TRIBHUVAN UNIVERSITY

INSTITUTE OF ENGINEERING

PULCHOWK CAMPUS

THESIS NO: 079/MSPSE/015

DEEP NEURAL NETWORK-BASED OPTIMAL POWER FLOW AND SECURITY ASSESSMENT UNDER CRITICAL CONTINGENCIES

By

PANAS BHATTARAI

A THESIS SUBMITTED TO THE DEPARTMENT OF ELECTRICAL ENGINEERING IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF MASTER OF SCIENCE IN POWER SYSTEM ENGINEERING

DEPARTMENT OF ELECTRICAL ENGINEERING

LALITPUR, NEPAL

APRIL 2025

त्रिभुवन विश्वविद्यालय
TRIBHUVAN UNIVERSITY
इन्जिनियरिङ्ग अध्ययन संस्थान
INSTITUTE OF ENGINEERING
पुल्चोक क्याम्पस
PULCHOWK CAMPUS
DEPARTMENT OF ELECTRICAL ENGINEERING
Pulchowk, Lalitpur

Accredited by University Grants Commission (UGC) Nepal 2020

Institute of Engineering Electrical Engineering Department Pulchowk Campus

CERTIFICATE OF APPROVAL

The undersigned certify that they have read and recommended to the Institute of Engineering for acceptance, a dissertation entitled **"Deep Neural Network-Based Optimal Power Flow and Security Assessment Under Critical Contingencies"**, submitted by **Panas Bhattarai** in partial fulfillment of the requirement for the award of the degree of **Master of Science in Power System Engineering**.

Assoc. Prof. Mahammad Badrudoza
Department of Electrical Engineering
Pulchowk Campus, Lalitpur
(Supervisor)

Asst. Prof. Anil Kumar Panjiyar
Deputy Head of Department
Department of Electrical Engineering
Pulchowk Campus, Lalitpur
(Supervisor)

Prof. Dr. Bhupendra Bimal Chhetri
Department of Electrical and Electronics Engineering
Kathmandu University, Dhulikhel, Kavre
(External Examiner)

Asst. Prof. Dr. Bishal Silwal
Program Coordinator
MSc in Power System Engineering
Department of Electrical Engineering
Pulchowk Campus, Lalitpur

Assoc. Prof. Dr. Basanta K. Gautam
Head of Department
Department of Electrical Engineering
Pulchowk Campus, Lalitpur

April 2025

COPYRIGHT

ACKNOWLEDGEMENT

I would like to express my deepest gratitude to my supervisors Associate Prof. Mahammad Badrudoza and Assistant Prof. Anil Panjiyar for their invaluable guidance, insightful comments, and continuous support and encouragement throughout the progress of this research work. Furthermore, I am immensely thankful to the whole team of Department of Electrical Engineering, Pulchowk Campus, Institute of Engineering for providing the necessary resources and environment conducive to academic growth and learning. Lastly, I would like to extend my appreciation to my friends and family for their unwavering encouragement and understanding during this challenging yet rewarding endeavor. Their moral support has been a constant source of strength throughout my academic pursuits. This thesis would not have been possible without the contributions and support of all those mentioned above. Thank you all for believing in me and for being part of this journey.

ABSTRACT

Modern power systems face growing operational challenges due to escalating complexity, demanding sophisticated tools for efficient management and security evaluation. This thesis introduces a data-driven framework leveraging Deep Neural Networks (DNNs) to address the Optimal Power Flow (OPF) problem and assess grid security during critical contingencies. An extensive dataset is generated by simulating diverse load profiles and contingency scenarios, forming the basis for model training and validation. High-risk contingencies—specifically the three most impactful transmission line outages and two severe generator outages—are systematically identified by evaluating their potential to induce operational violations.

Two distinct DNN architectures are designed: the first employs five networks trained on bus active and reactive power demand to forecast operational parameters (generator outputs, voltage magnitudes, bus angles, and line flows) under standard conditions. The second architecture integrates contingency status indicators (line/generator outages) as supplementary inputs, enabling robust prediction of post-contingency OPF solutions and real-time security evaluations. Compared to conventional iterative solvers, the DNN framework significantly reduces computational latency, supporting rapid decision-making while improving grid resilience.

Validation results confirm the models' accuracy in delivering reliable OPF solutions under normal operations and identifying security risks during contingencies. This research underscores the transformative role of machine learning in modernizing power system optimization and contingency preparedness, offering a pathway toward adaptive, intelligent grid management.

Keywords

Power Flow, Load Flow, Optimal Power Flow, Contingency Analysis, Security Assessment

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

AE	ABSOLUTE ERROR
AC-OPF	AC-OPTIMAL POWER FLOW
ANN	ARTIFICIAL NEURAL NETWORK
DNN	DEEP NEURAL NETWORK
LP	LINEAR PROGRAMMING
MAE	MEAN ABSOLUTE ERROR
MSE	MEAN SQUARED ERROR
OPF	OPTIMAL POWER FLOW
PF	POWER FLOW
QP	QUADRATIC PROGRAMMING
ReLU	RECTIFIED LINEAR UNIT
RMSE	ROOT MEAN SQUARED ERROR
SCOPF	SECURITY-CONSTRAINED OPTIMAL POWER FLOW

CHAPTER ONE: INTRODUCTION

1.1. Background and Motivation

The increasing complexity of modern power systems, driven by growing demand and integration of renewable energy sources, necessitates advanced techniques for ensuring optimal power flow (OPF) and system security. Traditional OPF methods, such as interior-point algorithms and Newton-Raphson techniques, efficiently solve optimization problems but often become computationally expensive when dealing with large-scale networks and real-time security assessments. Furthermore, power system contingencies, such as transmission line or generator outages, can significantly impact system stability, making it essential to develop fast and reliable methods for security assessment under critical contingencies.

Deep learning has emerged as a powerful tool for modeling complex, high-dimensional systems, offering real-time approximations of computationally intensive problems. In recent years, Deep Neural Networks (DNNs) have shown promising results in power system applications, including load forecasting, fault detection, and OPF solutions. By leveraging data-driven learning, DNNs can approximate the nonlinear relationships between power system variables with high accuracy while significantly reducing computation time. However, limited research has been conducted on integrating deep learning into OPF solutions that account for critical contingencies.

This study proposes a novel DNN-based framework for OPF and security assessment under critical contingencies. The approach involves generating a large dataset of OPF solutions using the MATPOWER toolbox, incorporating various system loading conditions. The critical line and generator outages that result in the most severe system violations are identified, and separate DNN models are trained to predict key power system parameters under both normal and contingency conditions. Specifically, two sets of DNN architectures are developed: A baseline DNN model that predicts optimal active power generation and optimal reactive power generation, voltages on buses, angles on buses, and power flow on lines based on load conditions. An extended DNN model that includes contingency conditions (critical line status and generator outages) as additional inputs, which enables real-time security assessment. The proposed method enhances computational efficiency by bypassing iterative OPF solvers while maintaining high accuracy. It provides a rapid assessment of system security, enabling grid operators to make proactive decisions for maintaining stability under critical contingencies. The effectiveness of the model is validated using extensive

simulations, demonstrating its potential for real-time power system operation and contingency management.

1.2. Problem Statement

Traditional OPF solvers, such as interior-point methods and Newton-Raphson techniques, provide accurate results but suffer from high computational costs, making them unsuitable for real-time power system operations, especially during critical contingencies such as transmission line and generator outages.

Existing contingency analysis methods rely on iterative power flow computations, which become computationally expensive for large-scale power networks. As power grids continue to expand, there is an urgent need for fast and scalable solutions that can accurately predict system behavior under both normal and contingency conditions without excessive computation time.

Machine learning, particularly Deep Neural Networks (DNNs), offers a promising alternative by approximating OPF solutions through data-driven learning, significantly reducing computational burdens. However, most existing research has focused on using DNNs for OPF under normal conditions, with limited attention to incorporating critical contingencies into the learning framework. The lack of real-time contingency-aware OPF models hinders proactive decision-making for grid operators, potentially compromising system security and stability.

This thesis addresses this gap by proposing a DNN-based OPF framework that integrates contingency analysis, allowing rapid security assessments under critical outages. The approach aims to:

1. Develop a baseline DNN model for OPF prediction under normal conditions.
2. Extend the model to incorporate contingency conditions, enabling real-time security assessment.
3. Validate the model using extensive simulations on standard test systems to demonstrate its effectiveness.

By bridging the gap between traditional OPF solvers and real-time machine learning approaches, this research aims to enhance the resilience, efficiency, and adaptability of modern power systems.

1.3. Main Objective

- To design a data-driven Deep Neural Network (DNN) framework capable of solving the AC-Optimal Power Flow Problem (AC-OPF) and evaluating power system security under high-risk contingency scenarios, aiming to enhance real-time operational reliability.

1.4. Specific Objectives

- To generate a comprehensive dataset of OPF solution under varying system loading conditions using MATPOWER toolbox.
- To identify critical contingencies by performing line outages and generator outages and selecting the most critical cases based on system violations
- To develop and train two sets of DNN models:
    1. A baseline DNN that predicts OPF variables under normal operating conditions.
    2. A contingency-aware DNN that incorporates the status of critical outages and provides OPF solutions under contingency conditions.
- To evaluate the performance of the proposed DNN models in terms of prediction accuracy, computational efficiency, and security assessment capability.
- To compare the DNN-based approach with traditional OPF solvers in terms of computational time and accuracy.

1.5. Scope of the Study

The research focuses on solving the OPF problem and performing security assessment using DNNs under both normal and contingency conditions. The study is conducted using the IEEE 30-bus test system, a benchmark power system model widely used for OPF and security analysis. The scope includes:

1. Optimal Power Flow (OPF) computation using traditional solvers and DNN-based approximations.
2. Contingency analysis, focusing on the most critical transmission line and generator outages.
3. Performance evaluation of the DNN models in terms of prediction accuracy and computational speed.
4. Comparison of DNN-based OPF solutions with those obtained from traditional solvers (MATPOWER).

1.6. Limitation of the Study

1. The study considers load variations between 20% and 100% of the base load condition. While this range covers typical operational scenarios, extreme low-load conditions (below 20%) or unexpected load fluctuations outside this range are not analyzed.
2. The power factor of loads is varied randomly between 0.75 lagging and unity, which introduces some variability.

1.7. Thesis Structure

The thesis is organized into five chapters, each covering different aspects of the study:

- Chapter 1: Introduction – Introduces the background, motivation, research objectives, research scope, research limitation and thesis structure.
- Chapter 2: Literature Review – Reviews existing research on OPF, contingency analysis, and machine learning application in power systems.
- Chapter 3: Mathematical Formulation of OPF – Provides the mathematical representation of OPF and discusses traditional optimization methods.
- Chapter 4: Deep Neural Networks (DNNs) – Explains the theory behind DNNs, including architecture, loss functions, activation functions, and training methodologies.
- Chapter 5: Methodology – Describes the dataset generation process, contingency simulation, and development of the DNN models.
- Chapter 6: Results and Discussion – Presents the experimental results, compares DNN predictions with traditional OPF solvers, and discusses the findings.
- Chapter 7: Conclusion and Future Work – Summarizes key contributions, discusses limitations, and suggests future research directions.

CHAPTER TWO: LITERATURE REVIEW

2.1. Introduction to Optimal Power Flow

Optimal Power Flow (OPF) is a fundamental problem in power system analysis, aiming to determine the optimal operating conditions of a power system while satisfying technical constraints such as generation limits, voltage stability, and network security. The OPF problem is traditionally solved using mathematical optimization techniques, including Newton-Raphson methods, linear programming, and interior-point methods. However, these approaches become computationally expensive for large-scale power systems, especially when considering security constraints under contingency conditions (Zimmerman et al., 2010).

2.2. Traditional Approaches to OPF and Contingency Analysis

The classical methods for solving OPF include:

Newton-Raphson Method: A widely used iterative method that solves the nonlinear power flow equations with high accuracy. However, it suffers from convergence issues when applied to large-scale networks (Saadat, 1999).

Interior-Point Methods: Efficient for large-scale power networks but computationally expensive for real-time security assessments (Phan & Kalagnanam, 2013).

Linear Programming (LP) and Quadratic Programming (QP): Used for specific OPF formulations but limited by the need for linearization, which reduces accuracy (Qiu et al., 2009).

For contingency analysis, traditional approaches rely on iterative simulations, such as

N-1 Contingency Analysis: Evaluates the impact of single-line or generator outages and determines if the system remains stable.

Security-Constrained OPF (SCOPF): Extends OPF to include contingency constraints but increases computational complexity.

These traditional methods, while accurate, are unsuitable for real-time applications due to high computational costs.

2.3. Deep Learning in Power System Optimization

The recent advancements in deep learning have led to new data-driven approaches for power system optimization, particularly using Deep Neural Networks (DNNs). Machine learning techniques have been successfully applied in power system applications, including:

Load Forecasting: Predicting future power demand using historical data (Samek et al., 2021).

Fault Detection: Identifying and diagnosing faults in power networks.

Optimal Power Flow (DeepOPF): Approximating OPF solutions using neural networks (Pan, 2021).

DNNs offer an efficient alternative to traditional OPF solvers by learning nonlinear relationships between system variables and providing near-instantaneous predictions.

2.4. Applications of Deep Neural Networks for OPF and Contingency Analysis

Several studies have explored the use of DNNs for OPF and contingency analysis:

- DeepOPF (Pan, 2021): A DNN-based model trained on OPF solutions, achieving high accuracy with significantly reduced computation time.
- Constraint-Guided DNN (Lotfi & Pirnia, 2022): Incorporates system constraints into the training process, improving reliability in real-world power systems.
- ANN-Based Contingency Analysis (Schafer et al., 2018): Uses neural networks to assess system stability under contingencies.

These studies demonstrate that DNN-based approaches can effectively approximate OPF solutions while enabling real-time security assessments.

CHAPTER THREE: MATHEMATICAL FORMULATION OF OPF

Mathematically, the optimal power flow problem can be formulated as follows.

$$\text{minimize } f(x) \tag{1}$$

$$subject\ to$$

$$g(x) = 0 \tag{2}$$

$$h(x) \leq 0 \tag{3}$$

$$x_{min} \leq x \leq x_{max} \tag{4}$$

The cost function $f(x)$ is the mathematical model of the polynomial cost of generator injections. The equality constraints $g(x)$ is the power balance equations. The inequality constraints $h(x)$ are the limits on the branch power flow. x_{min} and x_{max} bounds include reference bus angles, voltage magnitudes (for AC) and generator power limits from capability curves.

3.1. Standard AC-Optimal Power Flow

The decision variables x for the standard AC-OPF problem consists of the $n_b \times 1$ vectors of bus voltage angles Θ and bus voltage magnitudes V_m and the $n_g \times 1$ vectors of generator active power injections and generator reactive power injections P_g and Q_g.

$$x = \begin{bmatrix} \Theta \\ V_m \\ P_g \\ Q_g \end{bmatrix} \tag{5}$$

The cost function $f(x)$ in (1) is obtained from summing of individual polynomial cost function of real power injections for each generator.

$$f(P_g) = \sum_{i=1}^{n_g} A_i P_{gi}^2 + B_i P_{gi} \tag{6}$$

The terms $A_i\ and\ B_i$ are cost coefficients in quadratic fuel cost function of the generators. The constraints for representing equality in (2) are the full set of $2 \times n_b$ nonlinear real/active power balance equations and reactive power balance equations from (7) and (8).

$$g_p(\Theta, V_m, P_g) = P_{bus}(\Theta, V_m) + P_d - C_g P_g = 0 \tag{7}$$

$$g_p(\Theta, V_m, Q_g) = Q_{bus}(\Theta, V_m) + Q_d - C_g Q_g = 0 \tag{8}$$

The inequality constraints (3) consist of two sets of n_l branch flow limits as nonlinear functions of the bus voltage angles and magnitudes, one for the *from* end and one for the *to* end of branch:

$$h_f(\Theta, V_m) = \left|F_f(\Theta, V_m)\right| - F_{max} \leq 0 \tag{9}$$

$$h_t(\Theta, V_m) = |F_t(\Theta, V_m)| - F_{max} \leq 0 \tag{10}$$

Usually the flows are apparent power flows expressed in MVA, but it can be real/active power flows (in MW) or currents in kA, giving the following sets for the flow constraints:

$$F_f(\Theta, V_m) = \begin{cases} S_f(\Theta, V_m), apparent\ power \\ P_f(\Theta, V_m), real\ power \\ I_f(\Theta, V_m). current \end{cases} \tag{11}$$

The variable limits in equation (4) includes an equality constraint on any reference bus voltage angle and upper and lower limits on all bus voltage magnitudes and real/active generator injections and reactive generator injections:

$$\theta_i^{ref} \leq \theta_i \leq \theta_i^{ref}, \qquad i \in I_{ref} \tag{12}$$

$$V_m^{i,min} \leq V_m^i \leq V_m^{i,max}, \qquad i = 1 \dots n_b \tag{13}$$

$$P_g^{i,min} \leq P_g^i \leq P_g^{i,max}, \qquad i = 1 \dots n_g \tag{14}$$

$$Q_g^{i,min} \leq Q_g^i \leq Q_g^{i,max}, \qquad i = 1 \dots n_g \tag{15}$$

CHAPTER FOUR: DEEP NEURAL NETWORK

A Deep Neural Network (DNN) is a type of artificial neural network (ANN) composed of multiple hidden layers that learn complex non-linear mappings between inputs and outputs. A simple DNN architecture is shown in Figure 1. Mathematically, a DNN can be represented as a function $F: R^n \rightarrow R^m$, where n is the number of input features, and m is the number of output variables.

