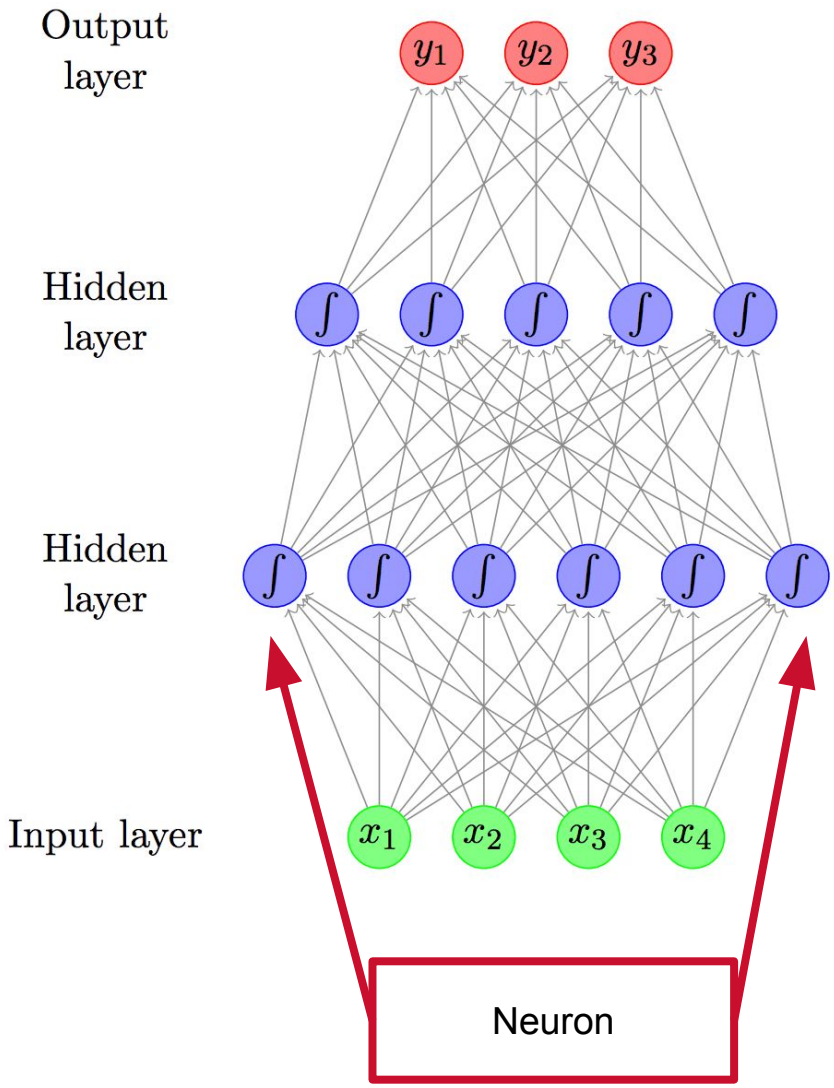
Explainable AI For Source Code Applications

