NuGraph2

A graph network for particle reconstruction

V Hewes (she/they)
NuGraph workshop
13th November 2023

NuGraph2

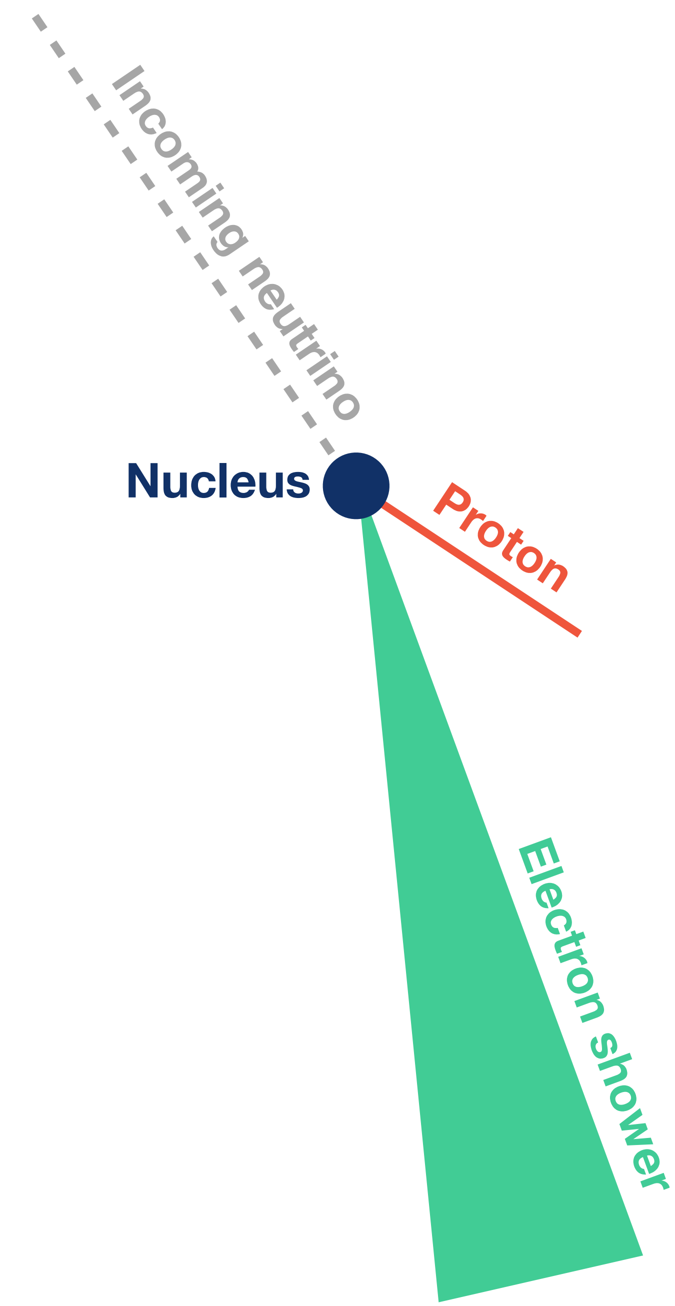
NuGraph2

Neutrino physics

- Neutrinos are neutral leptons, so **we do not observe them directly**.

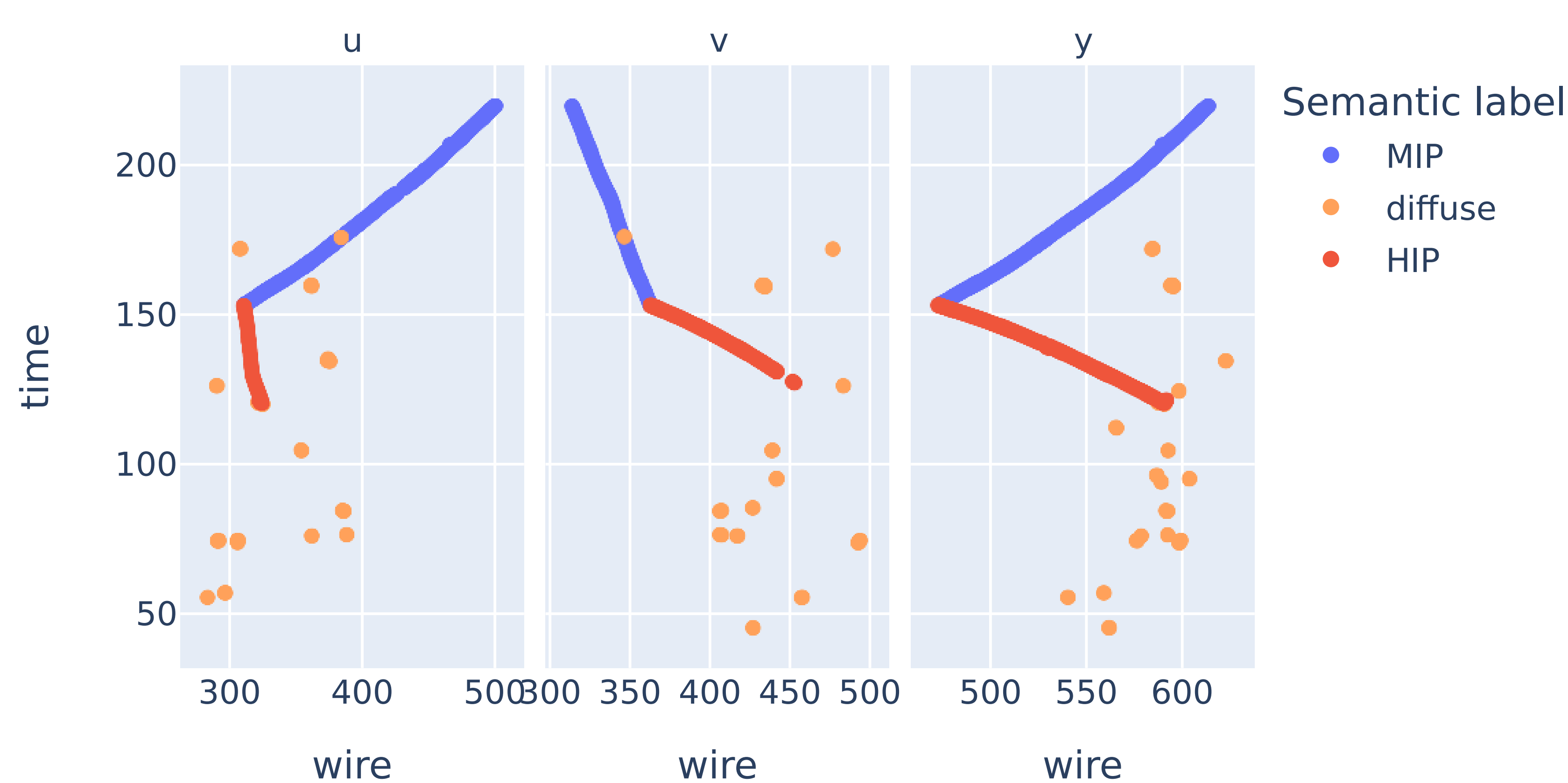
Neutrino physics

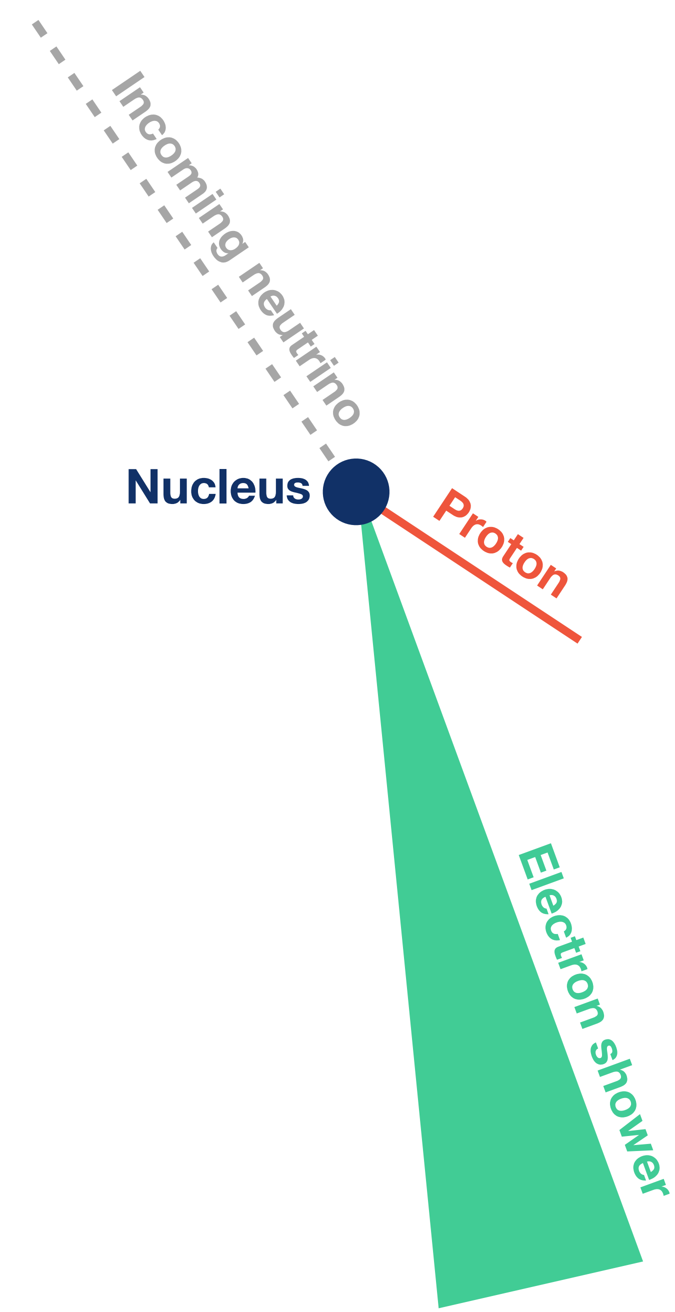
- Neutrinos are neutral leptons, so **we do not observe them directly**.
  - Instead, we search for the **visible particles** that are **produced when they interact**.



Neutrino physics

- Neutrinos are neutral leptons, so **we do not observe them directly**.
    - Instead, we search for the **visible particles** that are **produced when they interact**.





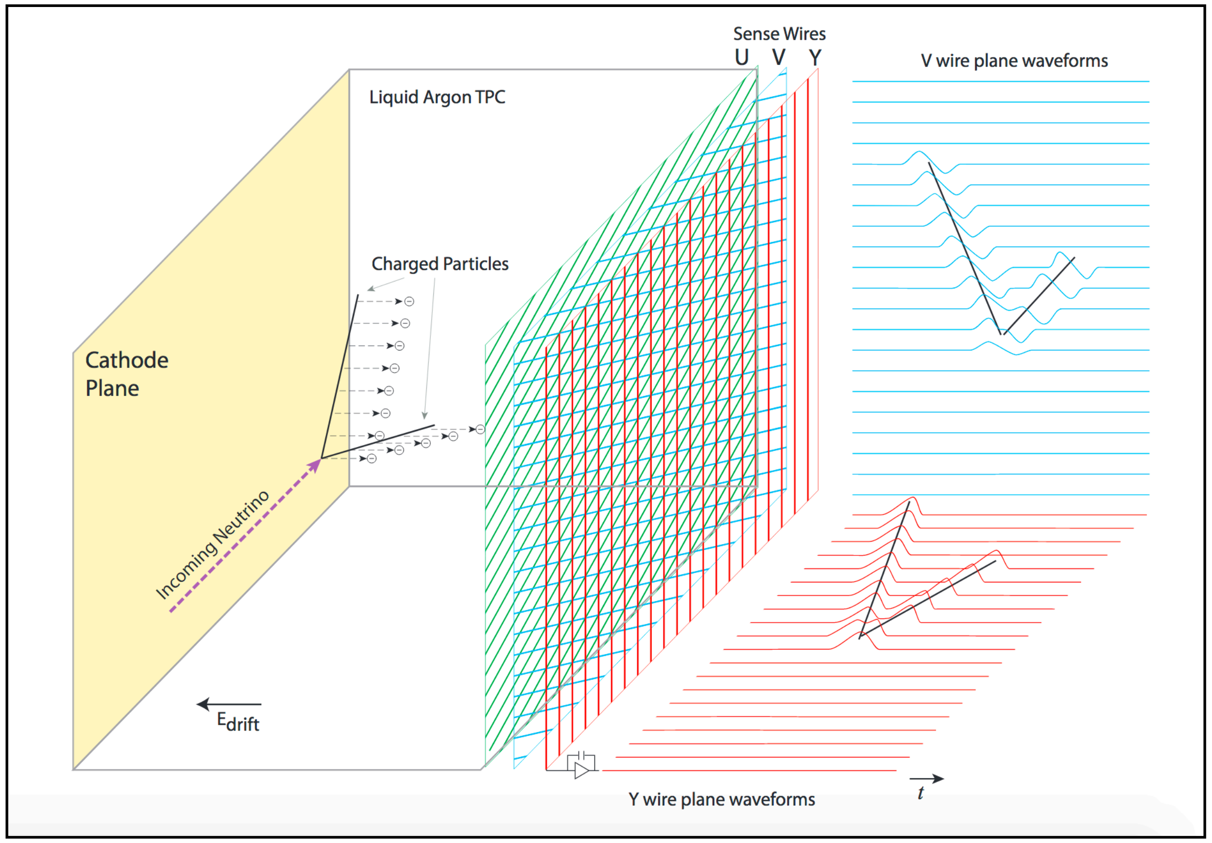
Liquid Argon TPCs

- Liquid Argon Time Projection Chambers (LArTPCs) currently a heavily utilised detector technology in neutrino physics.
  - At FNAL: MicroBooNE, Icarus, SBND.
  - Future: DUNE (70kT LArTPC deep underground, plus near detector).

Liquid Argon TPCs

- Liquid Argon Time Projection Chambers (LArTPCs) currently a heavily utilised detector technology in neutrino physics.
  - At FNAL: MicroBooNE, Icarus, SBND.
  - Future: DUNE (70kT LArTPC deep underground, plus near detector).

- Charged particles ionize liquid argon as they travel.
- Ionisation electrons drift due to HV electrode field, and are collected by anode wires.
- Wire spacing ~3mm – **high-resolution detector**.



MicroBooNE open dataset

- The work presented here today utilises **simulated neutrinos** from **MicroBooNE's Open Data Release** (link).
  - **Simulated neutrino interactions** with **cosmic data overlays**.

MicroBooNE open dataset

- The work presented here today utilises **simulated neutrinos** from **MicroBooNE's Open Data Release** (link).
  - **Simulated neutrino interactions** with **cosmic data overlays**.
- This network architecture was originally developed in the context of the **DUNE Far Detector** geometry.
  - Motivation: reconstructing complex and high-multiplicity atmospheric and ν_τ interactions.

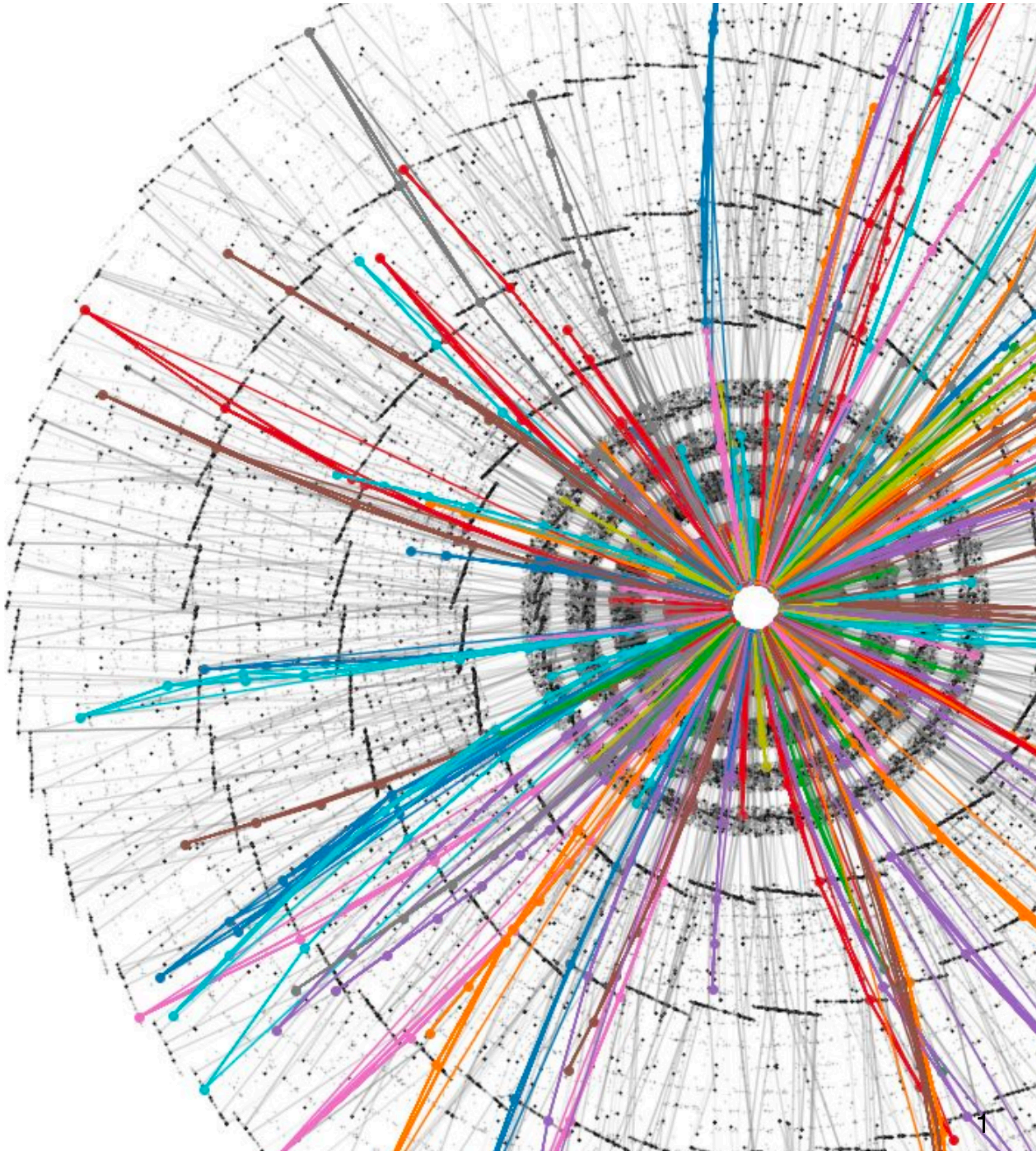
MicroBooNE open dataset

- The work presented here today utilises **simulated neutrinos** from **MicroBooNE's Open Data Release** (link).
  - **Simulated neutrino interactions** with **cosmic data overlays**.
- This network architecture was originally developed in the context of the **DUNE Far Detector** geometry.
  - Motivation: reconstructing complex and high-multiplicity atmospheric and ν_τ interactions.
- This network architecture is developed to have **broad applicability**, without being tied to any particular detector geometry.
  - NuGraph2 is a **general-purpose particle reconstruction tool**.
  - Developed for use in **neutrino detectors**, but can be deployed for other types of physics interactions!

NuGraph2

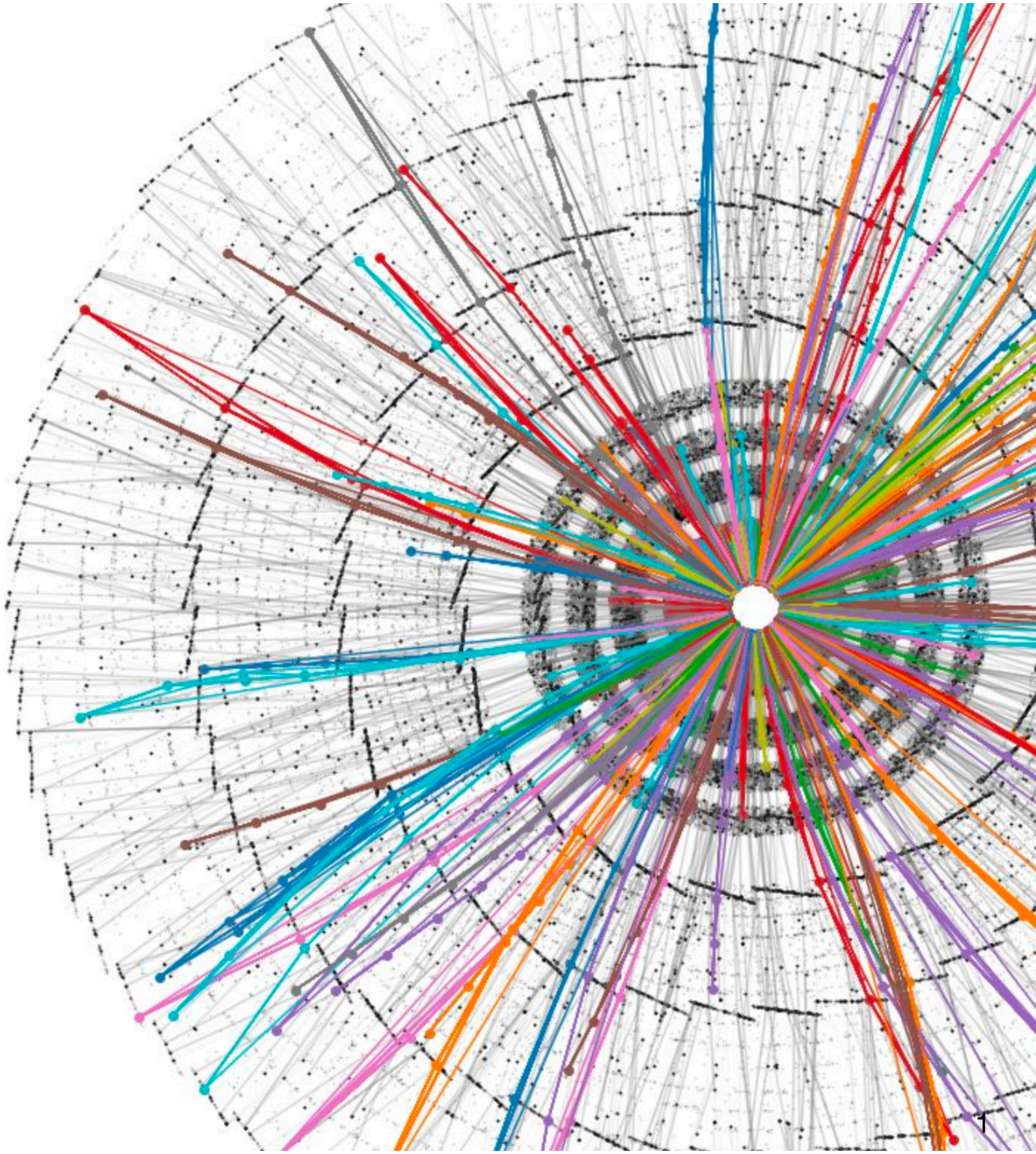
Exa.TrkX

- Exa.TrkX is a collaboration developing next-generation Graph Neural Network (GNN) reconstruction for HEP:



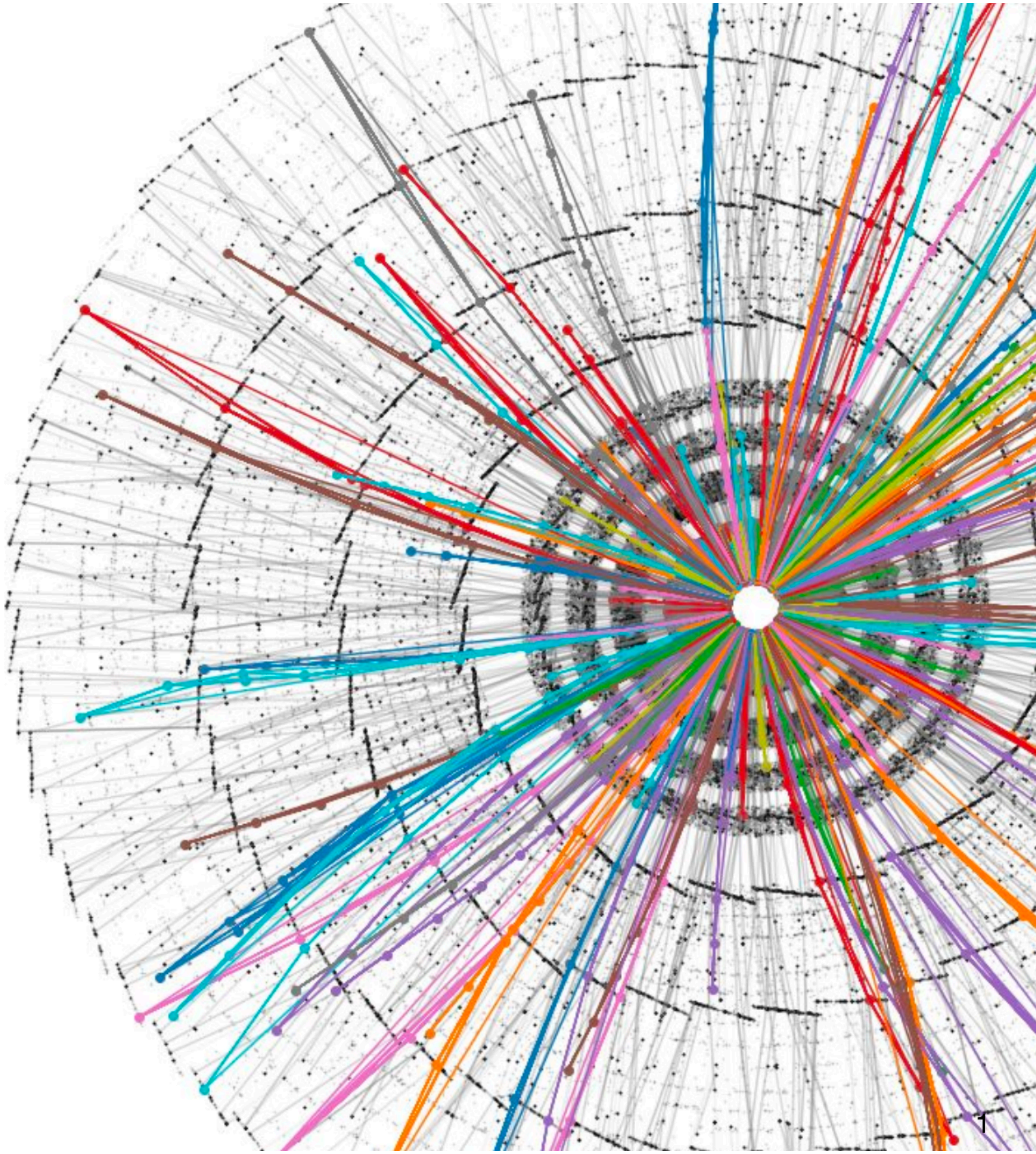
Exa.TrkX

- Exa.TrkX is a collaboration developing next-generation Graph Neural Network (GNN) reconstruction for HEP:
  - **Energy Frontier**
    - Expand on HEP.TrkX's prototype GNN for HL-LHC.
    - Incorporate into ATLAS's simulation and validation chain.