A DNN consists of multiple layers, each performing a weighted transformation of the input followed by a nonlinear activation function. The transformation at each layer can be expressed as follows:

$$z^{(l)} = W^{(l)}a^{(l-1)} + b^l \tag{16}$$

$$a^{(l)} = \sigma\left(z^{(l)}\right) \tag{17}$$

Where:

$z^{(l)}$ is the weighted sum of inputs at layer *l,*

$W^{(l)}$ is the weight matrix for layer *l,*

b^l is the bias vector for layer *l,*

$a^{(l)}$ is the activation output of layer *l,*

$\sigma(\cdot)$ is the activation function.

The final layer produces the predicted output $\hat{y}$:

$$\hat{y} = F(x;\theta) = W^{(L)}a^{(L-1)} + b^{(L)} \tag{18}$$

Where L is the number of layers, and $\theta = \{W^{(l)}, b^{(l)}\}$ represents the set of learnable parameters.

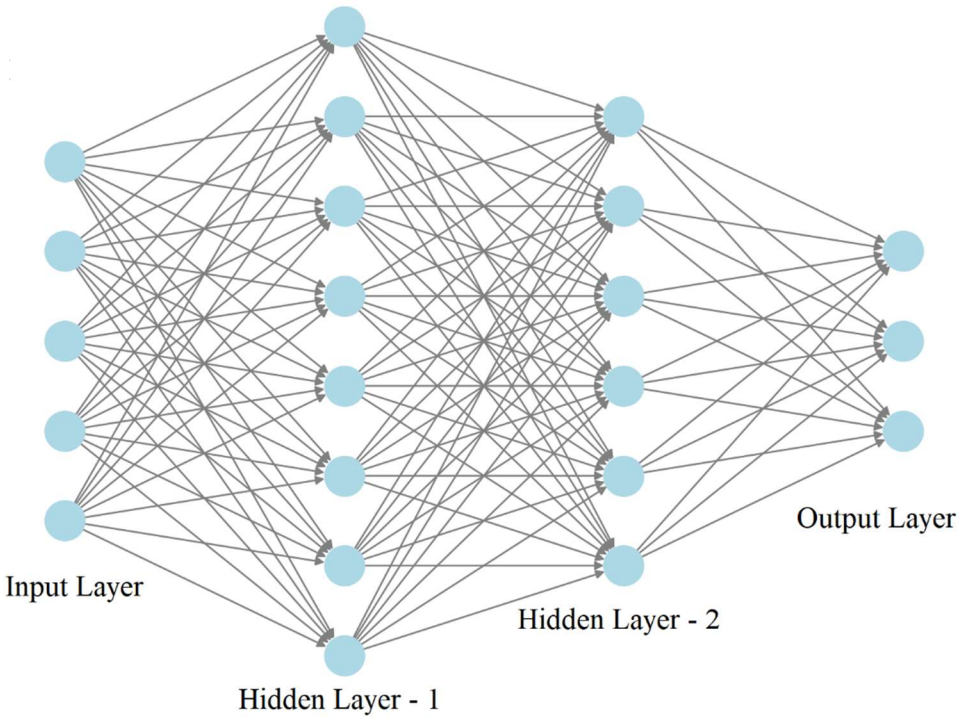

Figure 1: Deep Neural Network Architecture

4.1. Loss Function and Training

The DNN is trained by minimizing a loss function L, which measures the difference between the predicted values $\hat{y}$ and the actual values y. A common choice for regression problem in OPF prediction is the mean squared error (MSE):

$$L(\theta) = \frac{1}{N}\sum_{i=1}^{N}\left||y_i - \hat{y}_i|\right|^2 \tag{19}$$

Where N is the number of training samples. The optimization is performed using gradient-based methods such as the Adam optimizer:

$$\theta^{(t+1)} = \theta^{(t)} - \eta\nabla_\theta L(\theta) \tag{20}$$

Where η is the learning rate, and $\nabla_\theta L$ is the gradient of the loss function with respect to the network parameters.

4.2. Activation Functions

The activation function in artificial neuron gives nonlinearity to the neural network, which enables it to learn complex patterns. Most commonly used activation functions in DNN include:

- ReLU (Rectified Linear Unit):

$$\sigma(z) = \max(0, z) \tag{21}$$

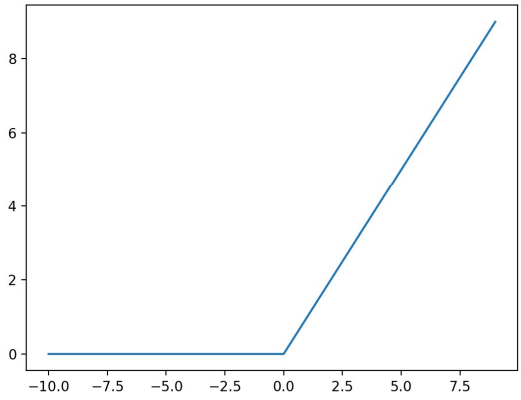

Figure 2: ReLU Activation Function

- Sigmoid:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

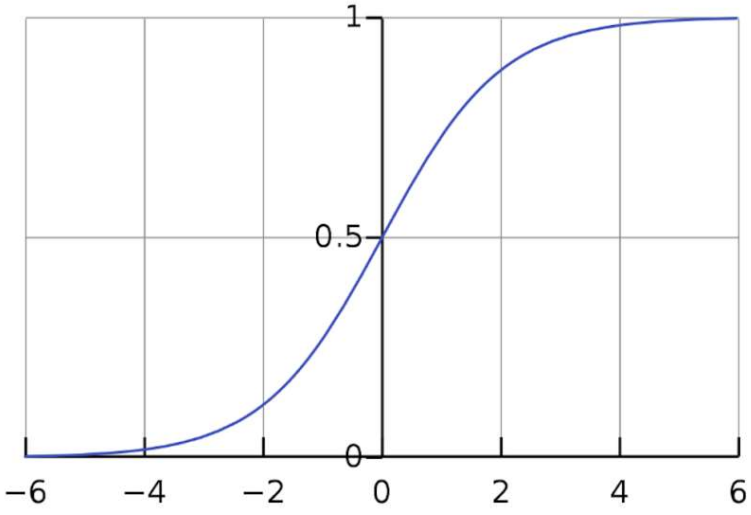

Figure 3: Sigmoid Activation Function

- Hyperbolic Tangent (Tanh)

$$\sigma(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

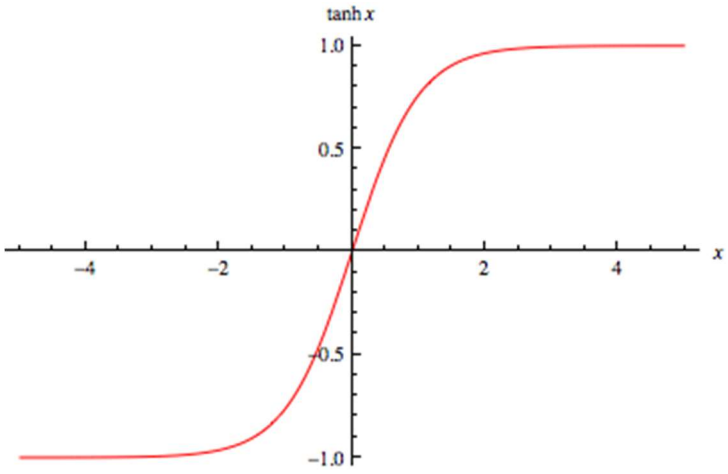

Figure 4: Hyperbolic tangent activation function

- Scaled Exponential Linear Unit (Selu)

$$y = elu(x) = f(x) = \begin{cases} x & x > 0 \\ a(e^x - 1) & x <= 0, \end{cases} \qquad a > 0$$

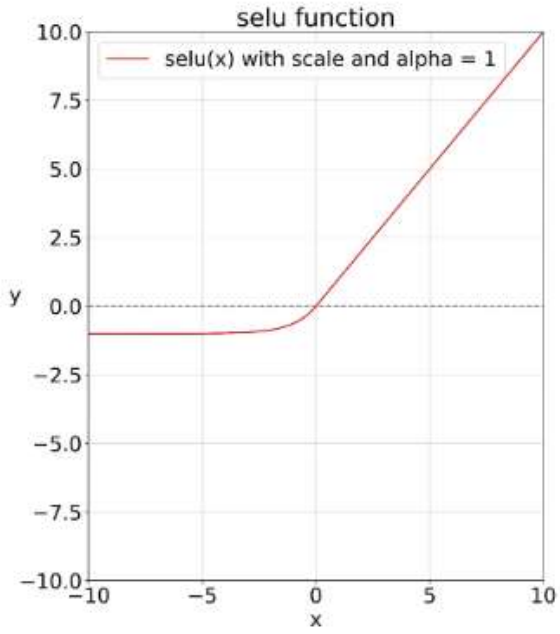

Figure 5: SELU activation Function

4.3. Regularization

To decrease the risk of overfitting, following regularization techniques are introduced.

4.3.1. Dropout

Certain percentage of neuron is randomly 'dropped' (set to zero) during training, effectively turning off different parts of the networks on each mini-batch. This forces network not to rely too heavily on any one neuron or path. This technique provides a form of ensemble averaging – each pass trains a slightly different sub-network, so the full model generalizes better. In OPF contexts, this can help the DNN avoid over-relying on certain patterns or line/generator features.

4.3.2. Batch Normalization

Batch Normalization normalizes activations in each mini-batch to have zero mean and unit variance. This technique stabilizes training, allowing higher learning rates and reducing sensitivity to weight initialization. While often seen as a technique for faster convergence, it also yields mild regularization effects. For large-scale DNNs modeling power systems, a stable training process is crucial—especially stacking multiple hidden layers.

4.4. Hyperparameter Tuning

The performance of a Deep Neural Network (DNN) is dictated by the choice of Hyperparameters – such as the learning rate, number of hidden units, number of neurons, dropout rates and regularization coefficients. The space of Hyperparameters is denoted as $\Theta \subset \boldsymbol{R^d}$, where *d* is the dimensionality of Hyperparameter vector. The goal is typically to minimize a scalar-valued loss function like MSE.

4.4.1. Bayesian Optimization

Bayesian optimization is sequential design strategy for global optimization of black-box functions, that does not assume any functional norms. It is usually employed to optimize expensive to evaluate functions. It has prominent use in machine learning problems for optimizing Hyperparameter values.

Bayesian optimization models the unknown objective function *f* using surrogate model, typically wtih Tree-structured Parzen Estimator (TPE) or Gaussian Process (GP).Then acquisition function is defined to decide which hyperparamter values is next. The typical algorithm for Bayesian Optimization is given below.

1. Initialize: Start with a few randomly sampled or hand-picked hyperparameter configurations. Train your DNN for each and record the validation losses.
2. Fit Surrogate: Build a probabilistic model (e.g., Gaussian Process or TPE) that approximates how hyperparameters relate to the observed validation loss.
3. Acquisition Function: Based on the surrogate's mean and variance predictions, pick the next promising hyperparameter set θ_{t+1}.
4. Evaluate: Train and validate the DNN with θ_{t+1}. Measure the new loss $f(\theta_{t+1})$.
5. Update: Incorporate the new data $(\theta_{t+1}, f(\theta_{t+1}))$. into the surrogate model.
6. Stopping: Continue until a set of iteration limit or convergence criterion is met. The best hyperparameters found so far are taken as the solution.

4.5. K-Fold Cross Validation

K-fold cross validation provides more robust, data-efficient method for both model selection when combined with hyperparameter tuning and performance assessment.

The general procedure for K-fold cross validation is given below.

1. Shuffle and Partition: Dataset of size N is randomly shuffled. It is then partitioned into k equal sized subsets. Each subset is referred to as fold.
2. Iterative Training and Validation: For each fold, the model is trained on all data except for that one fold, which is used as validation data. The performance metrics like MSE is computed for each fold training and validation process.
3. Aggregate Performance: After cycling through all folds, the overall performance metric is taken as average and standard deviation of MSE for each fold training and validation process.

CHAPTER FIVE: METHODOLOGY

The methodology of this research develops a feedforward framework based Deep Neural Network (DNN) which predicts Optimal Power Flow (OPF) solutions and power flow solutions under contingencies. The approach can be summarized on Figure 6 and consists of the following key steps.

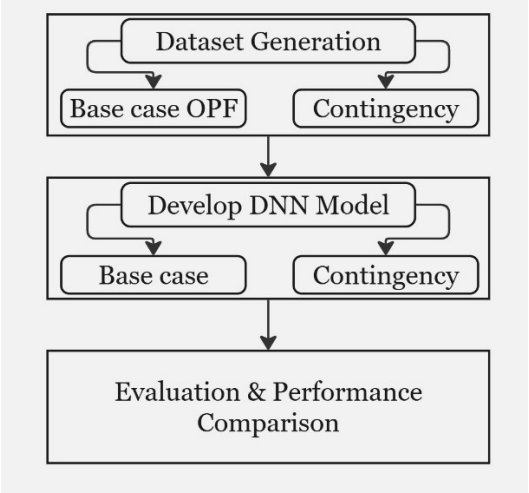

Figure 6: Methodology Summary

5.1. Data-set Generation

To train and validate the proposed DNN models, an extensive dataset of OPF solution is generated using MATPOWER, a MATLAB-based power system simulations tool.

5.1.1 Base-case OPF Data-set

OPF is solved for various load scenarios to generate dataset samples. The dataset consists of real and reactive power loads, optimal real/active and reactive power generations, bus voltages and transmission line flows.

5.1.2 Contingency Simulations

For each transmission line outage, the power flow is solved to identify violations of bus voltages and line flow. Newton Raphson method is used to solve the non-linear equations. For each generator outage, except for the slack/swing generator, the power flow is solved using Newton-Raphson method to assess bus voltages and line flow violations. The outage of line and generator are ranked on basis of causing the highest number of violations.

5.2. Development of DNN Models

Two sets of Deep Neural Network (DNN) models are designed for OPF prediction and security assessment under critical contingencies.

5.2.1. Baseline DNN for OPF Prediction

The input for this set are real and reactive power demands at each bus. The output is optimal real and reactive power generation at each generator, voltage magnitude and angles in all buses, power flow in transmission lines. The model architecture is chosen after performing hyperparameter tuning using Bayesian Optimization. The model is validated using K-Fold cross validation and finally trained on train dataset with pseudo validation data for callback.

5.2.2 Contingency-Aware DNN for Security Assessment

The input for the set are real and reactive power demands at all buses along with binary indicators for line and generator outage. Binary 0 means line and generator is in service, binary 1 means outaged has occurred. The output is real/active power generation of each generators and reactive power generation of each generators, bus voltage magnitude and bus voltage angles at each bus and line flows in transmission lines. The model architecture is chosen after performing hyperparameter tuning using Bayesian Optimization. The model is validated using K-Fold cross validation and finally trained on train dataset with pseudo validation data for callback.

5.3. Model Training, Validations and Testing

Dataset is split in training, validation and testing. 50% is used for training and validation and 50% is used for final model testing to determine performance metrics. For K-Fold cross validation, K is taken as 5. The metrics to measure performance includes Mean Squared Error (MSE), Mean Absolute Error (MAE), Coefficient of Determination and computation time comparison with traditional OPF solvers.