SDMAY25-30

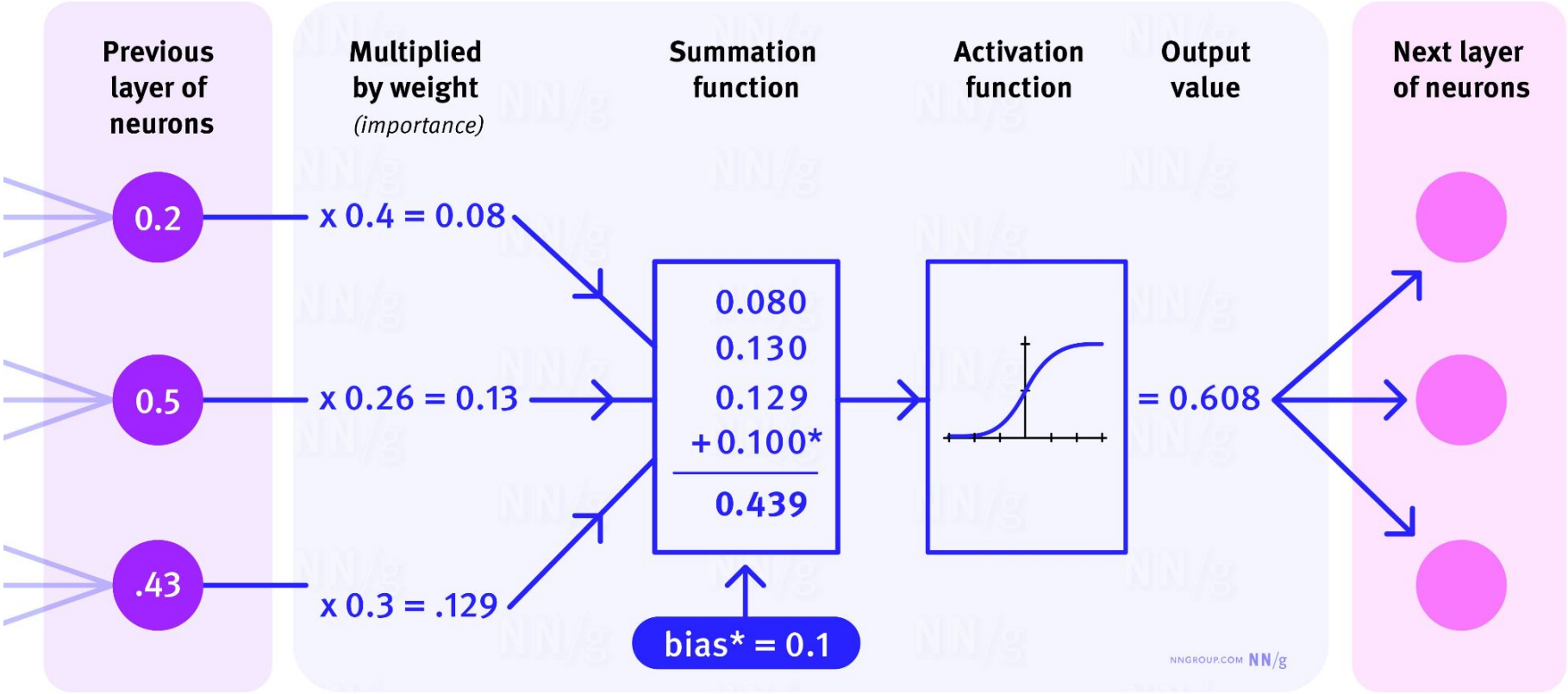
Manjul Balayar, Kellan Bouwman, Sam Frost, Akhilesh Nevatia, Ethan Rogers
Client: Dr. Ali Jannesari/ISU SwAPP Lab
Advisor: Arushi Sharma

IOWA STATE UNIVERSITY

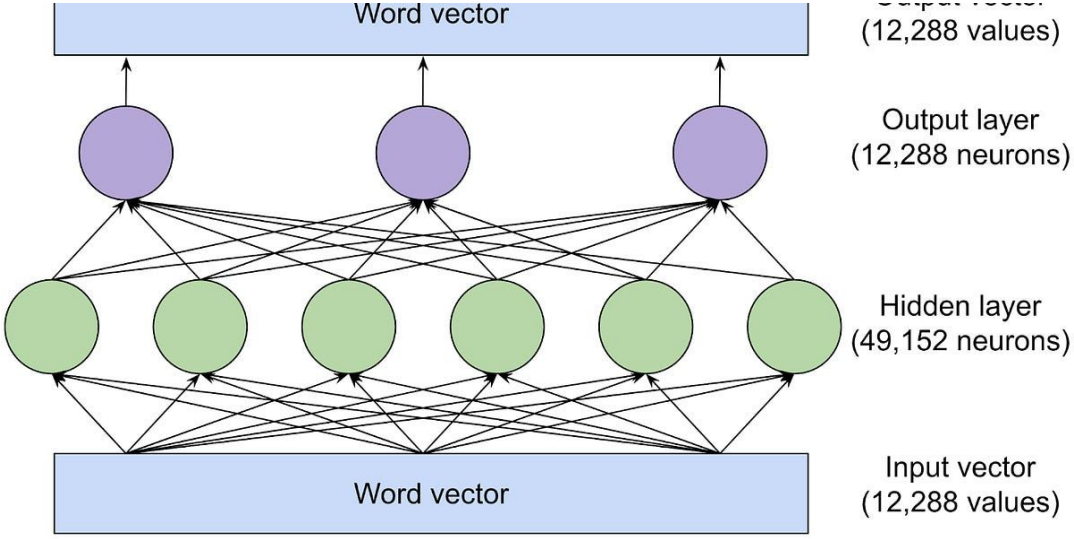
- Machine Learning / Statistical Learning:
  - Machine learning: A broad discipline that focuses on how computers can learn from data
  - Statistical learning: A branch of artificial intelligence that focuses on turning raw data into actionable information. Statistical learning theory is a framework that uses statistical and functional analysis to build models that can make predictions and draw conclusions from data
- Neuron:
  - A collection of a set of inputs, a set of weights, and an activation function.
  - It translates these inputs into a single output.
- Deep Learning:
  - Type of machine learning based on artificial neural networks in which multiple layers of processing are used to extract progressively higher level features from data