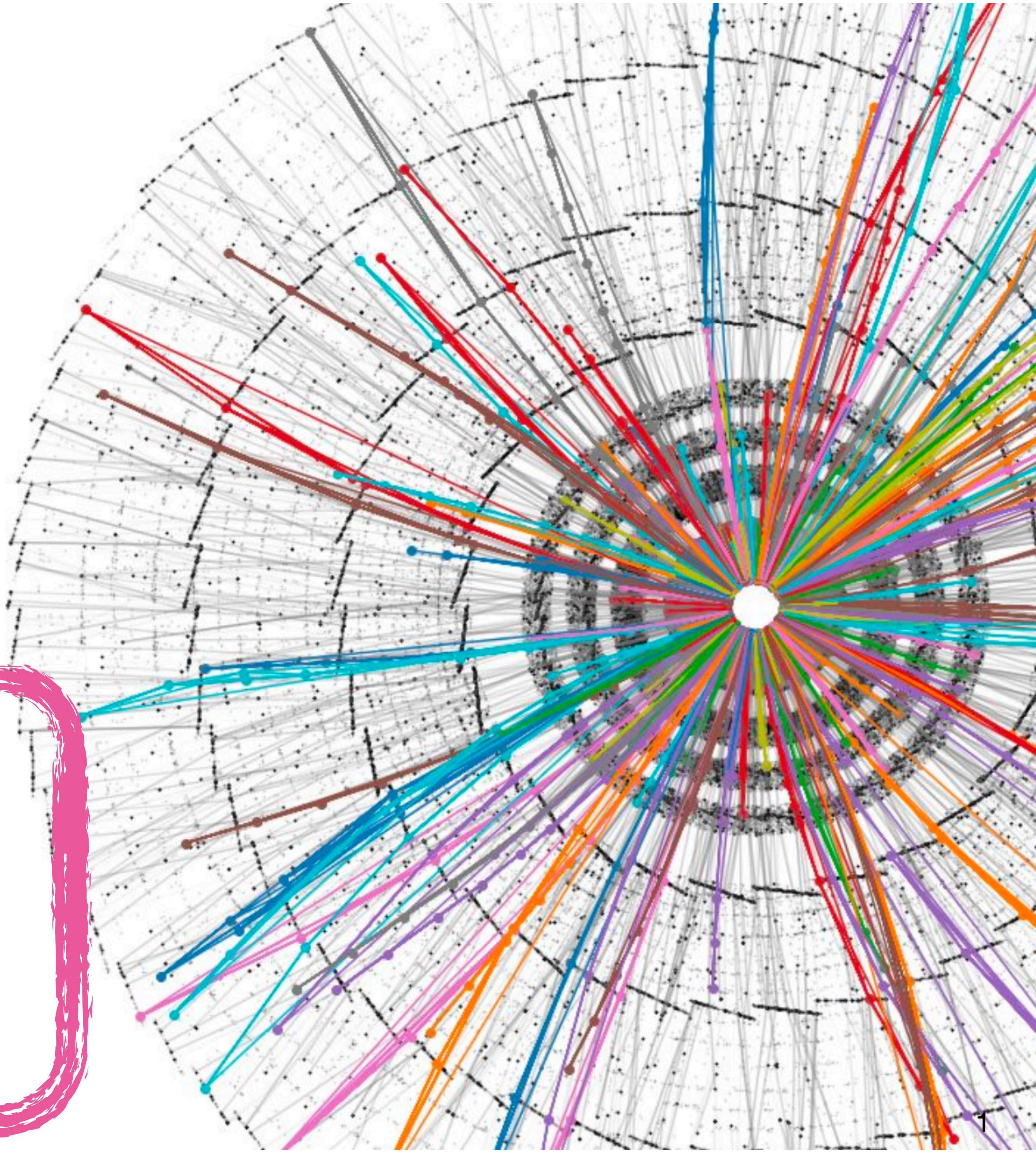
Exa.TrkX

- Exa.TrkX is a collaboration developing next-generation Graph Neural Network (GNN) reconstruction for HEP:
  - **Energy Frontier**
    - Expand on HEP.TrkX's prototype GNN for HL-LHC.
    - Incorporate into ATLAS's simulation and validation chain.
  - **Intensity Frontier**
    - Explore viability of HEP.TrkX network for neutrino physics.
    - Develop GNN-based reconstruction for Liquid Argon TPCs.



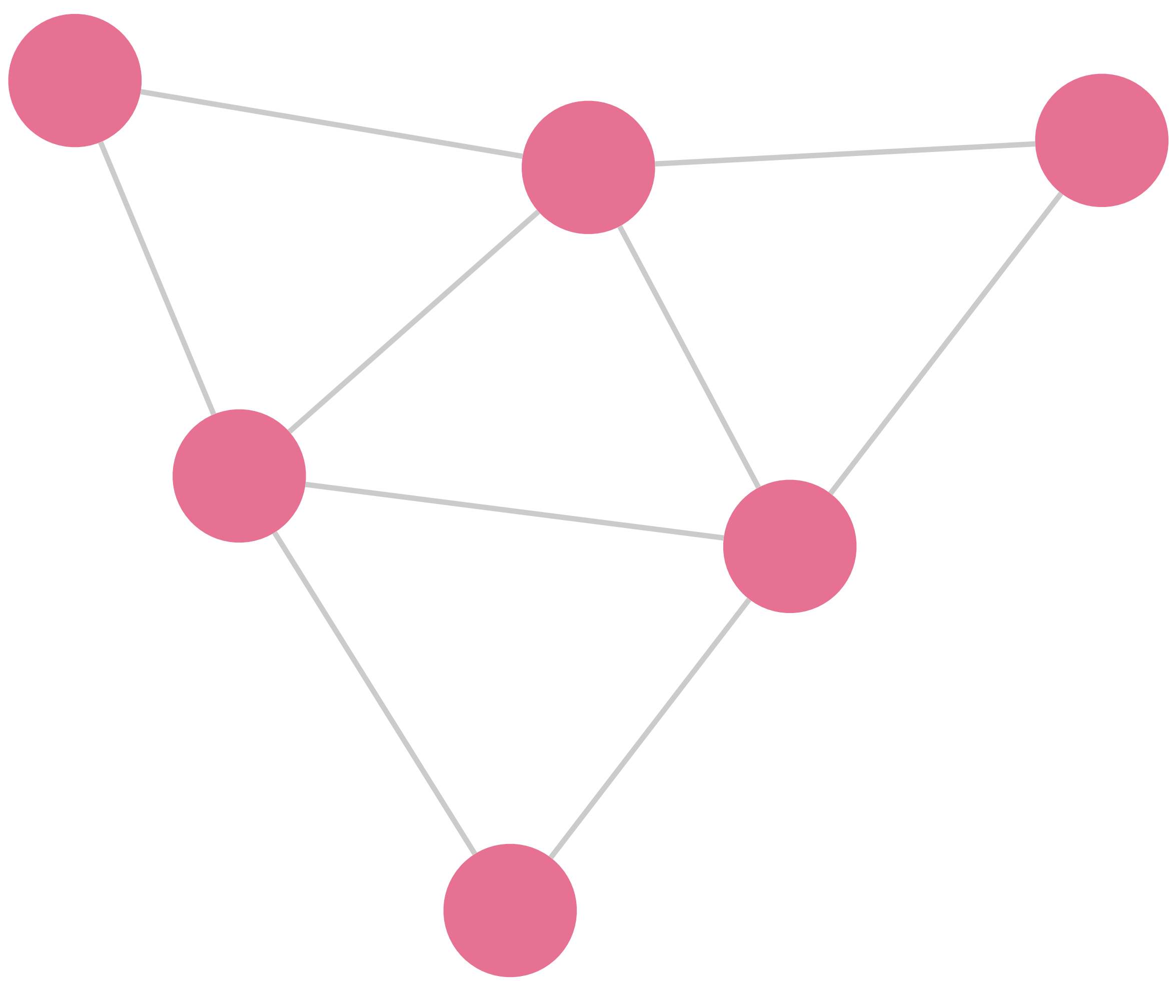
Exa.TrkX

- Exa.TrkX is a collaboration developing next-generation Graph Neural Network (GNN) reconstruction for HEP:
  - **Energy Frontier**
    - Expand on HEP.TrkX's prototype GNN for HL-LHC.
    - Incorporate into ATLAS's simulation and validation chain.
  - **Intensity Frontier**
    - Explore viability of HEP.TrkX network for neutrino physics.
    - Develop GNN-based reconstruction for Liquid Argon TPCs.

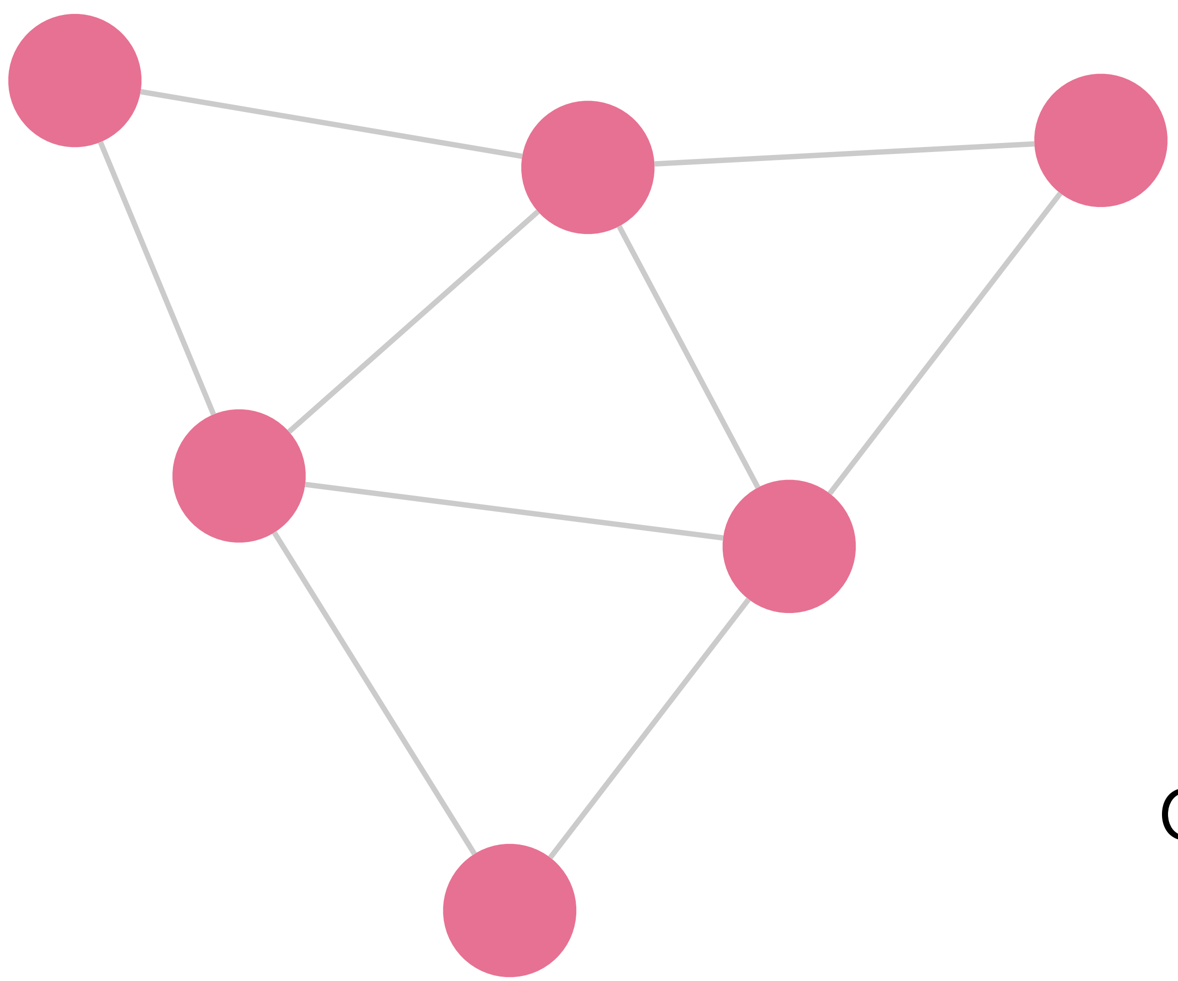


Graph Neural Networks

Construct **graph** where each **node** is a **detector hit**

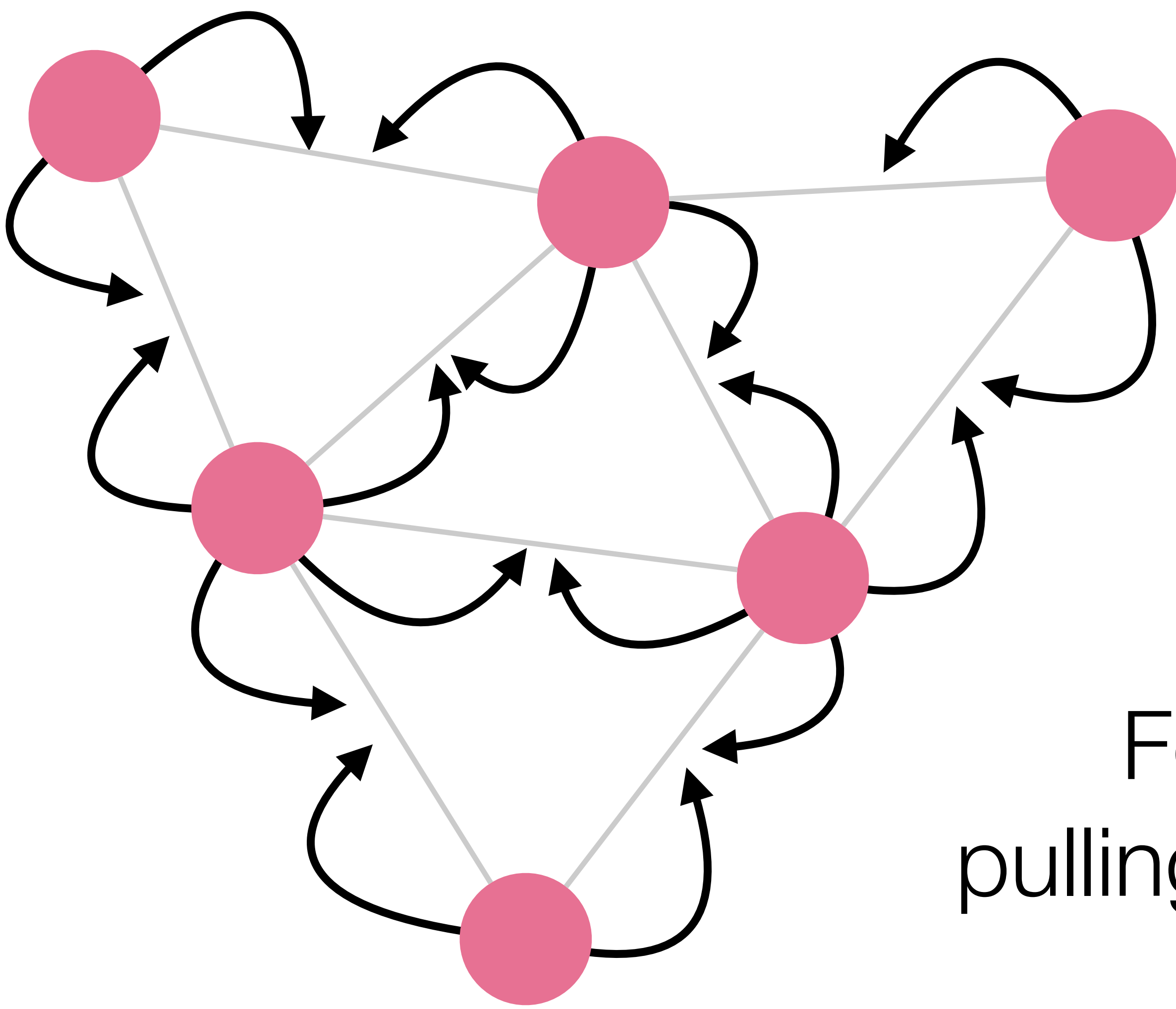


Graph Neural Networks



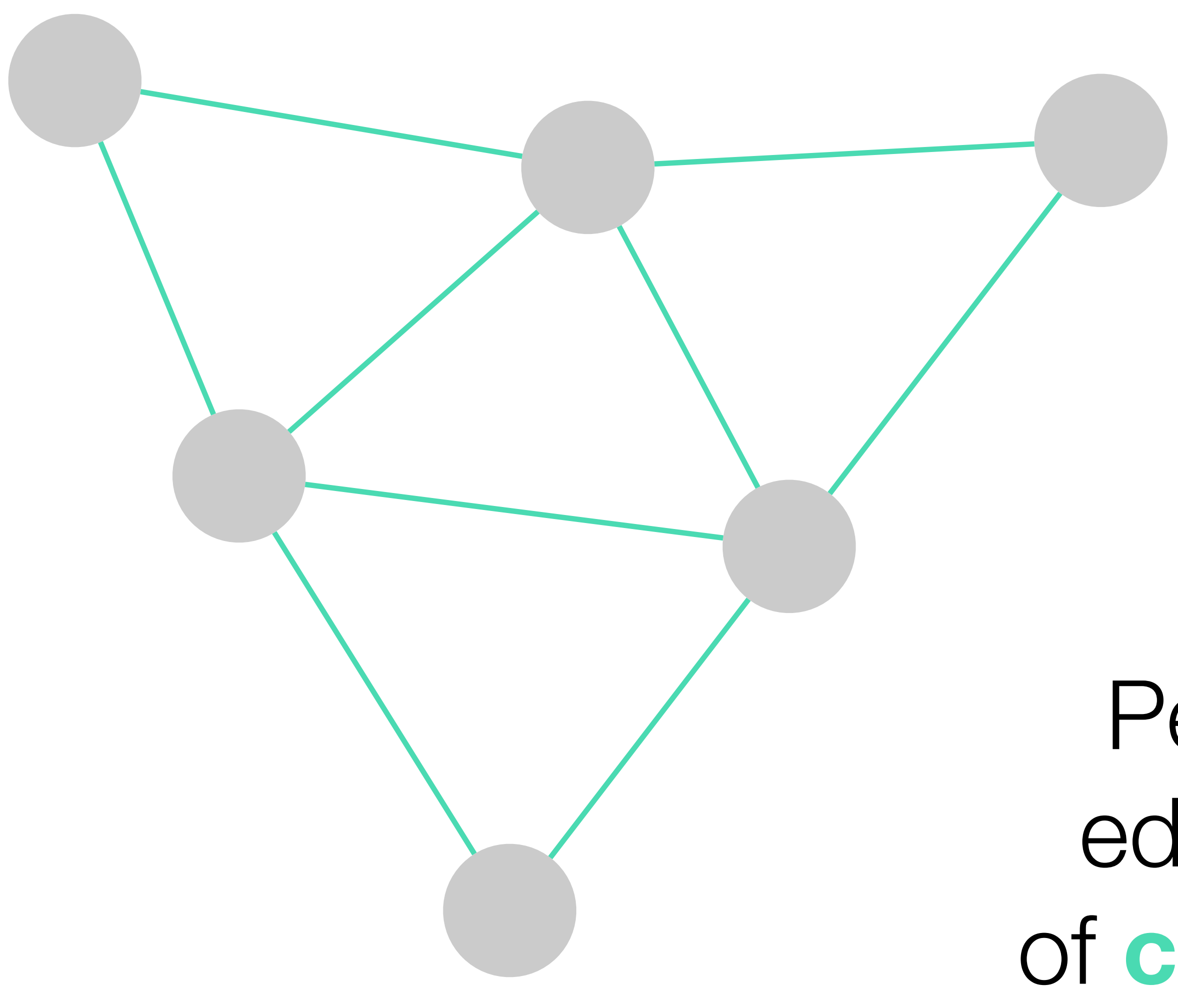
Node features:
wire index, time tick,
charge integral & RMS

Graph Neural Networks



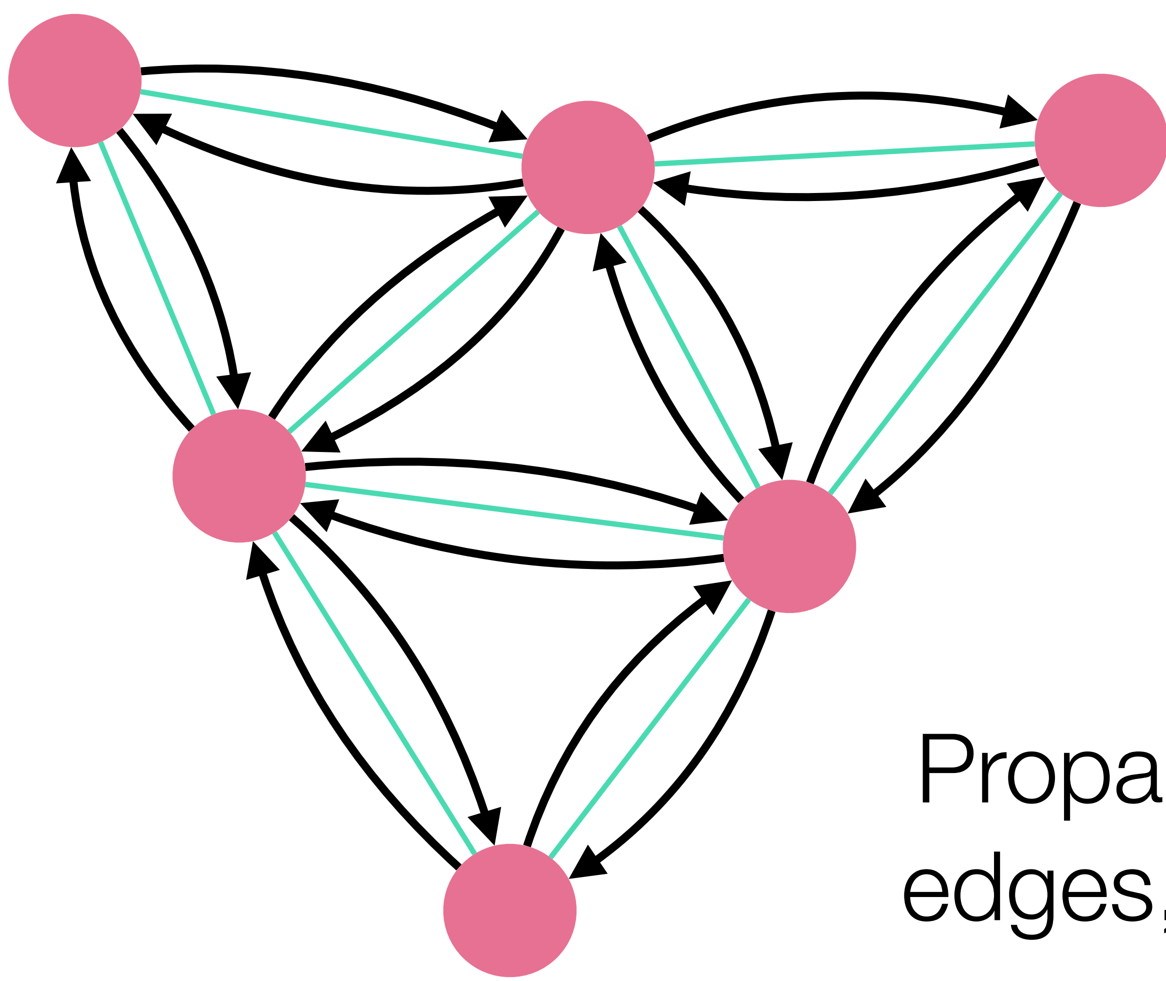
Form **edge features** by pulling in features from incoming and outgoing nodes

Graph Neural Networks



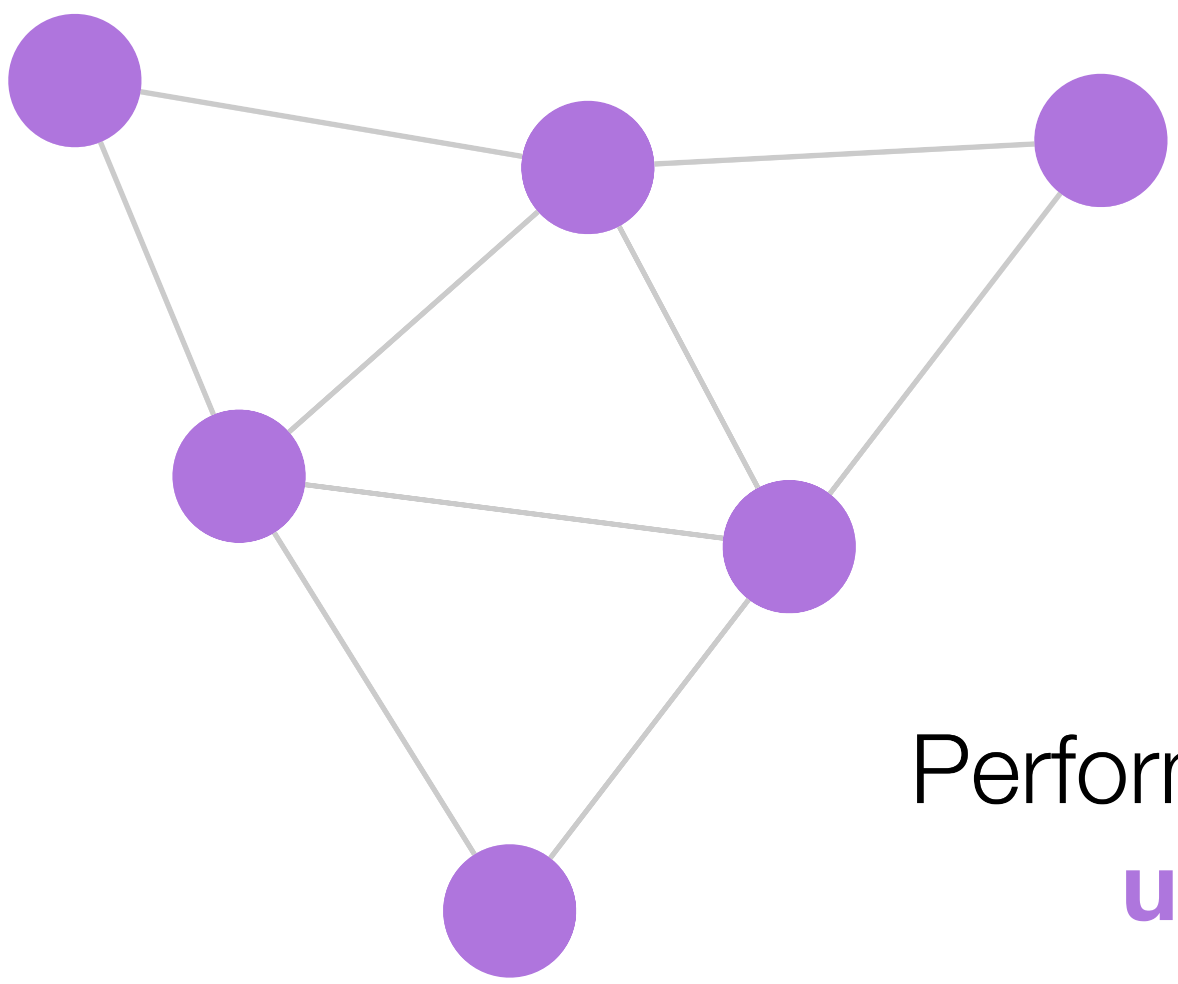
Perform convolutions on edge scores to form a set of **class-wise probabilities**