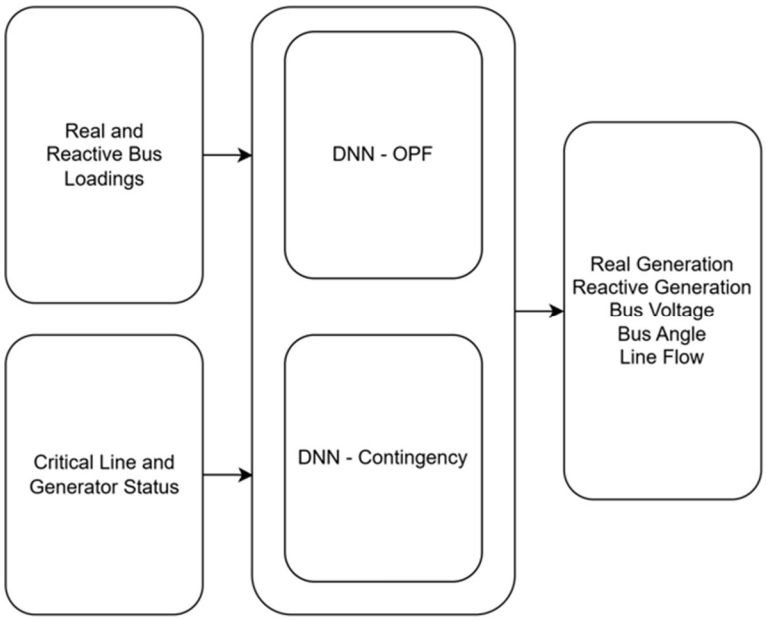

Figure 7: DNN Model

5.4. Hyperparamter Tuning: Bayesian Optimization

To determine the best set of hyperparameters, 50% of training & validation data is first passed through different model architectures under Bayesian Optimization procedure. The hyperparameters considered and its range are given in table below.

S.N.	Hyperparameters	Candidates
1.	Learning rate	Floating range between 1e-3 & 1e-4
2.	Batch size	{16, 32, 64, 128}
3.	Network Depth	2-4 hidden layers
4.	Neurons per layer	Integer range between 32 & 256
5.	Dropout rate	{0.1, 0.2, 0.3, 0.4, 0.5}
6.	Activation Function	Relu, Selu, Tanh

Table 1: Hyperparameter Tuning Candidates

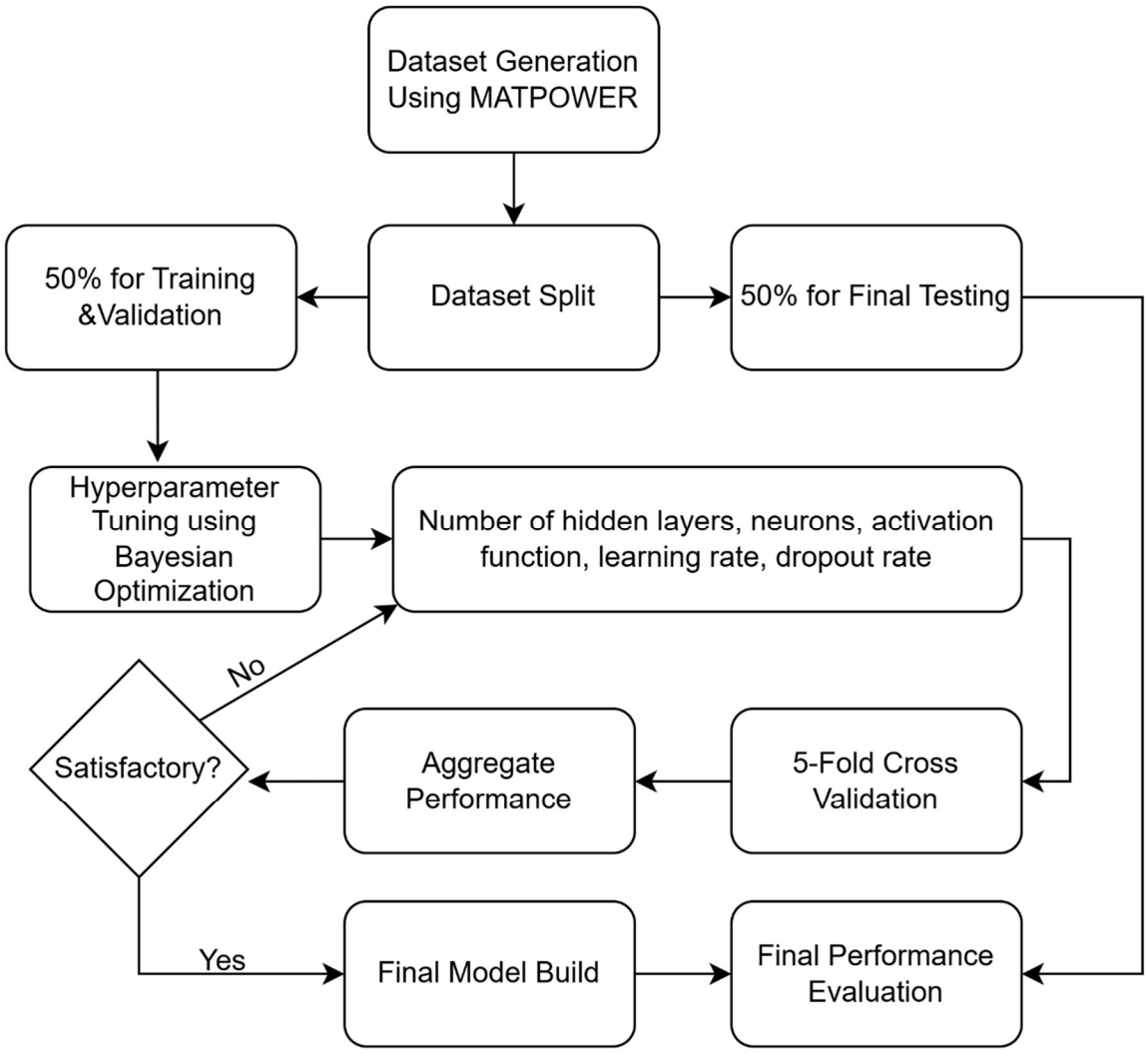

Figure 8: Detailed Methodology

CHAPTER SIX: RESULTS

The study for this research was conducted on IEEE 30-bus test system, which is well-established benchmark for power system optimization and security assessment studies. The system consists of 41 transmission lines, 6 generators, and 30 total buses, providing a realistic testbed for analyzing optimal power flow (OPF) and contingency scenarios.

6.1. Dataset Overview

For base case OPF, 10,000 data are generated using MATPOWER. The load is varied from minimum 20% to 100% of standard loading. The power factor for each load is varied from minimum 0.75 lag to unity.

For contingency cases, 1000 data are generated and three most critical line outage and two most critical generator outage are taken as critical contingency condition. Severe line and generator outage are given in table below.

Table 2: Line and Generator Outage for Critical Contingency

Type	ID	Violations
Line Outage	Line 8	1025
Line Outage	Line 10	418
Line Outage	Line 38	348
Generator Outage	Generator 2	403
Generator Outage	Generator 4	388

6.2. DNN Model Performance

The performance of each model during hyperparamter tuning, 5-fold cross validation and final evaluation is obtained and presented.

6.2.1 Base OPF – PQ – Pg

This model takes bus real/active loading and reactive loading as input and gives optimal generation of real power at generators as output. For hyperparamter tuning, the 50% of train & validation data is run 50 trials of architectures with 200 epochs each. The Bayesian Optimization gave the following best hyperparameters.

Activation	Selu
Number of hidden layers	2
Layer 1	Units: 256, dropout: 0.1
Layer 2	Units: 32, dropout: 0.1
Learning rate	0.001
Optimal batch size	128
Best validation loss	0.0019

Table 3: Hyperparameter tuning for PQ-Pg

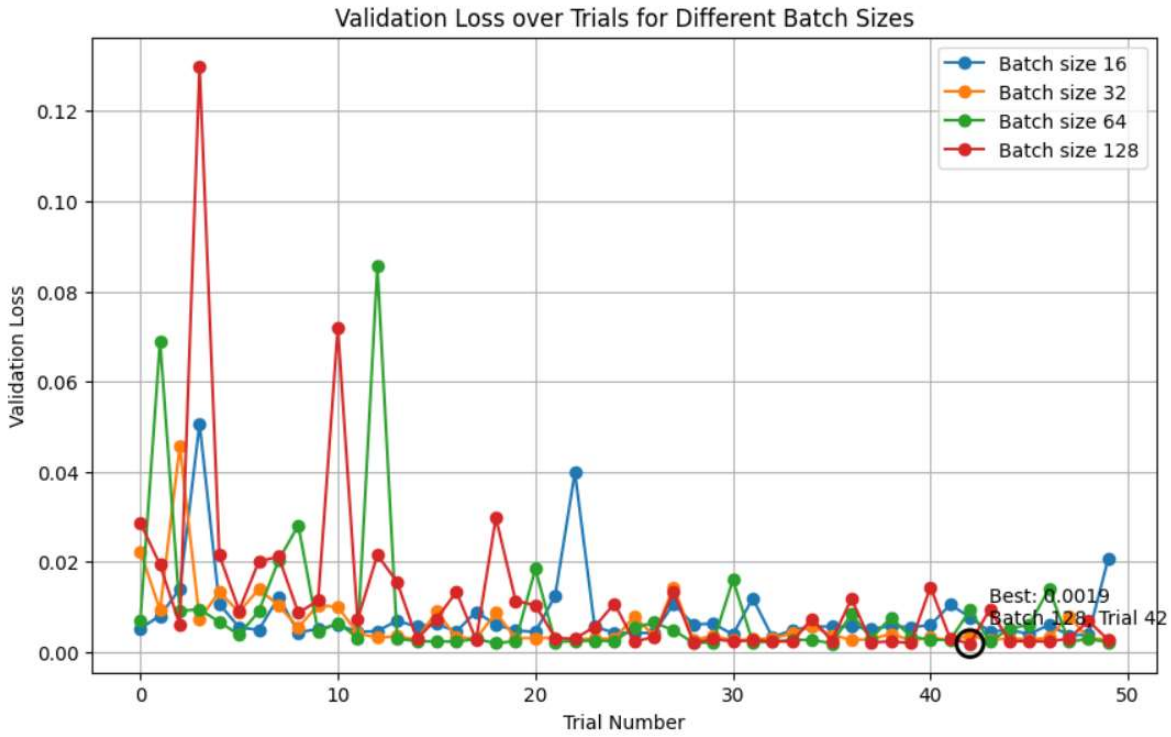

Figure 9: Hyperparameter Tuning: Loss vs. Trials for PQ-Pg

Using the best architecture, 5-fold cross validation over 350 epochs gives the following results.

Mean Validation Loss	0.004
Mean Validation MAE	0.041
Standard Deviation MAE	0.015
Mean Validation R-Squared	0.99585

Table 4: Model PQ-Pg 5-Fold CV

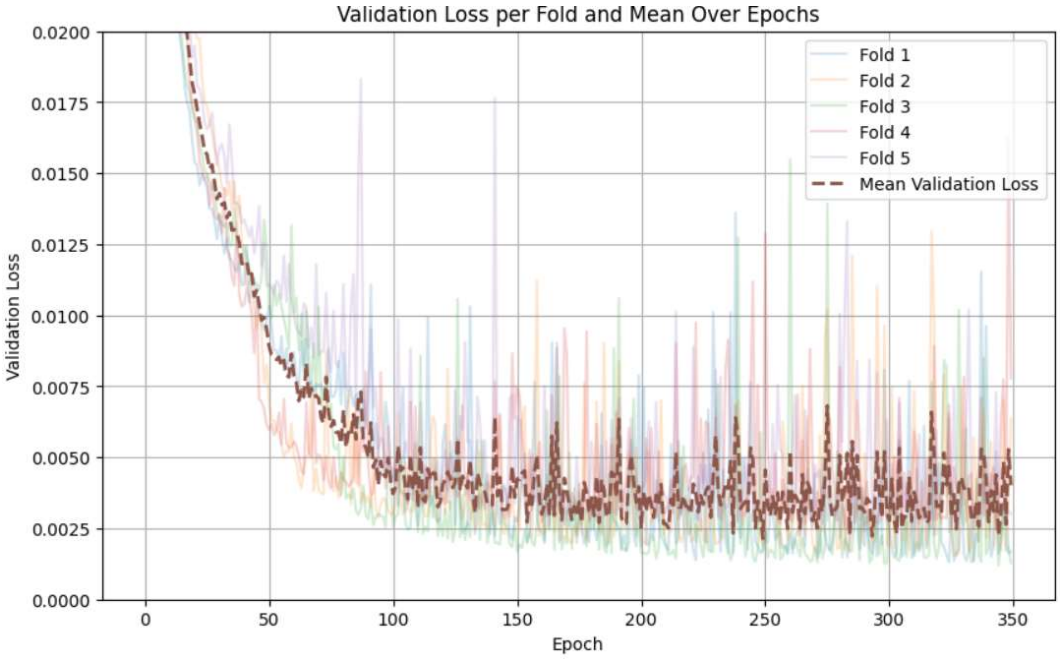

Figure 10: Model PQ-Pg 5-Fold CV

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Loss/Mean Squared Error (MSE)	0.013
Mean Absolute Error (MAE)	0.045
Standard Deviation of AE	0.104
R – Squared	0.9982

Table 5: PQ-Pg Model Performance for Test Data

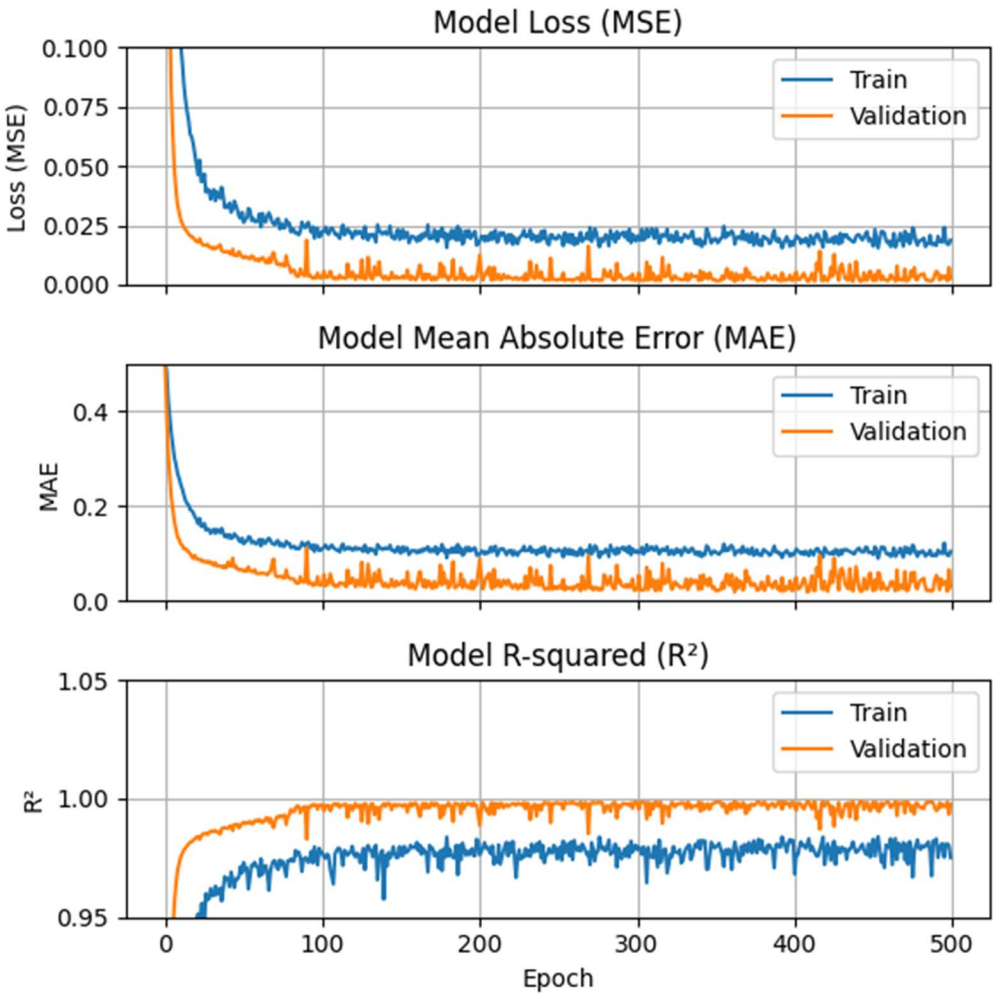

Figure 11: Model PQ-Pg Performance Metrics vs. Number of Epochs for Test Data

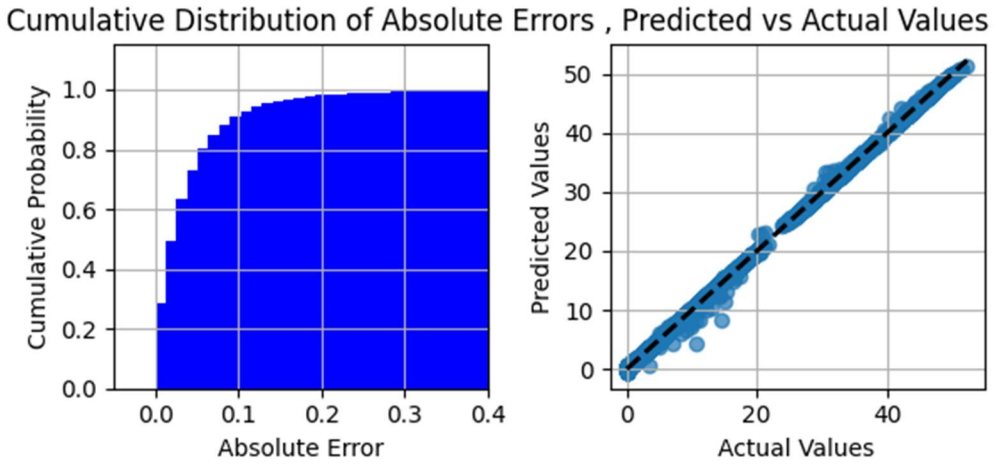

Figure 12: Model PQ-Pg Cumulative AE Distribution and Regression

The PQ-Pg model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.2 for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 0.013, MAE is 0.045, and R^2 is 0.9982, confirming that the PQ-Pg neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The x-co-ordinate axis in figure represents the value of absolute error, while the y-co-ordinate axis in figure represents the cumulative probability. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.3 suggests that almost all errors fall within this range.

