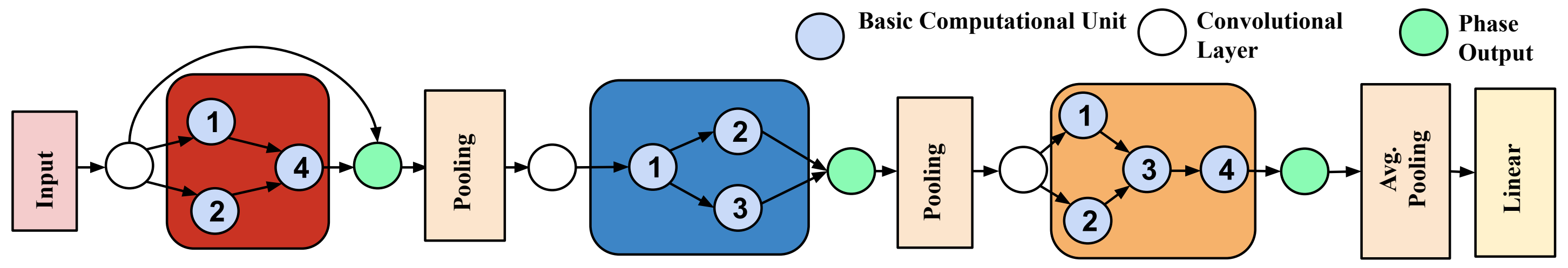
VINARCH: A Visual Analytics Interactive Tool for Neural Network Archaeology

Seoyoung An, Catherine Schuman, Michela Taufer
Department of Electrical Engineering and Computer Science, University of Tennessee, Knoxville, TN, USA

What is NAS and NSGA-Net?

- **Neural Architecture Search (NAS)** is an **architecture design for machine learning** that automates the process of designing network architectures.
- **NSGA-Net** is a **multi-objective genetic algorithm for NAS** to explore the space of potential neural network (NN) architectures with flexibility. By effectively optimizing prediction performance and computational complexity, it reduces error and required computational resources.
- **Network structure** created by NSGA-Net
  - [[0], [0, 0], [1, 1, 0], [1]], [[1], [1, 0], [0, 0, 0], [0]], [[0], [1, 1], [0, 0, 1], [0]]]
    - 0 = No connection between basic computational units
    - 1 = Connection between basic computational units
  - 3 phases
    - Up to 4 nodes per block for each phase



Challenge of NAS

NAS creates hundreds of networks throughout epochs and most of them are discarded except the final output.

We are introducing an aspect of novelty of **keeping** all generated networks using NAS and intelligently **exploring** it for better understanding of NAS process and reproducibility in the future.

VINARCH: Visual Interactive tool for Neural Network Archaeology

We design an **interactive tool for exploring NN data** through **analytics** and **visualization**. For analytics, VINARCH analyzes *common subsequences* within the structures, calculates the *distance between networks*, and summarizes *validation accuracy statistics*. For visualization, VINARCH visualizes the *network structures*, *validation accuracies*, and *network distances*.