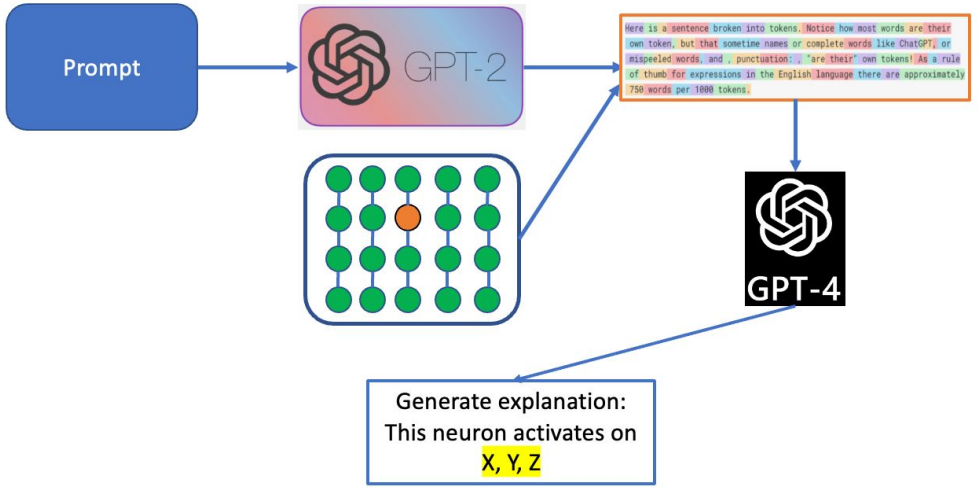


How a Single Artificial Neuron Works



- LLM:
  - Large Language Model
  - Form of Deep learning on NLP (Natural Language Processing)
- Generative AI:
  - A type of AI that uses generative models to create new content, such as text, images, videos, music, and audio
  - Based off an LLM (example: ChatGPT3.5)
- Neuron Activation
  - Non-linear function that we apply over the input data coming to a particular neuron and the output from the function will be sent to the neurons present in the next layer as input
  - A Path of Neurons





