `track_phirel` and `track_etarel`, with momentum once more on the z axis. Here, the relevance of a node is summarized by the norm of its features and depicted with size. The edge significance $\mathbf{S}_{\text{edge}}$ is highlighted in red above a threshold of relevance and a simple gray below, with a low uniform color intensity.

Momentum plays a major role, tying the readability of the graphs together, because it is well known that high momentum particles are often representative of a jet's character. And indeed, it is often the case that high momentum particles are associated with high relevance scores. Momentum sort is present in the axes of both plots to unify the sense of which particles and connections are most responsible for the classifications.

The LRP visualizations show a promising correlation to what is understood about the physics behind Higgs boson to b hadron decay. Many of the graphs are activated along two columns, which may suggest that the HIN is recognizing two major prongs in the jet, potentially representing the paths of each b hadron. When a given input is highly dependent on a single feature, it is often what a physicist would expect to be important: momentum, impact parameter, energy level. The model even recognizes the importance of electrons and muons—since these are frequently present in the decay of the b hadron, there is a good chance that a Higgs boson too may have been involved. These concepts are well understood by domain experts, but it is only with HIN-LRP that we can see through the eyes of the model itself.

Below, we have selected just 4 jets (among millions) that the HIN is most confident in labeling as Higgs boson signals to interpret and visualize below, in order to connect the major driving forces for the model's decisions with physics theory.

See Appendix B for feature definitions.

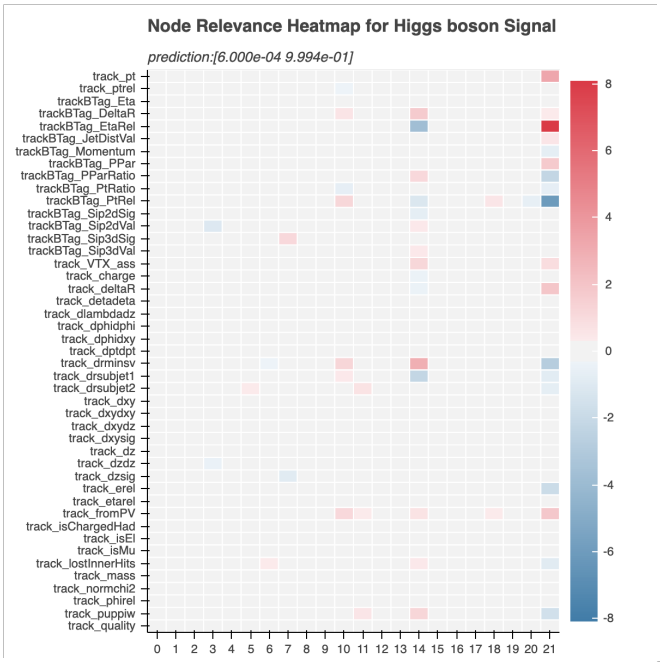

FIG. 6. relevance heat map of selected jet 1.