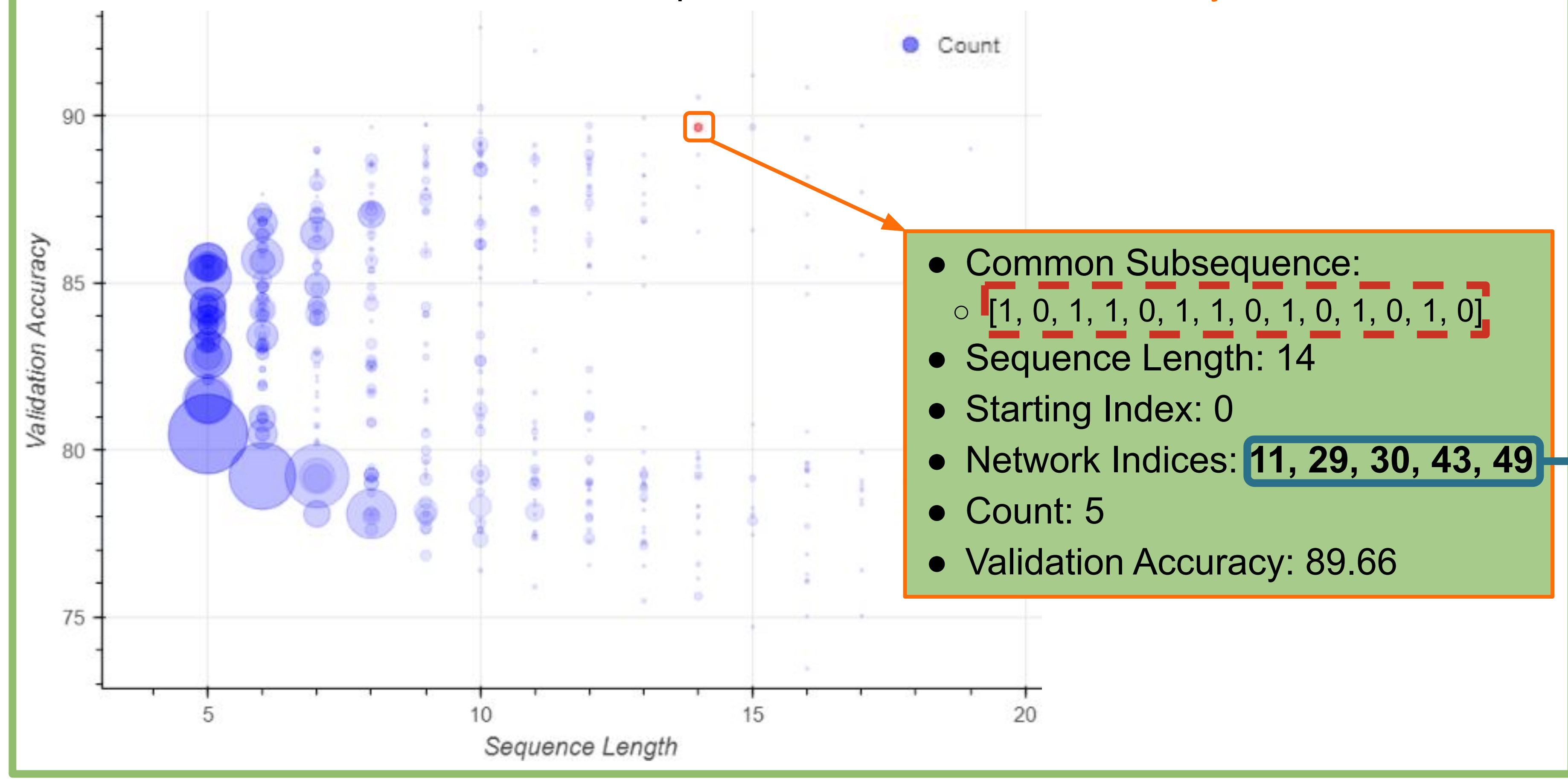
VINARCH Use Case

Analytics - Used Dataset

- **Common subsequences** of 2,167 network structures generated by NSGA-Net
  - Drop parenthesis and compare all networks with each other (Combination pair)
  - When binary matches on the same index, store:
    - *Structure of common subsequence*
    - *Length of the subsequence*: sequence length >= 5 were selected
    - *Starting index*: where does it start in the network
    - *Network index*: which networks have that subsequence
      - Unique networks (total 97) were numbered from 0 to 96
    - *Count*: how many unique networks have that subsequence
    - *Validation accuracy*: mean validation accuracy of all networks with that subsequence

