COVID-19 Case Prediction Using Deep Neural Networks

1. Introduction

This section provides context and sets the stage for the report.

- **Objective**:
  - Example: "The objective of this project is to predict the percentage of new COVID-19 positive cases based on survey data collected from Delphi's COVID-19 surveys using deep neural networks (DNN). The model predicts future cases based on symptoms, testing, and various behavioral and social factors."
- **Dataset Description**:
  - Example: "The dataset is derived from Delphi's COVID-19 surveys, which include a variety of features such as symptoms, testing results, social distancing measures, and mental health data for different states in the U.S."
- **Modeling Approach**:
  - Example: "A deep neural network was trained using PyTorch to solve this regression problem. The model is evaluated using Mean Squared Error (MSE) to determine its prediction accuracy."

2. Methodology

This section is crucial, as it explains the step-by-step process followed to train the model. Each part of the methodology should be backed by a clear rationale.

2.1 Data Preprocessing

Data Cleaning:

Example: "The dataset did not contain any missing values. Therefore, no imputation was necessary. However, we removed the id column as it is not relevant for prediction."

Feature Selection:

Example: "All features except for the id and tested_positive.2 columns were selected as input features. The tested_positive.2 column represents the target variable, which is the percentage of positive cases."

Train/Test Split:

Example: "The dataset was split into training (80%) and validation (20%) sets using scikit-learn’s train_test_split function to evaluate the model’s performance during training."

Feature Scaling:

Example: "Features were standardized using scikit-learn's StandardScaler to ensure that each feature had zero mean and unit variance, improving the training stability for the deep neural network."

2.2 Network Structure

- **Model Architecture**: Example:
  - Input layer: "The input layer has 94 neurons, corresponding to the 94 features in the dataset."
  - Hidden layers: "The network has three hidden layers with 128, 64, and 32 neurons, respectively, all using ReLU activation to introduce non-linearity."
  - Output layer: "The output layer has one neuron, which provides the predicted percentage of new positive COVID-19 cases."

Activation Function:

- Example: "ReLU was used in the hidden layers to introduce non-linearity to the model and help it learn complex patterns in the data."

2.3 Training Procedure

- **Loss Function**: Example: "Since this is a regression problem, the Mean Squared Error (MSE) was chosen as the loss function. MSE measures the average squared difference between predicted and actual values."

- **Optimizer**:
  - Example: "The Adam optimizer was used due to its ability to efficiently handle large datasets with sparse gradients. The learning rate was set to 0.001."
- **Training Process**:

    - Example: "The model was trained for 100 epochs using a batch size of 32. Training and validation losses were monitored throughout the process to prevent overfitting."

2.4 Hyperparameters

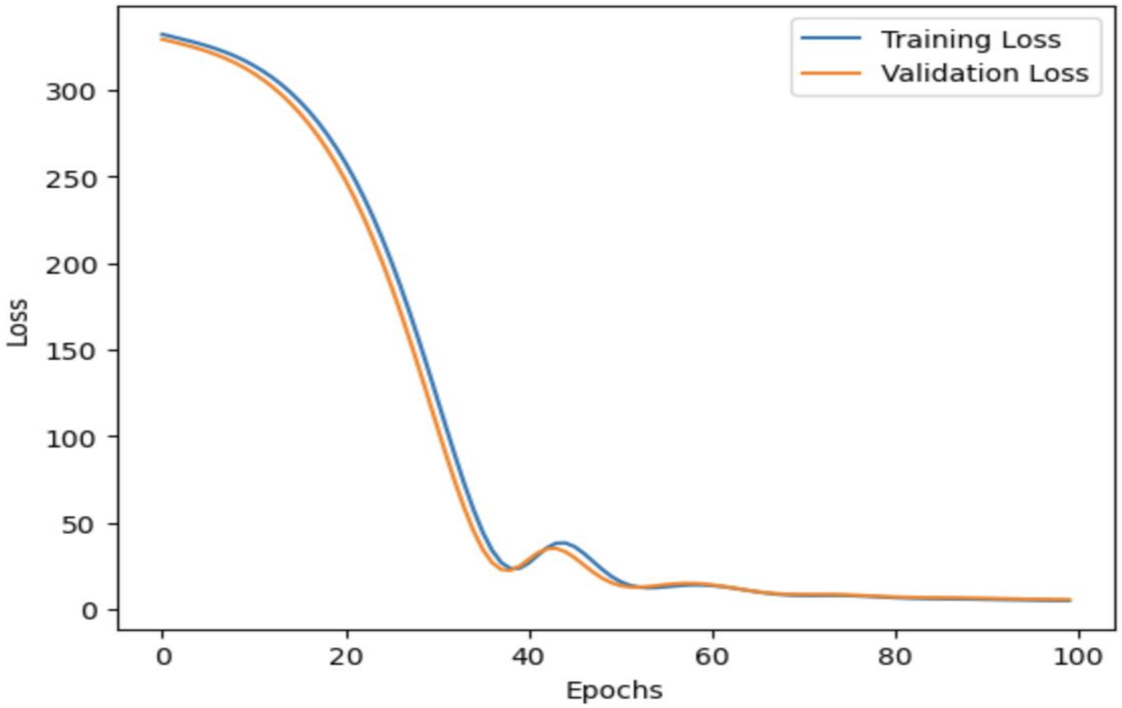
This part summarizes the hyperparameters used in training:

- **Learning Rate**: 0.001
- **Epochs**: 100
- **Batch Size**: 32
- **Optimizer**: Adam

3. Empirical Results and Evaluation

3.1 Training and Validation Loss

- **Loss Curves**: A graph of the training and validation losses over epochs.



    - **Discussion of Results**:
      Example: "The training loss steadily decreased over time, showing that the model was learning the patterns in the training data. The validation loss also decreased but started to plateau after 50 epochs, indicating that the model had reached a stable point."

3.2 Model Evaluation on Validation Data

- **Mean Squared Error (MSE)**:
    - Example: "The final validation Mean Squared Error (MSE) was **5.41**, indicating the average squared difference between the predicted and actual values in the validation set."

4. Conclusion

This section summarizes the insights gained from the project.

    - **Model Performance**:
      Example: "The model was able to predict the percentage of positive cases with a reasonable level of accuracy. However, there is room for improvement, particularly in handling complex relationships between features and improving generalization on unseen data."
    - **Challenges**:
      Example: "One challenge was selecting the right model architecture and tuning hyperparameters to minimize the validation loss without overfitting."
    - **Improvements and Future Work**:
    - Example: "Future work could focus on using more advanced models such as LSTMs for time-series forecasting or implementing dropout to reduce overfitting. Additionally, experimenting with larger datasets could further improve model performance."