Graph Neural Networks



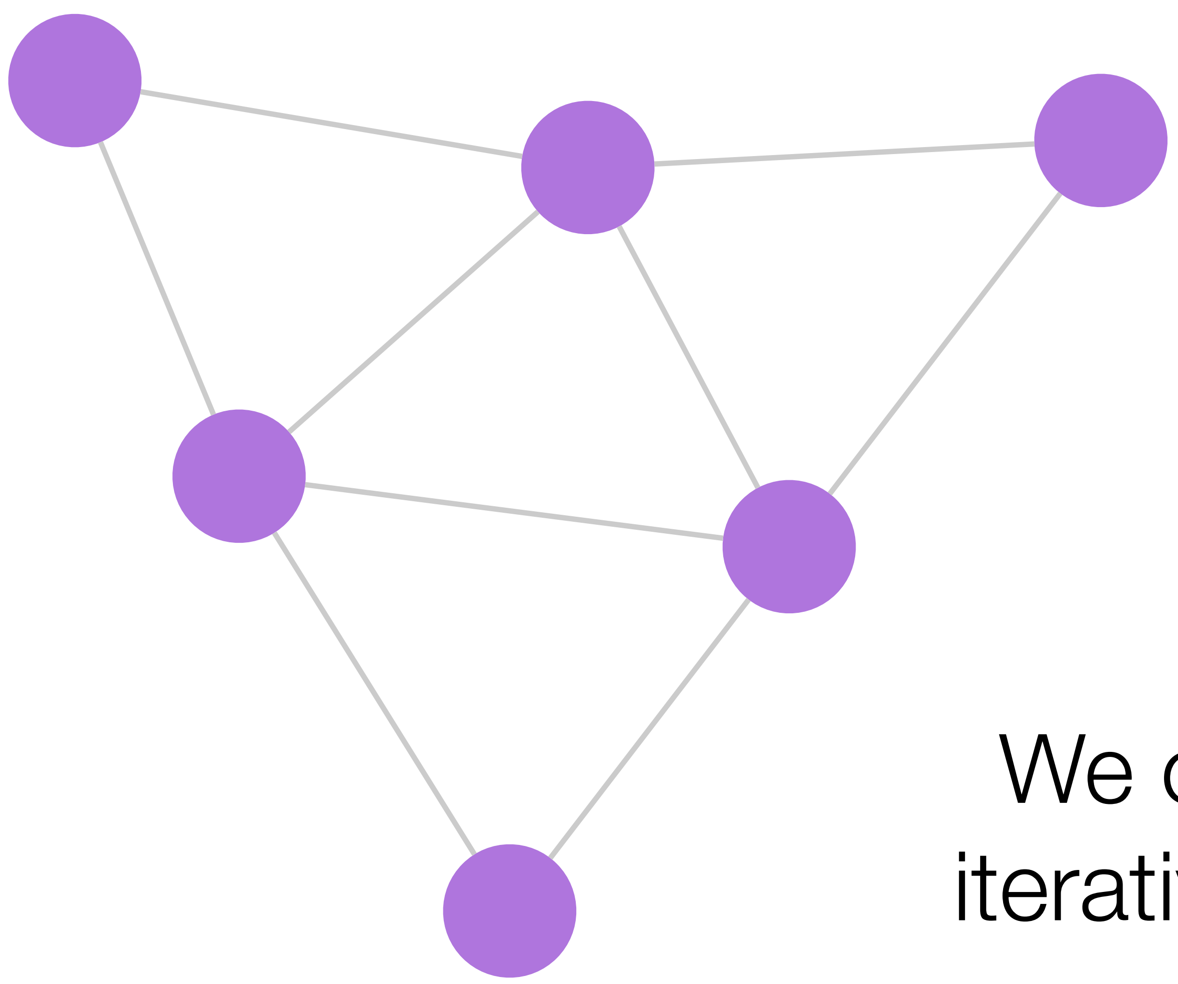
Propagate **node features** across edges, weighting by **edge scores**

Graph Neural Networks



Perform convolutions on nodes to **update node features**

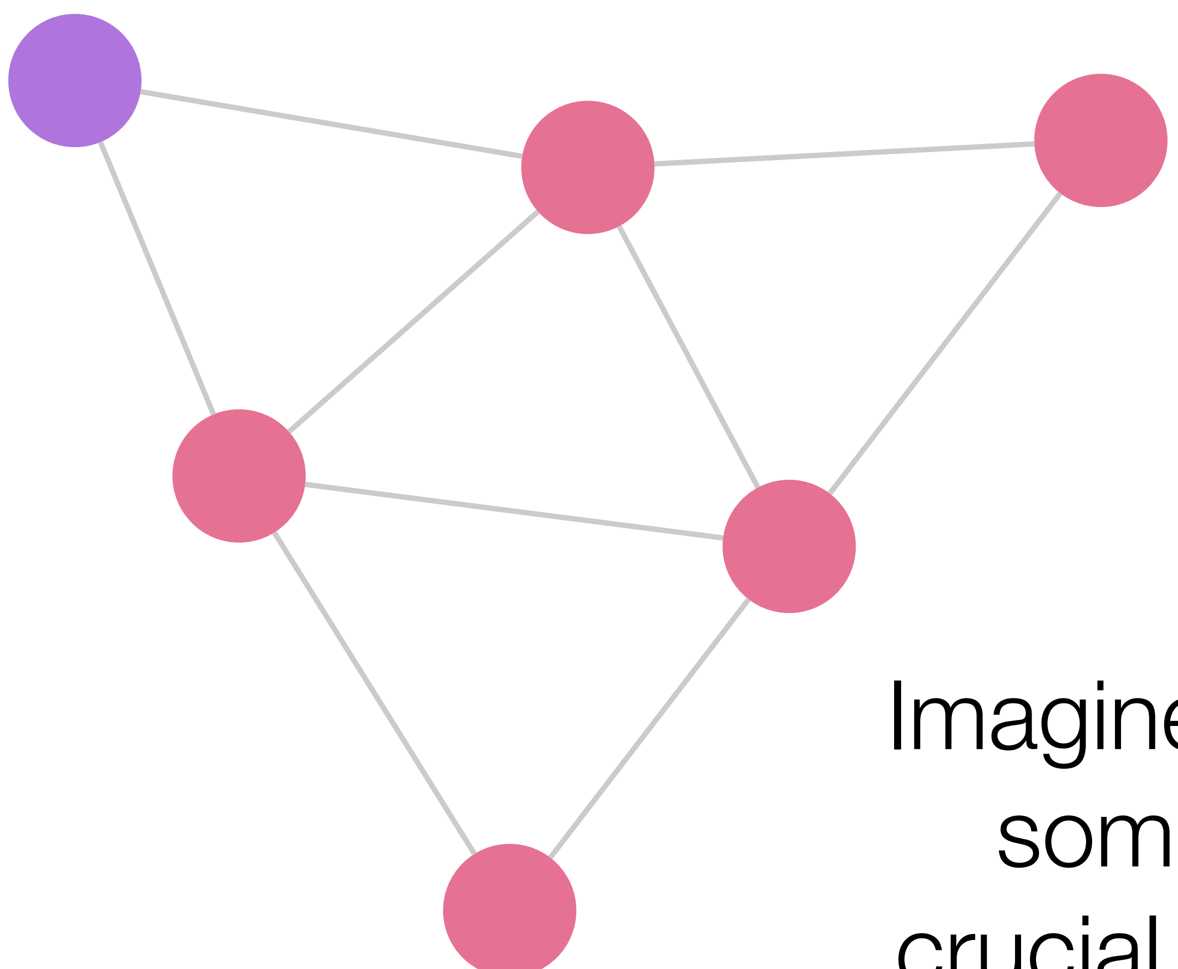
Graph Neural Networks



We can repeat this procedure iteratively to **"pass messages"**

Graph Neural Networks

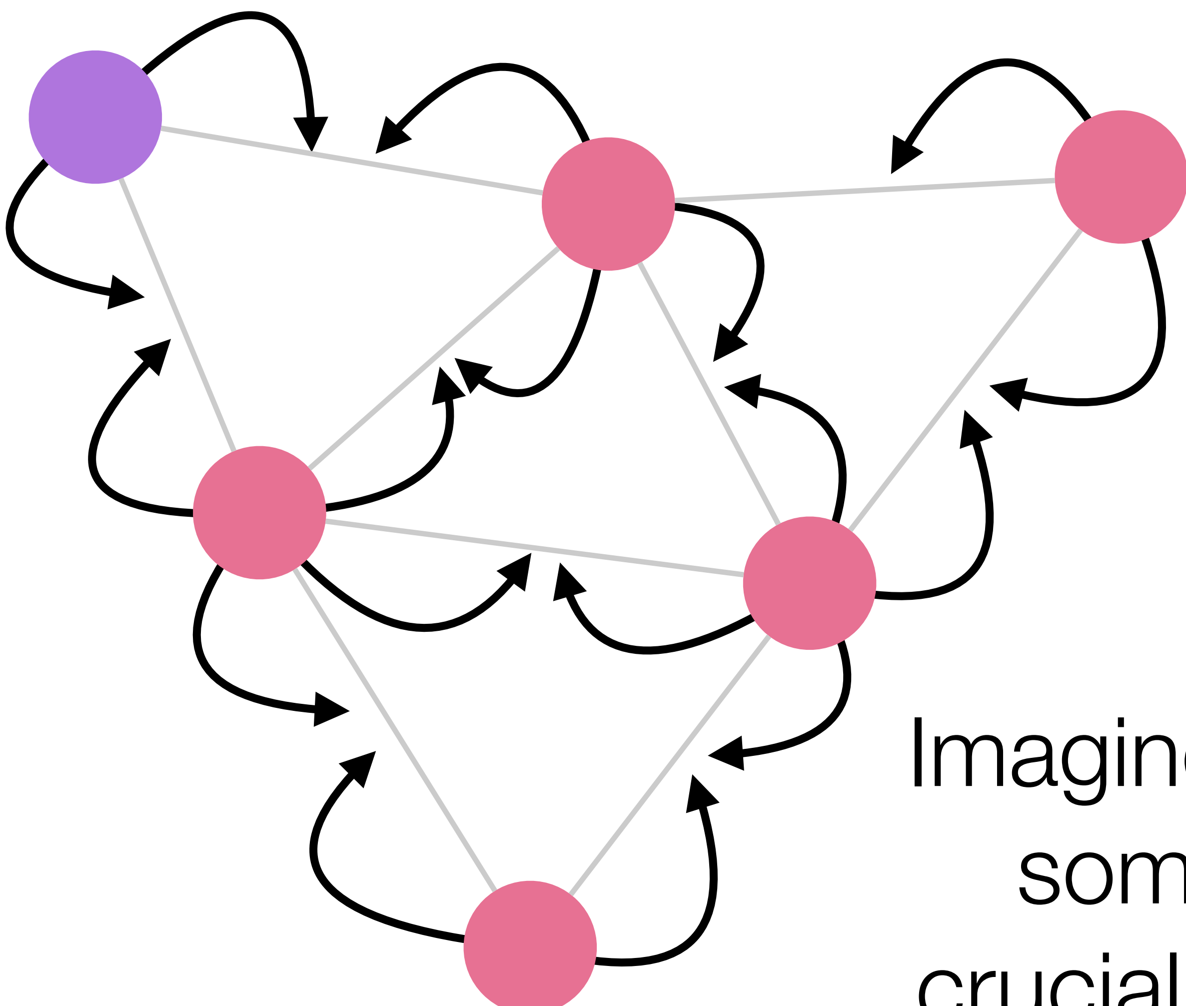
0 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

0 iterations

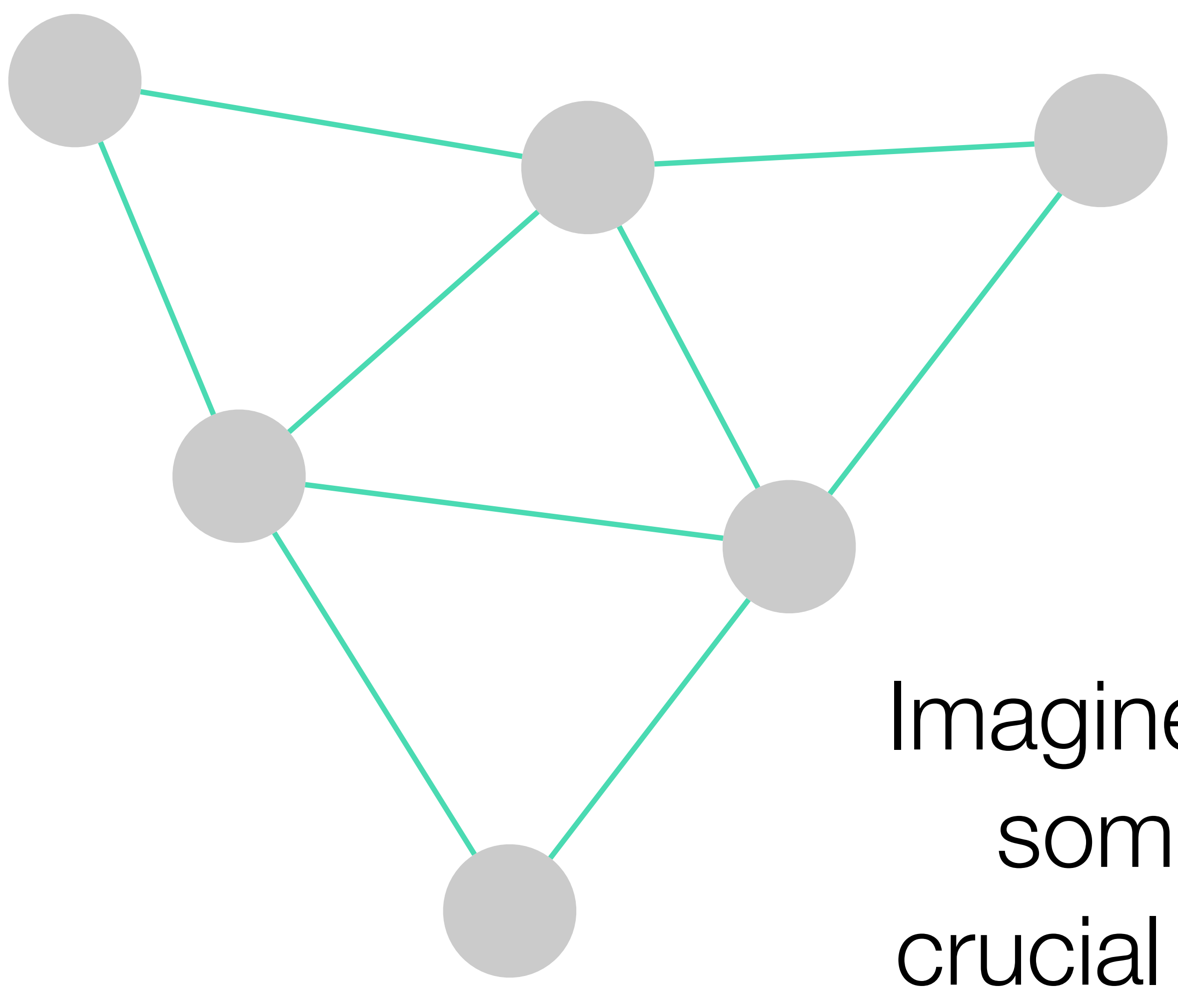


Imagine that one graph node holds some **key information** that's crucial for understanding the event

University of CINCINNATI

Graph Neural Networks

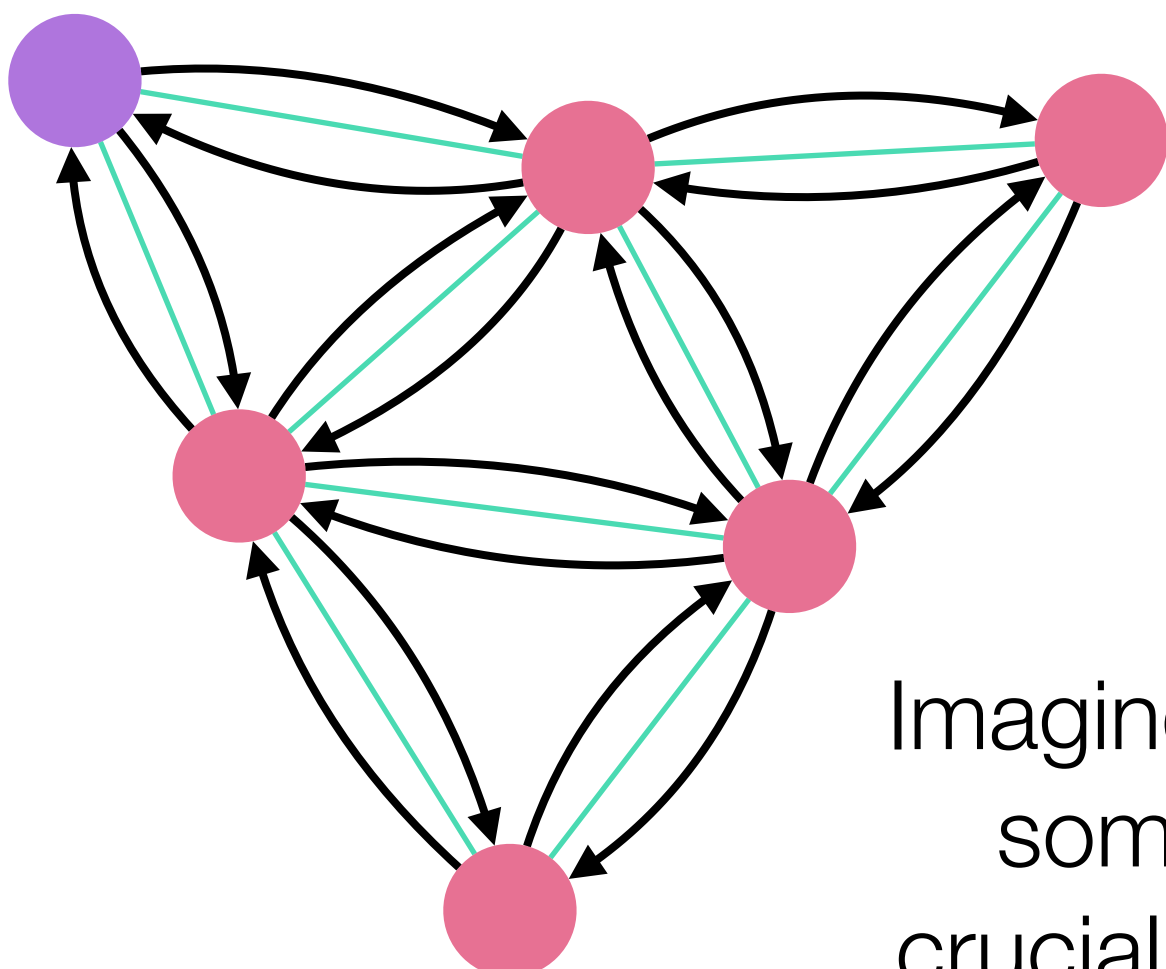
0 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

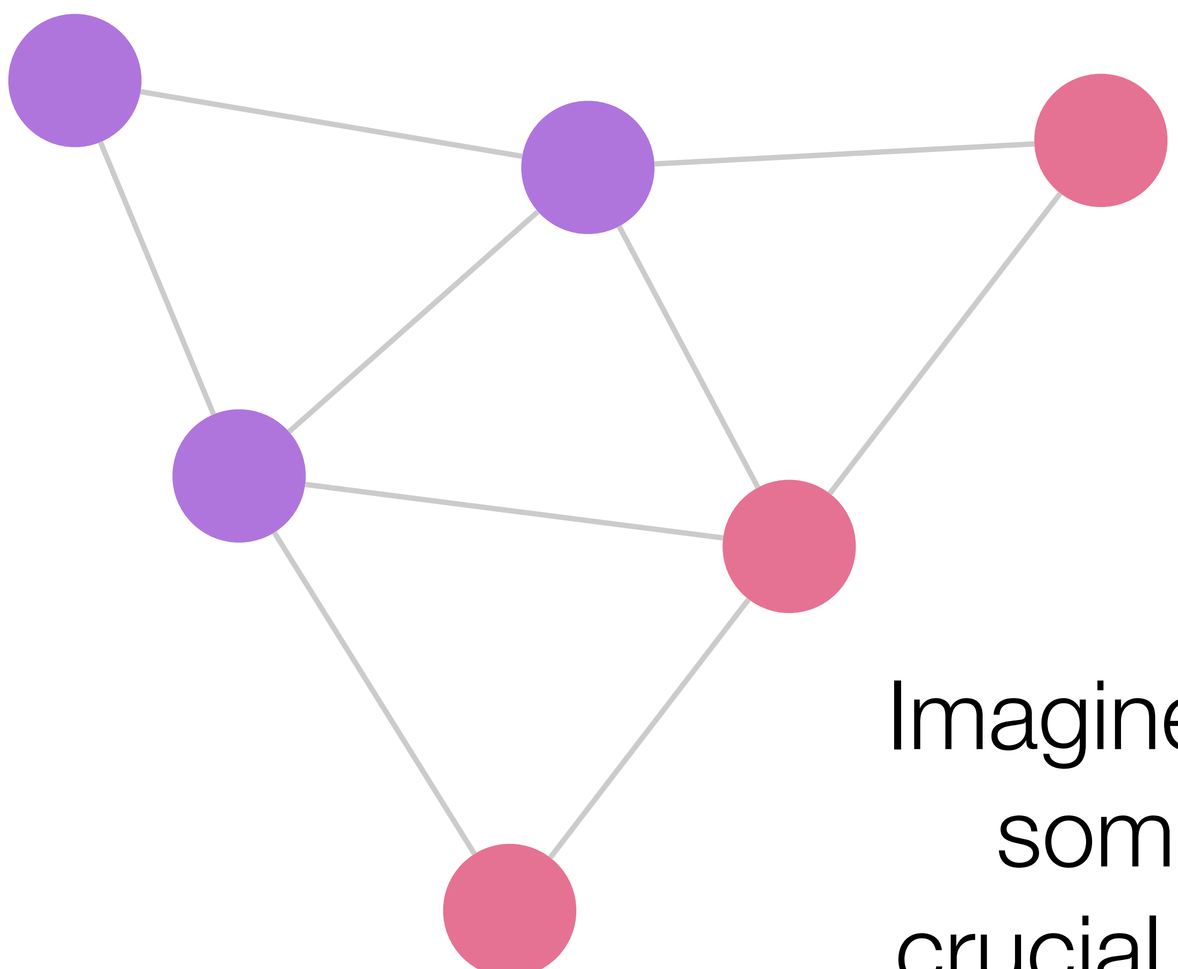
0 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

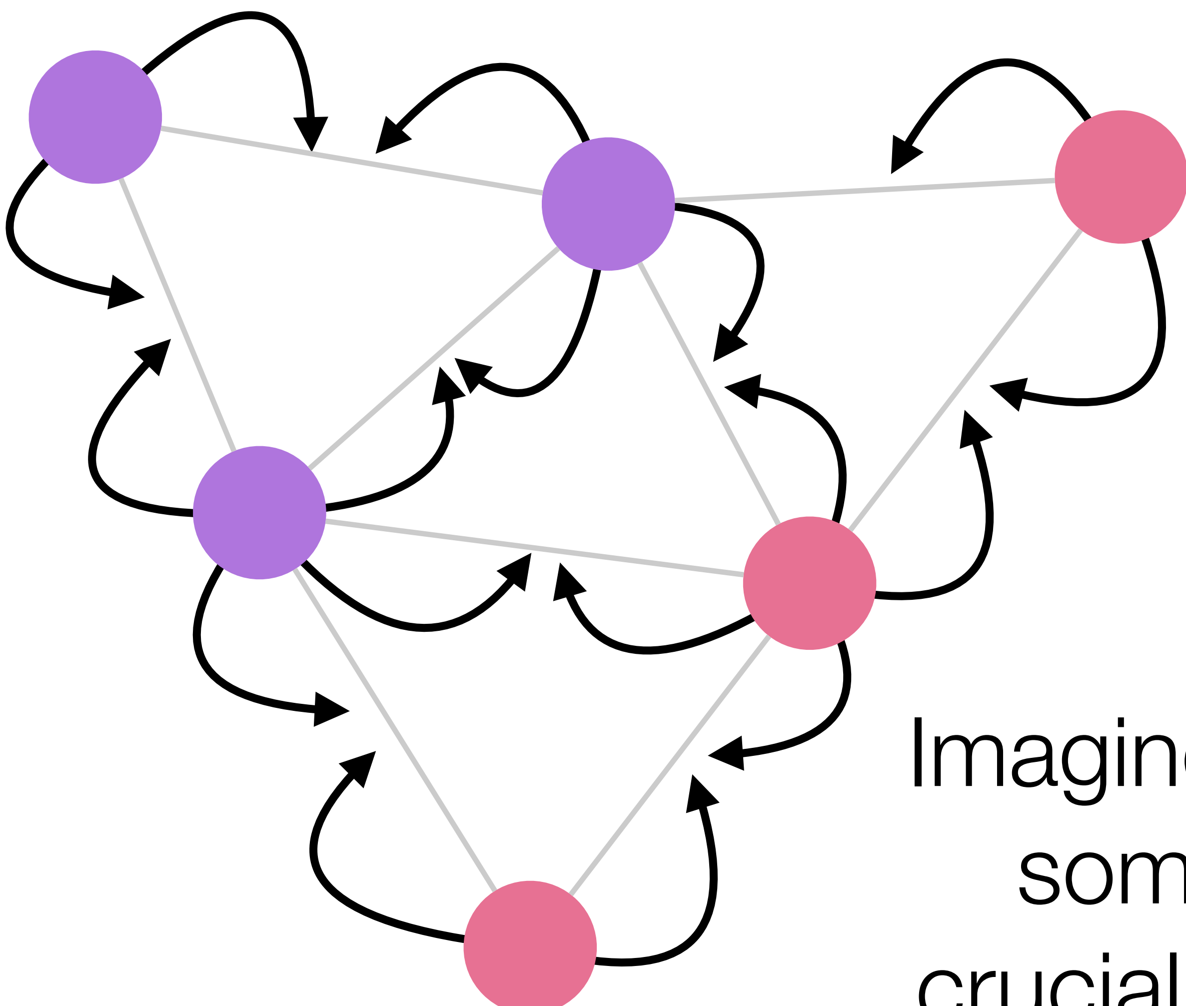
1 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

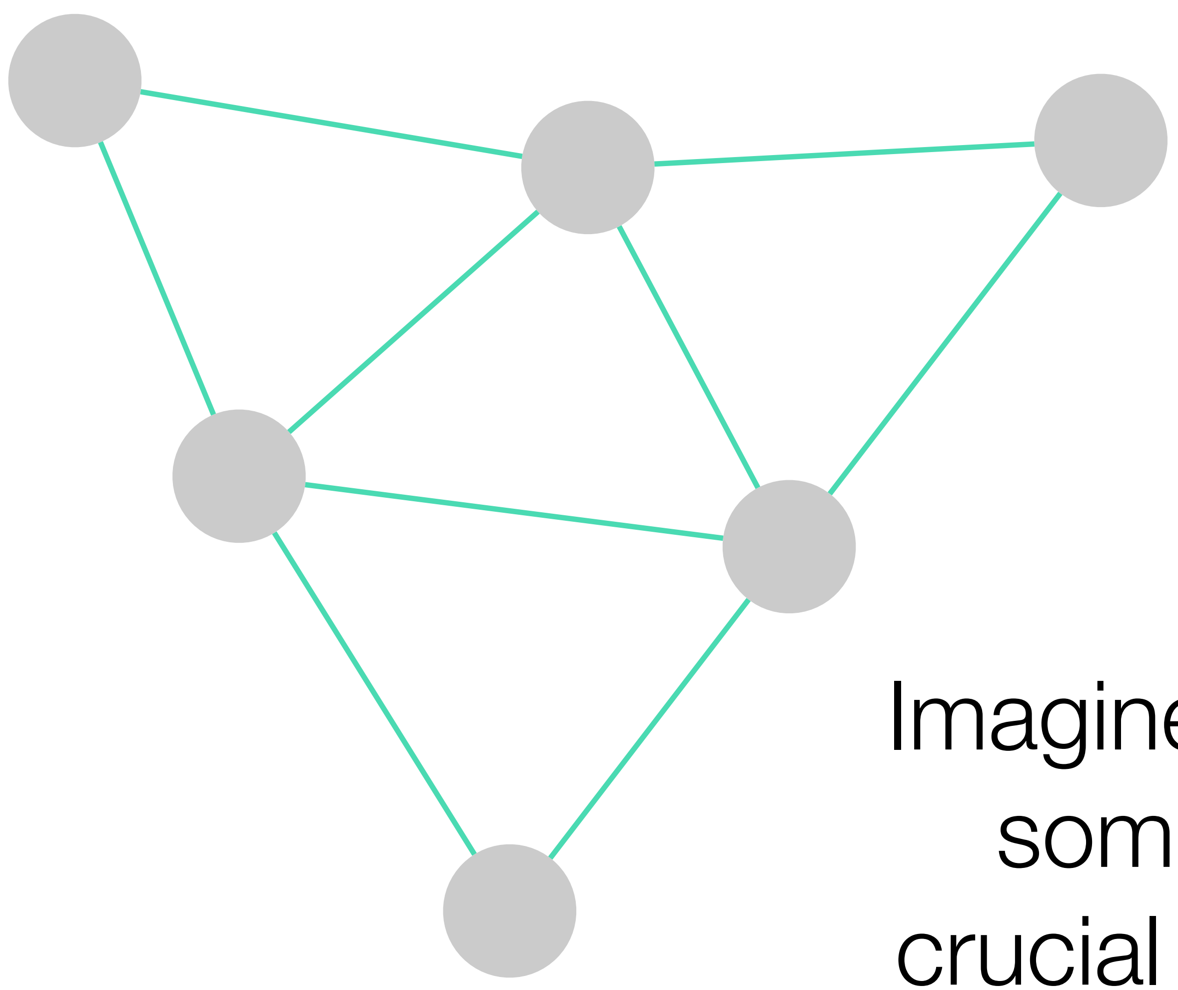
1 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