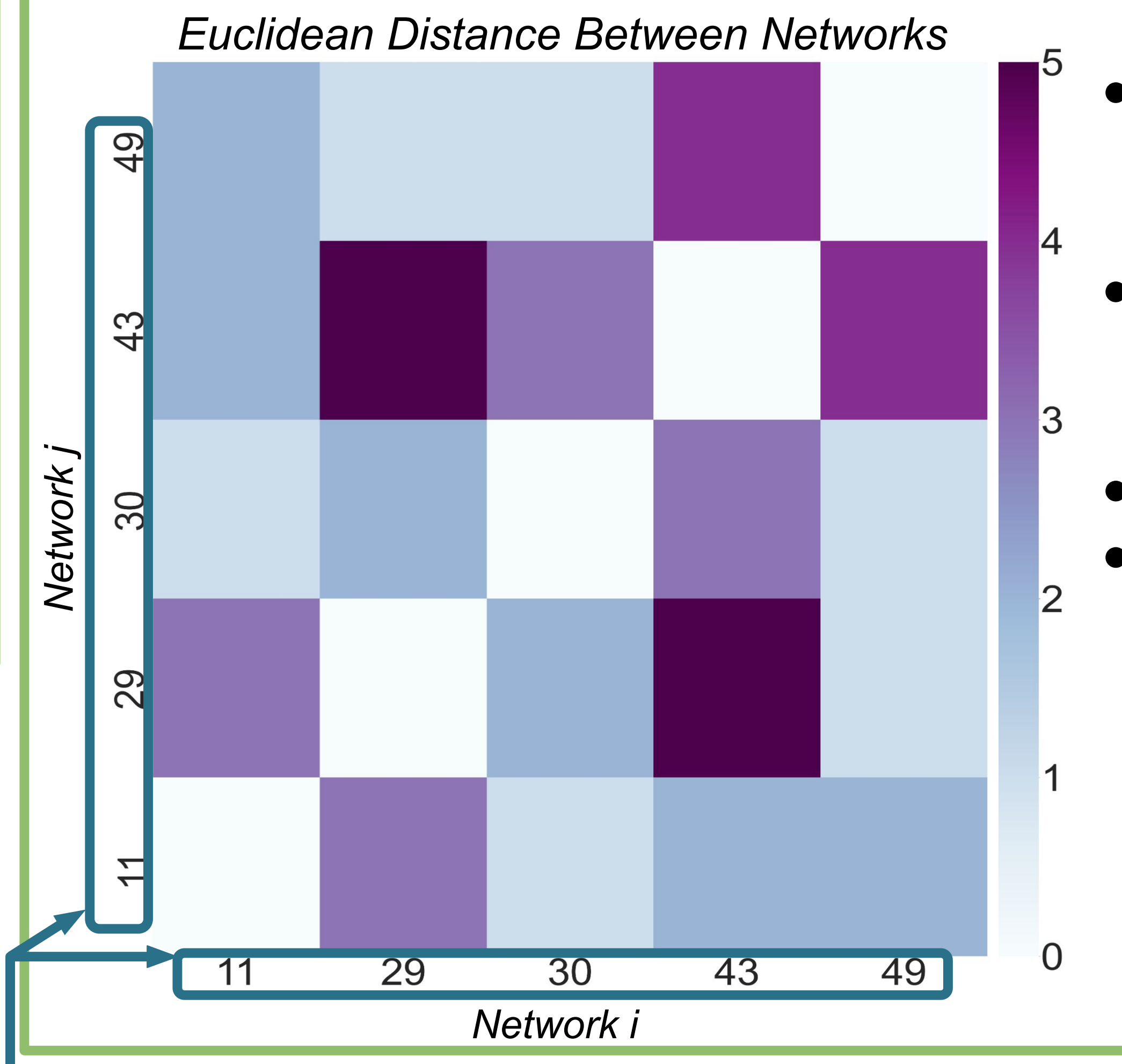
Visualization 1 - Common Network Subsequences

- Utilizing the summary CSV file, VINARCH generates an **interactive scatter plot**.
  - X-axis: sequence length
  - Y-axis: mean validation accuracy
  - Size of the point: number of unique network structures with common subsequences
  - Users can select data from the plot to see the detailed summary



- Common Subsequence:
  - [1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0, 1, 0]
- Sequence Length: 14
- Starting Index: 0
- Network Indices: 11, 29, 30, 43, 49
- Count: 5
- Validation Accuracy: 89.66

