**Performance and Resource Trade-Off Analysis of Neural Networks and Support Vector Machines on Pneumonia Detection – Project Proposal**

Mary Klimasewiski, Leonardo Pinheiro, and Sedat Touray

**1. Introduction**

The aim of this project is to evaluate the performance and computational efficiency of two machine learning approaches—Feed-Forward Neural Networks (FFNN) and Support Vector Machines (SVM)—on the RSNA Pneumonia Detection Challenge 2018 dataset. By comparing the models' runtime, resource utilization (CPU/GPU), and prediction accuracy, we will provide insights into trade-offs between computational efficiency and model performance in medical imaging tasks.

**2. Goals**

* Compare the classification performance of FFNN and SVM using metrics such as accuracy, precision, recall, and F1-score.
* Evaluate the computational costs associated with each model, including training/inference time and resource usage (CPU/GPU).
* Identify scenarios where one model might be preferred over the other based on performance-resource trade-offs.

**3. Actionable Steps**

Step 1: Data Acquisition and Preprocessing

* Download the RSNA Pneumonia Detection Challenge dataset from the official source.
* Perform preprocessing:

Handle missing values and inconsistencies.

Resize and normalize images for model compatibility.

Split the dataset into training, validation, and test sets.

Encode target labels and convert data to formats compatible with SVM and FFNN.

Step 2: Exploratory Data Analysis (EDA)

* Visualize the dataset to understand class distributions and data quality.
* Compute basic statistics on image intensities and feature representations.
* Identify potential challenges like class imbalance.

Step 3: Model Implementation

1. Feed-Forward Neural Network

* Implement a multi-layer FFNN using TensorFlow or PyTorch.
* Optimize hyperparameters (e.g., learning rate, number of layers, and units).
* Train and evaluate the model.

2. Support Vector Machine

* Extract features from the images using dimensionality reduction techniques.
* Implement and train an SVM classifier using a library like scikit-learn.
* Tune hyperparameters such as the kernel type and regularization parameter.

Step 4: Performance Evaluation

* Compute evaluation metrics (accuracy, precision, recall, F1-score) for both models on the test set.
* Analyze training/inference runtimes for FFNN and SVM.
* Record resource usage, including CPU/GPU utilization, using tools like NVIDIA System Management Interface (nvidia-smi).

Step 5: Trade-Off Analysis

* Compare the models based on performance and computational costs.
* Discuss the impact of hyperparameters, dataset size, and resource constraints on the results.
* Identify trade-offs and recommend the optimal approach based on the use case.

Step 6: Documentation and Presentation

* Document the methodology, results, and key findings in a report.
* Create a presentation summarizing the project goals, implementation, results, and recommendations.

**4. Expected Outcomes**

* Quantitative comparison of FFNN and SVM performance in pneumonia detection.
* Detailed analysis of computational costs associated with each model.
* Practical recommendations for selecting machine learning approaches in resource-constrained environments.

**5. Timeline**

|  |  |
| --- | --- |
| Course Week | Task |
| 3 | Data acquisition and preprocessing, Exploratory data analysis |
| 4 | Implement FFNN and SVM, Train and evaluate models |
| 5 | Analyze trade-offs and resource usage, Document findings and prepare the presentation |
| 6 | Prepare final report and presentation |

**6. Tools and Resources**

* Libraries: TensorFlow, PyTorch, scikit-learn, NumPy, pandas, matplotlib.
* Hardware: Access to GPU-enabled systems for training models.
* Dataset: RSNA Pneumonia Detection Challenge 2018 (<https://www.rsna.org/rsnai/ai-image-challenge/rsna-pneumonia-detection-challenge-2018>)

**7. Conclusion**

This project provides a comprehensive analysis of the trade-offs between performance and computational efficiency for FFNNs and SVMs in a medical imaging context. Insights gained will aid in selecting appropriate models for real-world applications with varying resource constraints.