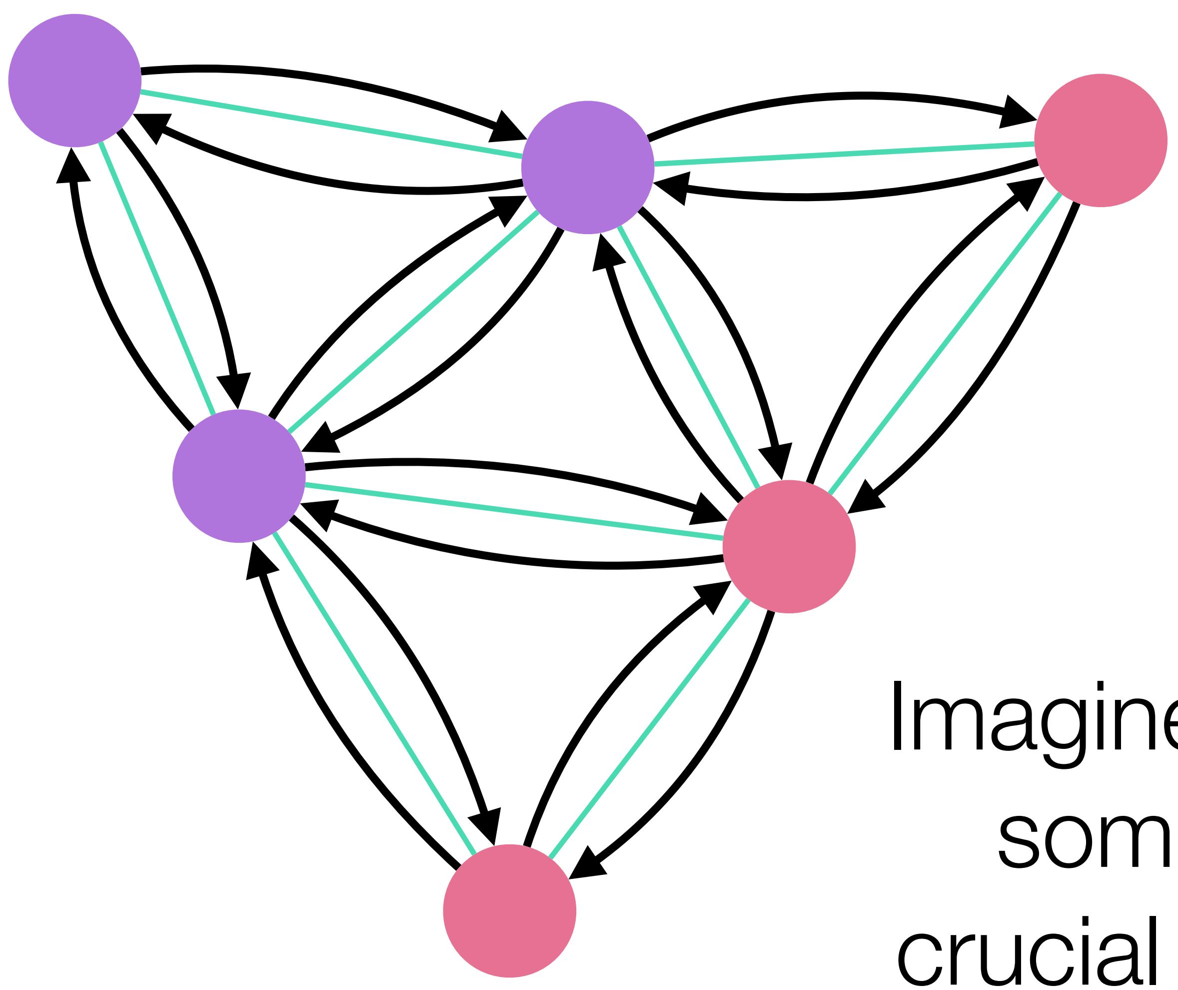
1 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

Graph Neural Networks

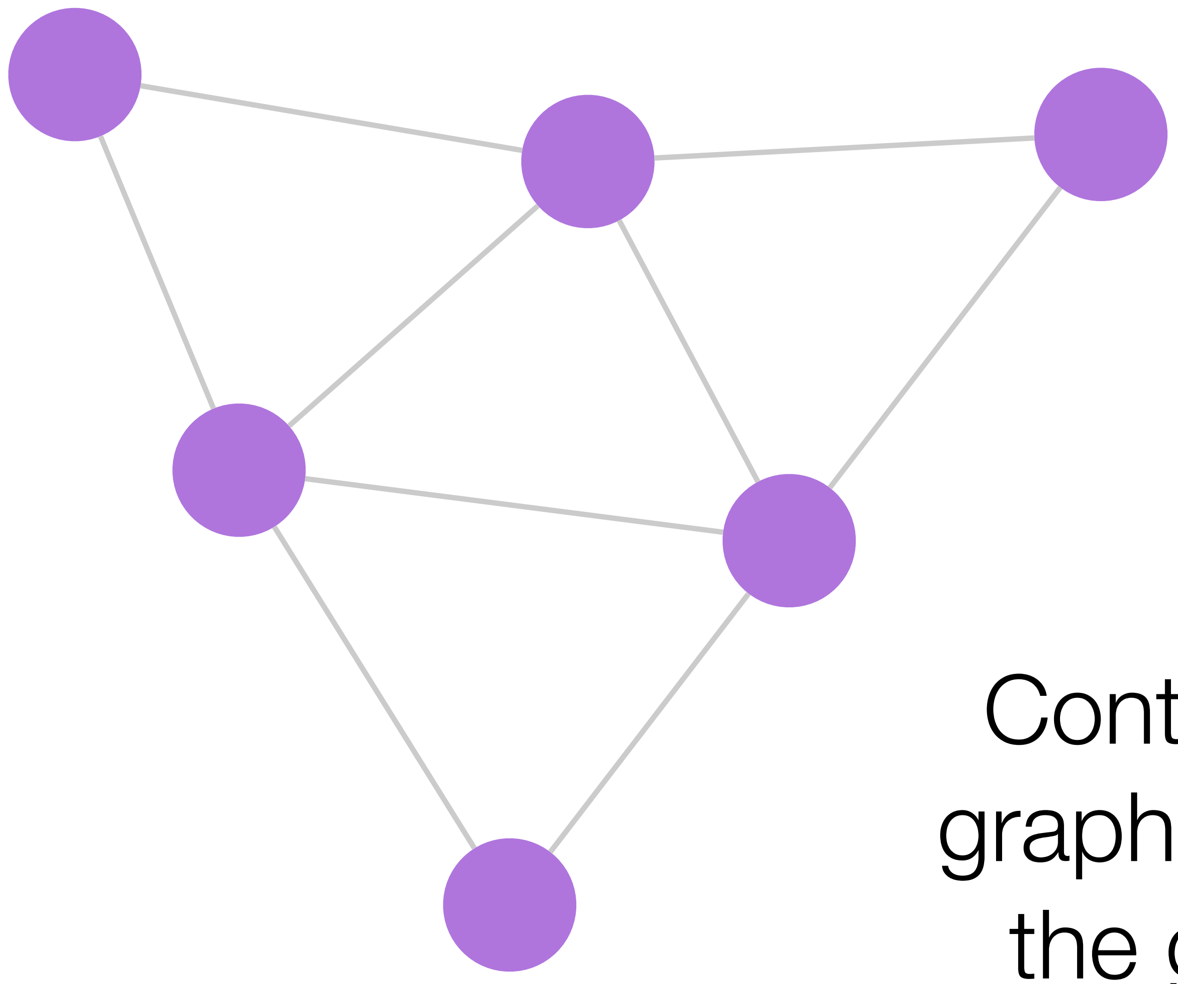
1 iterations



Imagine that one graph node holds some **key information** that's crucial for understanding the event

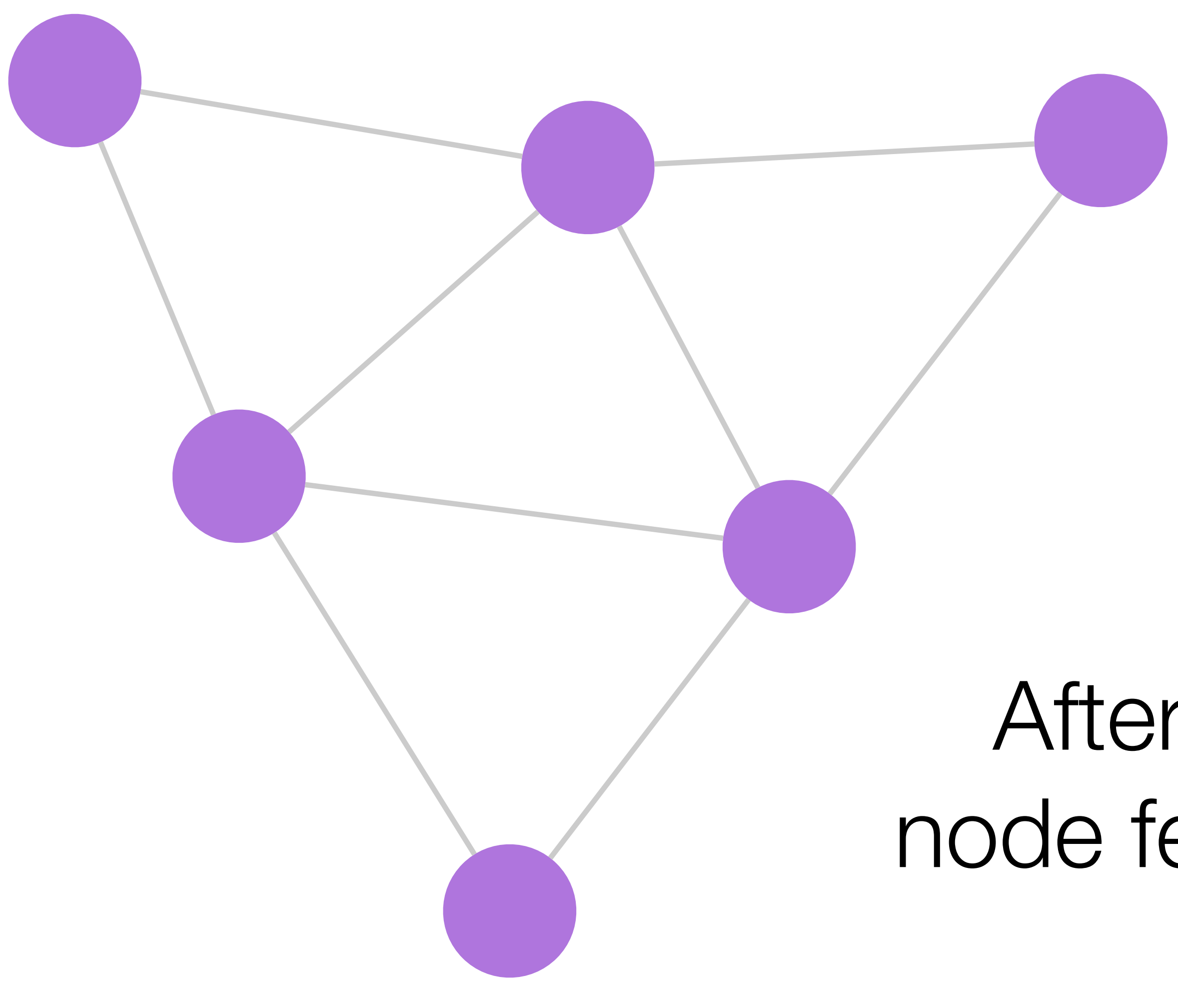
Graph Neural Networks

2 iterations



Context information from each graph node is dispersed through the graph with each iteration.

Graph Neural Networks



After multiple iterations, graph node features can be convolved to **label graph nodes**

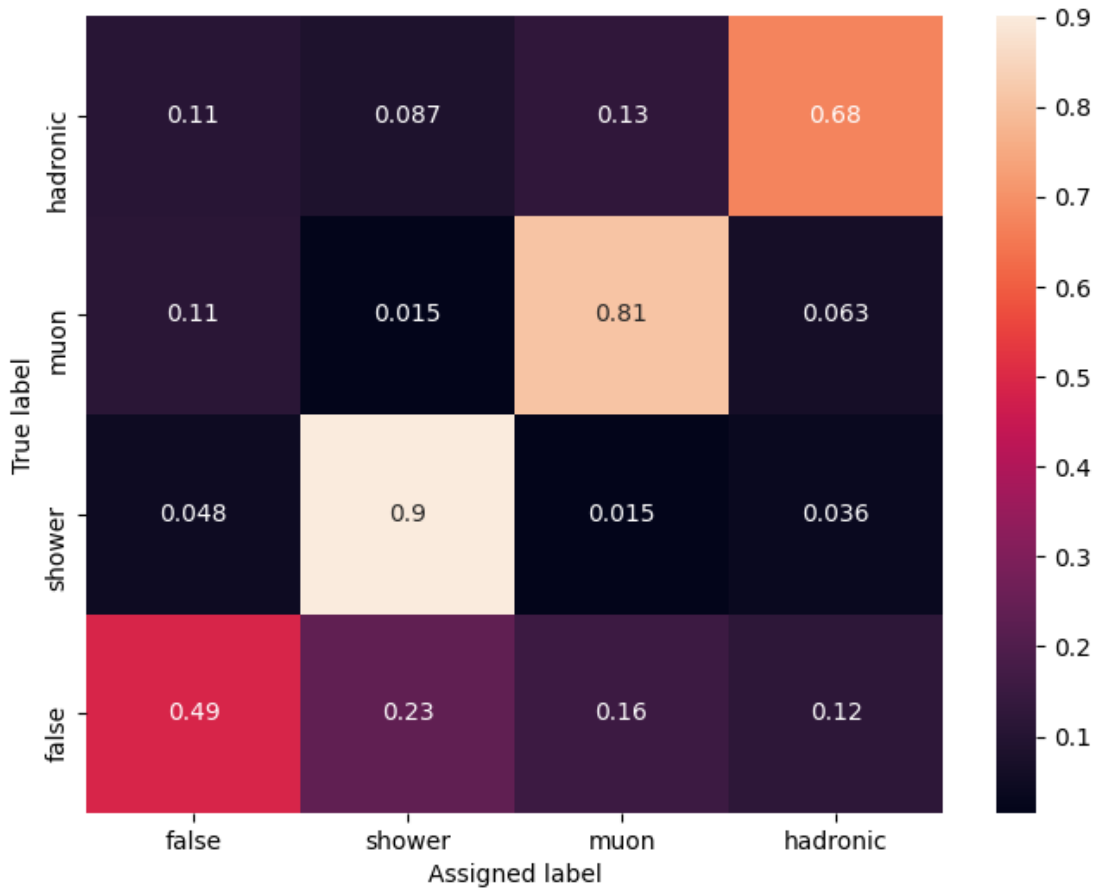
NuGraph2

NuGraph1

- First proof-of-concept model achieved 84% accuracy in **classifying graph edges**.
  - Reasonable performance on showers, struggled to correctly identify type of track.
  - See arxiv:2103.06233.

NuGraph1

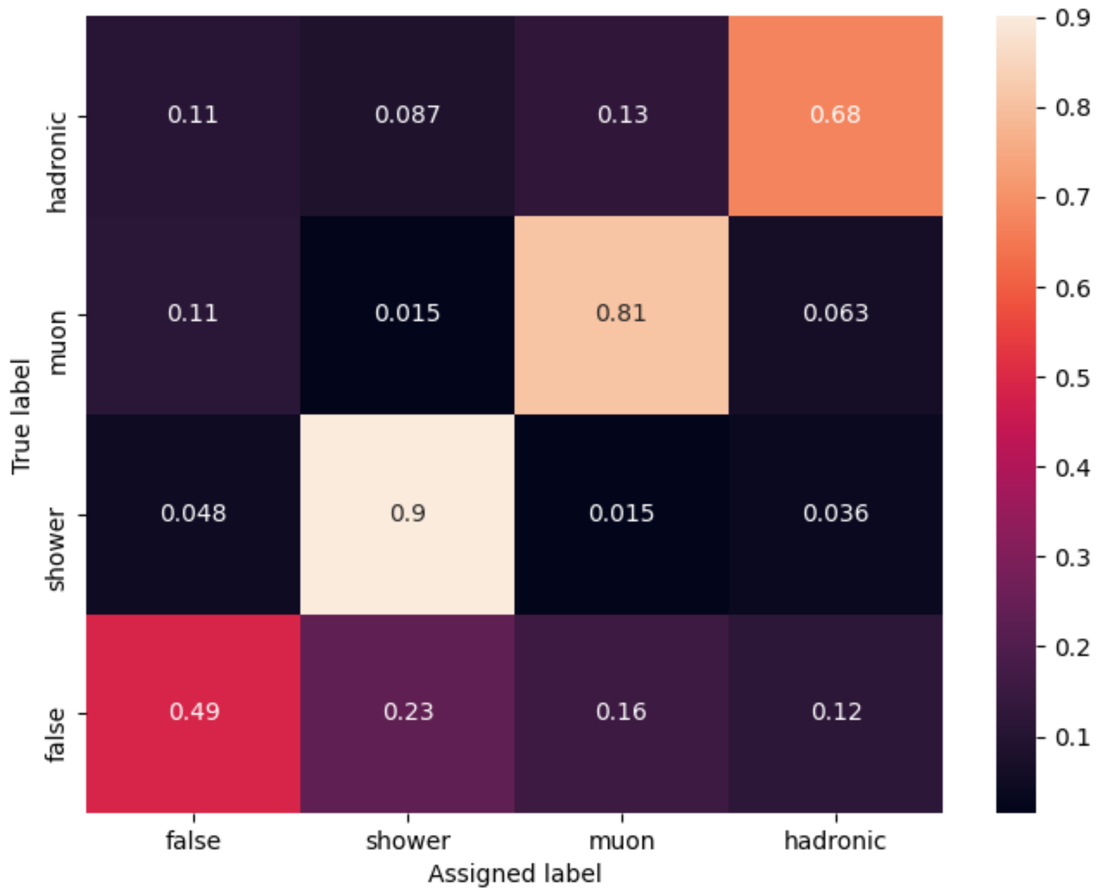
- First proof-of-concept model achieved 84% accuracy in **classifying graph edges**.
  - Reasonable performance on showers, struggled to correctly identify type of track.
  - See arxiv:2103.06233.

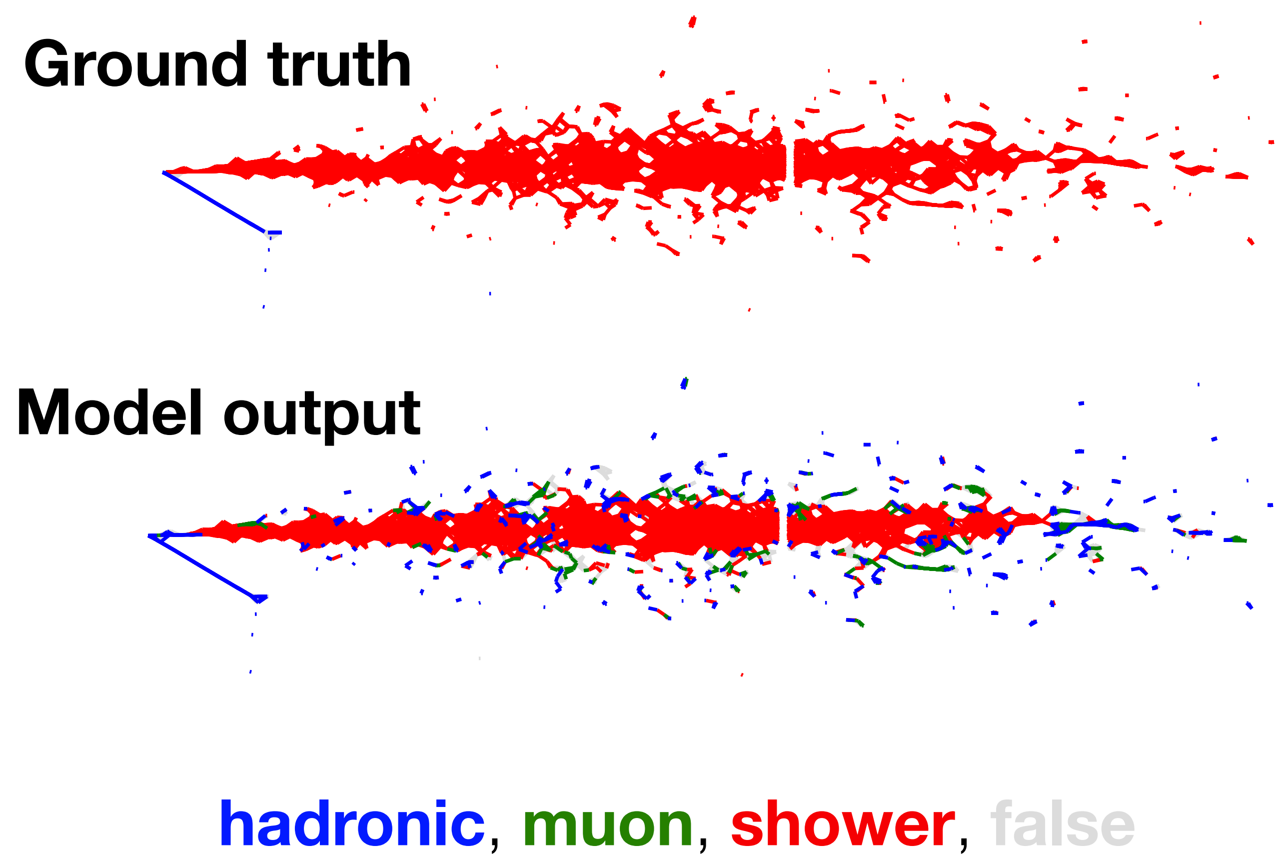


University of CINCINNATI

NuGraph1

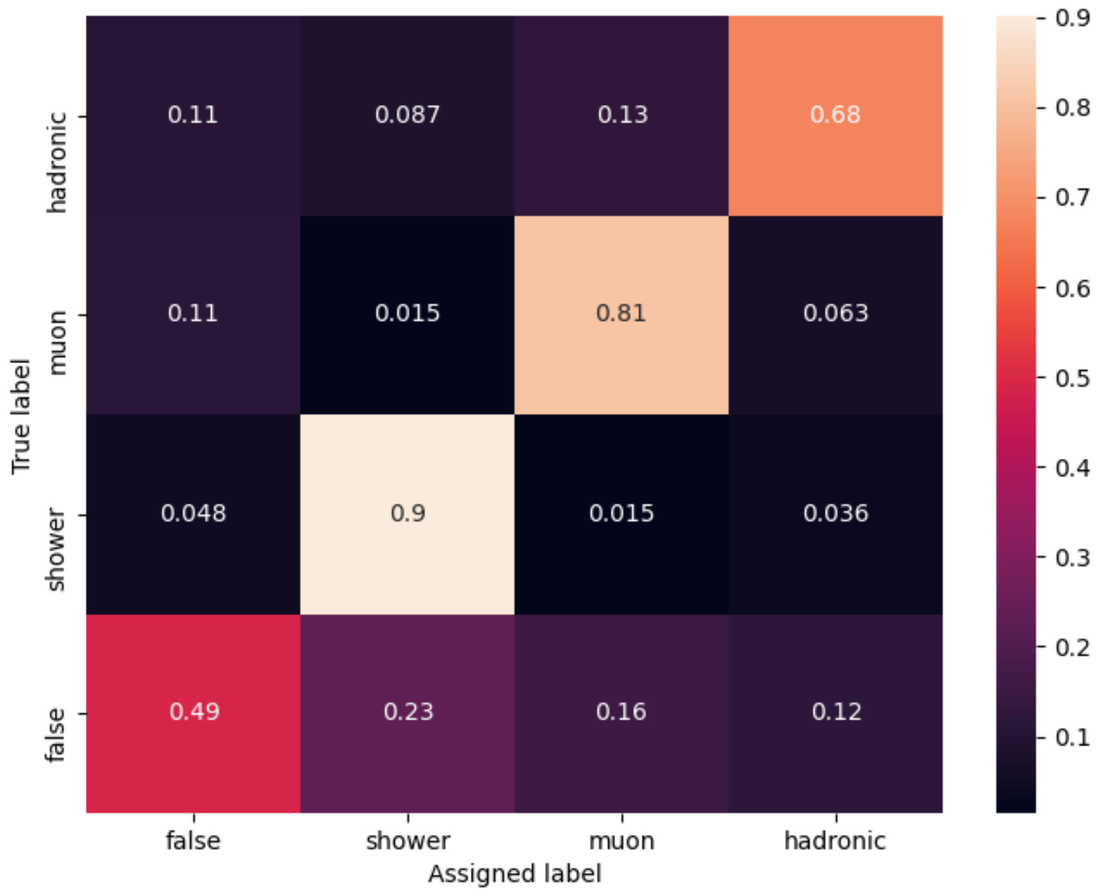
- First proof-of-concept model achieved 84% accuracy in **classifying graph edges**.
  - Reasonable performance on showers, struggled to correctly identify type of track.
  - See arxiv:2103.06233.