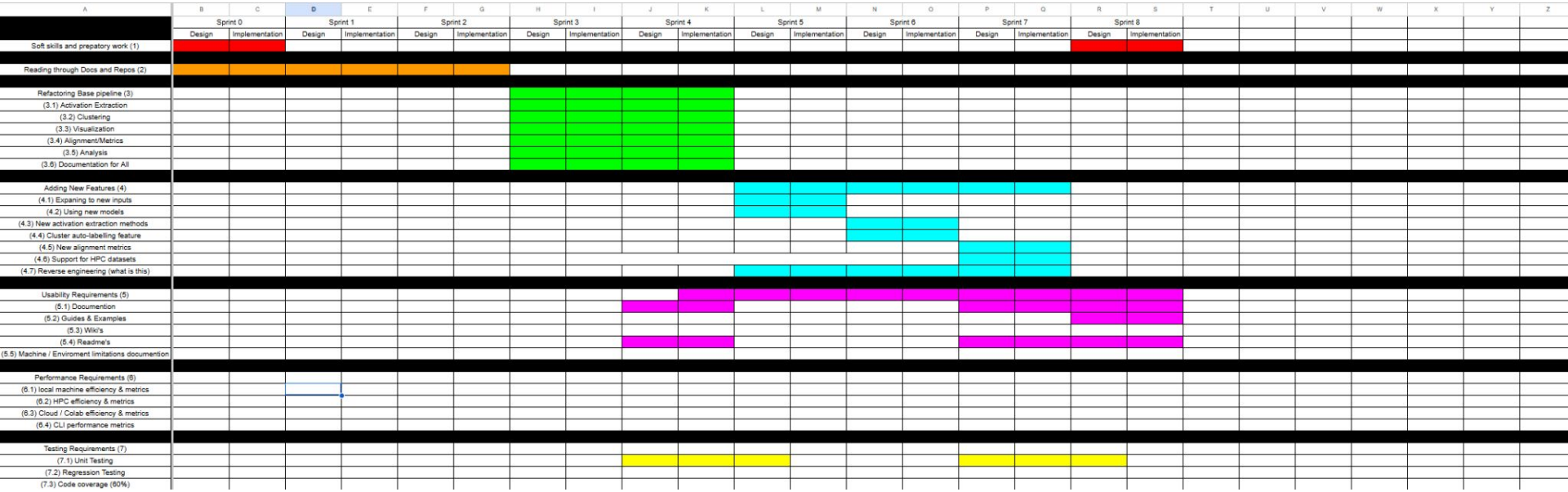
Project Overview

- Client
  - Dr. Ali Jannesari/ISU SwAPP Lab
- Abstract
  - Focus on auto-labeling code datasets using AST tools, regular expressions, and LLM-generated labels.
- Goal
  - Deepen the understanding of LLMs and generative AI by analyzing neuron activations and applying metrics and heuristics. The project will also unify two existing code bases into a flexible and scalable framework that can be deployed seamlessly across Colab, local environments, and HPC clusters.