The values that are predicted versus actual values scatter plot compares values that are predicted using DNN against actual values. Each point represents a test sample, with x-axis showing actual values and the y-axis showing predicted values. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQ-Pg appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.2 Base OPF – PQ – Qg

This model takes bus real/active power loads and reactive power loads as input and gives optimal generation of reactive power at generators as output. For hyperparamter tuning, the 50% of train & validation data is run 25 trials of architectures with 100 epochs each. The Bayesian Optimization gave the following best hyperparameters.

Activation	Selu
Number of hidden layers	2
Layer 1	Units: 256, dropout: 0.1
Layer 2	Units: 32, dropout: 0.1
Learning rate	0.001
Optimal batch size	64
Best validation loss	0.012

Table 6: Hyperparameter Tuning for PQ-Qg

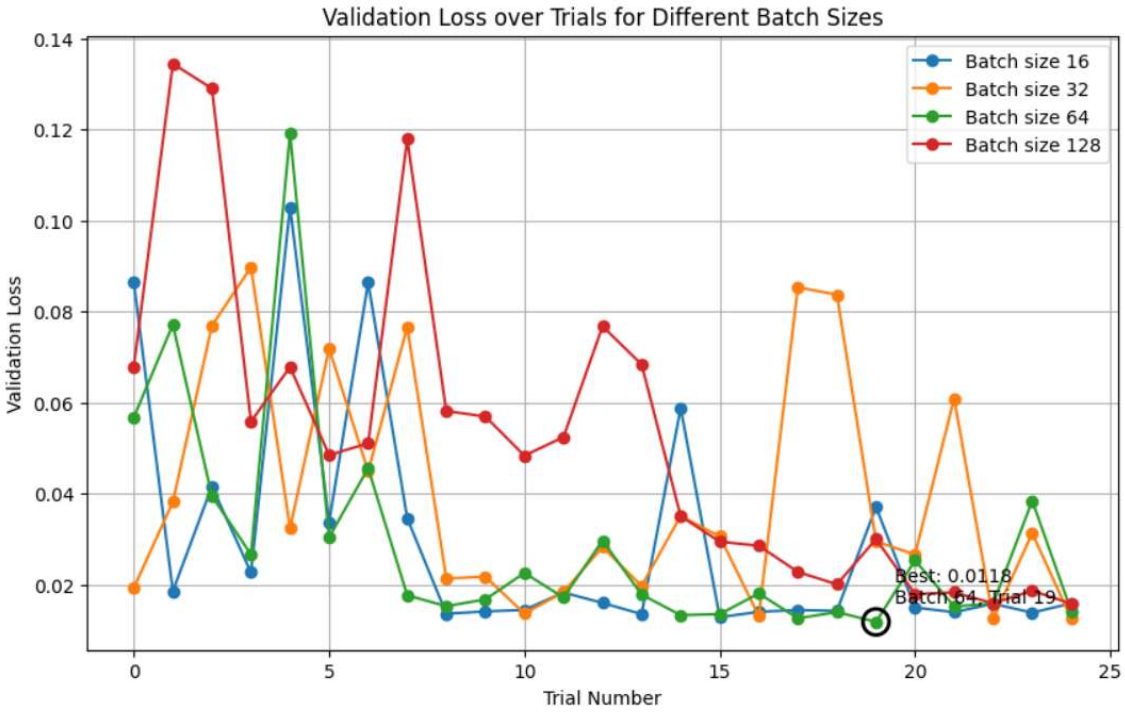

Figure 13: Hyperparameter Tuning: Loss vs. Trials for PQ-Qg

Using the best architecture, 5-fold cross validation over 350 epochs gives the following results.

Mean Validation Loss	0.014
Mean Validation MAE	0.054
Standard Deviation MAE	0.002
Mean Validation R-Squared	0.987

Table 7: Model PQ-Qg 5-Fold CV

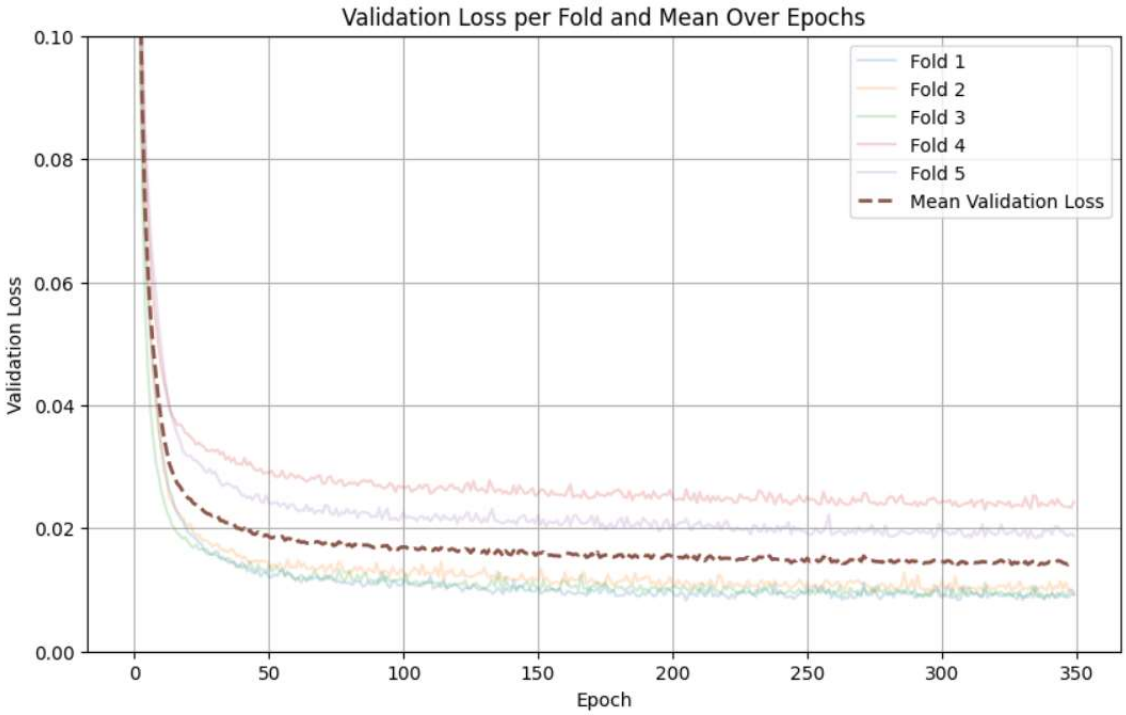

Figure 14: Model PQ-Qg 5-Fold CV

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Loss/Mean Squared Error (MSE)	0.108
Mean Absolute Error (MAE)	0.125
Standard Deviation of AE	0.304
R – Squared	0.99

Table 8: PQ-V Model Performance for Test Data

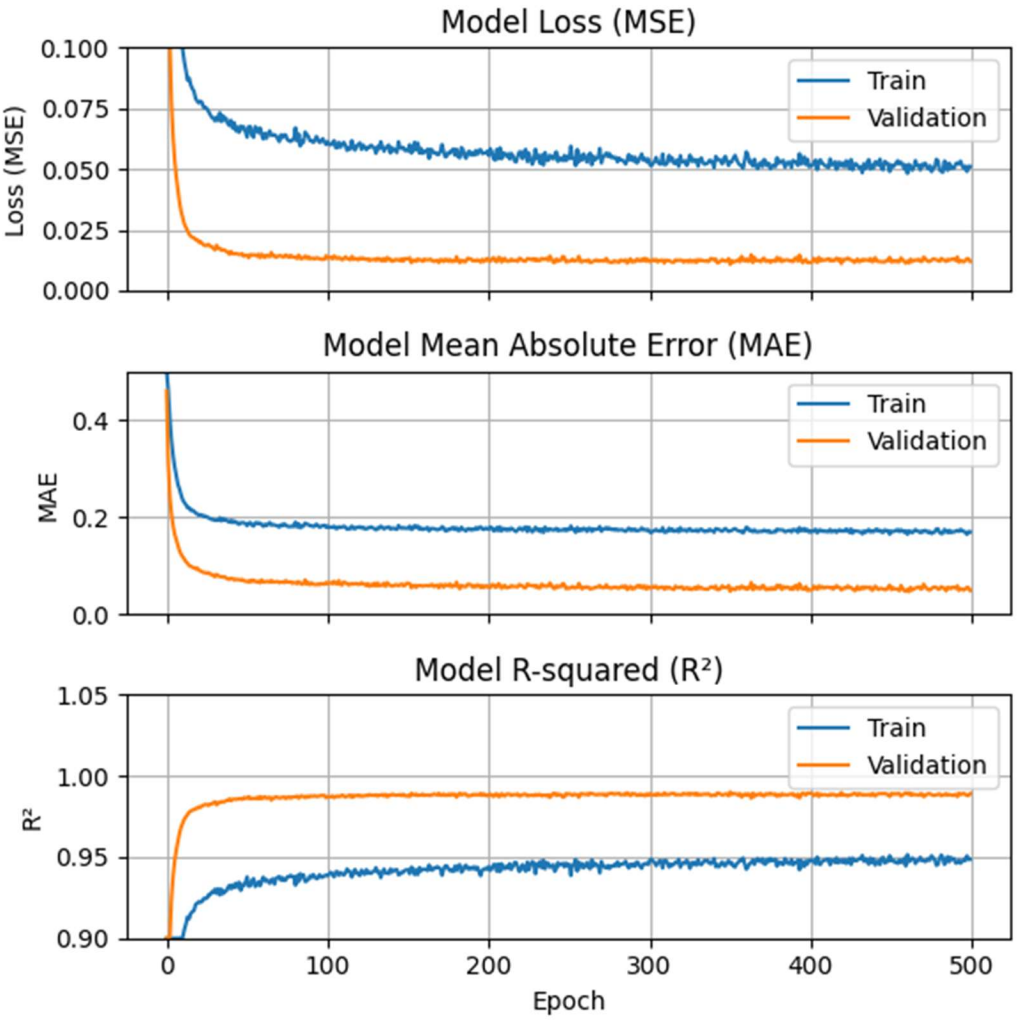

Figure 15: Model PQ-Qg Performance Metrics vs. Number of Epochs

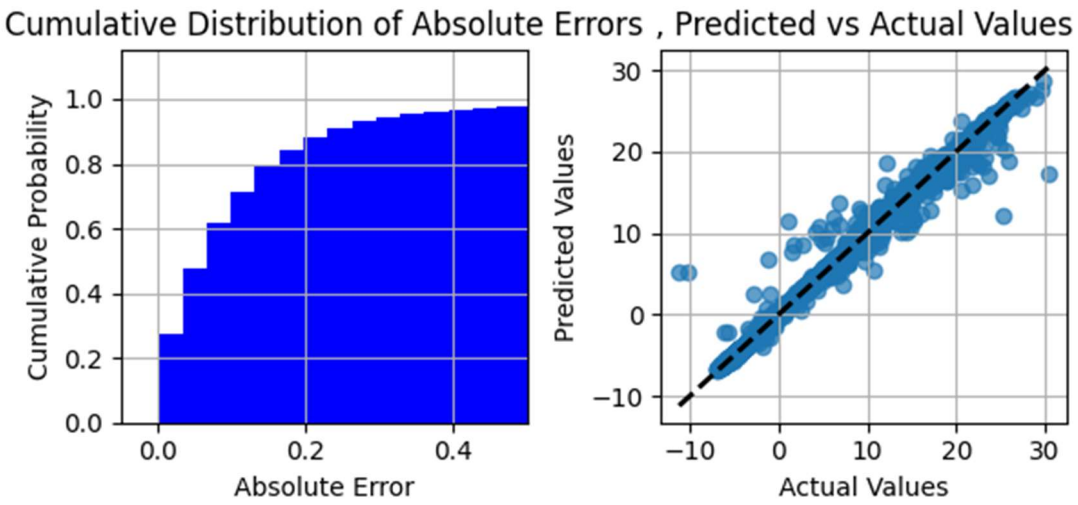

Figure 16: Model PQ-Qg Cumulative AE Distribution and Regression

The PQ-Qg model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.4 for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 0.108, MAE is 0.125, and R^2 is 0.99, confirming that the PQ-Qg neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The x-co-ordinate axis in Figure 16 represents the absolute error, while the y-co-ordinate axis shows the cumulative probability. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.5 suggests that almost all errors fall within this range.

The predicted vs. actual values scatter plot compares predicted values against actual values. Each point represents a test sample, with x-axis showing actual values and the y-axis showing predicted values. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQ-Qg appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.3 Base OPF – PQ – V

This model takes bus real/active power loads and reactive power loads as input and predicts bus voltages as output. For hyperparamter tuning, the 50% of train & validation data is run 25 trials of architectures with 100 epochs each. The Bayesian Optimization gave the following best hyperparameters.

Activation	Selu
Number of hidden layers	2
Layer 1	Units: 256, dropout: 0.1
Layer 2	Units: 256, dropout: 0.1
Learning rate	0.00066
Optimal batch size	32
Best validation loss	0.0564

Table 9: Hyperparameter Tuning for PQ-V

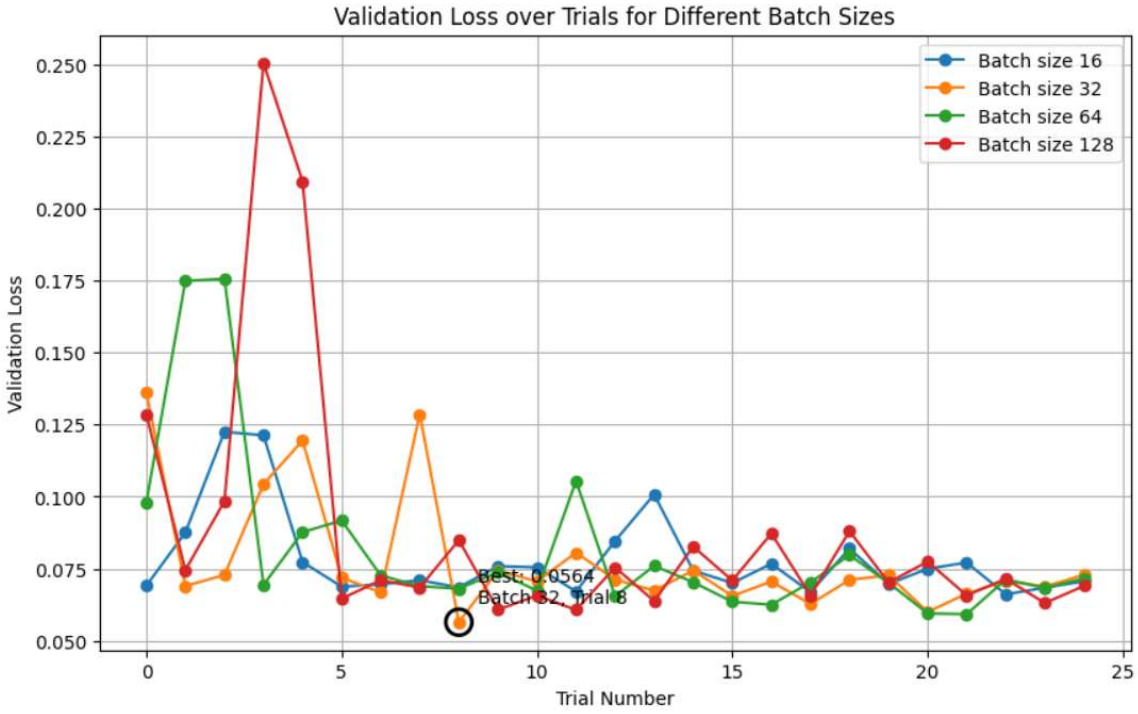

Figure 17: Hyperparameter Tuning: Loss vs. Trials PQ-V

Using the best architecture, 5-fold cross validation over 350 epochs gives the following results.