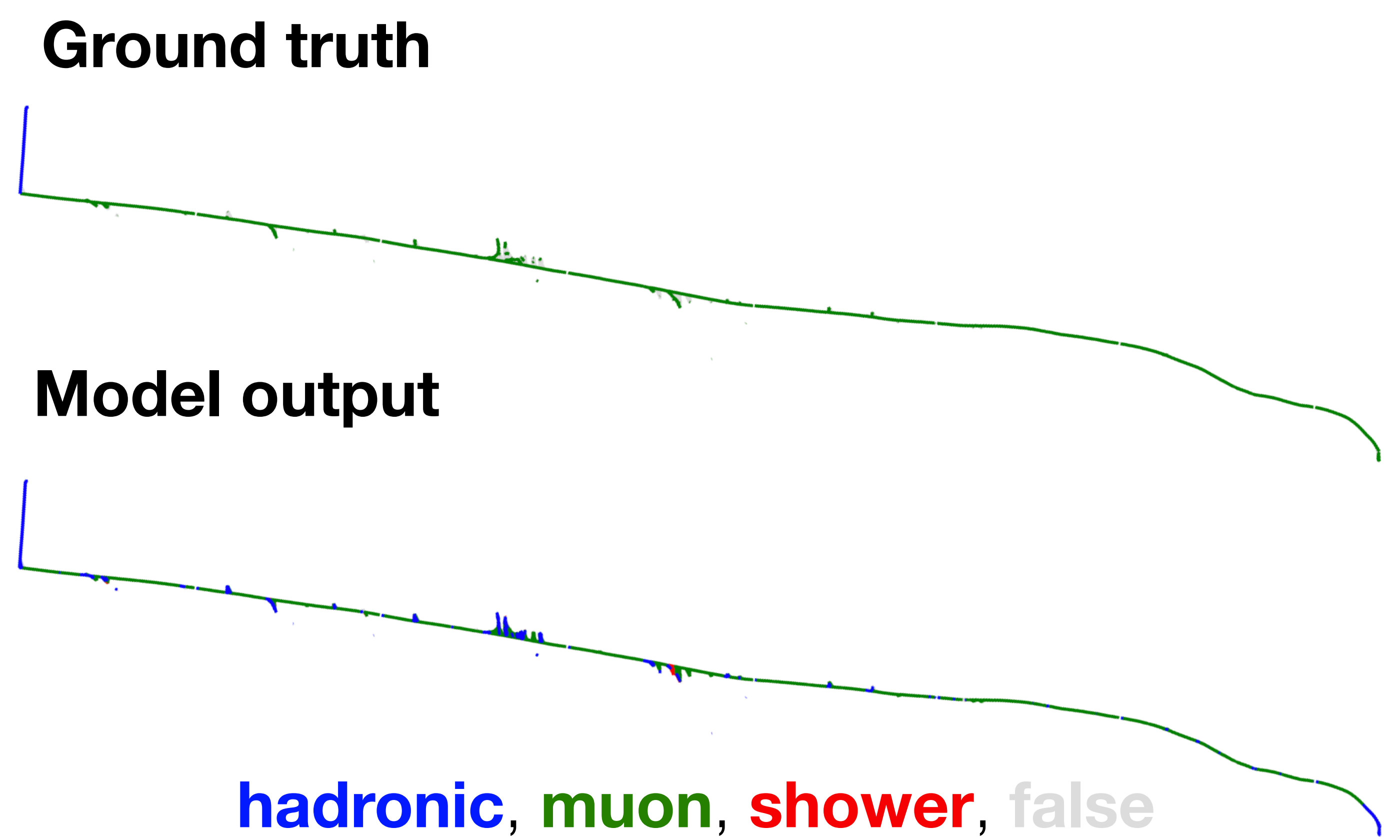




NuGraph1

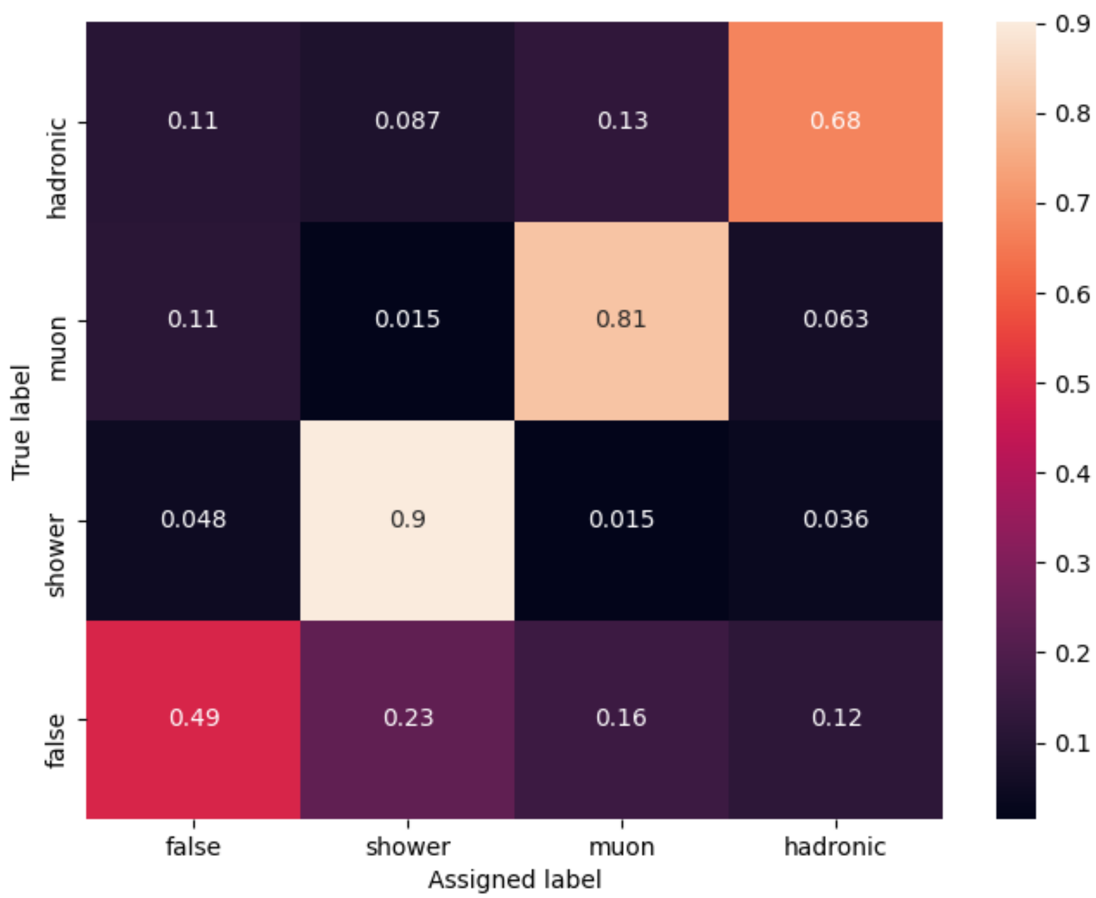
- First proof-of-concept model achieved 84% accuracy in **classifying graph edges**.
  - Reasonable performance on showers, struggled to correctly identify type of track.
  - See arxiv:2103.06233.

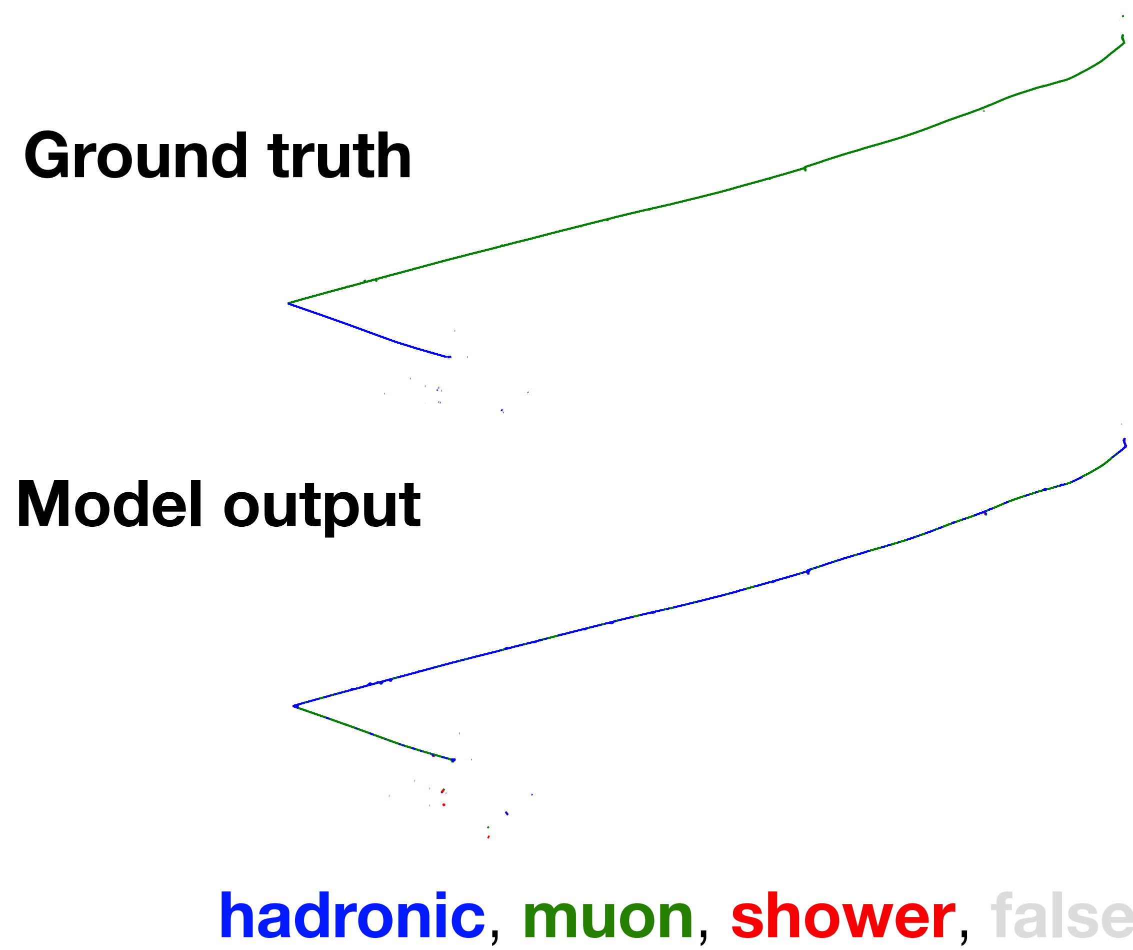




NuGraph1

- First proof-of-concept model achieved 84% accuracy in **classifying graph edges**.
    - Reasonable performance on showers, struggled to correctly identify type of track.
    - See arxiv:2103.06233.





University of CINCINNATI

NuGraph2

- **Second-generation model** incorporates a wide range of improvements over first proof-of-concept.

NuGraph2

- **Second-generation model** incorporates a wide range of improvements over first proof-of-concept.
- Move from **edge classification** to **node (ie. hit) classification**.
  - Graph edge classification for track forming is a natural choice for LHC detectors, where sequential layers provide a natural constraint on edges.
  - Dense LArTPC environment provides no such constraints, and number of edges explodes.

NuGraph2

- **Second-generation model** incorporates a wide range of improvements over first proof-of-concept.
- Move from **edge classification** to **node (ie. hit) classification**.
  - Graph edge classification for track forming is a natural choice for LHC detectors, where sequential layers provide a natural constraint on edges.
  - Dense LArTPC environment provides no such constraints, and number of edges explodes.
- Introduce **more sophisticated semantic labelling** which considers a wider variety of particle types, ie. EM showers, Michel electrons, diffuse EM activity.

NuGraph2

- **Second-generation model** incorporates a wide range of improvements over first proof-of-concept.
- Move from **edge classification** to **node (ie. hit) classification**.
  - Graph edge classification for track forming is a natural choice for LHC detectors, where sequential layers provide a natural constraint on edges.
  - Dense LArTPC environment provides no such constraints, and number of edges explodes.
- Introduce **more sophisticated semantic labelling** which considers a wider variety of particle types, ie. EM showers, Michel electrons, diffuse EM activity.
- Build a model which **classifies all views simultaneously**, instead of classifying each detector views as an independent event.
  - Furthermore, allow **information exchange between 2D views** to break degeneracies.

University of CINCINNATI

NuGraph2 outputs

- The primary goal of NuGraph2 is to **classify each detector hit according to particle type**.

NuGraph2 outputs

- The primary goal of NuGraph2 is to **classify each detector hit according to particle type**.
- Use five semantic categories:
  - **MIP: Minimum ionising particles (muons, pions etc)**
  - **HIP: Highly ionising particles (protons, nucleons etc)**
  - **EM showers**
  - **Michel electrons**
  - **Diffuse activity (Compton scatters, neutrons etc)**

NuGraph2 outputs

- The primary goal of NuGraph2 is to **classify each detector hit according to particle type**.

- Use five semantic categories:
  - **MIP: Minimum ionising particles (muons, pions etc)**
  - **HIP: Highly ionising particles (protons, nucleons etc)**
  - **EM showers**
  - **Michel electrons**
  - **Diffuse activity (Compton scatters, neutrons etc)**

- Also perform **hit filtering** to remove hits that are not part of the primary physics interaction.

NuGraph2 architecture

- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.

NuGraph2 architecture

- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.
  - Each particle category is provided with a separate set of embedded features, which are convolved independently.

NuGraph2 architecture

- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.
    - Each particle category is provided with a separate set of embedded features, which are convolved independently.
    - Context information is exchanged between different particle types via a **categorical cross-attention** mechanism.

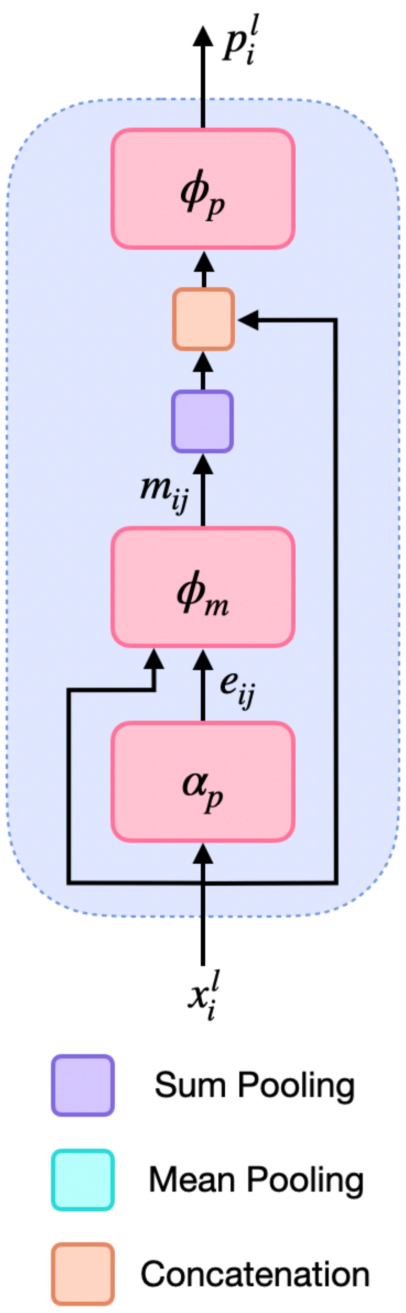
NuGraph2 architecture

- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.
  - Each particle category is provided with a separate set of embedded features, which are convolved independently.
  - Context information is exchanged between different particle types via a **categorical cross-attention** mechanism.
- Each message-passing iteration consists of two phases, the **planar** step and the **nexus** step:

NuGraph2 architecture

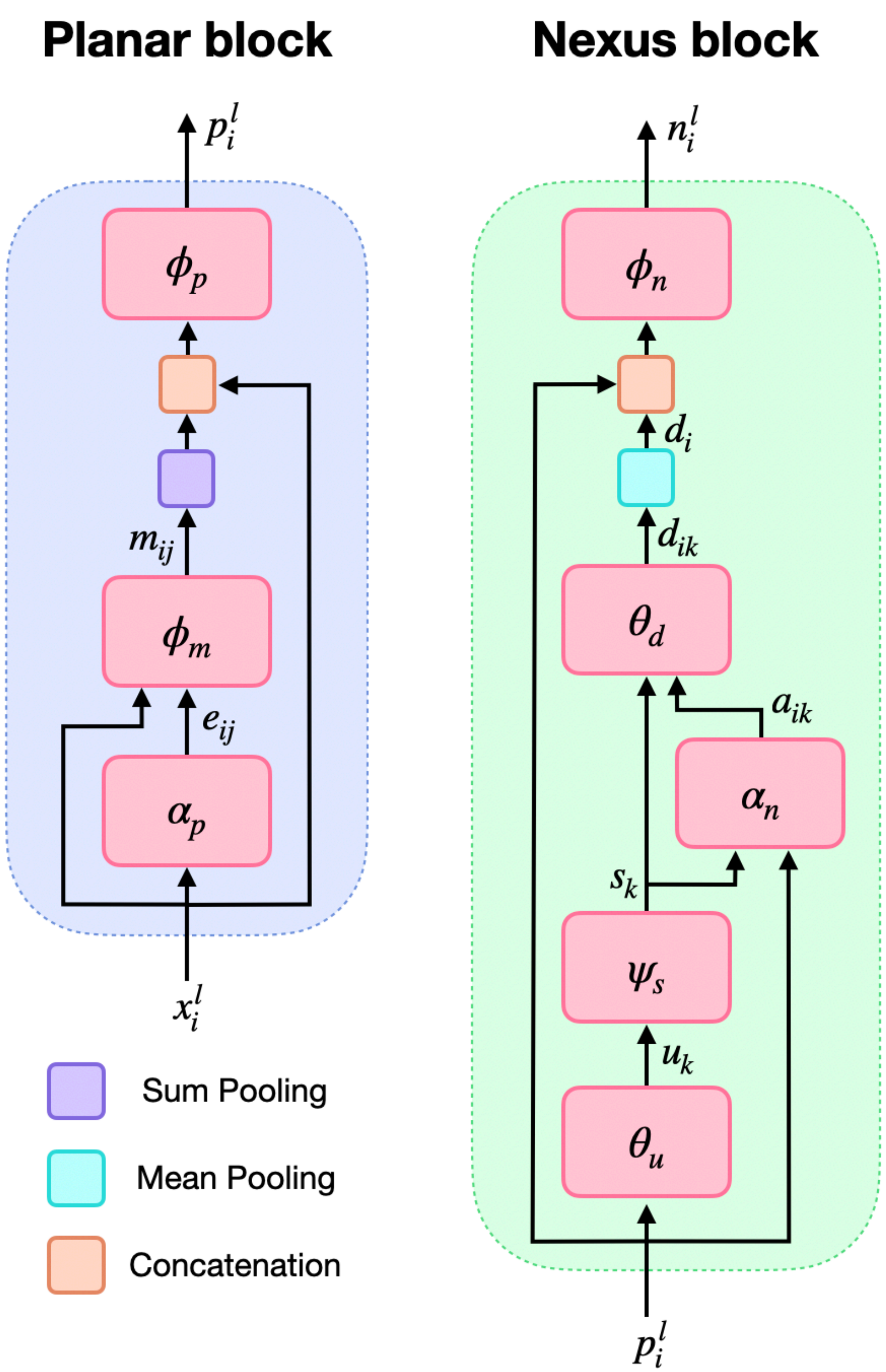
- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.
  - Each particle category is provided with a separate set of embedded features, which are convolved independently.
  - Context information is exchanged between different particle types via a **categorical cross-attention** mechanism.
- Each message-passing iteration consists of two phases, the **planar** step and the **nexus** step:
  - Pass messages internally in each plane.

Planar block



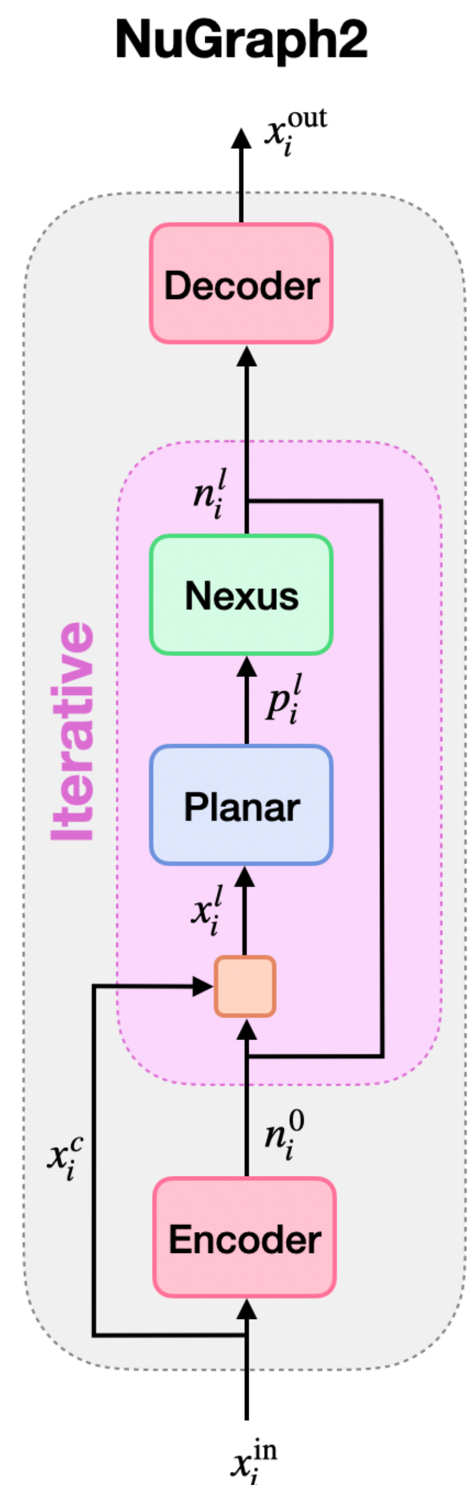
NuGraph2 architecture

- NuGraph2's core convolution engine is a **self-attention message passing network** utilising a **categorical embedding**.
    - Each particle category is provided with a separate set of embedded features, which are convolved independently.
    - Context information is exchanged between different particle types via a **categorical cross-attention** mechanism.
- Each message-passing iteration consists of two phases, the **planar** step and the **nexus** step:
    - Pass messages internally in each plane.
    - Pass messages up to 3D nexus nodes to share context information.



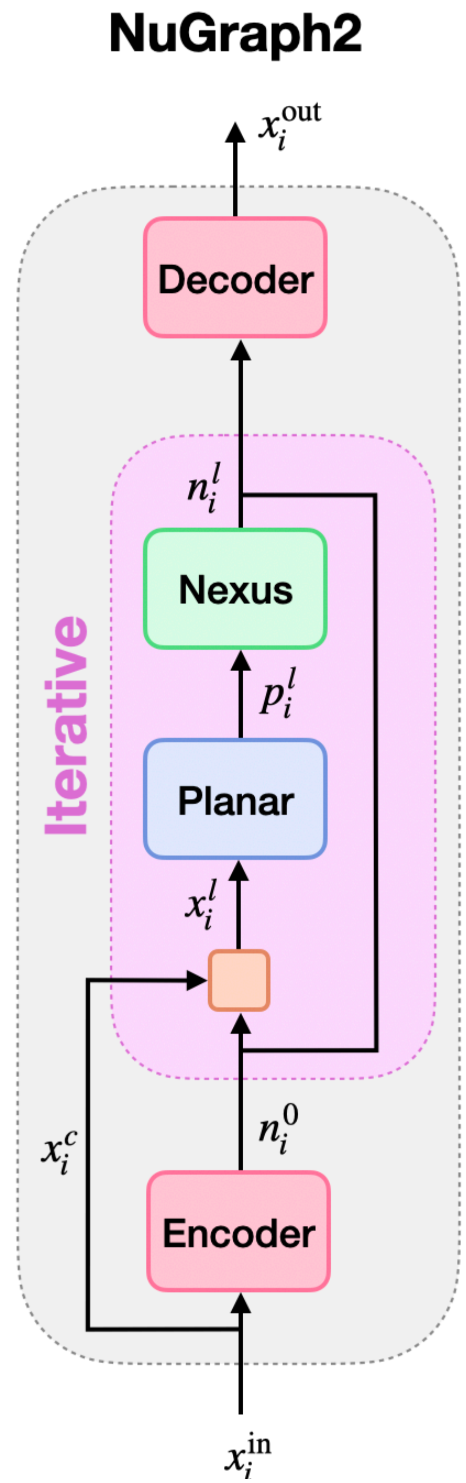
NuGraph2 architecture

- This **iterative two-step message-passing engine** forms the backbone of the NuGraph2 architecture.



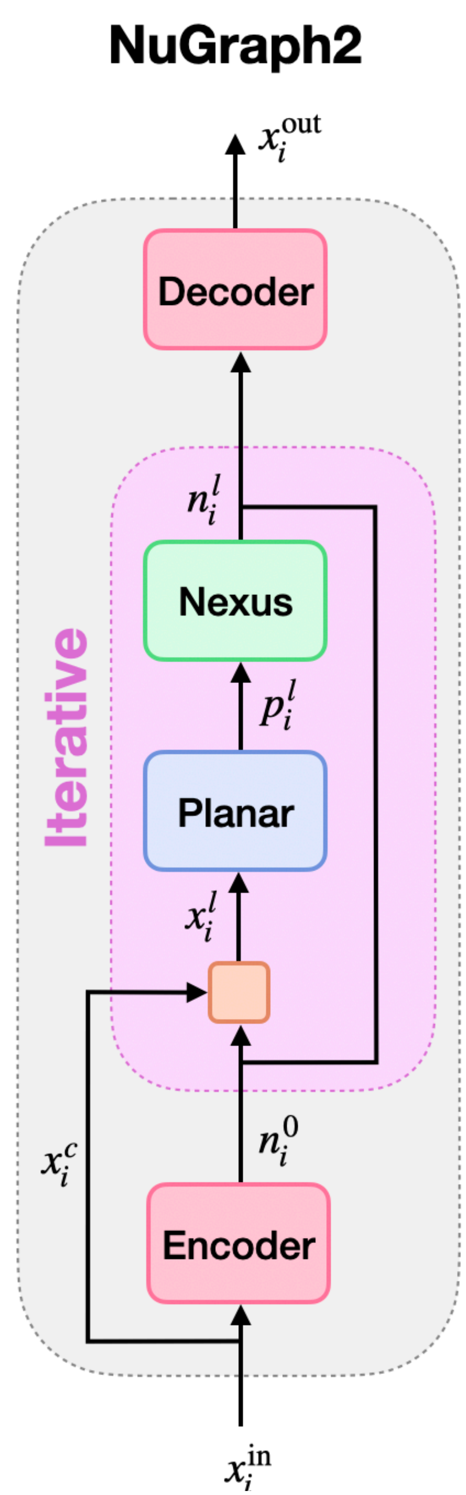
NuGraph2 architecture

- This **iterative two-step message-passing engine** forms the backbone of the NuGraph2 architecture.
- An **initial encoding step** generates a learned embedding for each graph node.



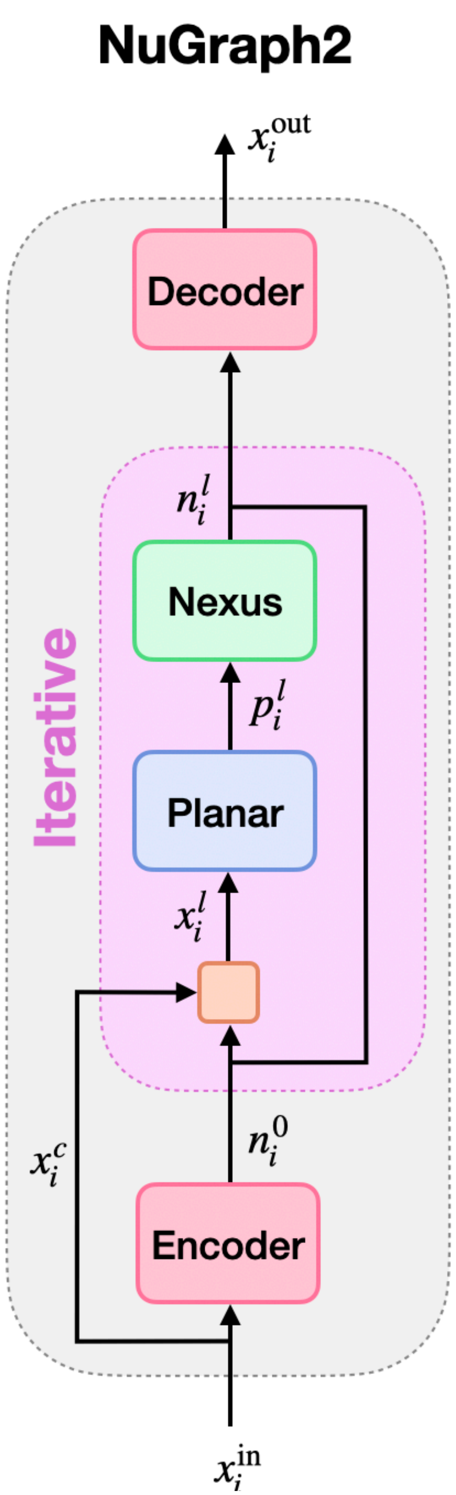
NuGraph2 architecture

- This **iterative two-step message-passing engine** forms the backbone of the NuGraph2 architecture.
- An **initial encoding step** generates a learned embedding for each graph node.
- The output of the message-passing engine can be forwarded into **any number of decoders** for a variety of tasks.