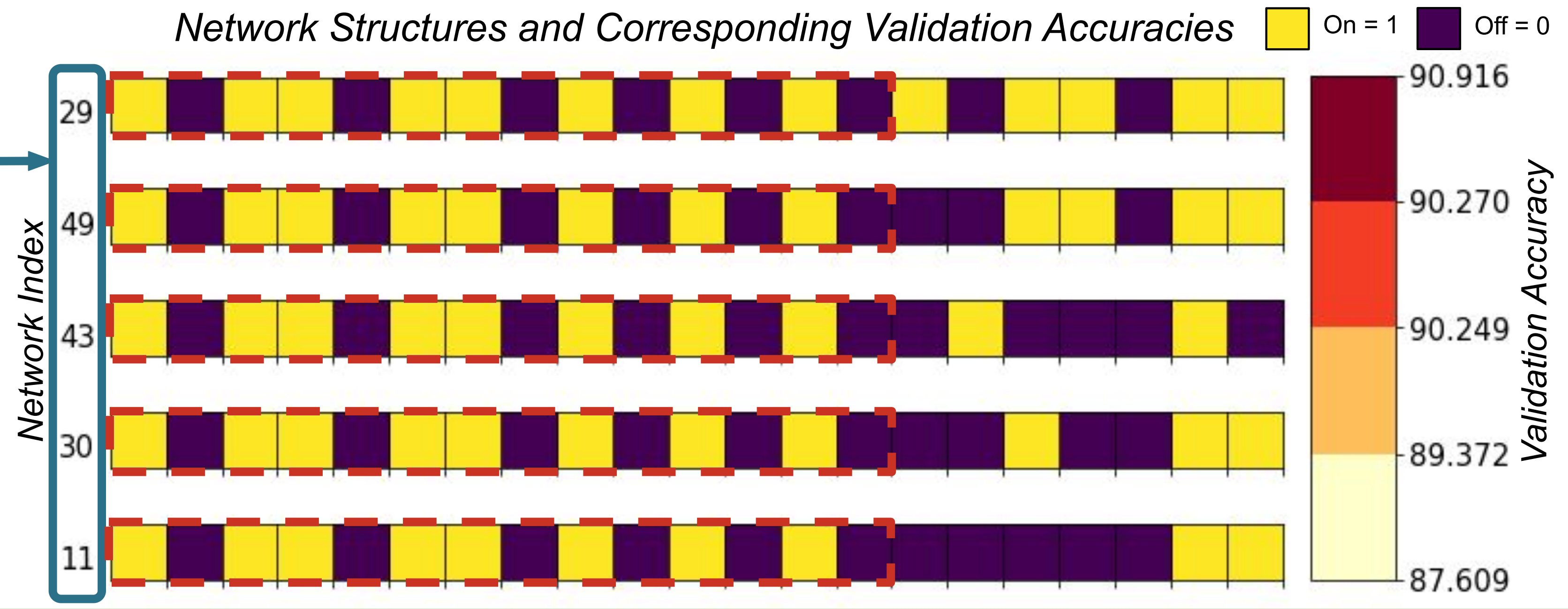
Visualization 2 - Relationship Between Networks



- From the network index of selected data, VINARCH creates a **distance heatmap**.
- 2 choices of calculation
  - Euclidean distance
  - City-block distance
- Axis: all network indices
- Distance depicts the similarities and differences between networks
  - Higher: less similar
  - Lower: more similar

Visualization 3 - Network Structures

- VINARCH visualizes the **full network structures** of selected networks in ascending order of corresponding **validation accuracies**.
- Compare or contrast the full network structures
- Distinguish common subsequence structure



Preliminary Conclusion

- VINARCH can help researchers to explore and analyze the data generated by NAS through multiple visualization plots.
- We hope to use VINARCH to help us with the explainability of produced network structures and answer questions about the decision-making processes.

Future Work

- We hope to expand VINARCH to other NAS algorithms and add more visualization and analytics features such as NetworkX.
- Furthermore, we hope to expand the ability to interact such as drag-and-drop for sorting the networks so researchers can analyze and interact with their data with more flexibility.

THE UNIVERSITY OF
TENNESSEE
KNOXVILLE

References:

- T. Elsken, J. H. Metzen, and F. Hutter, “Neural architecture search: A survey,” The Journal of Machine Learning Research, vol. 20, no. 1, pp.1997–2017, 2019.
- Z. Lu, I. Whalen, V. Boddeti, Y. Dhebar, K. Deb, E. Goodman, and W. Banzhaf, “Nsga-net: neural architecture search using multi-objective genetic algorithm,” in Proceedings of the genetic and evolutionary computation conference, 2019, pp. 419–427.
- A. Keller Rorabaugh, S. Caino-Lores, T. Johnston, and M. Taufer, “Building high-throughput neural architecture search workflows via a decoupled fitness prediction engine,” IEEE Transactions on Parallel and Distributed Systems, vol. 33, no. 11, pp. 2913–2926, 2022.
- G. Channing, R. Patel, A. Rorabaugh, P. Olaya, S. Caino-Lores, C. Schuman, O. Miyashita, F. Tama, and M. Taufer, “Composable Workflow for Accelerating Neural Architecture Search Using In Situ Analytics for Protein Characterization,” in Proceedings of the 52nd International Conference on Parallel Processing (ICPP). Salt Lake City, UT, USA: ACM, 8 2023, pp. 1–10.
- R. Patel, A. K. Rorabaugh, P. Olaya, S. Caino-Lores, G. Channing, C. Schuman, O. Miyashita, F. Tama, and M. Taufer, “A methodology to generate efficient neural networks for classification of scientific datasets,” in 2022 IEEE 18th International Conference on e-Science (e-Science), 2022, pp. 389–390.

Acknowledgements:
This research was supported by the National Science Foundation (NSF) under grant numbers 2223704.