

Hyperparameter Tuning for Deep Learning Neural Networks

Students: Martin Macho, Jacob Roman, Jose Lopez

Mentor: Yanzhao Wu, Florida International University

Instructor/Faculty: Masoud Sadjadi, Florida International University

PROBLEM

Machine Learning algorithms are more accessible than ever to more people thanks to advancements in technology and computational power; deep learning neural networks are particularly popular due to their wide range of capabilities and ease of development. However, deep learning algorithms are still frequently slow to train and test due to the high volumes of data they must process. To ensure the optimization of training high accuracy deep learning models, developers must tune hyperparameters such as learning rate and batch size to minimize the loss of the model. This application is an exploration into how model training can be impacted by tuning batch size, and see if there is a way to automatically tune batch size based on other, easier to understand parameters.

SUMMARY

The Hyperparameter Tuning app is intended to use algorithms to tune the batch size of a machine learning automatically based on user priorities and input so as to ensure the most optimized and accurate possible model training. The application could greatly optimize and simplify the process of training machine learning models, as a value that could be difficult to understand and optimize for the average user would be abstracted away.

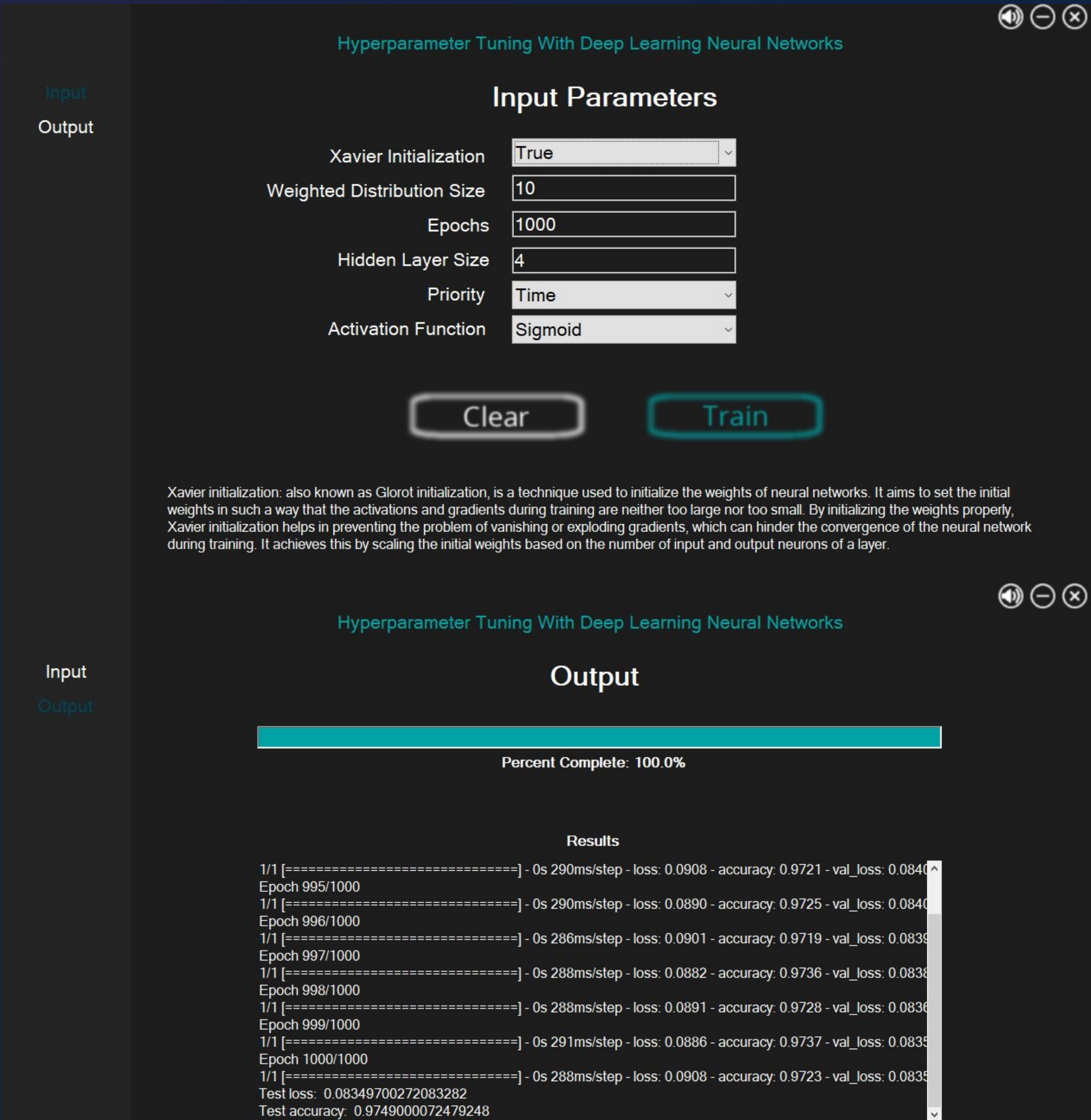
SYSTEM & OBJECT DESIGN

The backend of the app leverages Python, particularly the Keras machine learning framework by TensorFlow. This allows for quickly and easily training a machine learning model to test and utilize the application. The backend is connected to a front end developed in C# with Windows Forms, to ensure a smooth and easy to understand User Experience, connected to the Python backend through Windows Sockets.

VERIFICATION

Visual Studio 2022 was used to test and run the application, allowing for quick and simple debugging, testing, and experimentation at runtime.

IMAGES



CONTRIBUTION/IMPLEMENTATION

- Main Team Lead, ensuring communication between team and product owner and management of meetings and development
- Developed Python backend to manage training of model
- Adjusted front end to meet product specifications

Requirements

The application currently only runs of Windows OS.

- Python
- Visual Studio 2022
- .NET Framework

CURRENT SYSTEM

The current system utilizes the Keras framework to train a sequential neural network model on the MNIST dataset (future iterations would allow for different models and datasets) with a batch size dependent on whether the user wants to prioritize time or accuracy of training and how many epochs the user would like to train the model for. The frontend is a Windows Form, allowing for ease of access for any user running Windows OS.

ACKNOWLEDGEMENT

The material presented in this poster is based upon the work supported by XXXXX (*name of the project sponsor: federal, state or local government, or corporate partners, where applicable*). We/I thank XXXX for his/her/their assistance and mentorship that I/we received throughout the senior design project.