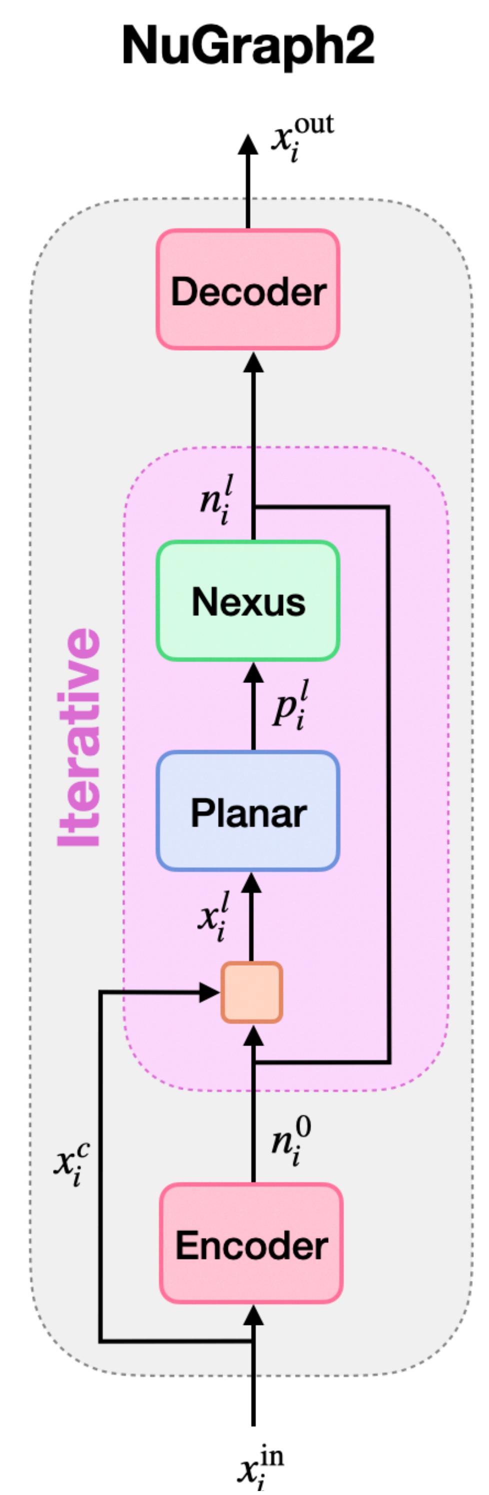
NuGraph2 architecture

- This **iterative two-step message-passing engine** forms the backbone of the NuGraph2 architecture.
- An **initial encoding step** generates a learned embedding for each graph node.
- The output of the message-passing engine can be forwarded into **any number of decoders** for a variety of tasks.
  - In this talk, we present two tasks: **semantic hit segmentation** and **background filtering**.



NuGraph2 architecture

- This **iterative two-step message-passing engine** forms the backbone of the NuGraph2 architecture.
- An **initial encoding step** generates a learned embedding for each graph node.
- The output of the message-passing engine can be forwarded into **any number of decoders** for a variety of tasks.
  - In this talk, we present two tasks: **semantic hit segmentation** and **background filtering**.
  - This engine can be leveraged for much more than just these two tasks. See the **NuGraph3** talk later for more details ;)

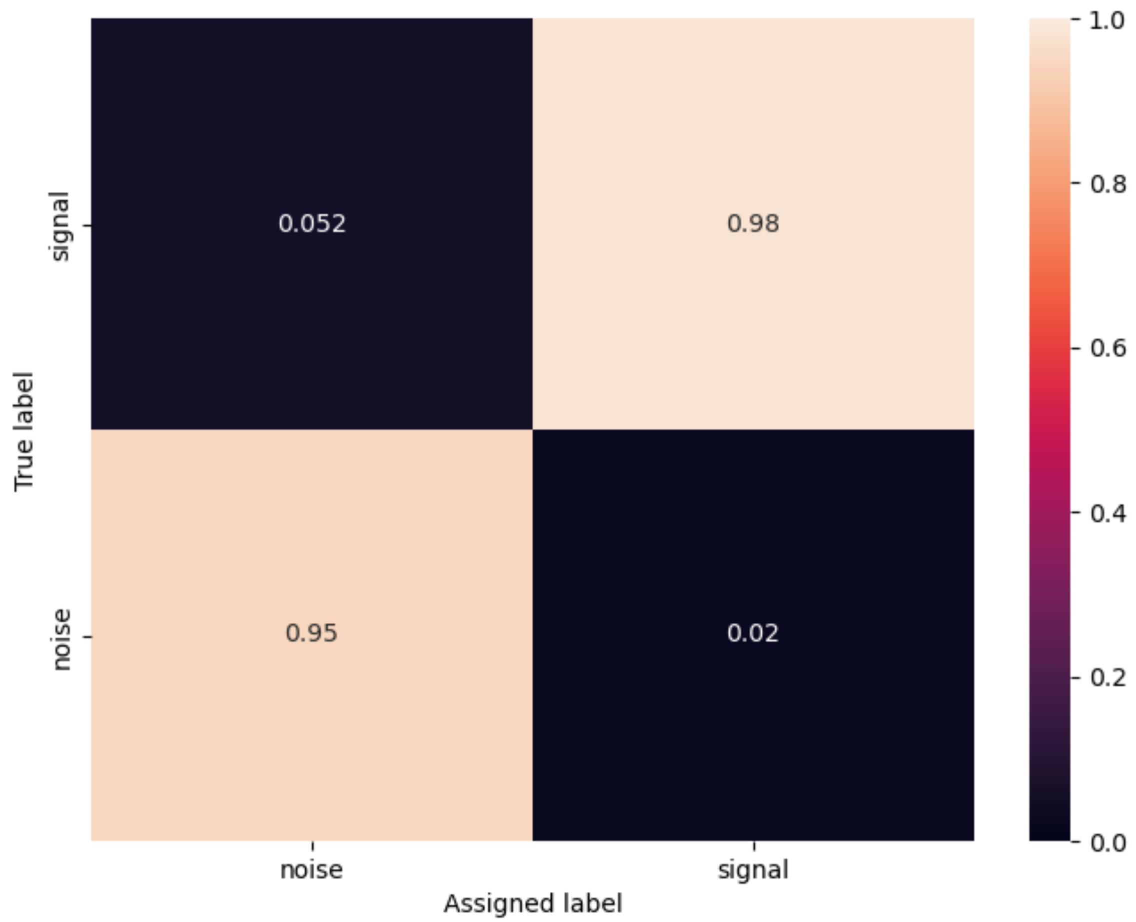


NuGraph2

Background filtering

- Performance metrics: **recall 0.978**, **precision 0.977**.

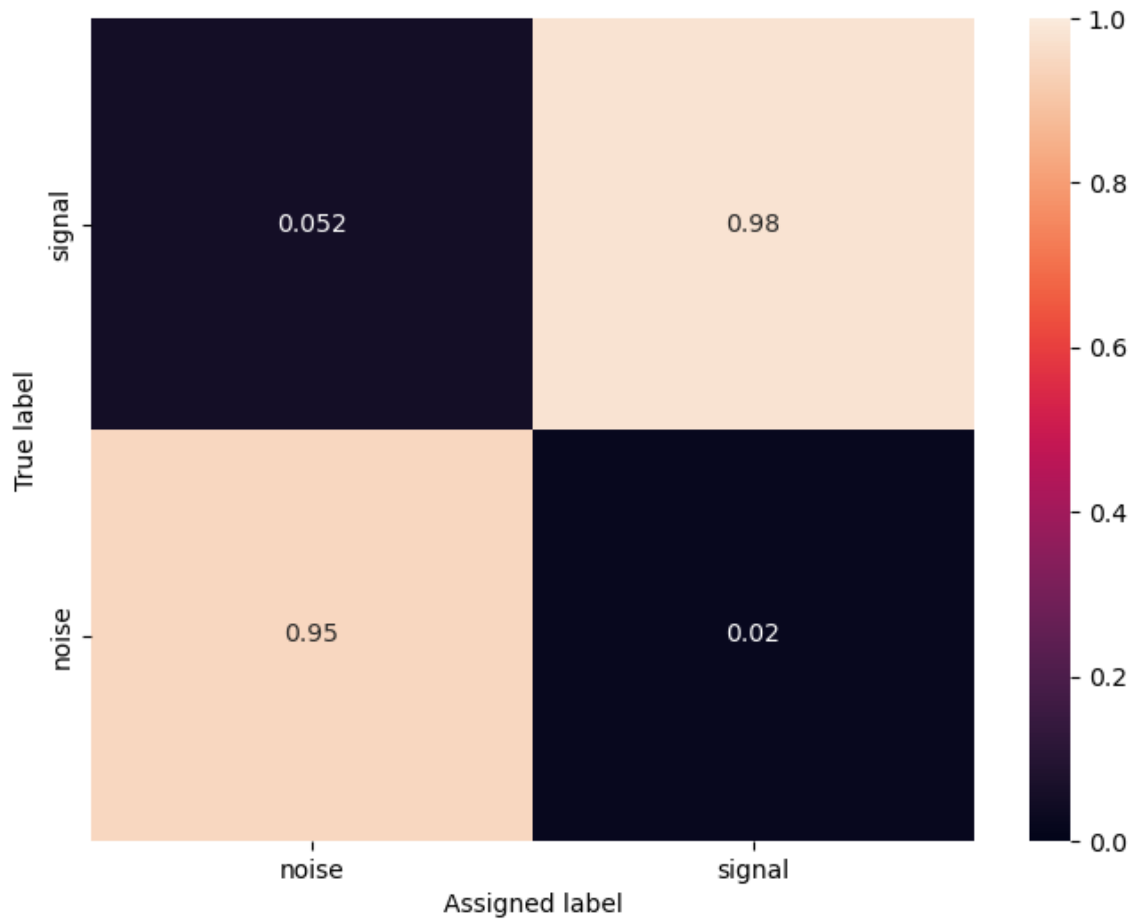
Recall matrix



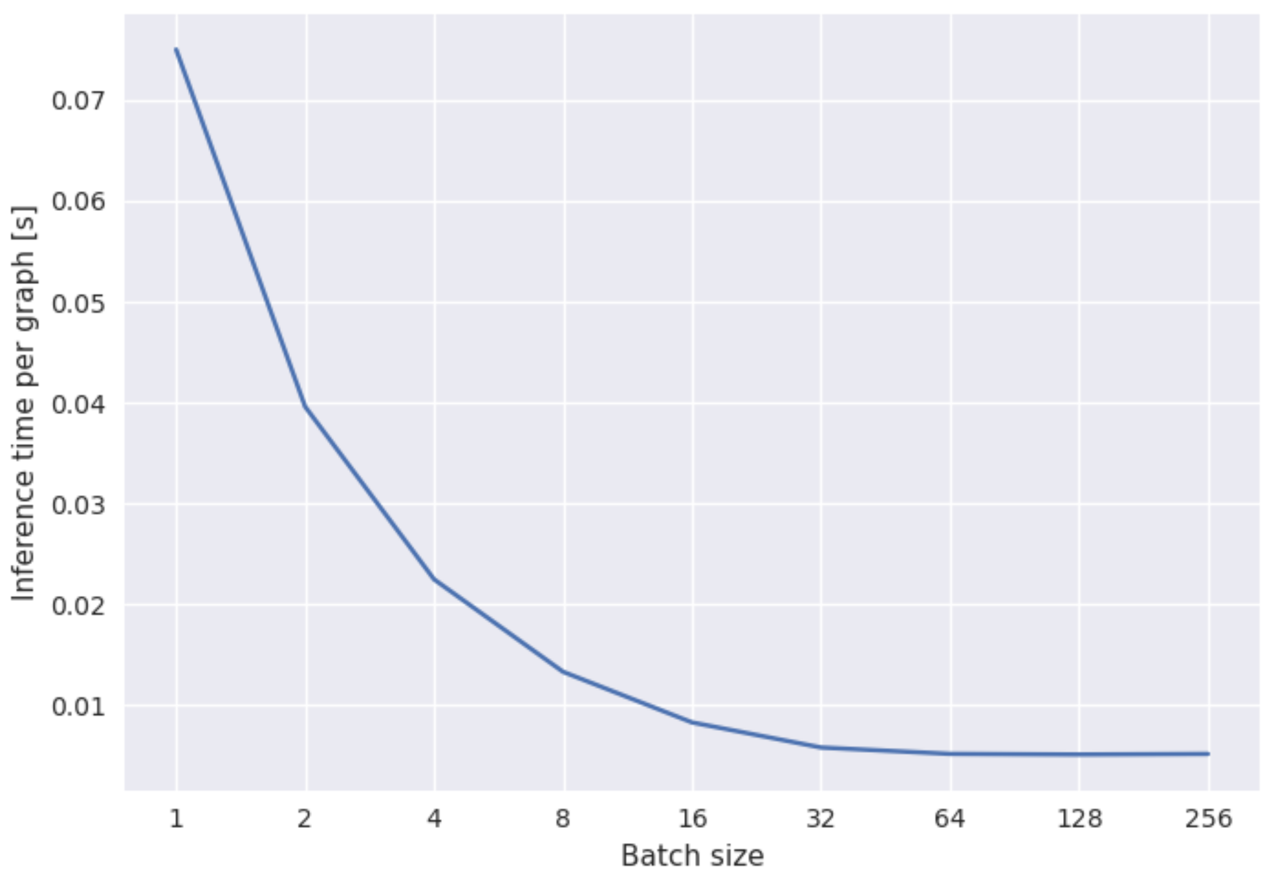
Background filtering

- Performance metrics: **recall 0.978**, **precision 0.977**.
- Inference time: **0.12 s/evt on CPU**, **0.005s/evt batched on GPU**

Recall matrix



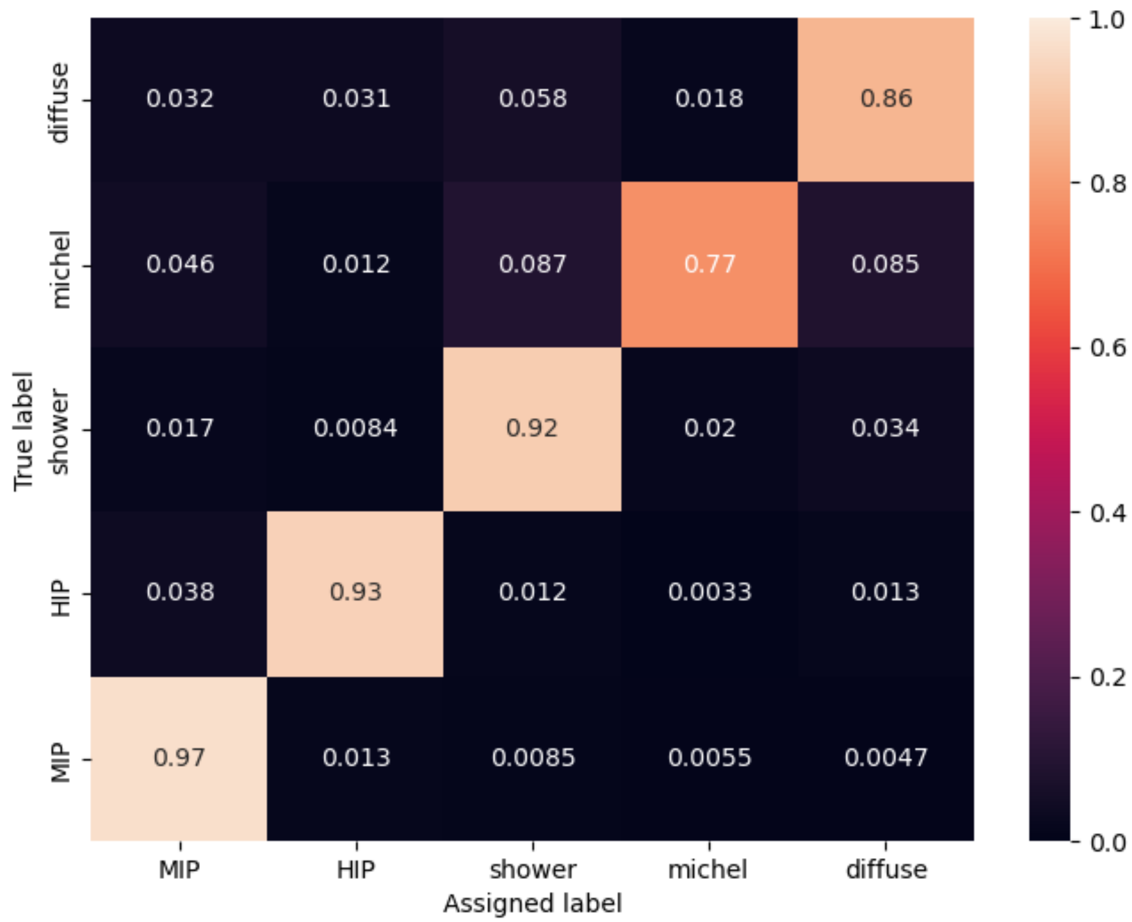
GPU inference time vs batch size



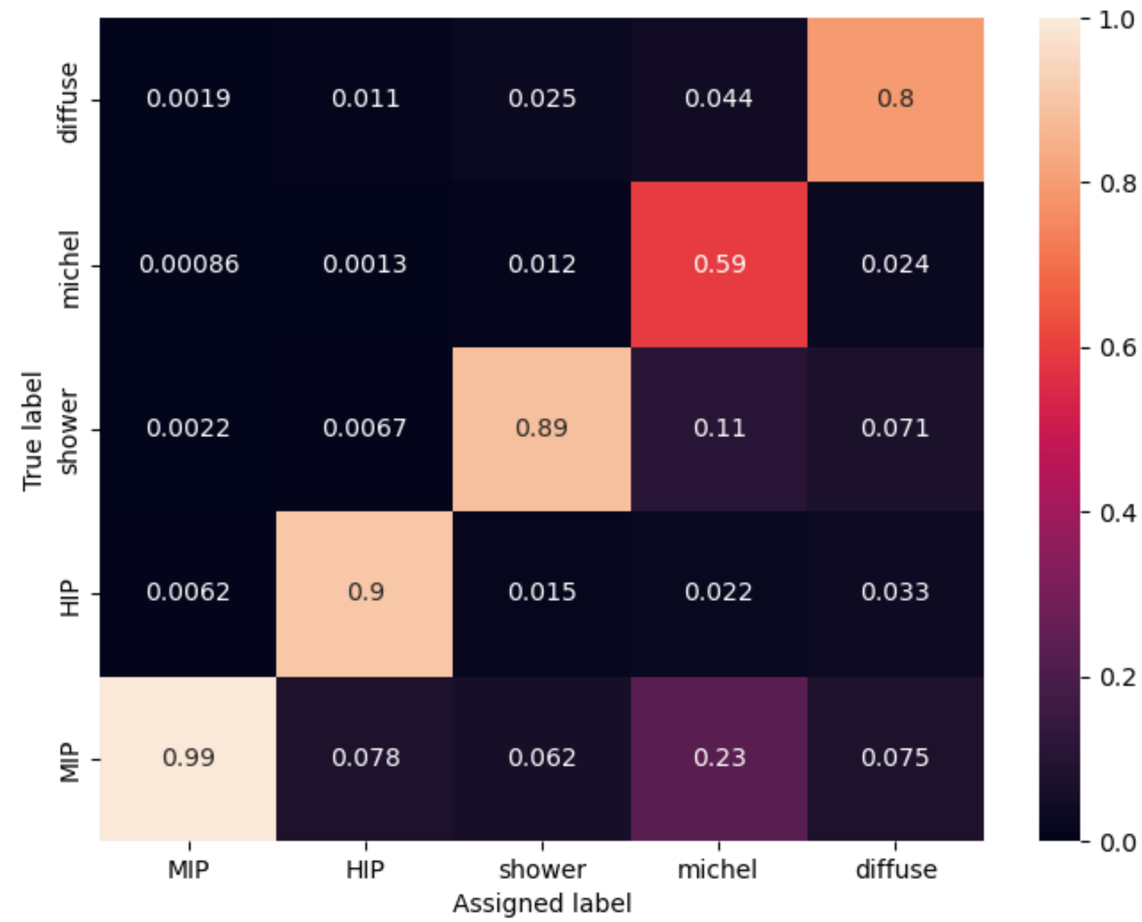
Hit classification

- Performance metrics: **recall 0.948**, **precision 0.948**.
- Recently improved performance by enhancing ν_μ component of dataset, and using **recall loss** to counteract class imbalance.