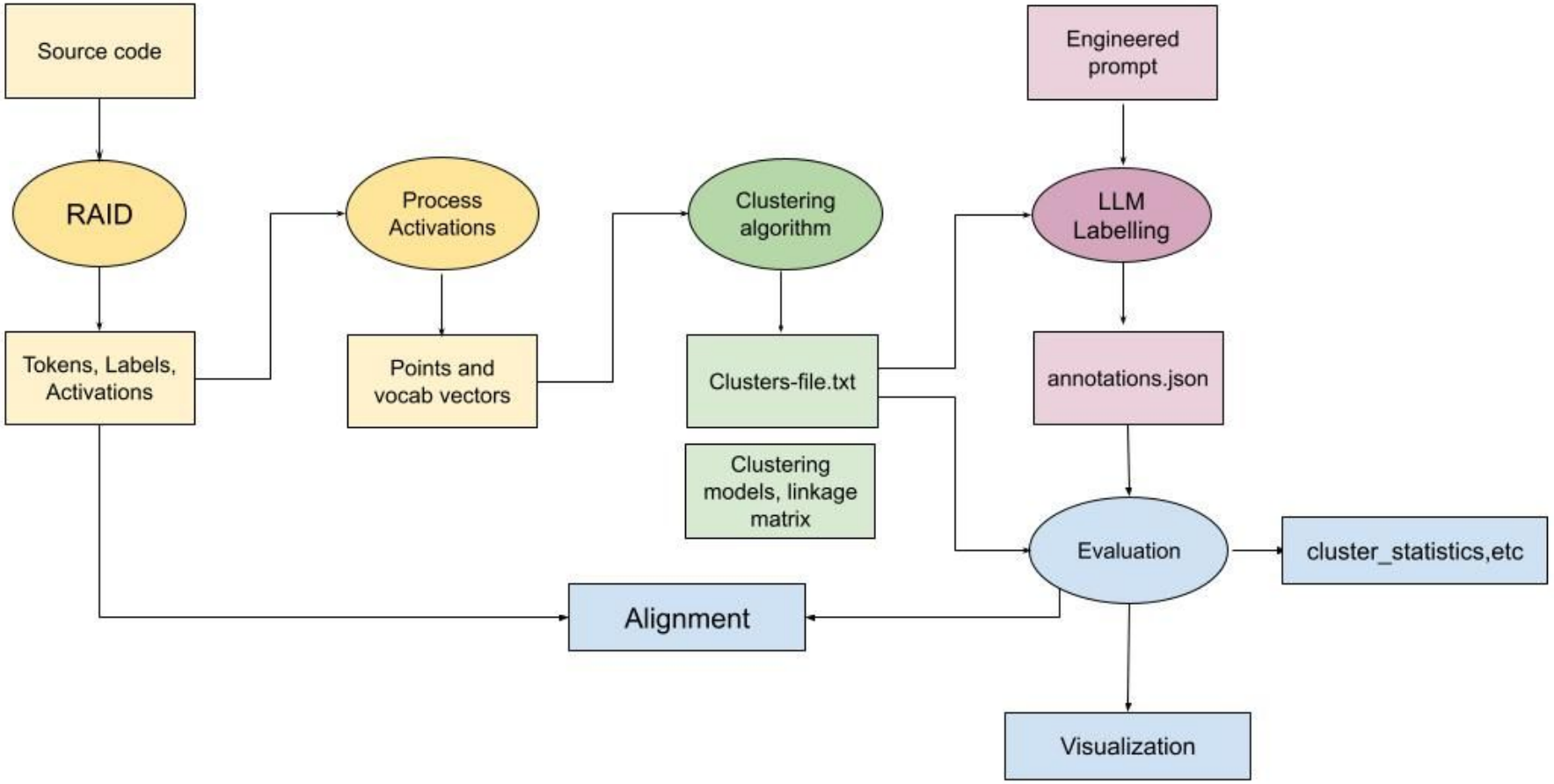
Project Overview - Continued

- Users
  - Researchers
    - ML Engineers / Researchers
    - Prompt Engineers / Researchers
    - Computer Scientists
  - Students
    - Graduate Students
    - Undergraduate Students
  - Industry Professionals
- We aim to include students, researchers, and industry professionals as key users. Our documentation will prioritize accessibility and clarity for all experience levels, while the codebase will remain flexible and scalable to meet both academic and industry needs.

Artifacts - Gantt Chart



Artifacts - Detailed Design



Artifacts - Product Research

Product Services and Design *What is the product?*	**Unique Value Proposition** *What makes this product unique?*	**Product Advantages** *What are the things that provide a leg up?*	**Product Disadvantages** *Where might drawbacks exist?*	**User Pros** *What do users like about the product?*	**User Cons** *What do users NOT like about the product?*
CodeConceptNet Analyzing Encoded Concepts in Transformer Language Models ConceptX is a framework designed to analyze the inner workings of pre-trained language models, specifically how they encode and structure linguistic information. This is achieved by clustering similar representations of words into "encoded concepts" and comparing them to established linguistic categories, termed "human-defined concepts	ConceptX differentiates itself from other language model analysis tools by going beyond simple prediction accuracy. It directly examines the encoded concepts learned by these models, offering a deeper, more nuanced understanding of their linguistic capabilities	Unsupervised Analysis: Unlike "probing" methods that rely on training auxiliary classifiers, ConceptX employs unsupervised clustering, directly analyzing the representations without external task-specific training. Comprehensive Linguistic Analysis: ConceptX utilizes a broad spectrum of human-defined concepts, encompassing lexical, morphological, syntactic, semantic, and psycholinguistic categories, providing a holistic view of the model's linguistic knowledge. Scalability: The framework is designed to be scalable, capable of analyzing a variety of transformer-based language models with differing architectures and training data. This is demonstrated by its application to seven distinct models	Computational Demands: Clustering high-dimensional word representations, especially from large datasets, can be computationally intensive and memory-heavy, potentially limiting the scale of analysis. Limited Explanatory Power for Novel Concepts: While ConceptX excels at aligning encoded concepts with established linguistic categories, it might struggle to explain novel, multifaceted concepts learned by the models that fall outside these predefined categories	Deeper Understanding of Language Models: Users gain insights into the internal representations of language models, going beyond surface-level performance metrics and understanding how different linguistic concepts are encoded. Qualitative and Quantitative Insights: ConceptX provides both qualitative visualizations of encoded concept clusters and quantitative alignment scores, allowing for a comprehensive interpretation of the model's linguistic knowledge	Technical Expertise Required: Utilizing and interpreting the results from ConceptX likely requires a certain level of technical expertise in natural language processing and machine learning. Potential for Misinterpretation: Without proper understanding of the methodology and limitations, there's a risk of misinterpreting the alignment results and drawing inaccurate conclusions about the model's capabilities.

Artifacts - Product Research (cont.)

SD 4910
Product Services and Design Research

Team Identifier: sdmay25-proj030

NeuroX wiki NeuroX is a framework designed to increase the transparency of deep NLP models. It provides a toolkit and other projects for analyzing and interpreting neural networks, primarily for NLP applications.	NeuroX offers a unique approach by focusing on neuron-level interpretation of deep NLP models, going beyond traditional input-feature analysis. This allows for more granular explanations of model predictions and inner workings	NeuroX supports popular transformer models, offers tools for activation extraction, probe training, neuron analysis, and visualization, and simplifies complex NLP model interpretation.	depending on the complexity and size of the model being analyzed, NeuroX could be computationally intensive. This might be a potential drawback. This is not mentioned in the sources and you may want to verify it independently.	Users appreciate NeuroX's ability to extract activations, train probes, extract and visualize neurons, and its user-friendly Python library	Some users of NeuroX would like to see an extension of its features. Specifically, an easier time integrating the library with high performance computing.