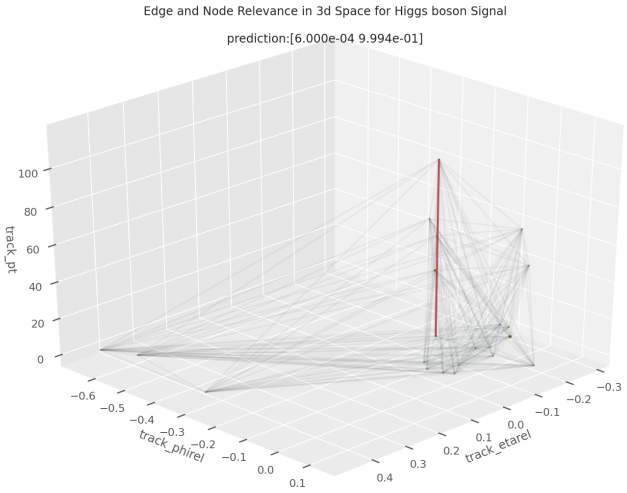

FIG. 7. edge significance and node relevance of selected jet 1 in 3D space

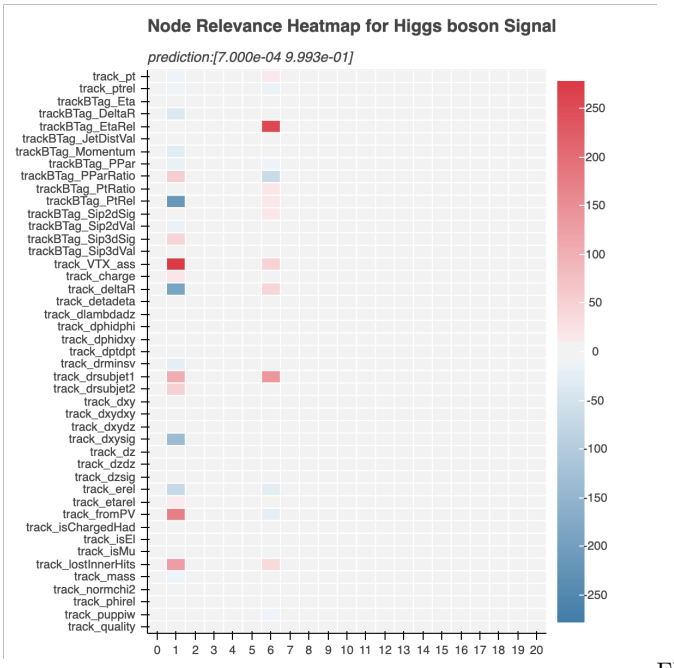

FIG. 8. relevance heat map of selected jet 2

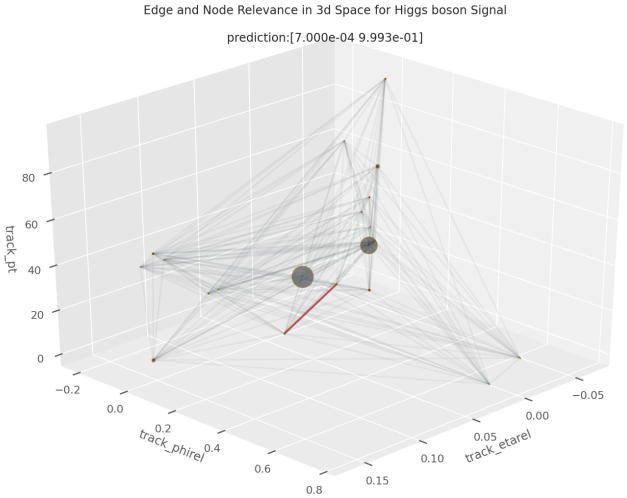

FIG. 9. edge significance and node relevance of selected jet 2 in 3D space

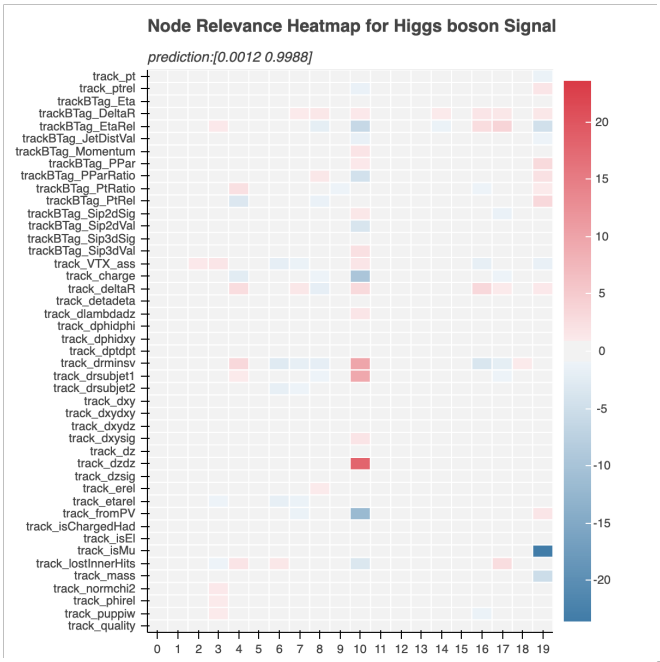

FIG. 10. relevance heat map of selected jet 3

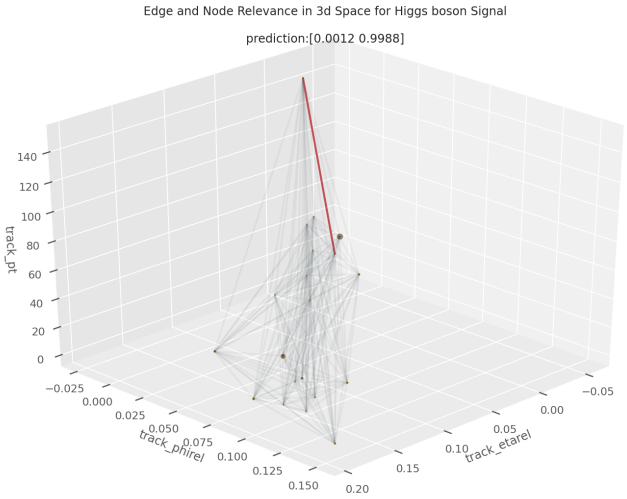

FIG. 11. edge significance and node relevance of selected jet 3 in 3D space

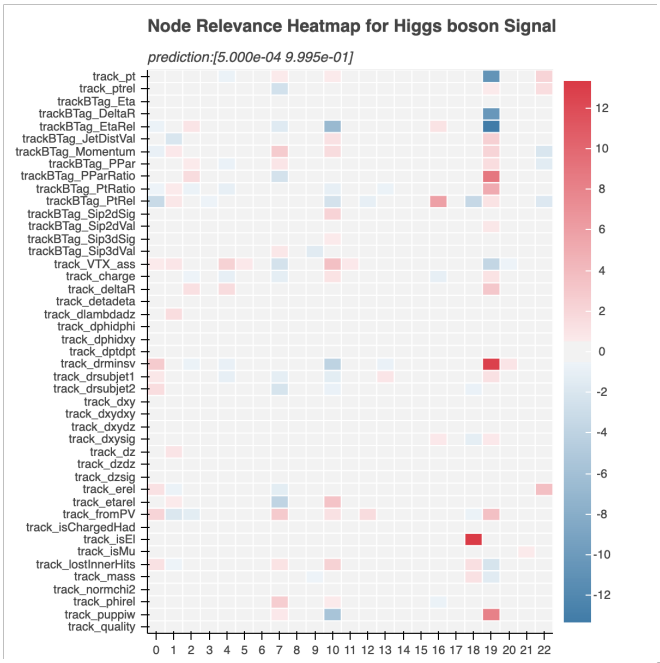

FIG. 12. relevance heat map of selected jet 4

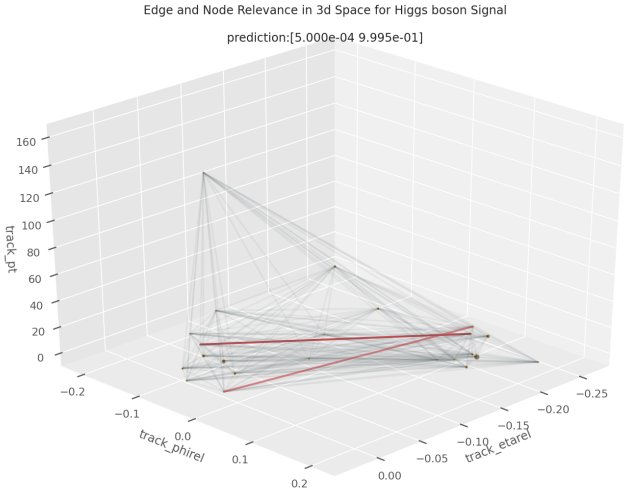

FIG. 13. edge significance and node relevance of selected jet 4 in 3D space

B. Visualizations

Figure 6 and 7 show the interpretation result of selected jet 1, for which the model is more than 99.9% confident in its prediction. The colored cells are mostly distributed in columns 14 and 21, suggesting that those two nodes are most relevant to the prediction. This aligns with the intuition that the model is perceiving the two pronged nature of Higgs bosons decaying into 2 b hadrons. Note how the highest momentum node is the most important, and in the 3D space we see the edge connecting to this node to be highly activated. Physicists have shown that Higgs boson decay products are likely to have larger transverse momentum value relative to the jet axis [9].