Mean Validation Loss	0.164
Mean Validation MAE	0.092
Standard Deviation MAE	0.006
Mean Validation R-Squared	0.932

Table 10: Model PQ-V 5-Fold CV

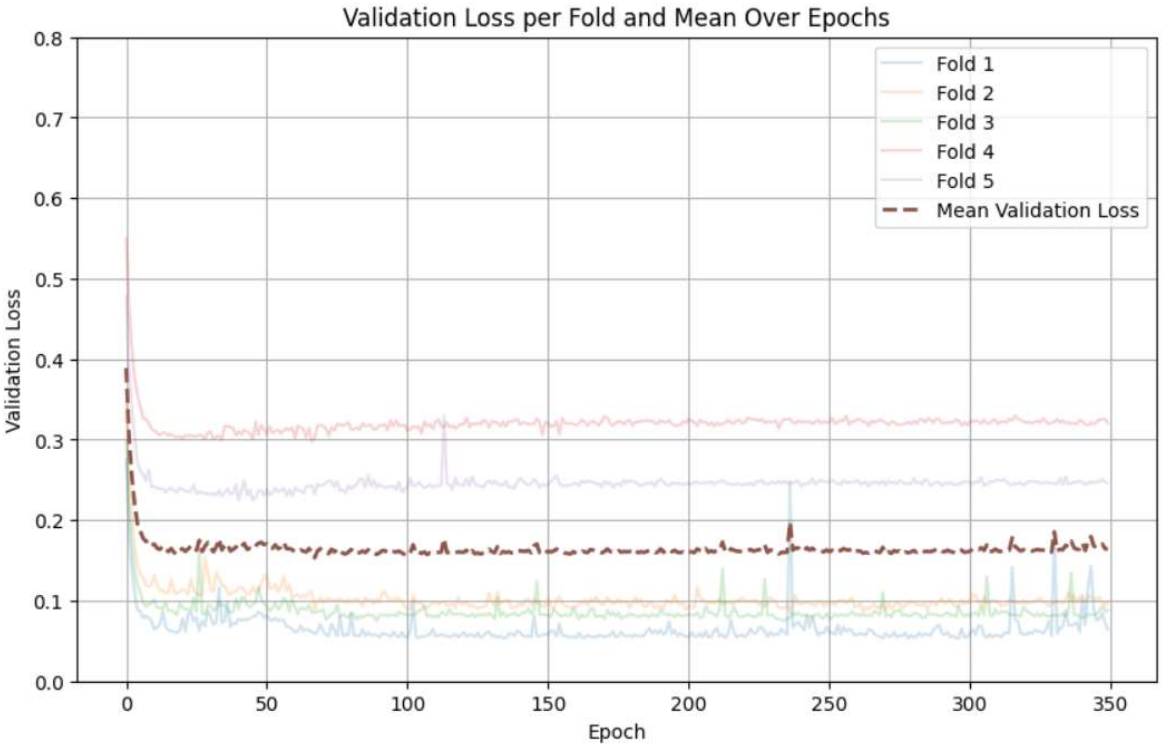

Figure 18: Model PQ-V 5-Fold CV

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Loss/Mean Squared Error (MSE)	0.000264
Mean Absolute Error (MAE)	7.5e-7
Standard Deviation of AE	0.0008
R – Squared	0.95

Table 11: PQ-V Model Performance for Test Data

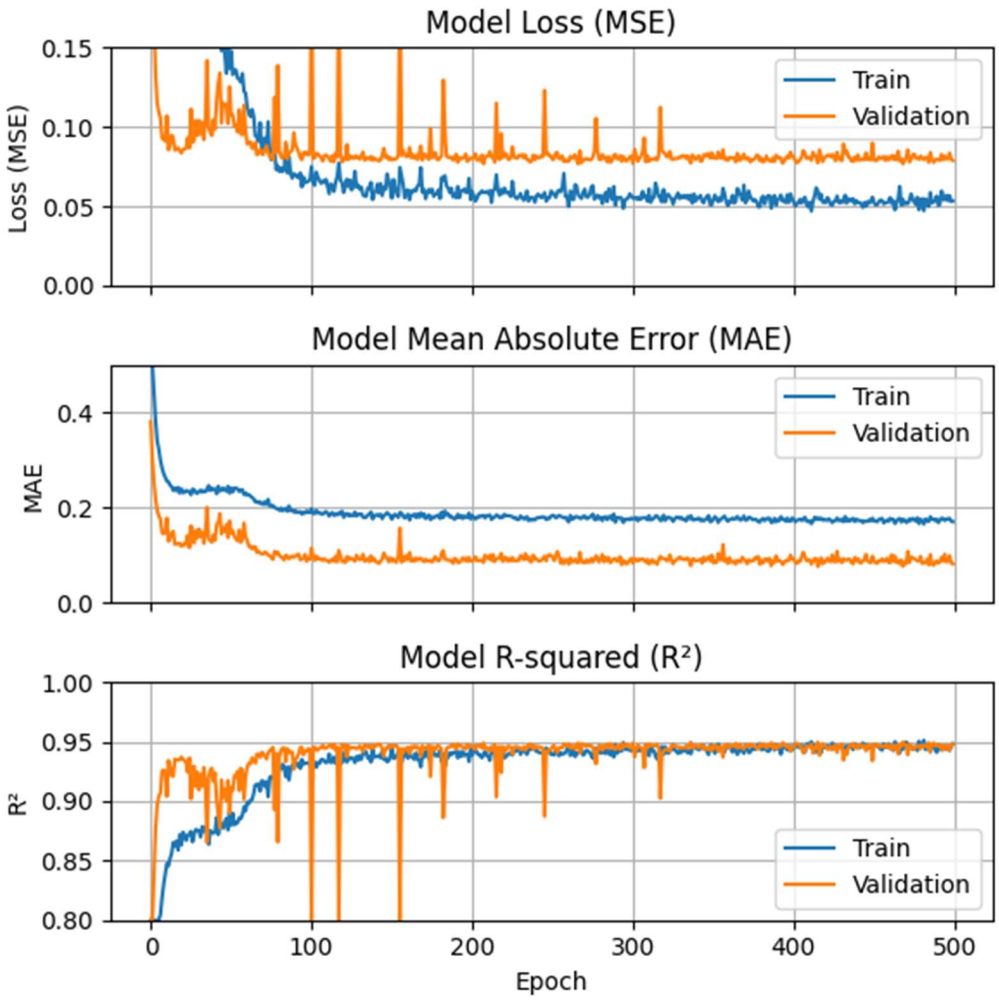

Figure 19: Model PQ-V Performance Metrics vs. Number of Epochs

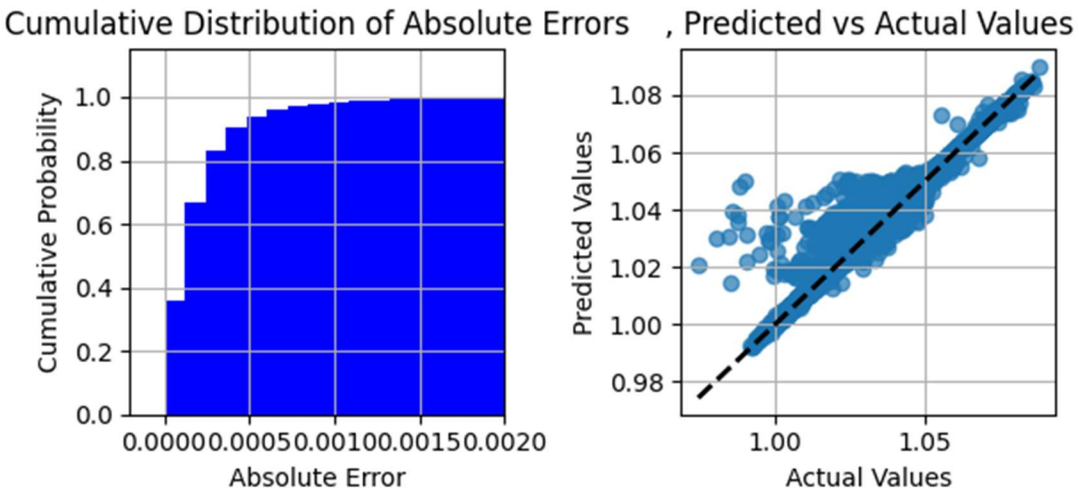

Figure 20: Model PQ-V Cumulative AE Distribution and Regression

The PQ-V model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.2 for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 2.64e-4, MAE is 7.5e-7, and R^2 is 0.95, confirming that the PQ-V neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The horizontal axis displays the magnitude of absolute errors, and the vertical axis corresponds to the cumulative probability of distribution. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.0015 suggests that almost all errors fall within this range.

The scatter plot visualizes the alignment between model predictions and ground-truth measurements. Each data point corresponds to a test instance, with the horizontal axis indicating observed (actual) values and the vertical axis displaying predicted outputs. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQ-V appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.4. Base OPF – PQ – Angle

This model takes bus real/active power loads and reactive power loads as input and predicts bus angles as output. For hyperparamter tuning, the 50% of train & validation data is run 25 trials of architectures with 100 epochs each. The Bayesian Optimization gave the following best hyperparameters.

Activation	Selu
Number of hidden layers	2
Layer 1	Units: 128, dropout: 0.1
Layer 2	Units: 96, dropout: 0.1
Learning rate	0.001
Optimal batch size	32
Best validation loss	0.005

Table 12: Hyperparameter Tuning for PQ-Angle

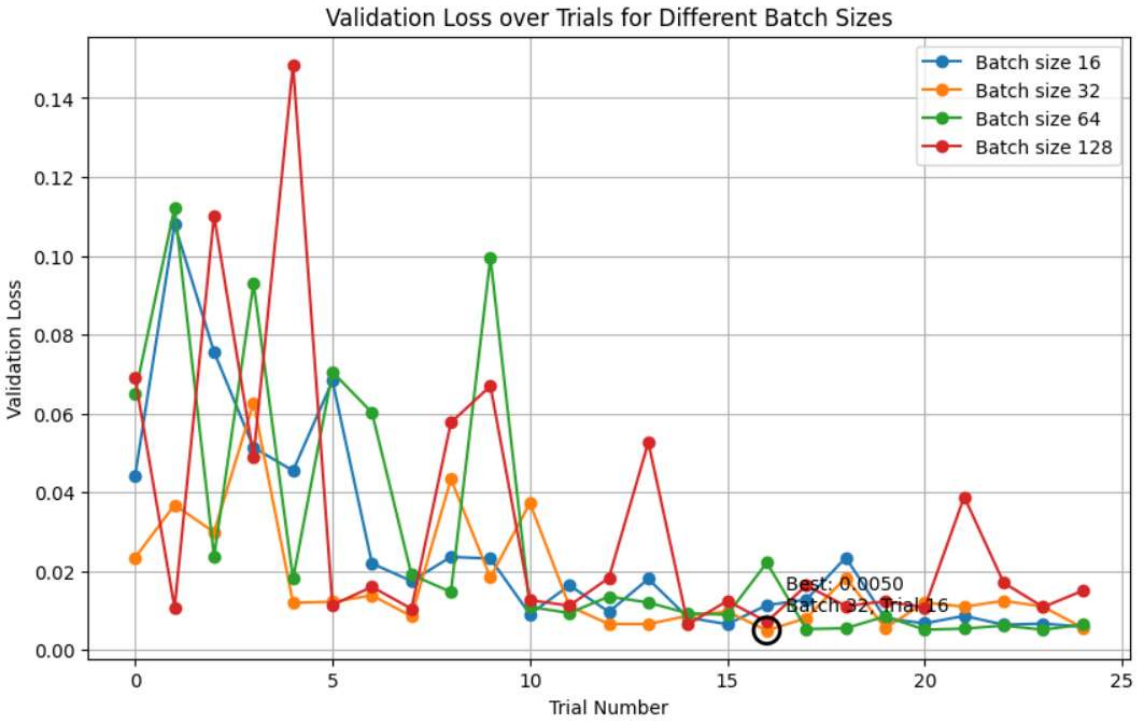

Figure 21: Hyperparameter Tuning: Loss vs Trials PQ-Angle

Using the best architecture, 5-fold cross validation over 350 epochs gives the following results.

Mean Validation Loss	0.006
Mean Validation MAE	0.04
Standard Deviation MAE	0.0025
Mean Validation R-Squared	0.9941

Table 13: Model PQ-Angle 5-Fold CV

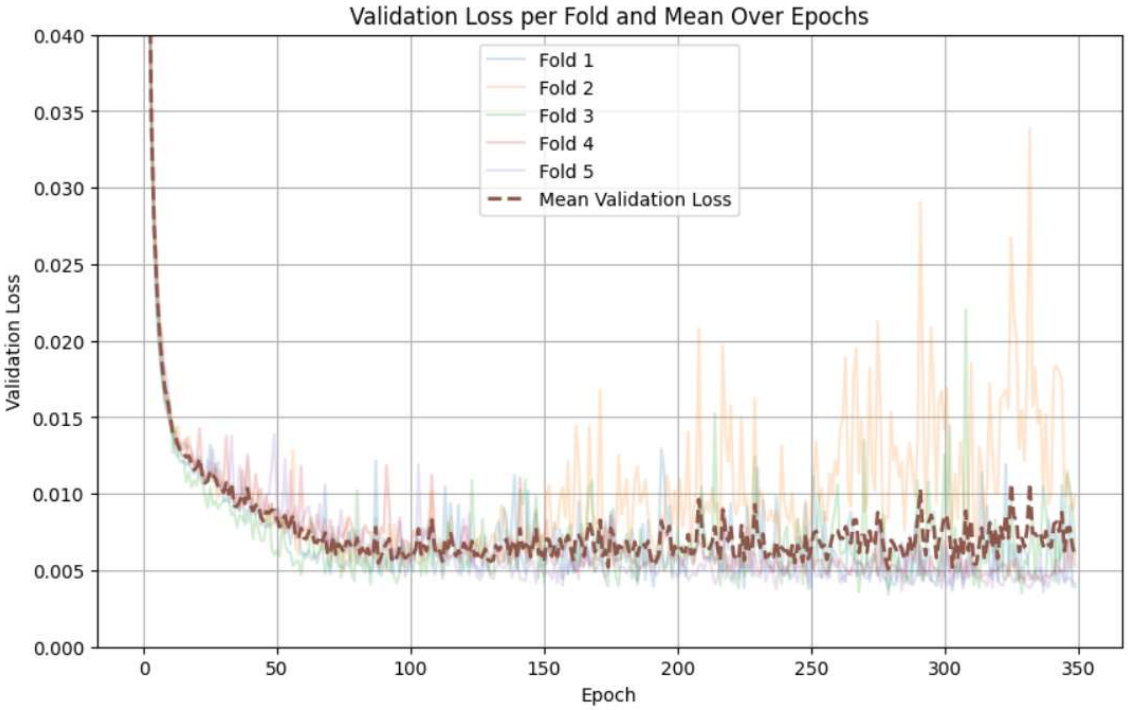

Figure 22: Model PQ-Angle 5-Fold CV

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Loss/Mean Squared Error (MSE)	0.0009
Mean Absolute Error (MAE)	0.0186
Standard Deviation of AE	0.024
R – Squared	0.995