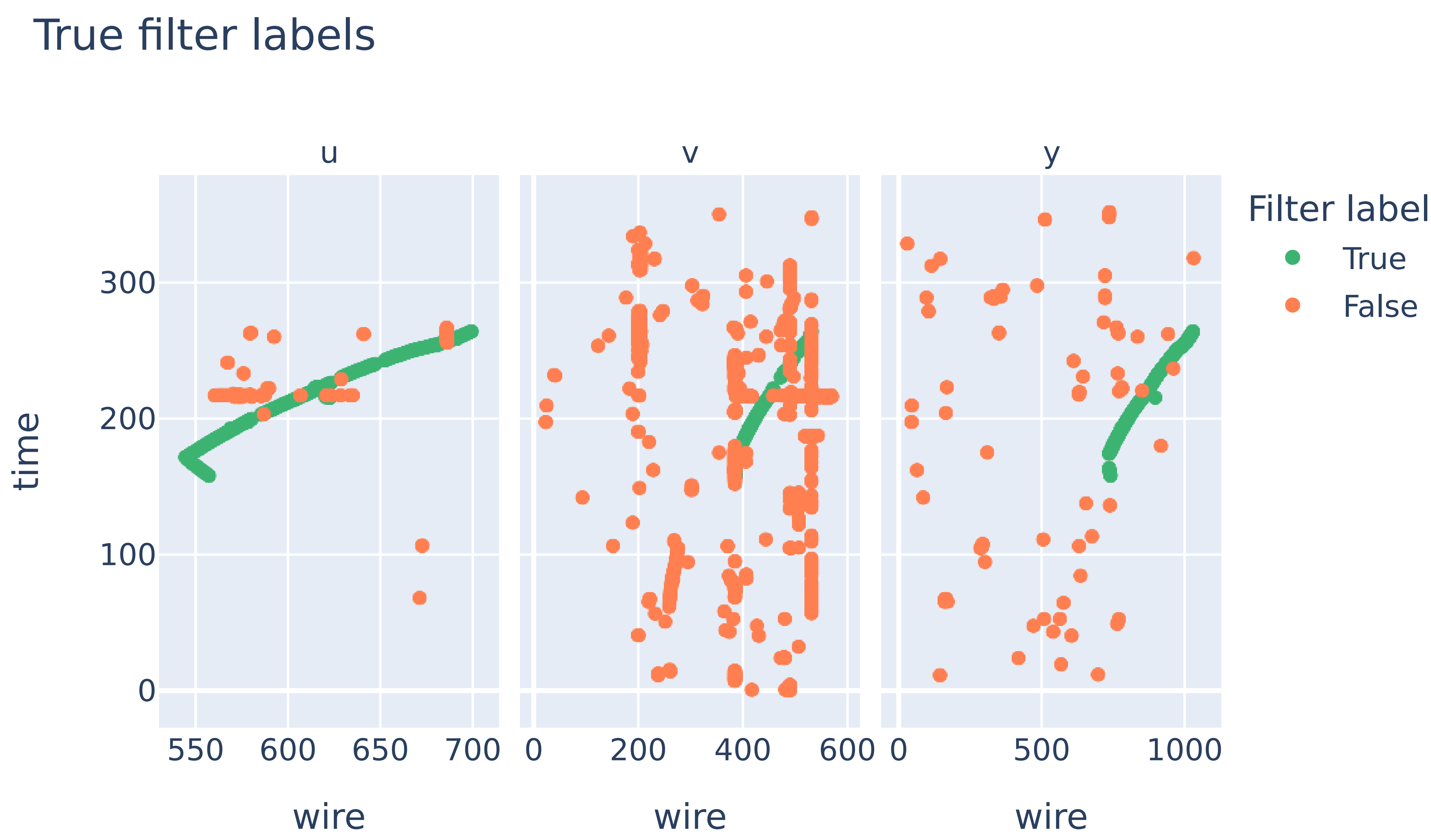
Recall matrix



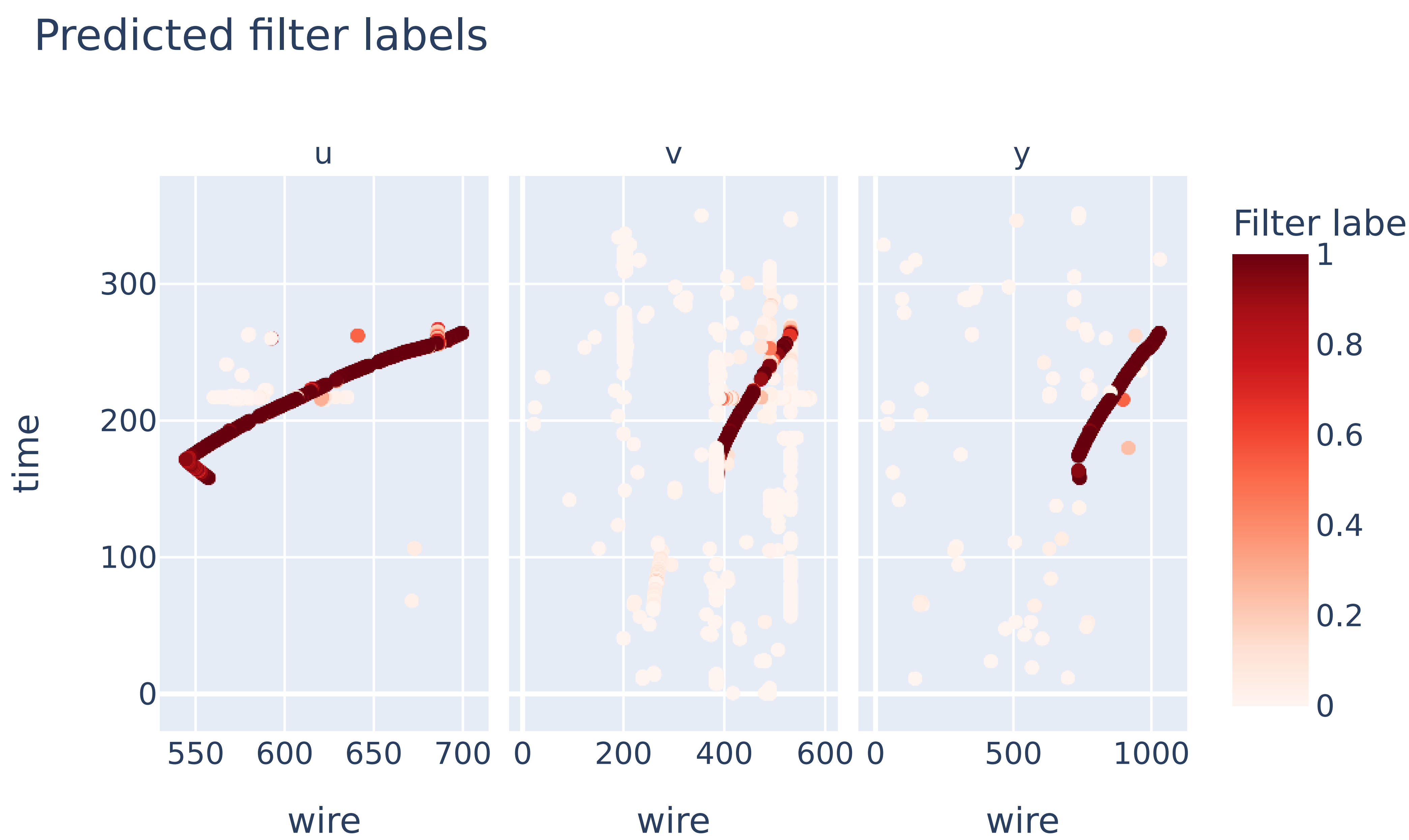
Precision matrix



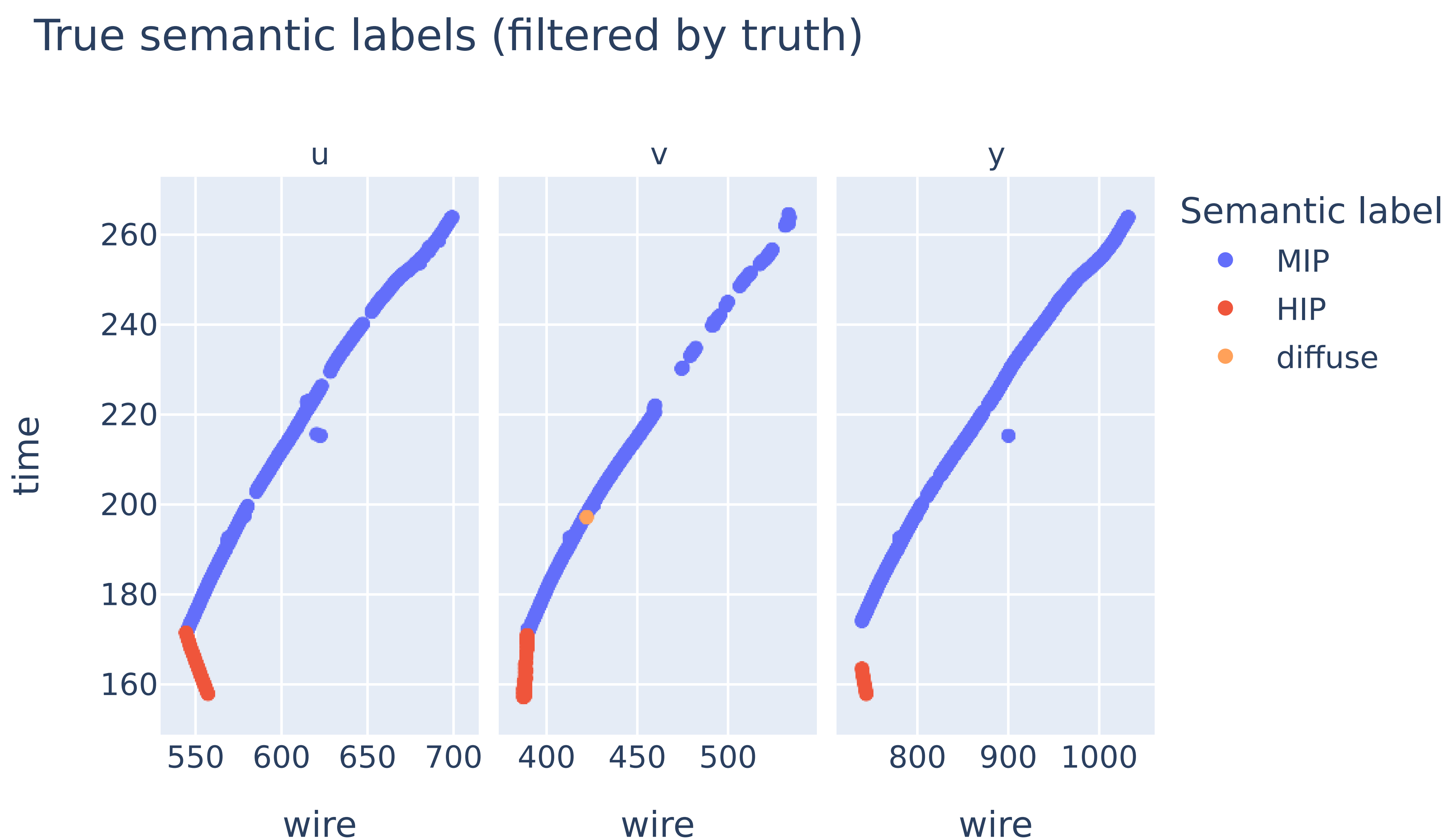
Example event #1



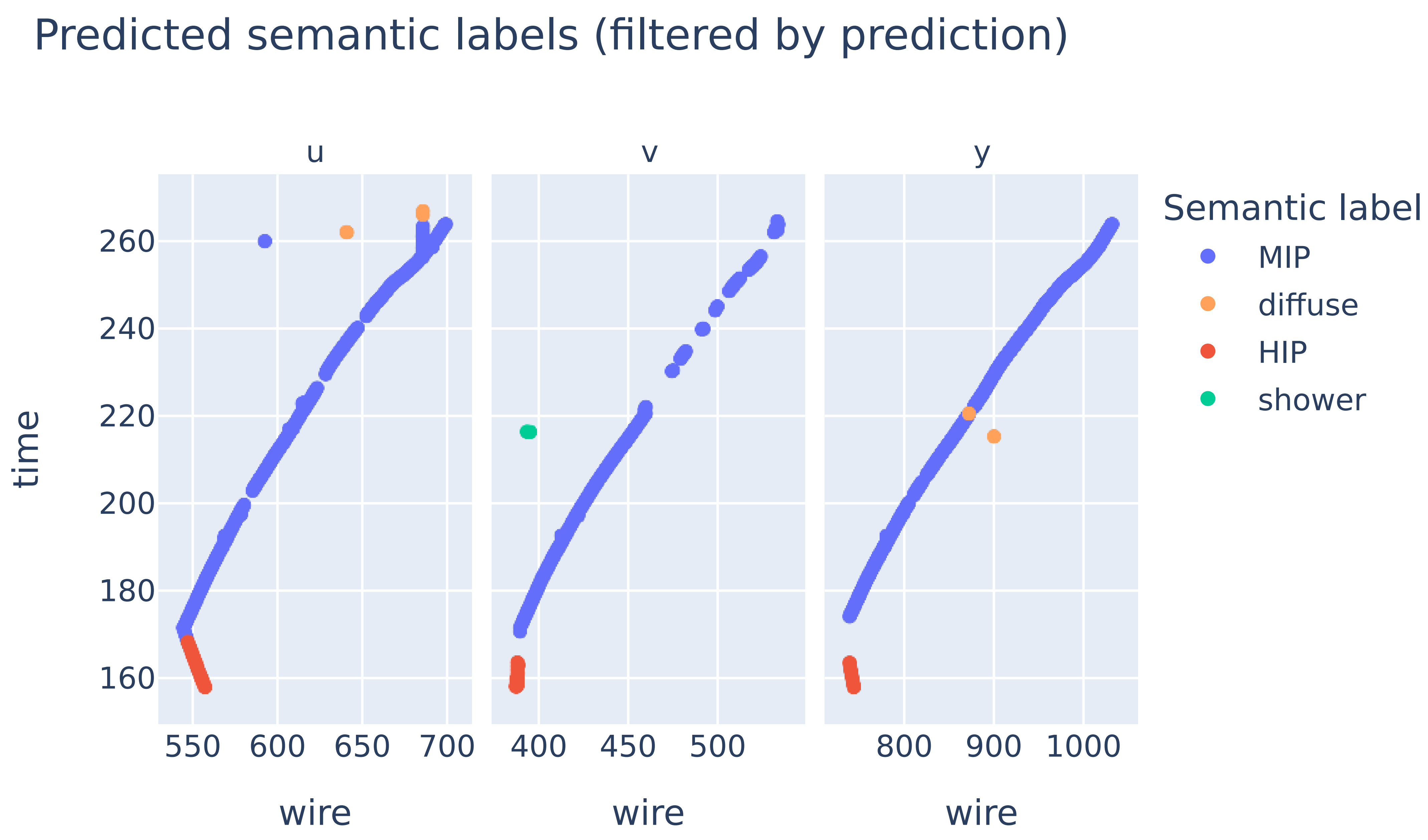
Example event #1



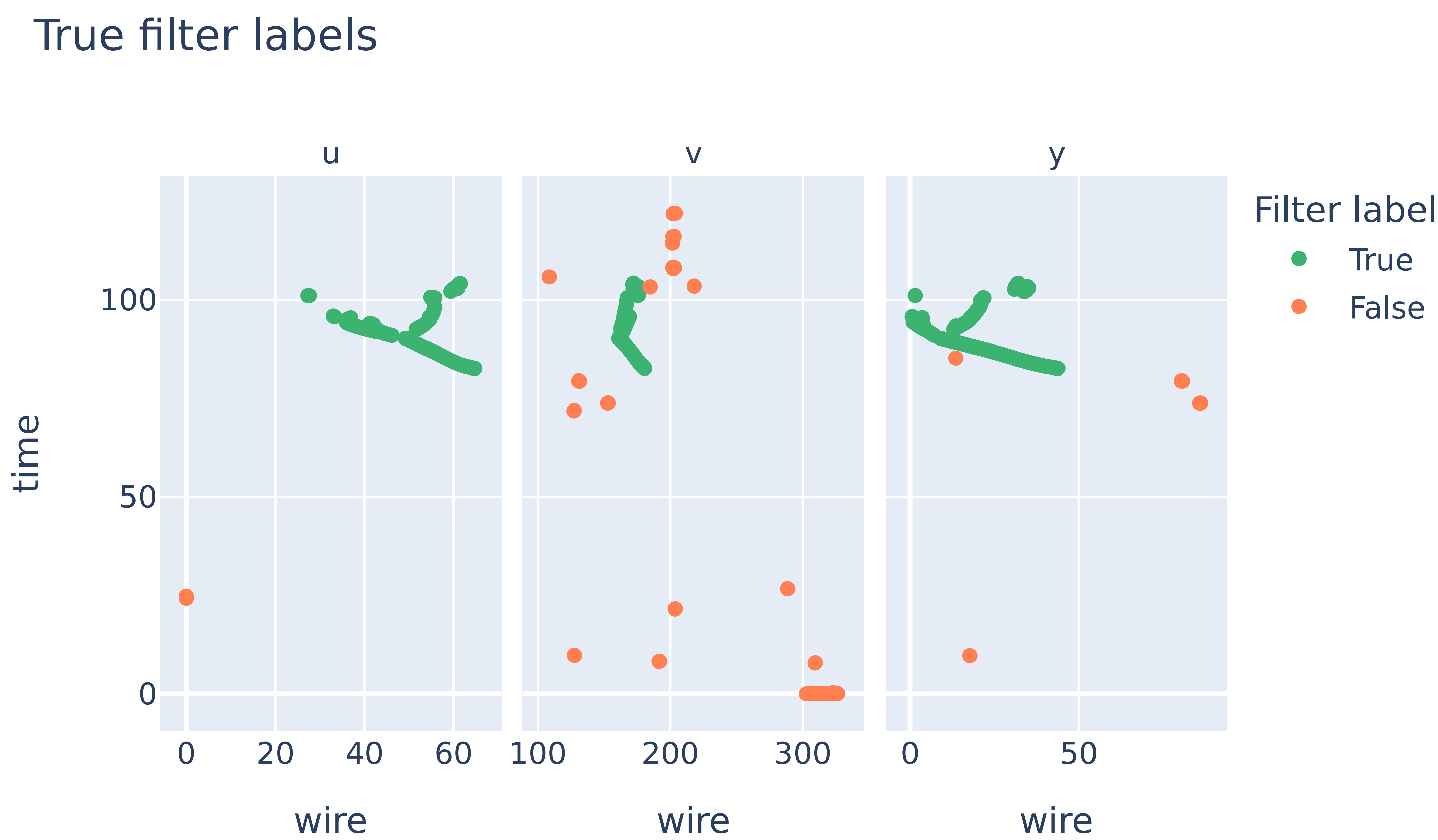
Example event #1



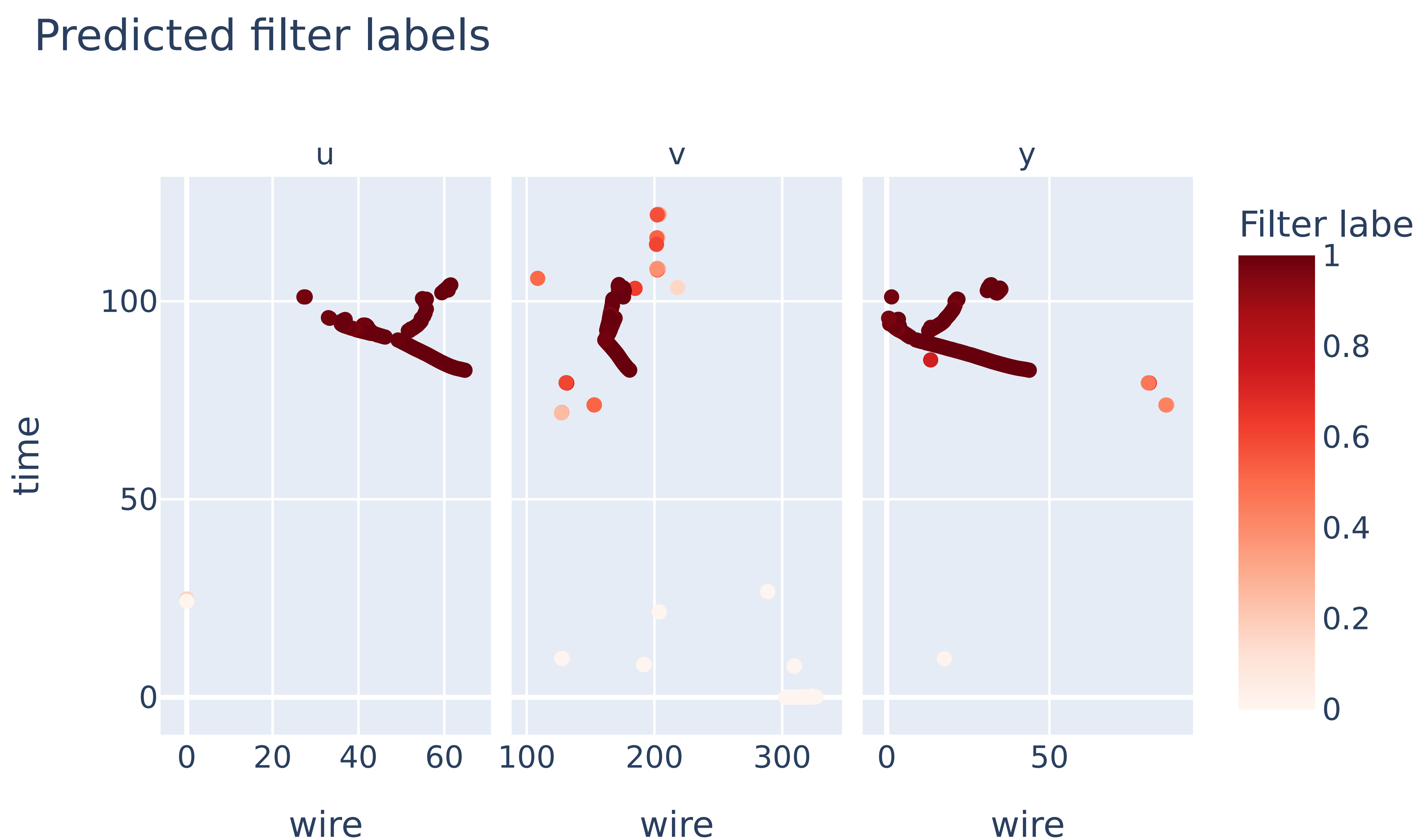
Example event #1



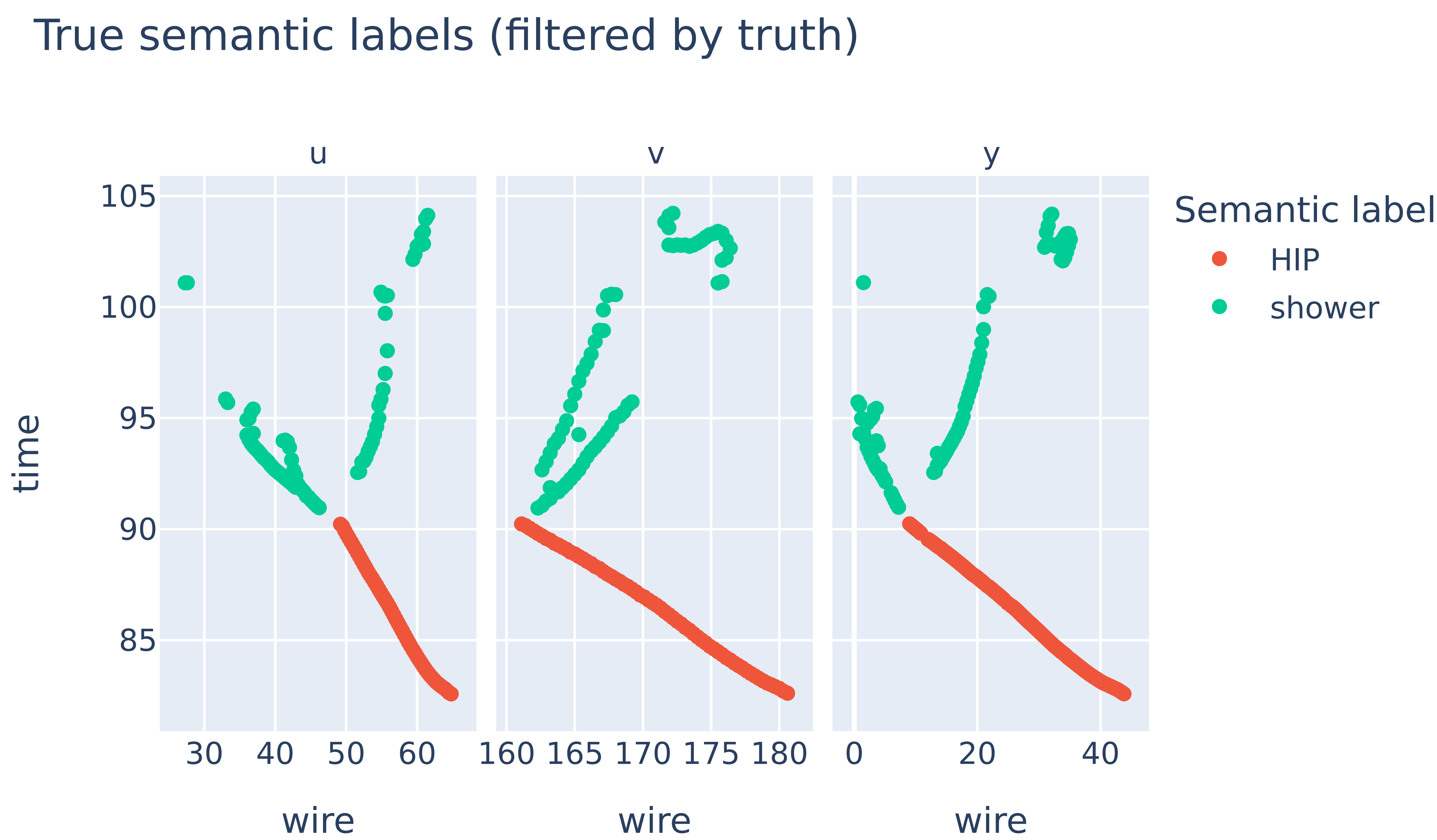
Example event #2



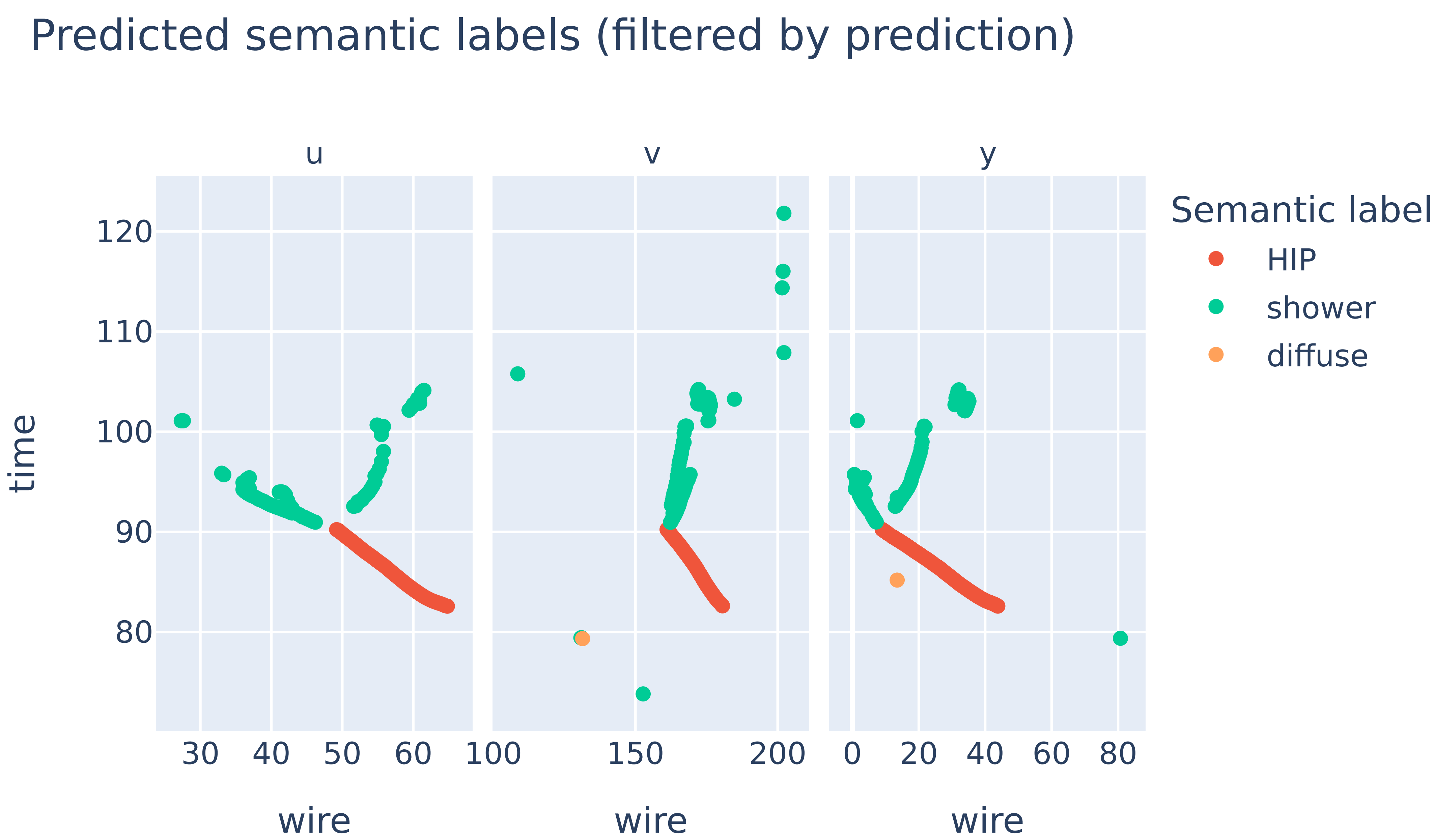
Example event #2



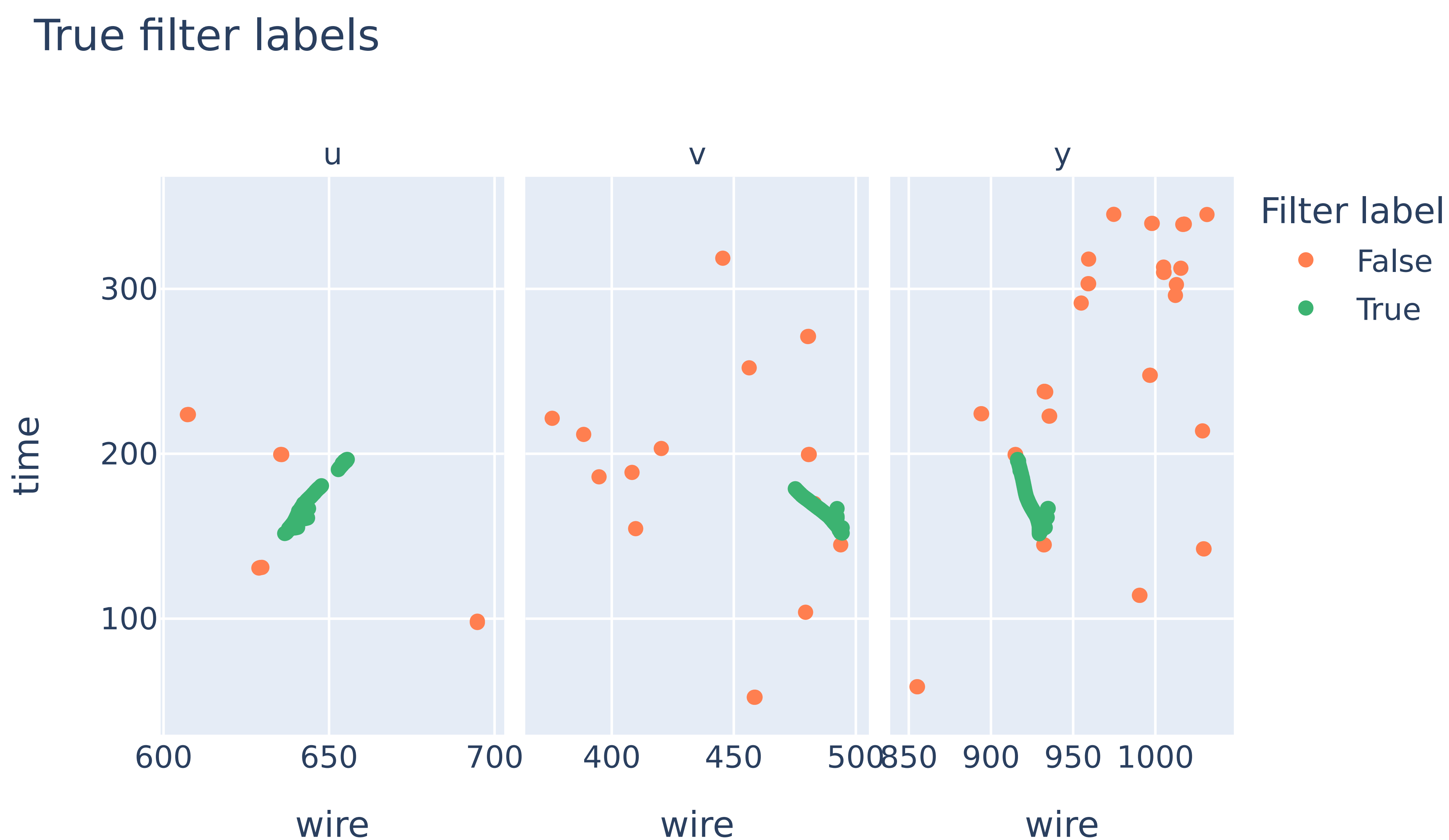
Example event #2



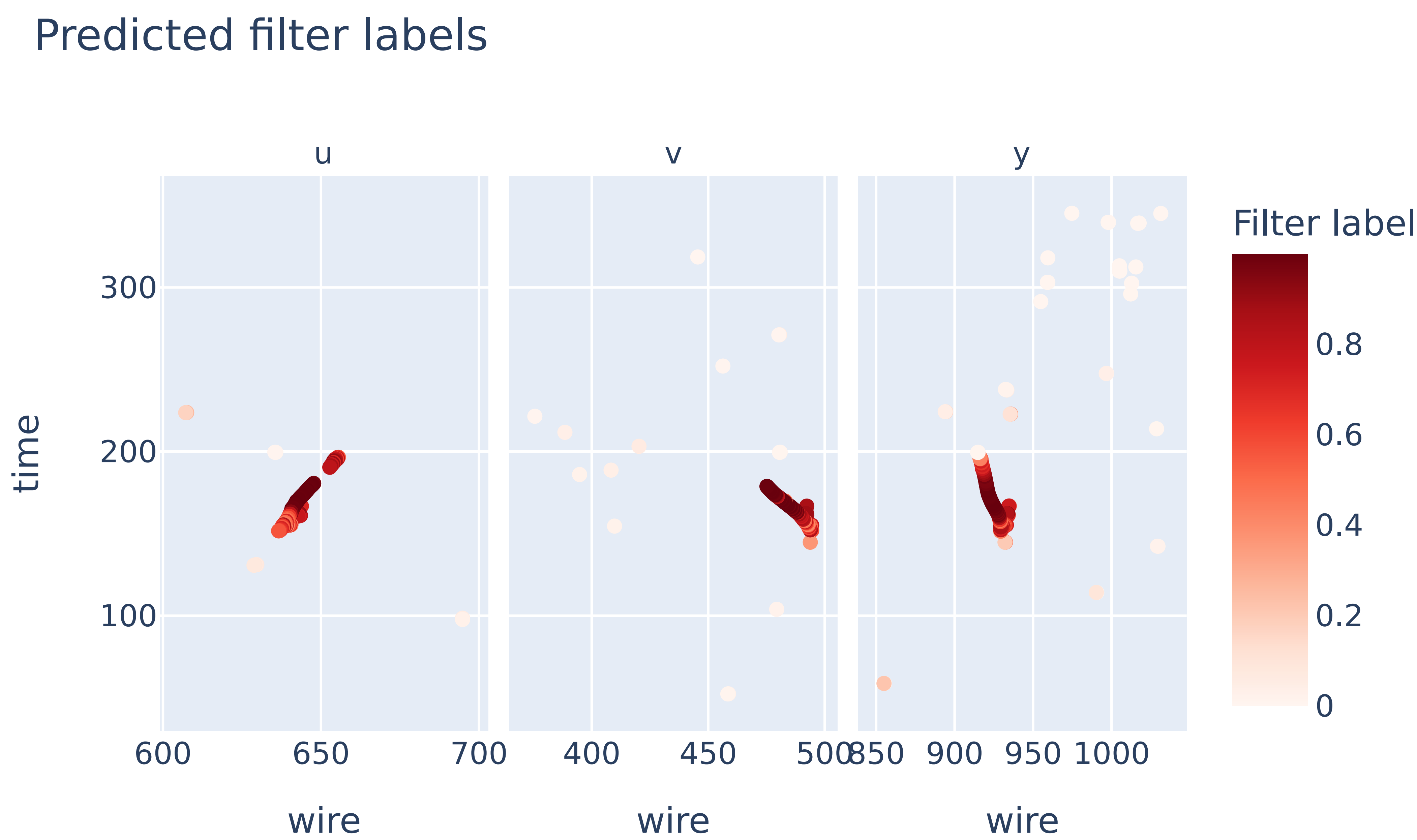
Example event #2



Example event #3

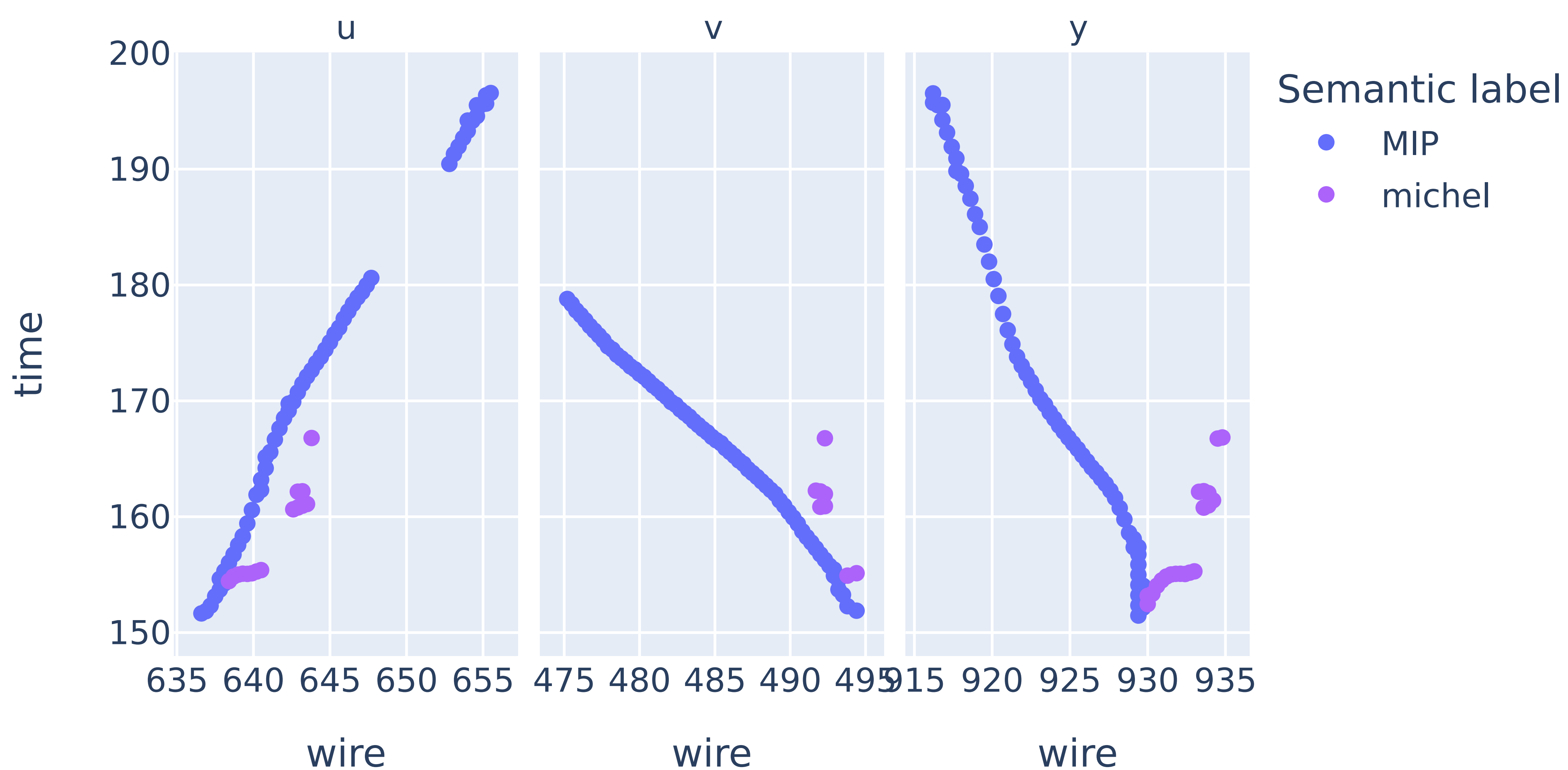


Example event #3

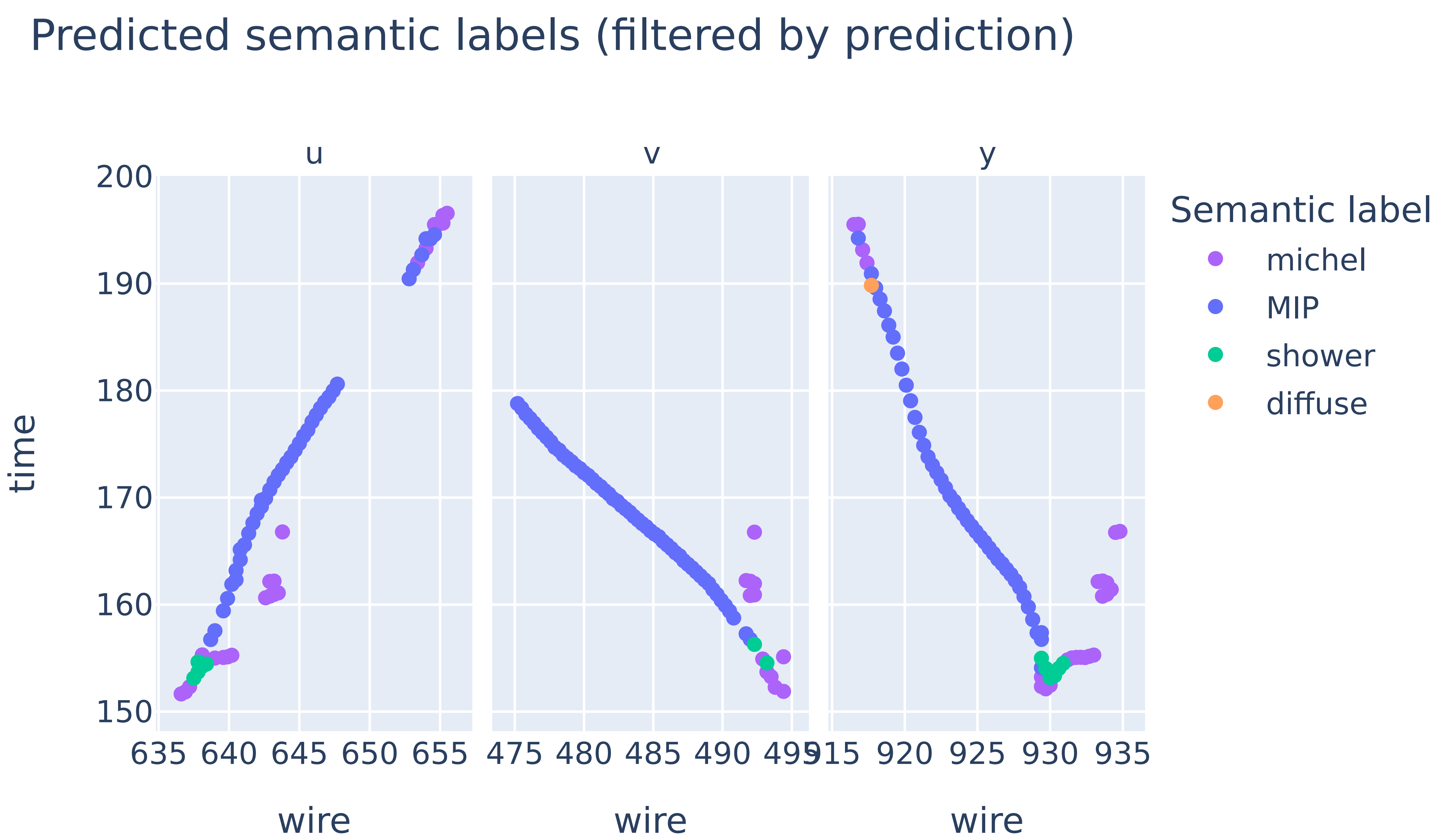


Example event #3

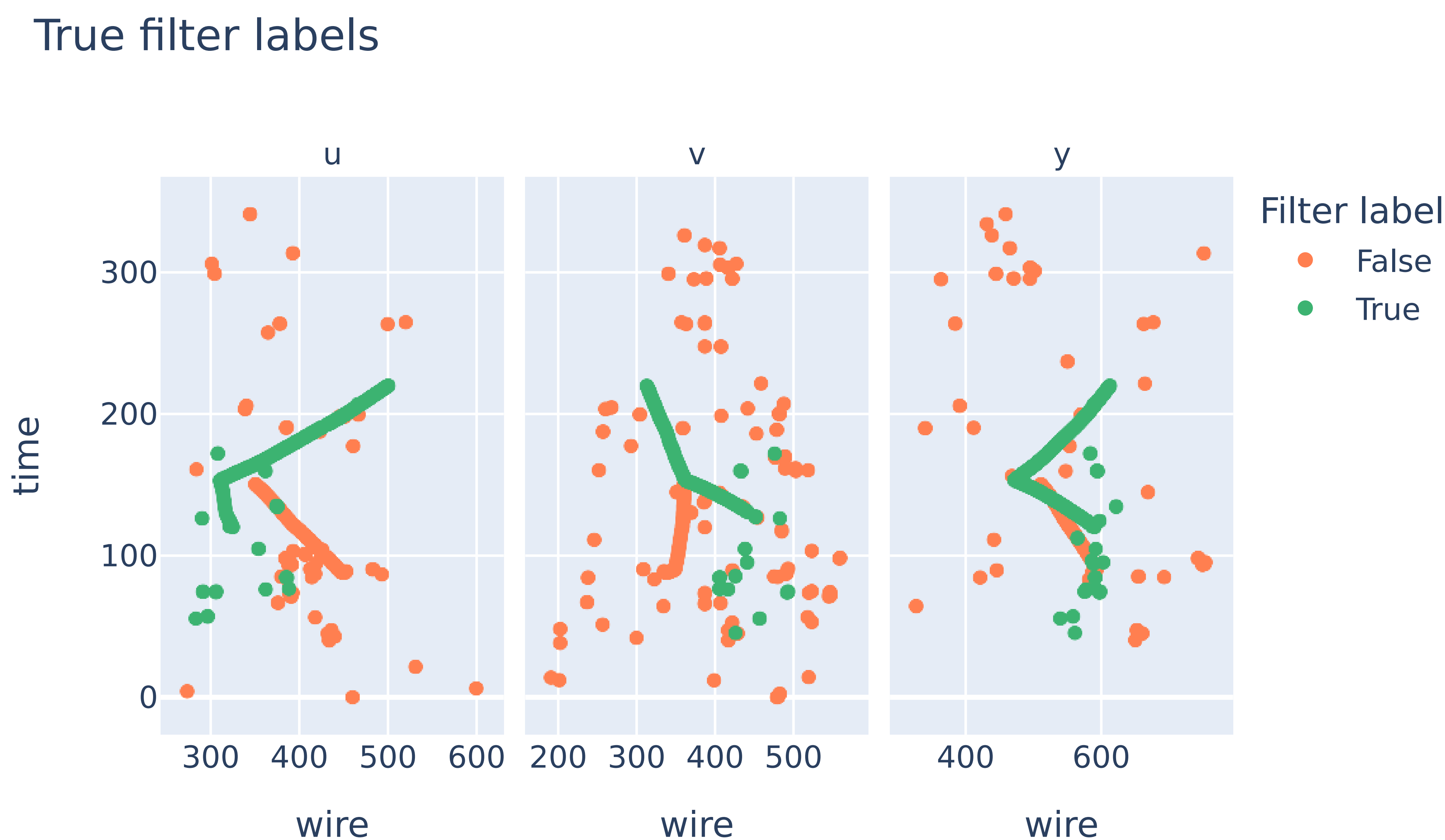
True semantic labels (filtered by truth)



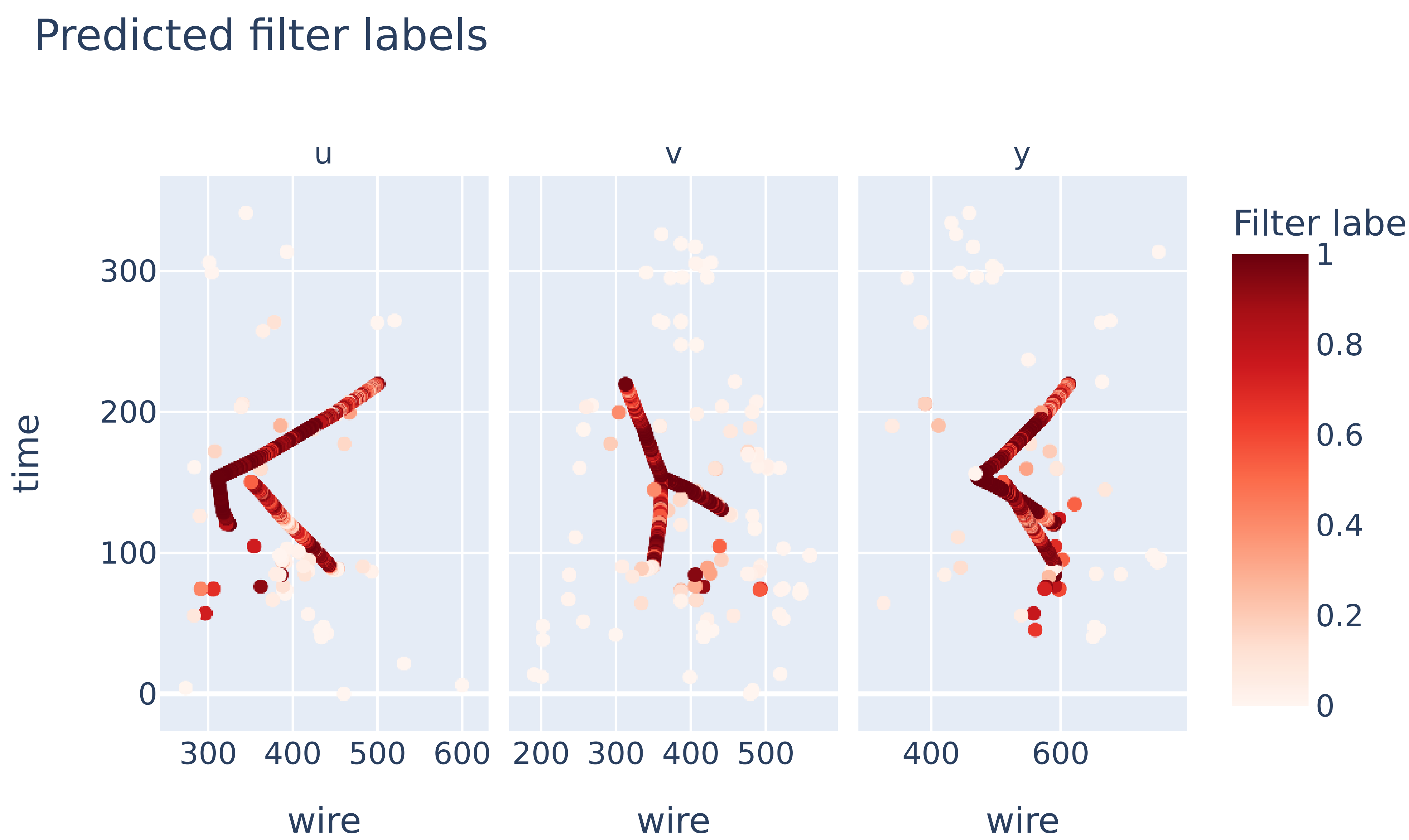
Example event #3



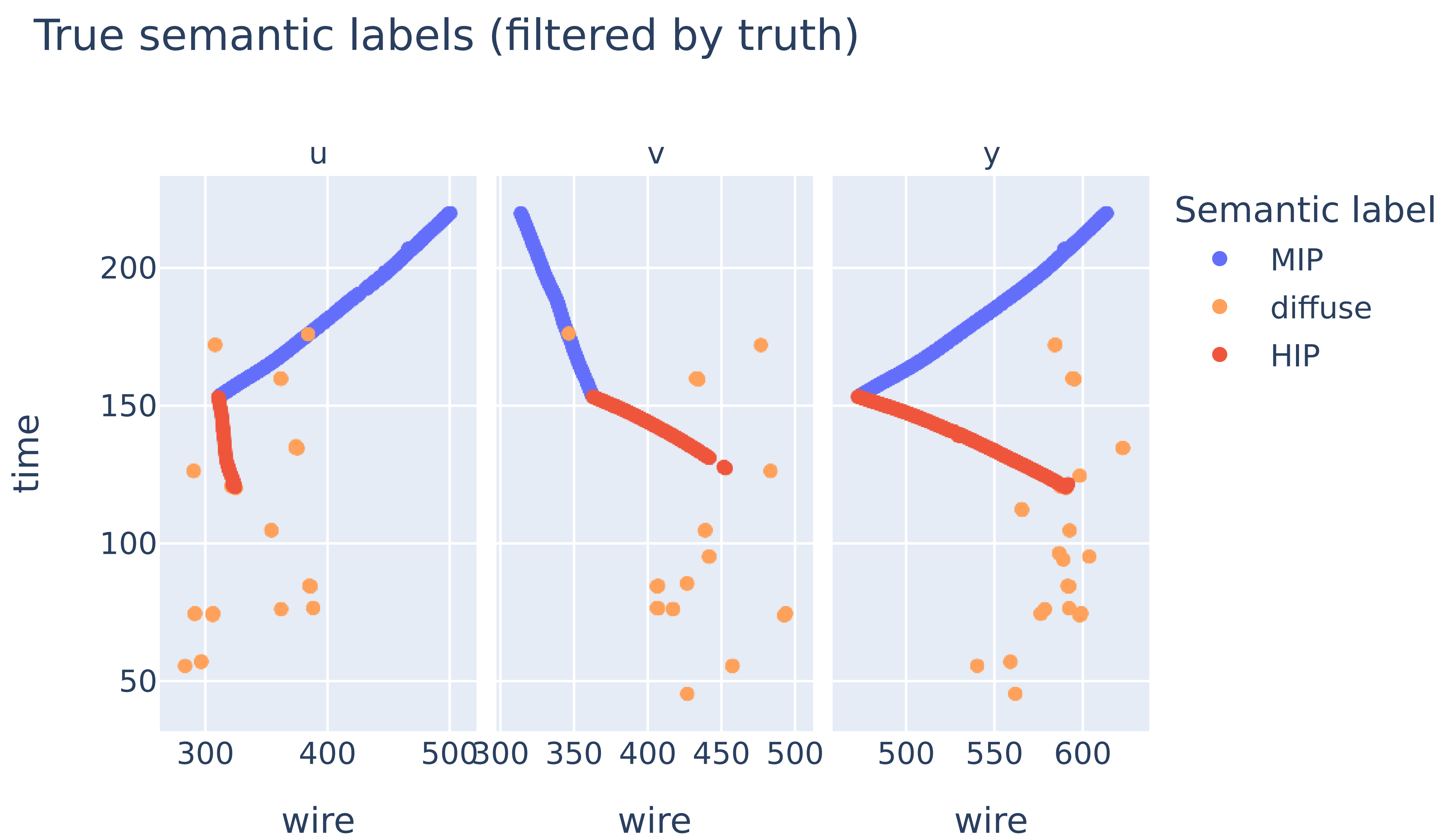
Example event #4



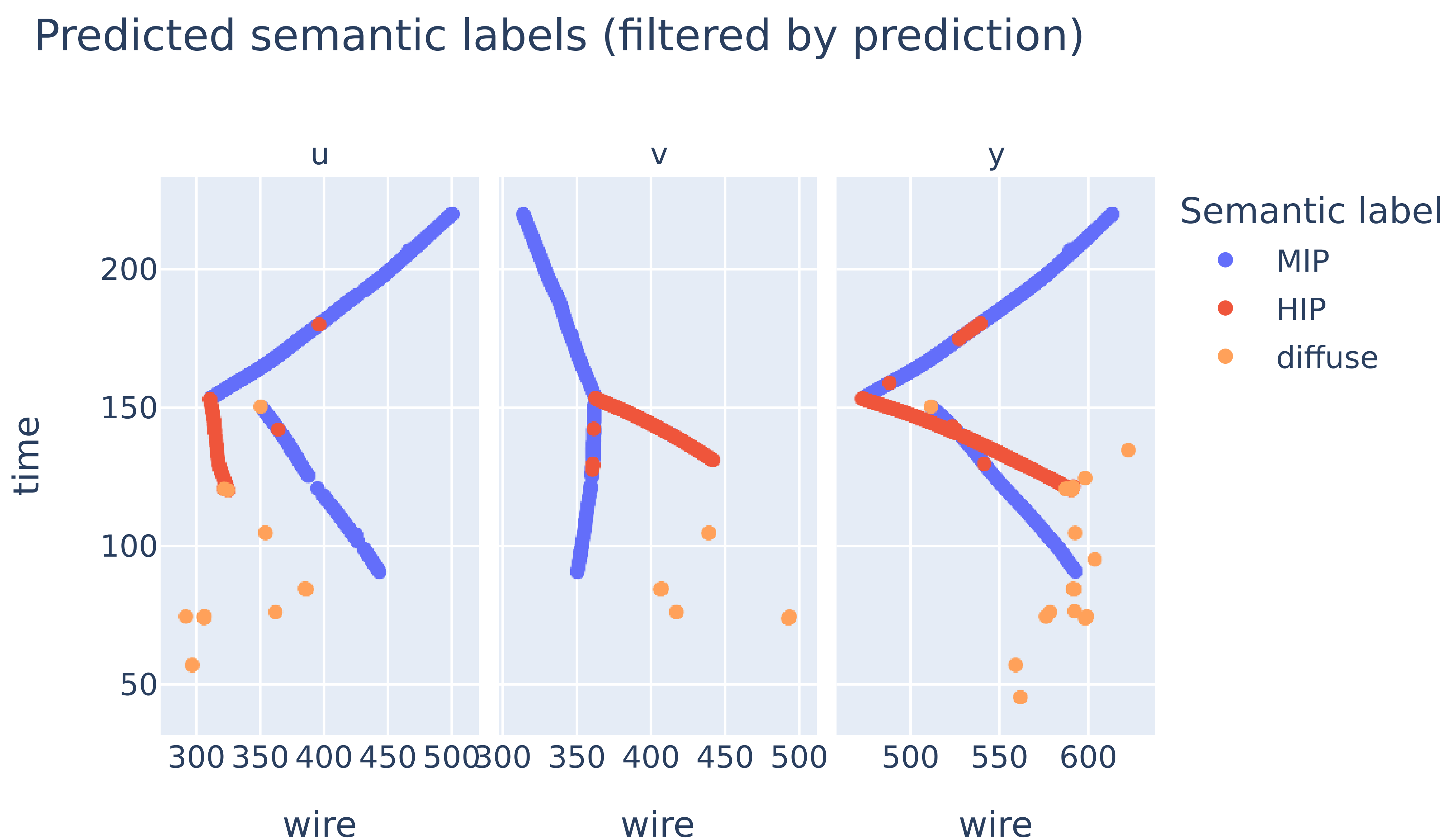
Example event #4



Example event #4

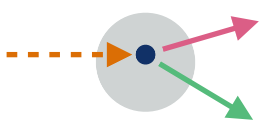
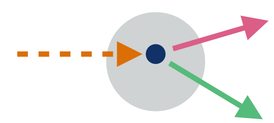
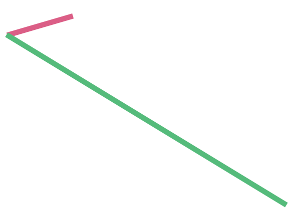
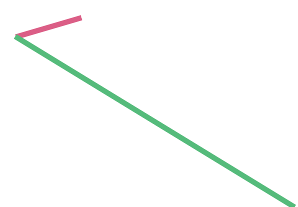
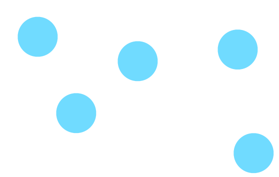
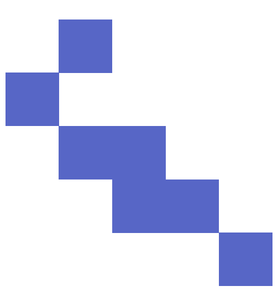
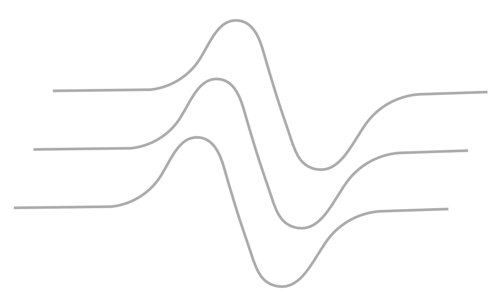


Example event #4



Common abstraction for neutrino experiments

- Although the details of many neutrino physics experiments vary, the majority of them share a common paradigm at a high level.

NOvA		MicroBooNE		Shared structure
	Neutrino generator (GENIE)		Neutrino generator (GENIE)	**Event information**
	Particle simulation (Geant4)		Particle simulation (Geant4)	**True particles**
	True light depositions		True ionization electrons	**True energy deposits**
	Photoelectrons on APDs		Pulses on TPC wires	**Detector hits**

University of CINCINNATI

The NuML ecosystem

- The **NuML** LArSoft package (github) produces HDF5 files containing tables summarising basic data products.

The NuML ecosystem

- The **NuML** LArSoft package (github) produces HDF5 files containing tables summarising basic data products.
- The **ph5concat** package (github, conda) merges many event HDF5 files into a single file in parallel.

The NuML ecosystem