Figure 8 has all the relevance scores attributed among only two columns, again alluding to the two pronged decay into b hadrons. In other words, those two particles are the most significant nodes in the prediction, something that is clearly affirmed in Figure 9's 3D plotting. Notably, it is the lower momentum tracks in this jet that are more responsible for classification, but even in this situation momentum itself is a highly relevant feature.

Figures 10 and 11 show the interpretation result of a Higgs boson jet that the model believes with 99.9% confidence is a signal. Note that in bottom right of Figure 10, the cell corresponding to node 19 and feature `track_isMu` is colored a rather dark shade of blue, suggesting that this entry has been particularly important for the model to classify the jet as a signal. A quick examination of the raw data shows that the entry is 1 in the corresponding position, meaning that node 19 is a muon. This matches with the theoretical expectation that the presence of a muon among the decay products is a strong indicator of the jet being related to a Higgs boson [9].

Figure 12 shows how, in track 19, the HIN utilizes a host of positional information like momentum, `track_pt`, angle from jet (`track_DeltaR`, `track_EtaRel`), angle from secondary vertex (`track_drminsv`), all to classify the Higgs boson sginal with great confidence. Alongside this positional information in track 19 is the boolean for whether the track is "pileup-like", or more colloquially, an intrusive track from separate jet decay. This indicates that the model understands that the aforementioned positional features are only helpful given that this track is actually native to the jet being analyzed. In track 18, the binary feature `track_isEl` is also hugely important for making the classification decision. `track_isEl` encodes whether the given particle is an electron, which, similar to muons, is a strong indicator of the jet being related to a Higgs boson when it is present [9].

VI. CONCLUSION

We presented a specific way to interpret the inner workings of the Higgs boson interaction network through layerwise relevance propagation, and have found that the methodology does, to a significant extent, reflect the foundations of theoretical physics that the model is trained on. With a simple application of just the LRP-ϵ rule, we now have the ability to visualize the relevance of any given jet graph input with respect to how likely it is to be a Higgs boson signal. Our implementation code can be found here [10].

The Higgs boson interaction network itself was trained on a simulated dataset from the CMS Collaboration's Open Data Portal. It was implemented with PyTorch Geometric and directly built from prior work establishing the primacy of interaction networks for Higgs boson signal classification [3].

This is first and foremost a foray into LRP GNN interpretation for a hyper specific purpose: Higgs bosons decaying into b hadrons. HIN-LRP successfully asserts the value of layerwise relevance propagation for providing transparency regarding how well GNNs perceive the rules governing particle collision and decay. Our techniques can potentially be applied to more deeply understand a model, optimize or reduce features necessary for training, and, within the particle physics domain, thoroughly compare the essence of varied jet inputs. HIN-LRP could benefit greatly from a way to generalize the results across the millions of jet entries to paint a bigger picture. All said, GNN interpretation is constantly innovating, and the tactics we have outlined can be further refined alongside the latest methodologies—varied LRP rules [4], generalization heuristics [2], relevant walks [5]—to gain an even deeper understanding of how deep learning comes to understand the laws of physics.

ACKNOWLEDGMENTS

We wish to acknowledge the support of our domain mentors Javier Duarte and Frank Würthwein, our domain TA Farouk Mokhtar, and our Capstone instructor Aaron Fraenkel. The support of our section peers Sharmi Mathur, Nathan Roberts, and Darren Chang was greatly appreciated. We also thank Justin Eldridge and Chen Cai for providing critical insight on graph theory.

Appendix A: Higgs Boson Interaction Network

1. Data

This Higgs boson interaction network was trained on a monte carlo of the CMS collaboration's collision