Table 14: PQ-Angle Model Performance for Test Data

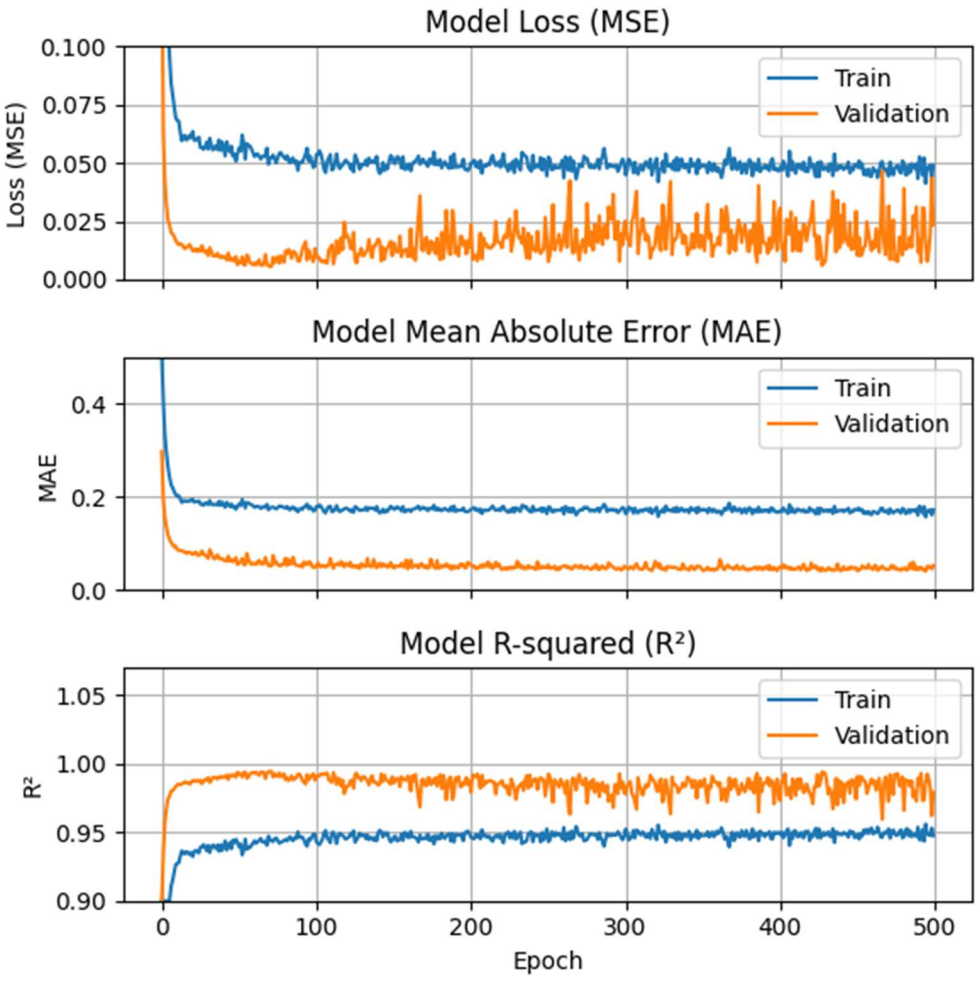

Figure 23: Model PQ-Angle Performance Metrics vs. Number of Epochs

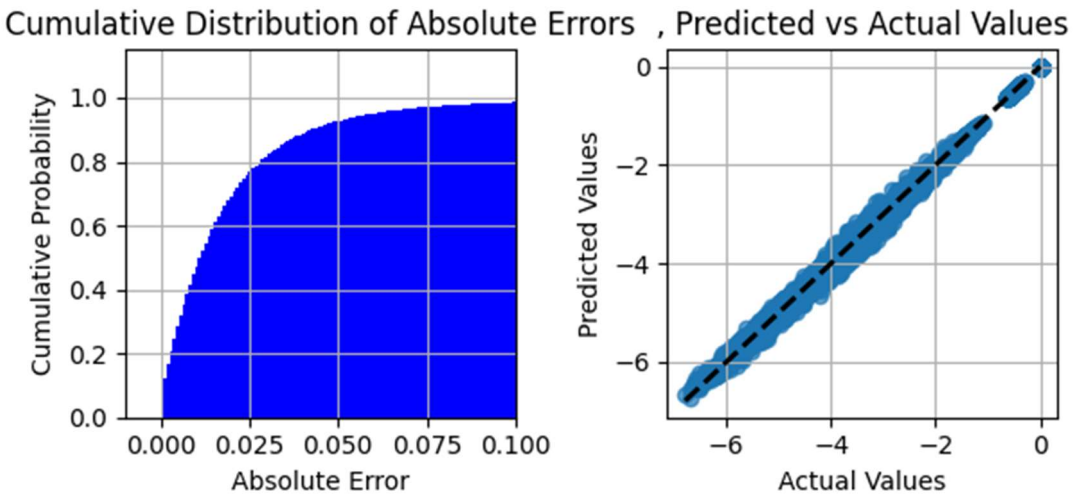

Figure 24: Model PQ-Angle Cumulative AE Distribution and Regression

The PQ-Angle model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.075 and 0.2 respectively for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 0.0009, MAE is 0.0186, and R^2 is 0.995, confirming that the PQ-Angle neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The horizontal axis displays the magnitude of absolute errors, and the vertical axis corresponds to the cumulative probability of distribution. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.1 suggests that almost all errors fall within this range.

The scatter plot visualizes the alignment between model predictions and ground-truth measurements. Each data point corresponds to a test instance, with the horizontal axis indicating observed (actual) values and the vertical axis displaying predicted outputs. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQ-Angle appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.5 Base OPF – PQ – Line Flow

This model takes bus real/active power loads and reactive power loads as input and predicts line flows as output. For hyperparamter tuning, the 50% of train & validation data is run 25 trials of architectures with 100 epochs each. The Bayesian Optimization gave the following best hyperparameters.

Activation	Selu
Number of hidden layers	2
Layer 1	Units: 256, dropout: 0.1
Layer 2	Units: 256, dropout: 0.1
Learning rate	0.001
Optimal batch size	64
Best validation loss	0.0123

Table 15: Hyperparameter Tuning for PQ-Line Flow

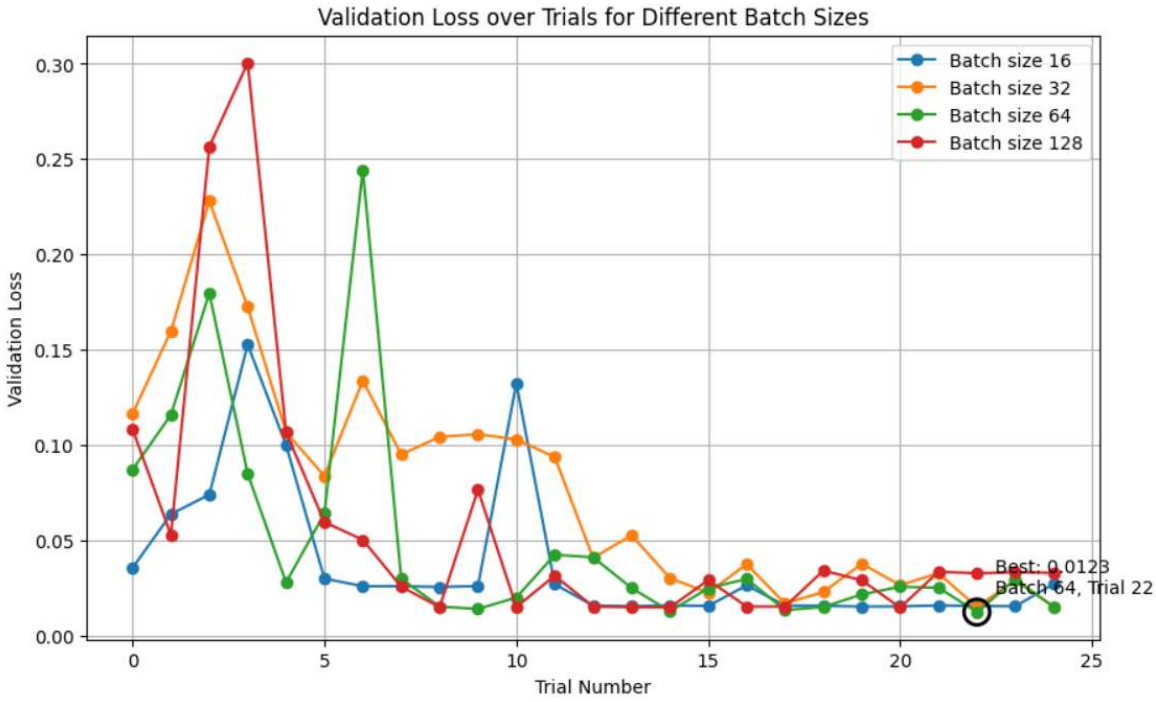

Figure 25: Hyperparameter Tuning: Loss vs. Trials PQ-Line

Using the best architecture, 5-fold cross validation over 350 epochs gives the following results.

Mean Validation Loss	0.01
Mean Validation MAE	0.063
Standard Deviation MAE	0.001
Mean Validation R-Squared	0.99

Table 16: Model PQ-Line Flow 5-Fold CV

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Loss/Mean Squared Error (MSE)	0.026
Mean Absolute Error (MAE)	0.091
Standard Deviation of AE	0.1314
R – Squared	0.992

Table 17: Model PQ-Line Performance for Test Data

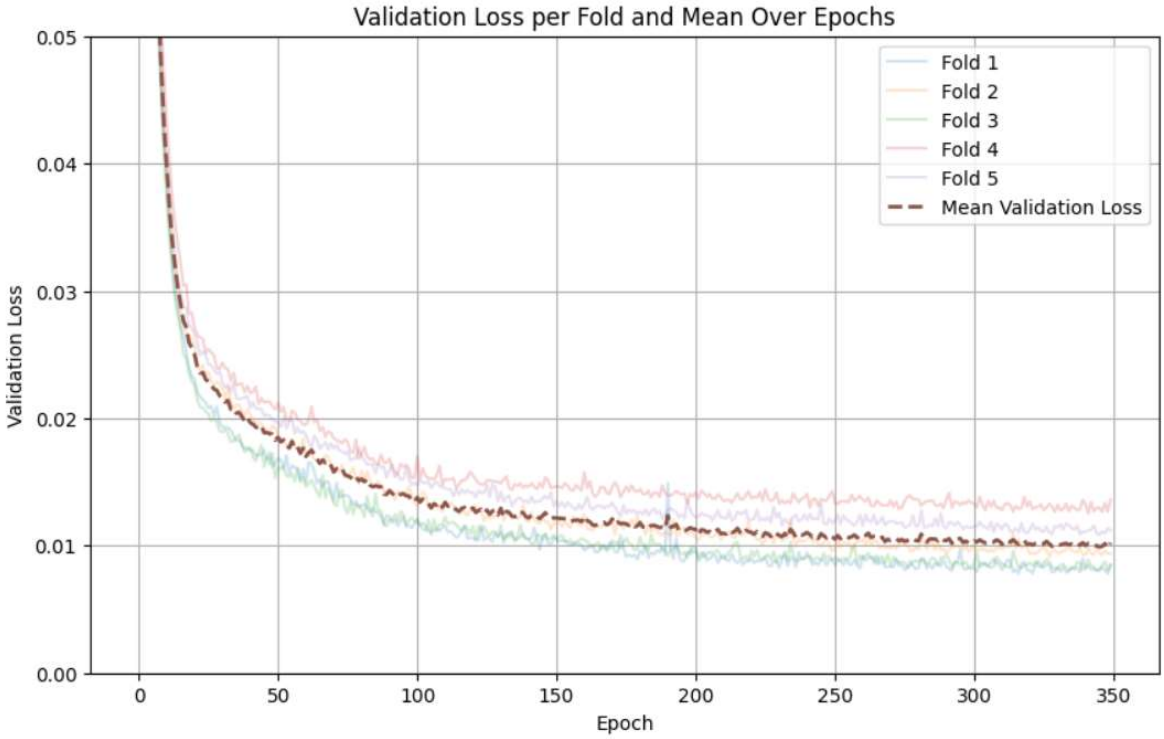

Figure 26: Model PQ-Line Flow 5-Fold CV

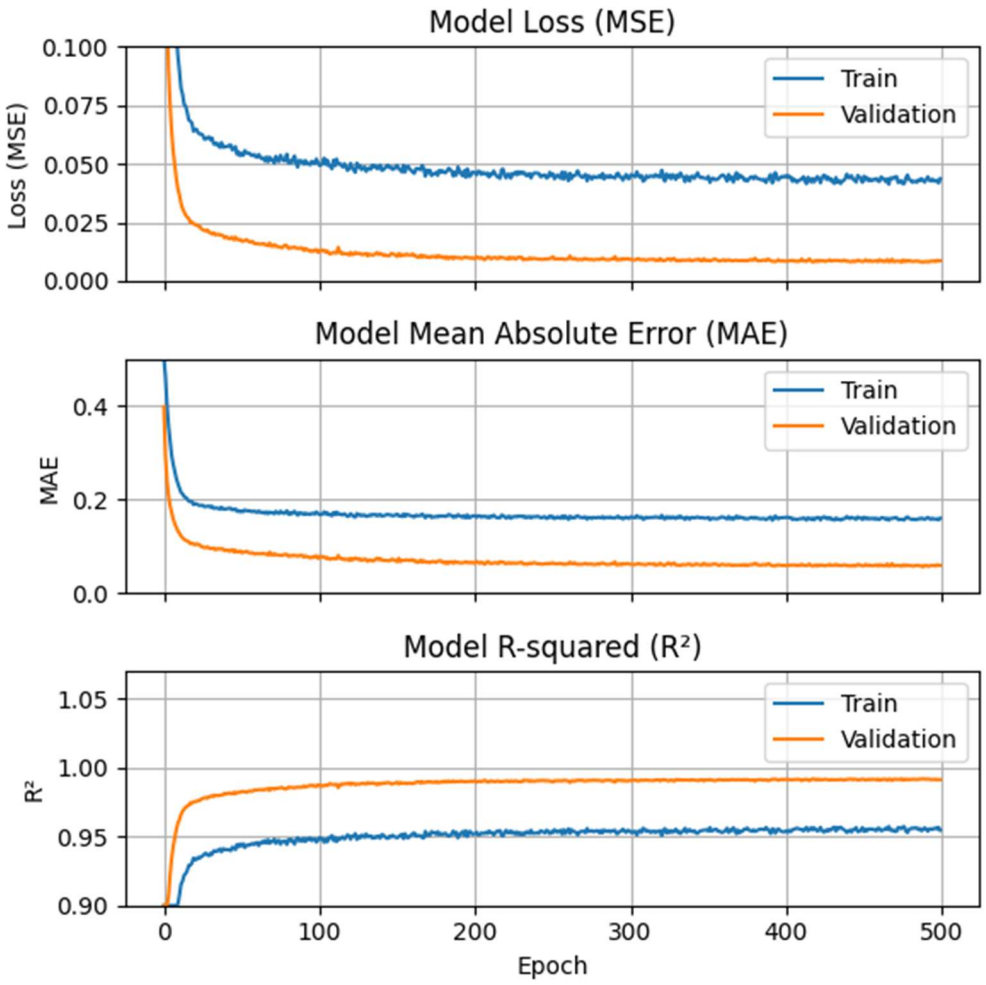

Figure 27: Model PQ-Line Performance Metrics vs. Number of Epochs

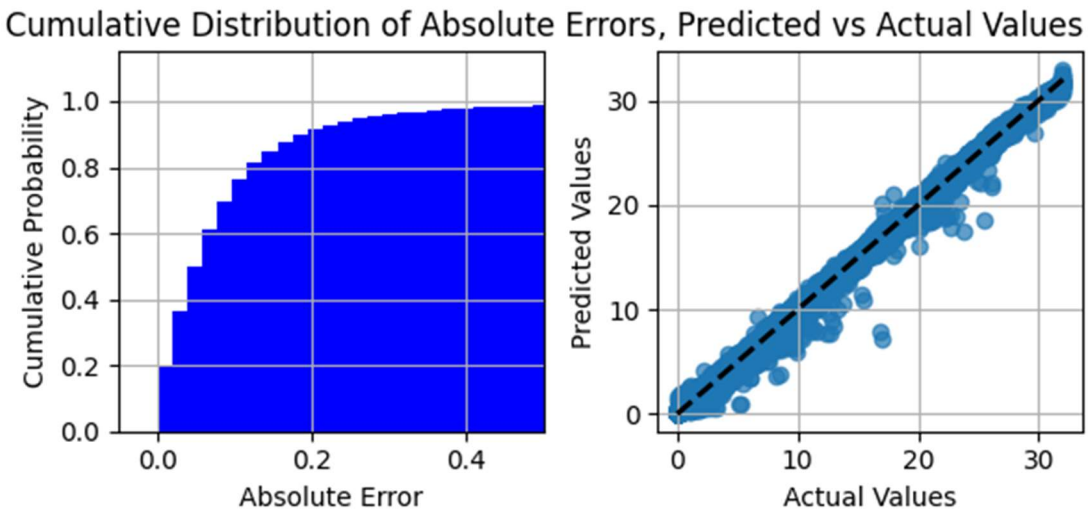

Figure 28: Model PQ-Line Cumulative AE Distribution and Regression

The PQ-Line model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.05 and 0.2 respectively for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 0.026, MAE is 0.091, and R^2 is 0.992, confirming that the PQ-Line neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The horizontal axis displays the magnitude of absolute errors, and the vertical axis corresponds to the cumulative probability of distribution. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.6 suggests that almost all errors fall within this range.