- The **NuML** LArSoft package (github) produces HDF5 files containing tables summarising basic data products.
- The **ph5concat** package (github, conda) merges many event HDF5 files into a single file in parallel.
- The **pynuml** package (github, pypi, conda) generates truth labels and efficiently preprocesses event HDF5 files into graph datasets.

The NuML ecosystem

- The **NuML** LArSoft package (github) produces HDF5 files containing tables summarising basic data products.
- The **ph5concat** package (github, conda) merges many event HDF5 files into a single file in parallel.
- The **pynuml** package (github, pypi, conda) generates truth labels and efficiently preprocesses event HDF5 files into graph datasets.
- The **NuGraph** package (github, pypi, conda) contains a PyTorch implementation of the network architecture, as well as data loaders for the graph dataset.

The NuML ecosystem

- The **NuML** LArSoft package (github) produces HDF5 files containing tables summarising basic data products.
- The **ph5concat** package (github, conda) merges many event HDF5 files into a single file in parallel.
- The **pynuml** package (github, pypi, conda) generates truth labels and efficiently preprocesses event HDF5 files into graph datasets.
- The **NuGraph** package (github, pypi, conda) contains a PyTorch implementation of the network architecture, as well as data loaders for the graph dataset.
- A **conda environment** to easily install these packages and all their dependencies is available here.

Summary

- **NuGraph2** is a multi-purpose GNN architecture for reconstructing neutrino interactions in MicroBooNE, DUNE and elsewhere.
  - **Efficiently reject background detector hits**.
  - **Classify detector hits according to particle type**.
  - Future: **vertexing**, **clustering**, **hierarchical graphs**!

Summary

- **NuGraph2** is a multi-purpose GNN architecture for reconstructing neutrino interactions in MicroBooNE, DUNE and elsewhere.
  - **Efficiently reject background detector hits**.
  - **Classify detector hits according to particle type**.
  - Future: **vertexing**, **clustering**, **hierarchical graphs**!
- **NuML** toolkit for standardising the process of producing ML inputs from HEP data for general use.
  - Utilised for MicroBooNE's public data release.
  - Open-source, easy-to-install code packages.