TABLE I. Definitions of the features mentioned in this paper

track_pt	Transverse momentum of the charged PF candidate
track_etarel	Pseudorapidity $\Delta\eta$ of the track relative to the jet axis
track_phirel	Azimuthal angular distance $\Delta\phi$ between the charged PF candidate and the AK8 jet axis
track_isMu	Boolean that is 1 if the charged PF candidate is classified as a muon
track_isEl	Boolean that is 1 if the charged PF candidate is classified as an electron
trackBTag_EtaRel	Pseudorapidity $\Delta\eta$ of the track relative the AK8 jet axis
trackBTag_PtRel	Component of track momentum perpendicular to the AK8 jet axis
track_DeltaR	Pseudoangular distance (ΔR) between the charged PF candidate and the AK8 jet axis
track_drminsv	Minimum pseudoangular distance (ΔR) between the associated SVs and the charged PF candidate

simulation data. The CMS collaboration simulates collision events in a ground up fashion based on confirmed physics theory [11]. A benefit of this is that we can also lessen the rarity of the Higgs boson signal, such that it's actually useful to train this model. After many events are generated with the simulators, the most relevant jets are filtered through with a particle-flow algorithm that removes a certain amount of collision noise. Simulations are the preferred training data because the LHC produces approximately 10 quadrillion collisions per year, resulting in petabyte level amounts of particle data that are impractical to train on. Additionally, the actual data has an extremely imbalanced class ratio (approximately 99 to 1), as the Higgs boson is an extremely rare occurrence in real collision events.

Training models on the actual data would potentially result in a model with high accuracy, but low precision. Also, by using simulated data, we can concentrate solely on jets, because the actual LHC data often seeks to observe many other kinds of data. With all these considerations, we sourced the simulated data from the CERN Open Data Portal, and pulled out approximately 3 million jet entries. The particular Higgs boson event we draw from is the decay into b hadron pairs ($H \to \bar{b}b$), with background collision noise (QCD).

The Interaction Network ultimately trained on approximately 2 million jet entries, and evaluated on a random subset of 128000 jets due to time complexity. IN is trained with batch size of 128 jet particle-particle interaction graphs for 10000 minibatches per epoch (20% of the mini batches are put aside for validation purpose) with Adam optimizer and an initial learning rate of 10-4. We had planned for 150 epochs for the model with early stopping, but it actually converged quickly to a desirable result in less than 10 epochs.

2. Model

The edge, node, and global block structure is facilitated heavily by PyTorch Geometric abstraction, and further context for can be found in the documentation for PyTorch Geometric's `Metalayer` function [12], as well as the paper it was based on: "Relational inductive biases, deep learning, and graph networks" [13].

Appendix B: Selected Feature Definitions

See Table 1, or the complete CERN feature sheet at: [14]

Appendix C: Project Proposal

Deep learning (DL) has commonly been regarded as a black box. By black box, we mean that we lack full understanding of how it works, despite knowing that it can produce outstanding performance. After implementing a successful GNN classifier for identifying Higgs bosons, we are left wondering how exactly our model makes the decisions. While there is an awareness in the physics domain that mass and momentum are large factors, it is difficult to understand concretely how the graphs weigh the features, especially since the more commonly understood features are usually decorrelated. We can look at the very model we just created, and with the same data, see if we can explain the model's understanding of Higgs boson jets. Is there potential for furthering the physics domain's understanding of the problem by explaining the model's composition?

This approach to deep learning is not exclusive to particle physics, it also applies to general applications of DL in different fields. Explainable DL models are becoming more and more popular recently, as people begin to notice its potential to understand various problems when the neural network's structure is more transparent.

Deep learning is still quite new, and deep learning interpretability is even newer, but quite a lot research has been done to interpret deep learning models. Many papers have already been written on understanding CNNs through the visualization of the network's activation, producing images that peel back the curtain on the neural net's mind. Unfortunately, we have far less understanding of how GNN works—the graph

representation of the hidden layers are not as intuitively visualizable as feature maps of images in CNNs. And so we lack a comprehensive explanation of the Interaction Network we implemented for jet identification.

We know that a major advantage of IN is its ability to learn from low level features that are closer to the raw measurements from the collider, i.e. it does not depend as much on expert knowledge as the previously used classical ML solutions. We propose that by understanding a trained IN, we could potentially gain more insight into both the physics problem itself and possible direction for further improvement of the model. We hypothesize that we could render a "most signal looking input" or the graph that our model most strongly associates with the Higgs to b hadron decay.

