Proposal for MSE Capstone Project

Project Title: A Generalized Web-Based Application for Neural Network Visualization

Student Name: Bennett Wendorf

Faculty Advisor: Dr. Jason Sauppe

Date of Submission: April 16, 2023

A Generalized Web-Based Application for Neural Network Visualization

Objective

The primary goal of this project is to develop a web-based tool that will allow users to visualize neural networks. The tool will be designed to be as general as possible, enabling users to visualize networks from multiple machine learning frameworks, including TensorFlow, Keras, and PyTorch.

Background

Neural networks are notoriously difficult to visualize for students and experienced engineers alike due to their scale and complexity. There are a handful of existing tools for visualizing neural networks, but they are often limited in their scope. For example, TensorBoard is a tool developed by Google for visualizing TensorFlow models. However, TensorBoard is only natively compatible with TensorFlow and Keras models. In addition, many existing solutions are only able to visualize one of simple feed-forward networks, convolutional neural networks, and recurrent neural networks. There is a distinct lack of tools that are able to visualize multiple types in one application. In researching existing tools including TensorBoard, Netron, and ENNUI, it was found that none of the tools were able to animate the visualizations in a way that would allow users to see how the network changes over time as it would during the training process.

Current Project

The current project focuses on developing an application that will allow users to visualize multiple types of neural networks created using several popular machine learning frameworks. The final product must meet the following requirements:

- The application must be able to visualize neural networks created using TensorFlow, Keras, and PyTorch, through the models uploaded by users.
- Users must be able to upload models created using the specified frameworks into the application for visualization.
- At a minimum, the application must be able to visualize the structure of simple feed-forward networks and convolutional neural networks.
- The application must be able to visualize the structure of the network, as well as the weights and biases in the network.

Time permitting, the following features will be added to the application:

- The application will be able to animate the visualization of the network, allowing users to see how the network changes over time.

- Users will be able to change the values of the inputs to the neural network and update the visualization in real time using input sliders and text boxes.
- The application will be able to export the configuration of the network to code to generate the network in the specified frameworks.
- The application will be able to export the network visualization or animation to a series of tikz graphs for use in LaTeX documents like class slides.
- The application will be able to visualize the structure of recurrent neural networks.
- Users will be able to visualize ML.NET and scikit-learn models.

Challenges

The following are primary challenges that are expected to be encountered during the development of the application:

- Supporting multiple different model formats will require a complex model parser and a large amount of research into each of the frameworks.
- Visualization of arbitrary networks is a complex graphical problem that will require either finding or building a complex graph building library.
- Visualizing simple feed-forward networks is a relatively simple problem, but to be competitive, more complex network types must be supported.
- Adding inputs and export of code will require a large amount of research into the frameworks and their APIs as well as well-standardized model representation in the application.

Project Schedule

Phase	**From**	**To**	**Credits**
Develop requirements document and problem analysis	Sept 1, 2023	Sept 31, 2023	1
Produce MVP	Oct 1, 2023	Dec 31, 2023	5
Add stretch features	Jan 1, 2024	Feb 31, 2024	3
Refine and test	Mar 1, 2024	Mar 31, 2024	2
Demonstration and project report	Apr 1, 2024	May 10, 2024	1
		Total:	12

Resources

The student will use his personal computer for development of the application. Application hosting will be determined as needed. Testing data will come from popular demo datasets, such as MNIST and Iris.