The scatter plot visualizes the alignment between model predictions and ground-truth measurements. Each data point corresponds to a test instance, with the horizontal axis indicating observed (actual) values and the vertical axis displaying predicted outputs. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQ-Line appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.5 Contingency Analysis Model

This set has five different models. Each of these model take real and reactive loading on buses as input as well as status of critical line and generator. The output is corresponding real power generation, reactive power generation, bus voltage, bus angle and line flows. The output is the system condition after outage has occurred. It does not represent optimal power flow in the system. The information that we obtain form the outputs are to check if the system violated any cases like generation overflow, voltage deviation or line overflow. Since, the most important ones are line over flow and voltage violations, these models is discussed here. The architecture of all the models are also presented.

Hyperparameter tuning for each of the model is done using Bayesian Optimization. The total trials for each model is 10 with 100 epochs each. The best hyperparameter obtained for each of the model is presented in Table below.

Model	No. of Hidden Layers	Number of Neurons & Dropout	Activation Function	Best Validation Loss	Learning Rate & Batch Size
PQS-Pg	4	(96,0.1),(256,0.5),(192,0.1),(32,0.1)	Selu	0.0062	(0.001,64)
PQS-Qg	2	(256,0.1),(256,0.1)	Selu	0.01	(0.00034,32)
PQS-V	2	(256,0.1),(256,0.1)	Selu	0.0184	(0.001, 16)
PQS-A	2	(256,0.2),(244,0.1)	Selu	0.007	(0.001, 16)
PQS-L	2	(256,0.1),(256,0.1)	Selu	0.019	(0.001,32)

Table 18: Best Model Architecture for Contingency Scenario

For each of the model under best architecture, 5-fold cross validation is done over 350 epochs. The results obtained is presented in table below.

Model	Mean MSE	Mean MAE	Std. MAE	Mean R-squared
PQS-Pg	0.0033	0.0403	0.0031	0.997
PQS-Qg	0.0034	0.043	0.00075	0.997
PQS-V	0.024	0.073	0.0036	0.985
PQS-A	0.004	0.047	0.0023	0.996
PQS-L	0.544	0.074	0.005	0.5

Table 19: Contingency Models 5-Fold CV

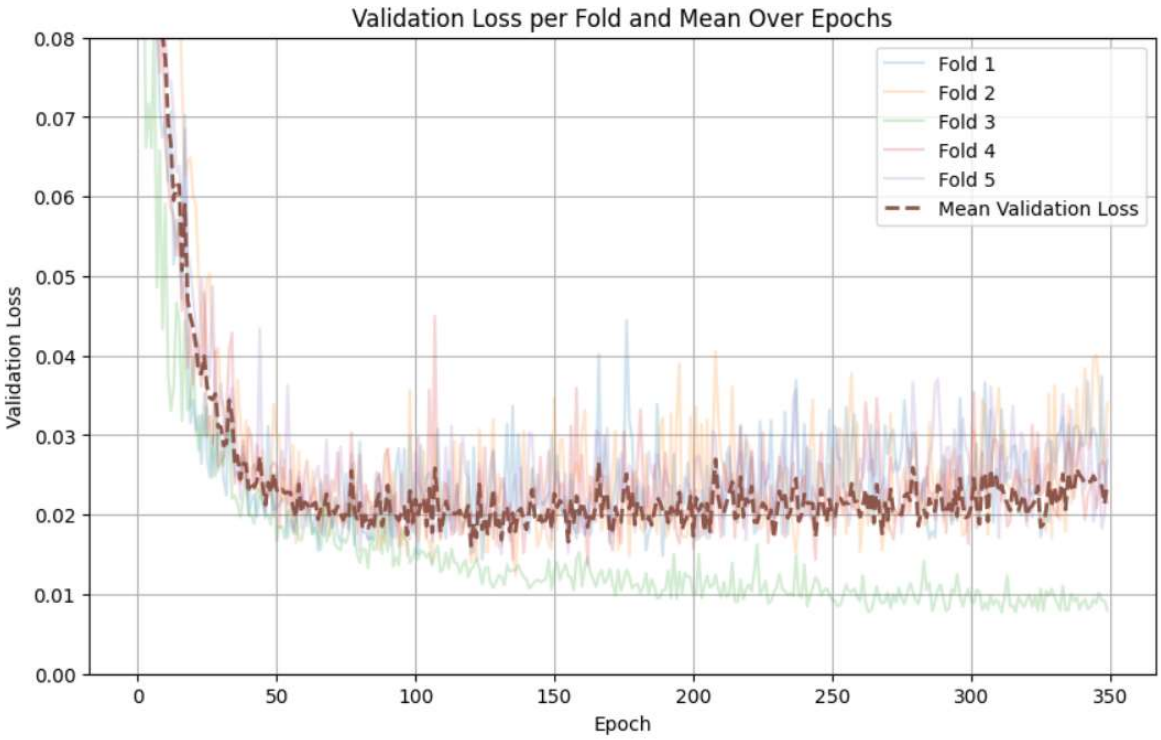

Figure 29: Model PQS-V 5-Fold CV

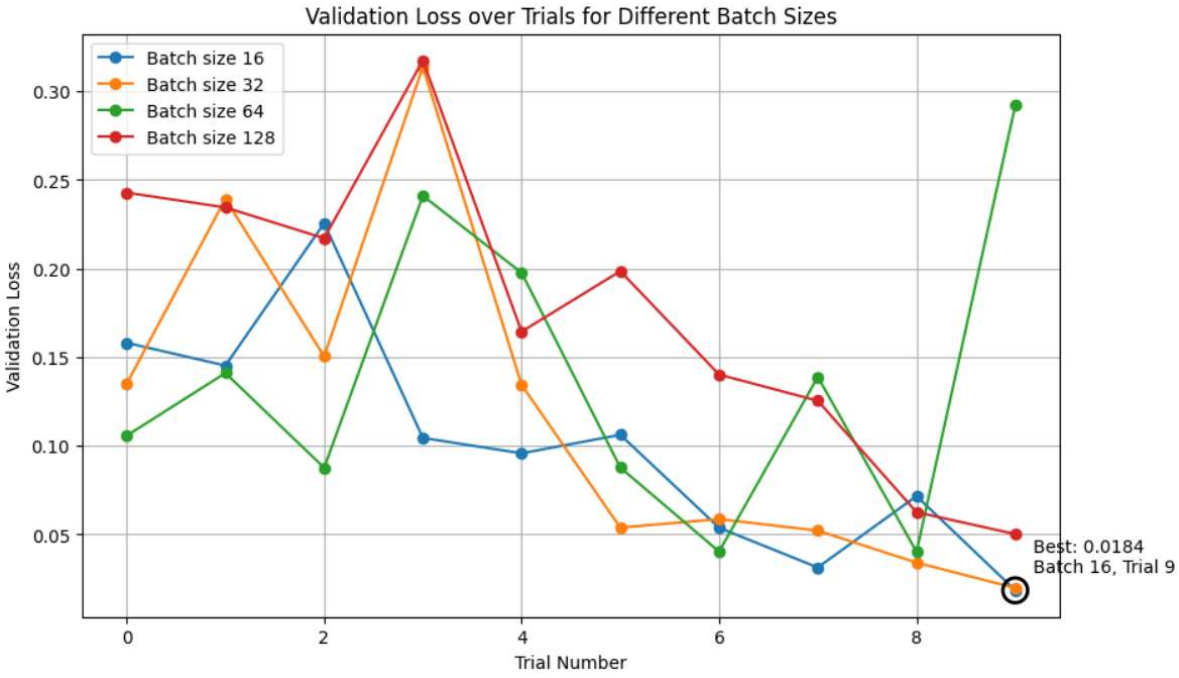

Figure 30: Hyperparameter Tuning: Loss vs. Trials PQS-V

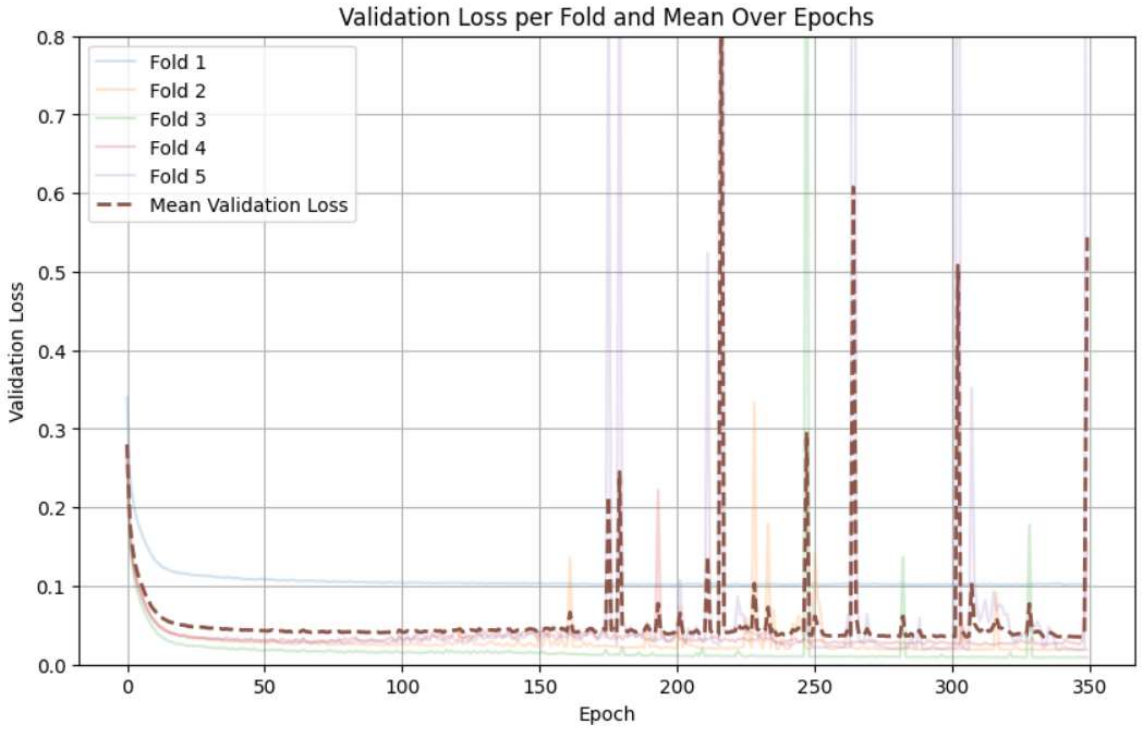

Figure 31: Model PQS-Line Flow 5-Fold CV

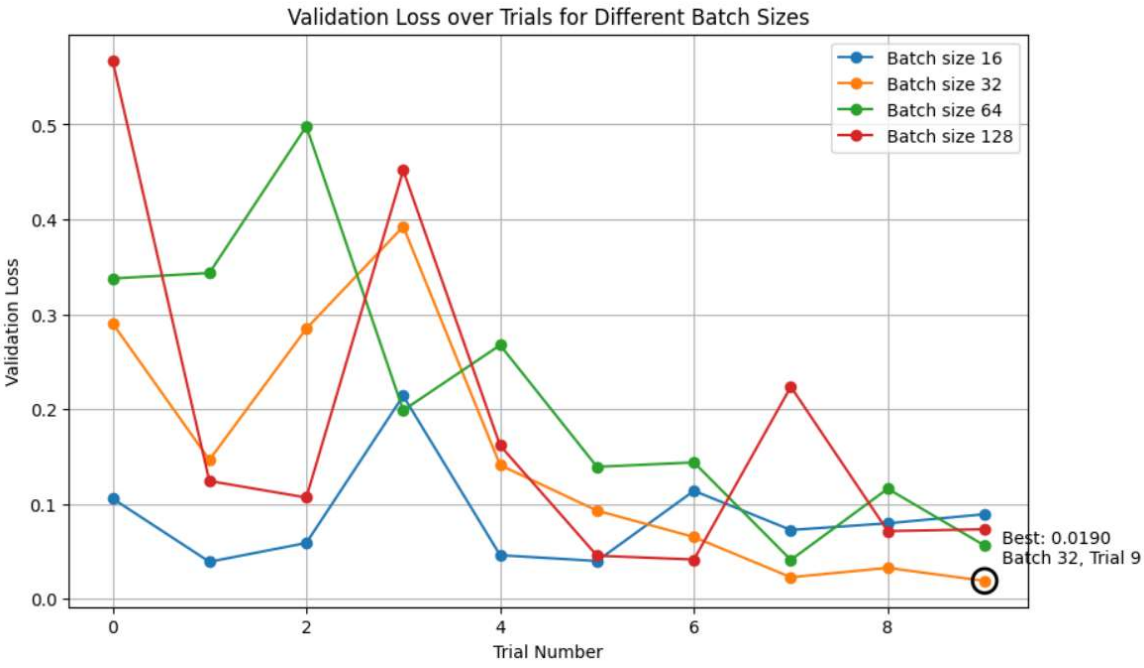

Figure 32: Hyperparameter Tuning: Loss vs. Trials PQS-Line Flow

The final model is trained for 500 epochs on that 50% train & validation set with 10% pseudo validation for best model callback. The performance metrics is given below.

Model	MSE	MAE	Std. AE	R-squared
PQS-Pg	0.1	0.181	0.26	0.9982
PQS-Qg	0.048	0.147	0.163	0.9976
PQS-V	3.21e-7	0.0003	0.0005	0.993
PQS-A	0.0013	0.024	0.027	0.997
PQS-L	0.065	0.155	0.2026	0.979