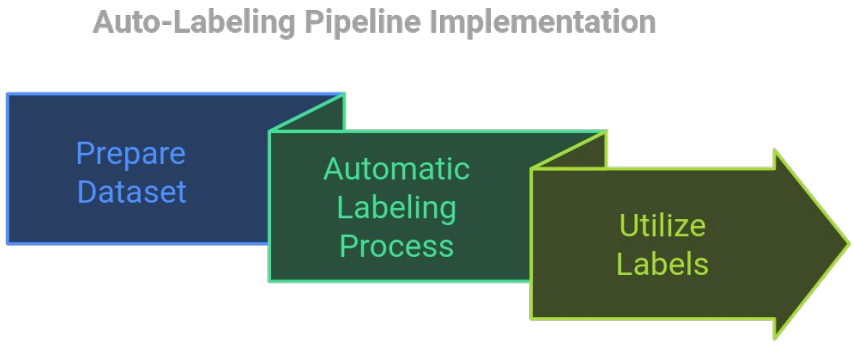
Human - Addressing User Needs

Current Suitability:

- Enhances interpretability of code-trained LLMs for users like software engineers and data scientists.
- Provides a user-friendly web interface for visualizing clustering results and evaluation metrics.
- Empowers users to understand how models perceive code patterns, increasing trust and usability.

Potential Improvements:

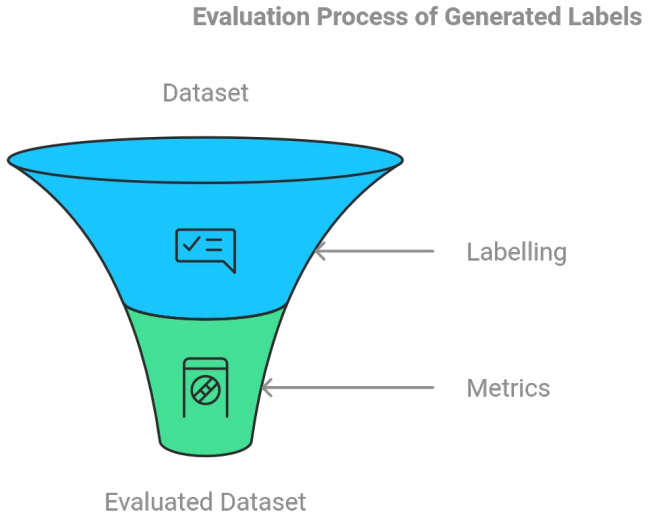
- **User Feedback Loop:**
  - Allow users to flag or label misinterpreted clusters.
  - Improve model alignment with human understanding.
- **Accessibility Features:**
  - Add customizable views and color-blind friendly options.
  - Include detailed tooltips for better user experience.
- **Documentation and Tutorials:**
  - Provide onboarding tutorials for new users.
  - Enhance support through comprehensive documentation.



Economic - Comparison with Existing Solutions

Advantages Over Existing Solutions:

- **Cost Efficiency:**
  - Utilizes open-source tools and models, reducing licensing fees.
- **Customization Potential:**
  - Flexible approach adaptable to various code datasets.
- **Improved Accuracy:**
  - Auto-labeling pipeline offers detailed, domain-relevant tags.
  - Potentially better alignment scores for specific code properties.



Drawbacks and Mitigations:

- **High Initial Setup Complexity:**
  - **Mitigation:** Simplify setup with step-by-step instructions and pre-trained models.
- **Limited Direct Support:**
  - **Mitigation:** Establish community support or partner with university labs.

Technical - Justifying Complexity

Internal Complexity:

- **Advanced Clustering Algorithms:**
  - Demonstrates expertise in machine learning and clustering methods.
- **Auto-Labeling Pipeline Integration:**
  - Combines AST parsing, regular expressions, and LLMs.
  - Requires deep technical knowledge of multiple components.

External Complexity:

- **Interoperability with Diverse Datasets:**
  - Develops preprocessing modules for various code structures.
  - Showcases ability to design scalable, adaptable solutions.
- **Scalable Evaluation Framework:**
  - Establishes robust metrics applicable across multiple datasets.
  - Enables consistent and comparative analysis.