Instead of studying a general question on how to explain the behavior of graph nets, we want to focus on the interpretation of the GNN based IN. A unique and good thing about studying IN is that, unlike many other GNNs, we have a rough expectation on its behavior based on existing particle physics theories. Probing into a trained IN, we expect to see that it actually learns the expert crafted variables from the raw input data, and it learns to distinguish the significance of different features; we also expect to discover something new or unexpected about the interaction of certain low level features. There is a potential to perceive actual physics phenomena by understanding how the GNN gets trained in a more coherent way.

Though explaining GNN is a relatively untapped market, there are plenty of new and exciting reference points for us to reasonably build from. There is a long history of visualizing jets with image abstractions, so the hypothetical "most signal input" would have a straightforward extant visualization method. PyTorch Geometric does provide avenues to explain GNNs based on a paper that highlights and visualizes the most activated subsets of GNNs, which would synergize nicely with our existing PyTorch model. And just the other week a physics domain related paper was published about "Explainable AI for ML jet taggers using expert variables and layerwise relevance propagation" [7], which closely relates to the classical ML model we implemented with XGBoost in our replication.

[1] R. Ying, D. Bourgeois, J. You, M. Zitnik, and J. Leskovec, Gnnexplainer: Generating explanations for graph neural networks (2019), arXiv:1903.03894 [cs.LG].

[2] D. Luo, W. Cheng, D. Xu, W. Yu, B. Zong, H. Chen, and X. Zhang, Parameterized explainer for graph neural network (2020), arXiv:2011.04573 [cs.LG].

[3] E. A. Moreno, T. Q. Nguyen, J.-R. Vlimant, O. Cerri, H. B. Newman, A. Periwal, M. Spiropulu, J. M. Duarte, and M. Pierini, Interaction networks for the identification of boosted h→bb¯ decays, Physical Review D **102**, 10.1103/physrevd.102.012010 (2020).

[4] W. Samek, M. Gregoire, A. Vedaldi, L. K. Hansen, and K.-R. Muller, *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning* (Springer International Publishing, 2019).

[5] T. Schnake, O. Eberle, J. Lederer, S. Nakajima, K. T. Schütt, K.-R. Müller, and G. Montavon, Higher-order explanations of graph neural networks via relevant walks (2020), arXiv:2006.03589 [cs.LG].

[6] H. Cho, E. K. Lee, and I. S. Choi, Interactionnet: Modeling and explaining of noncovalent protein-ligand interactions with noncovalent graph neural network and layerwise relevance propagation (2020), arXiv:2005.13438 [q-bio.BM].

[7] G. Agarwal, L. Hay, I. Iashvili, B. Mannix, C. McLean, M. Morris, S. Rappoccio, and U. Schubert, Explainable ai for ml jet taggers using expert variables and layerwise relevance propagation (2020), arXiv:2011.13466 [hep-ph].

[8] M. Fey and J. E. Lenssen, Fast graph representation learning with pytorch geometric (2019), arXiv:1903.02428 [cs.LG].

[9] A. Sirunyan, A. Tumasyan, W. Adam, F. Ambrogi, E. Asilar, T. Bergauer, J. Brandstetter, E. Brondolin, M. Dragicevic, J. Erö, and et al., Identification of heavy-flavour jets with the cms detector in pp collisions at 13 tev, Journal of Instrumentation **13** (05), P05011–P05011.

[10] Y. Xiao and A. Luo, Hin-lrp, `https://github.com/HIN-LRP/Interpret-InteractionNetwork` (2021).

[11] .

[12] Source code for torch_geometric.nn.meta.

[13] P. W. Battaglia, J. B. Hamrick, V. Bapst, A. Sanchez-Gonzalez, V. Zambaldi, M. Malinowski, A. Tacchetti, D. Raposo, A. Santoro, R. Faulkner, C. Gulcehre, F. Song, A. Ballard, J. Gilmer, G. Dahl, A. Vaswani, K. Allen, C. Nash, V. Langston, C. Dyer, N. Heess, D. Wierstra, P. Kohli, M. Botvinick, O. Vinyals, Y. Li, and R. Pascanu, Relational inductive biases, deep learning, and graph networks (2018), arXiv:1806.01261 [cs.LG].

[14] J. Duarte, Sample with jet, track and secondary vertex properties for hbb tagging ml studies higgstobbntuple$_h iggstobb_q cd_r unii_1 3tev_m c$(1970).