Table 20: Contingency Models Test Data Performance Metrics

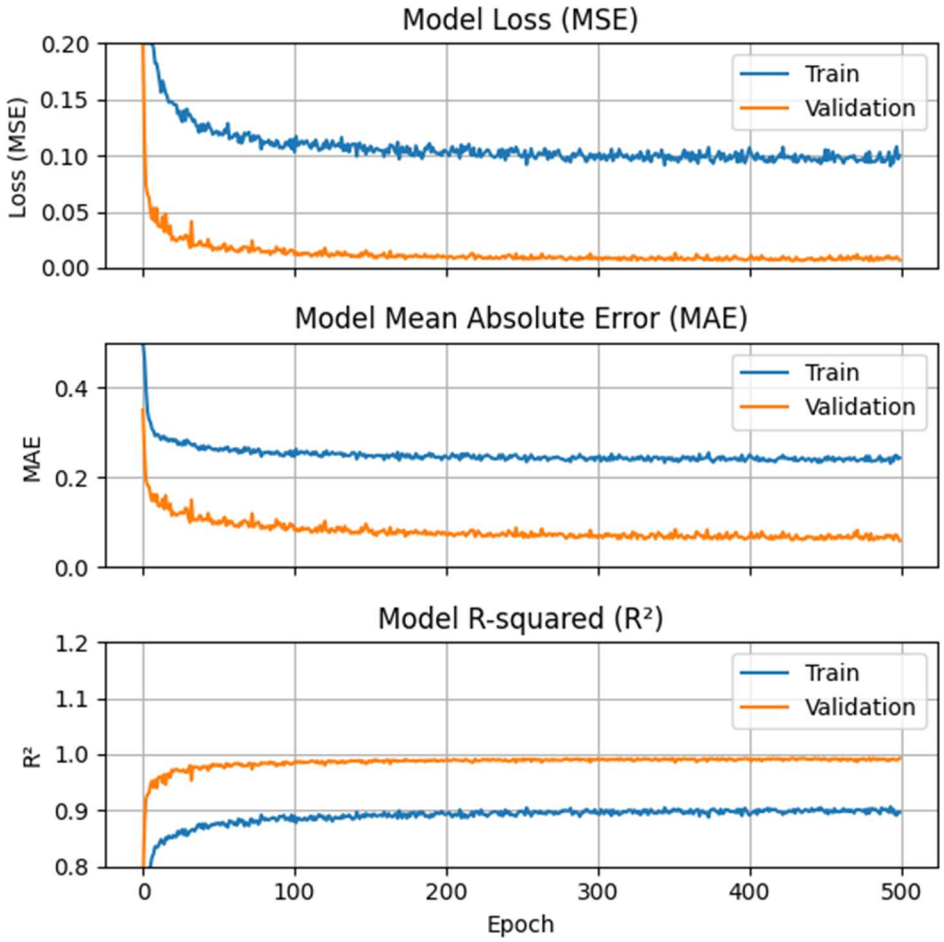

Figure 33: Model PQS-V Performance Metrics vs. Number of Epochs

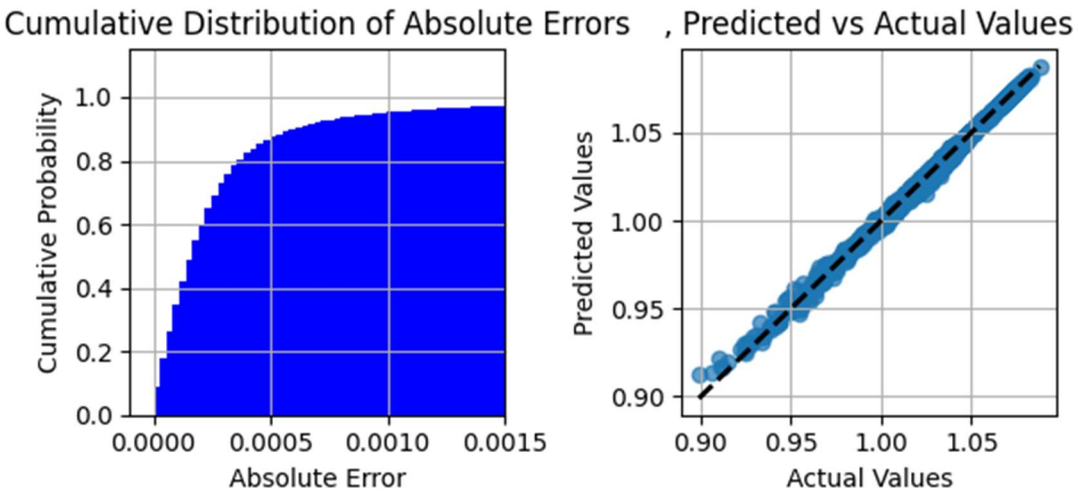

Figure 34: Model PQS-V Cumulative AE Distribution and Regression

The PQS-V model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.15 and 0.4 respectively for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 3.24e-7, MAE is 3e-4, and R^2 is 0.993, confirming that the PQS-V neural network performs exceptionally well and makes highly accurate predictions.

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The horizontal axis displays the magnitude of absolute errors, and the vertical axis corresponds to the cumulative probability of distribution. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.0015 suggests that almost all errors fall within this range.

The scatter plot visualizes the alignment between model predictions and ground-truth measurements. Each data point corresponds to a test instance, with the horizontal axis indicating observed (actual) values and the vertical axis displaying predicted outputs. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQS-V appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

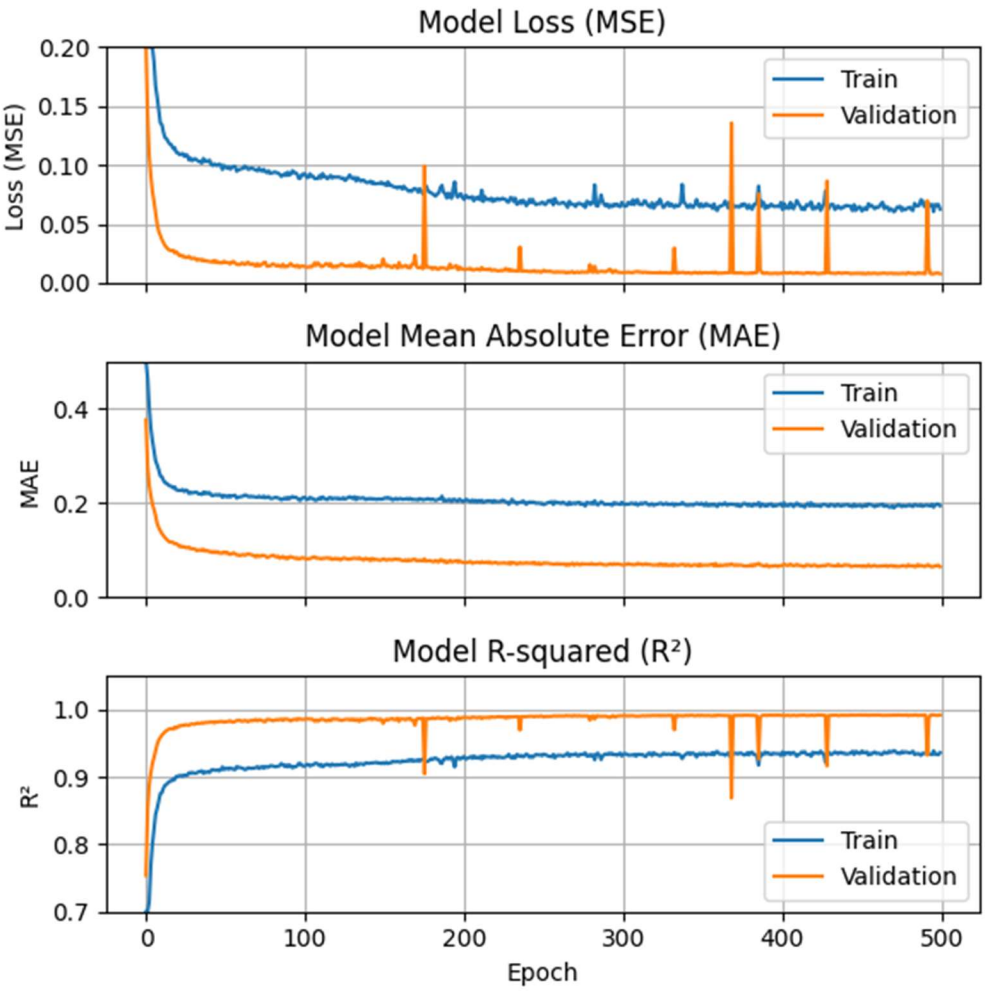

Figure 35: Model PQS-Line Performance Metrics vs. Number of Epochs

The PQS-Line model's loss (MSE) and Mean Absolute Error (MAE) steadily converged over epochs, staying below 0.15 and 0.3 respectively for both training and validation, indicating effective learning. The R^2 metric remained close to 1, showing that the model explains nearly all variance in the data. For test data, MSE is 0.07, MAE is 0.2, and R^2 is 0.979, confirming that the PQS-Line neural network performs exceptionally well and makes highly accurate predictions.

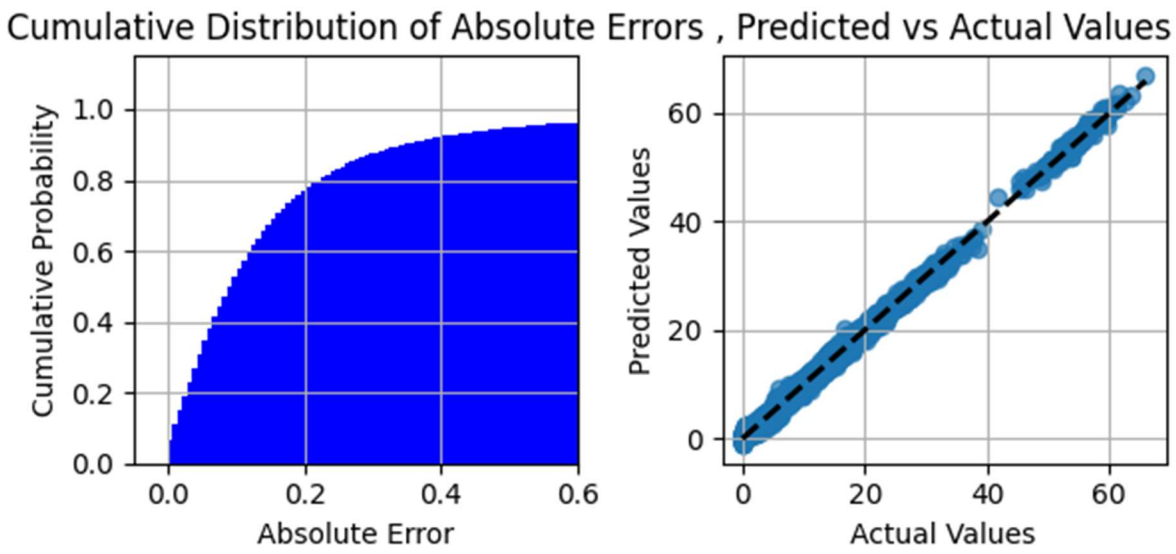

Figure 36: Model PQS-Line Cumulative AE Distribution and Regression

The cumulative distribution of error shows the cumulative probability of absolute errors in predictions. The horizontal axis displays the magnitude of absolute errors, and the vertical axis corresponds to the cumulative probability of distribution. A steep rise in the curve at lower error values suggests the most prediction errors are small. The plot reaching nearly 1 or (100%) at around 0.6 suggests that almost all errors fall within this range.

The scatter plot visualizes the alignment between model predictions and ground-truth measurements. Each data point corresponds to a test instance, with the horizontal axis indicating observed (actual) values and the vertical axis displaying predicted outputs. The black dashed line represents the ideal 1:1 relationship (i.e. perfect predictions). The points closely following dashed line indicate that the model's predictions are quite accurate. Minimal deviation from the line suggests that the model has low bias and high accuracy.

The model PQS-Line appears to have a strong predictive performance. The majority of absolute errors are small, indicating low overall prediction error. The predicted values align closely with the actual values, reinforcing the model's accuracy.

6.2.7 Performance on Arbitrary Data

An arbitrary input is given to the model and its performance is noted with respect to actual values.

Optimal Power Flow Performance:

Table 21: PQ - Pg and PQ - Qg Model Performance on Arbitrary Data

Gen	DNN PQ - Pg		AE	DNN PQ - Qg		AE
	Predicted	Actual		Actual	Predicted	
1	29.66	29.71	0.04	-5.19	-5.22	0.03
2	41.47	41.52	0.05	8.61	8.36	0.25
3	17.94	17.95	0.01	13.83	13.78	0.05
4	1.47	1.56	0.09	8.07	8.15	0.08
5	5.35	5.38	0.03	5.57	5.58	0.02
6	5.08	5.13	0.05	14.40	14.46	0.06

Generator	Cost Coefficients			Actual Gen	Predicted Gen	Actual Cost	Predicted Cost	AE
	a	b	c					
1	0.02	2	0	29.71	29.66	77.06	76.93	0.13
2	0.0175	1.75	0	41.52	41.47	102.83	102.66	0.18
3	0.0625	1	0	17.95	17.94	38.08	38.04	0.04
4	0.00834	3.25	0	1.56	1.47	5.09	4.81	0.29
5	0.025	3	0	5.38	5.35	16.85	16.77	0.08
6	0.025	3	0	5.13	5.08	16.04	15.89	0.15
Total				101.24	100.97	255.95	255.09	0.86

Table 22: PQ-Pg Model Performance on Generation Costing

The proposed deep neural network (DNN) model was evaluated on arbitrary power load inputs to assess its performance against actual optimal power flow (OPF) solutions. The model demonstrated high accuracy in predicting active power generation (Pg), with absolute errors ranging from 0.0039 to 0.0364 across all generators. Reactive power predictions (Qg) were also close to actual values, though a slightly higher error was observed for Generator 6 (AE = 0.3001). Additionally, the model accurately estimated generation costs, with a total absolute error of 0.46, indicating a strong alignment with actual cost values. These results confirm the performance of the DNN in approximating OPF problem solutions while maintaining cost estimation accuracy.

Bus	DNN PQ - V		% Err	DNN PQ - Angle		% Err
	Predicted	Actual		Predicted	Actual	
1	1.05	1.05	0.0176	0.00	0.00	-
2	1.05	1.05	0.0224	-0.48	-0.48	-0.05
3	1.04	1.04	0.0152	-1.51	-1.51	-0.04
4	1.04	1.04	0.0195	-1.81	-1.81	-0.01
5	1.04	1.04	0.0269	-1.62	-1.62	-0.37
6	1.04	1.04	0.0172	-2.12	-2.11	-0.13
7	1.03	1.03	0.0356	-2.25	-2.24	-0.34
8	1.04	1.04	0.0111	-2.31	-2.31	-0.14
9	1.04	1.04	0.0145	-2.84	-2.85	-0.11
10	1.04	1.04	0.0133	-3.22	-3.23	-0.19
11	1.04	1.04	0.0145	-2.84	-2.85	-0.11
12	1.05	1.05	0.0110	-3.43	-3.46	-0.82
13	1.07	1.07	0.0097	-3.06	-3.09	-1.03
14	1.03	1.03	0.0082	-3.84	-3.86	-0.48
15	1.04	1.04	0.0045	-3.80	-3.83	-0.75
16	1.04	1.04	0.0201	-3.47	-3.48	-0.24
17	1.04	1.04	0.0105	-3.41	-3.42	-0.35
18	1.03	1.03	0.0132	-4.10	-4.12	-0.39
19	1.02	1.02	0.0122	-4.22	-4.26	-0.95
20	1.03	1.03	0.0067	-4.00	-4.04	-0.93
21	1.05	1.05	0.0155	-3.18	-3.19	-0.23
22	1.05	1.05	0.0070	-3.11	-3.13	-0.38
23	1.05	1.05	0.0031	-3.58	-3.61	-0.80
24	1.04	1.04	0.0175	-3.50	-3.50	-0.10
25	1.04	1.04	0.0108	-3.82	-3.82	-0.03
26	1.04	1.04	0.0150	-4.19	-4.21	-0.39

27	1.05	1.05	0.0056	-3.80	-3.80	-0.02
28	1.04	1.04	0.0146	-2.36	-2.35	-0.20
29	1.03	1.03	0.0040	-4.31	-4.30	-0.35
30	1.02	1.02	0.0335	-4.71	-4.72	-0.12

Table 23: PQ - V and PQ - Angle Model Performance on Arbitrary Data

The deep neural network (DNN) model was further evaluated for voltage magnitude (V) and voltage angle (θ) predictions across the 30-bus system. The results in Table 9 demonstrate that the model effectively approximates actual values with minimal errors. For voltage magnitude prediction, percentage errors remain relatively low, with most buses exhibiting errors below 0.2%. However, slightly higher errors are observed at certain buses, such as Bus 18 and Bus 26, with errors of 0.1985% and 0.1844%, respectively. For voltage angle prediction, the model maintains a close match with actual values, though percentage errors vary across buses. The highest observed angle error is 4.5297% at Bus 2, while most other buses have errors below 1%. The results here indicates that the model gives a reliable approximation of bus voltages and angles, reinforcing its effectiveness in predicting system states for arbitrary input conditions.

The deep neural network (DNN) model's performance was also evaluated for transmission line parameters, particularly for power flow. The results indicated in Table 10 shows that the model effectively approximates actual line performance with minimal deviations. Most predicted values closely match the actual values, demonstrating the model’s ability to capture power flow dynamics accurately. Small errors are observed in certain lines, but overall, the model maintains high reliability in predicting line power transfers. These findings reinforce the DNN’s capability to provide an accurate and computationally efficient alternative to traditional power flow solutions.

Line	DNN PQS - Line		AE
	Predicted	Actual	
1	15.13	15.12	0.00
2	15.28	15.29	0.00
3	14.88	14.89	0.01
4	14.25	14.28	0.04
5	11.23	11.24	0.01
6	17.37	17.41	0.04
7	14.98	15.09	0.11
8	11.43	11.48	0.05
9	8.38	8.48	0.10
10	9.32	9.54	0.22
11	6.75	6.66	0.09
12	3.86	3.80	0.05
13	0.00	0.00	0.00
14	6.76	6.67	0.09
15	12.50	12.35	0.15
16	15.10	15.07	0.03
17	5.53	5.49	0.04
18	9.13	9.05	0.08
19	2.42	2.28	0.13
20	0.85	0.83	0.02
21	1.80	1.92	0.12
22	4.09	4.09	0.00
23	3.08	3.23	0.15
24	7.08	7.07	0.00
25	9.00	9.10	0.10
26	5.52	5.49	0.03
27	4.74	4.77	0.02
28	4.17	4.26	0.08
29	15.22	15.40	0.17
30	4.42	4.42	0.00
31	3.95	3.92	0.03
32	0.97	0.86	0.11
33	1.50	1.61	0.11
34	2.10	2.18	0.08
35	1.50	1.28	0.22
36	7.19	7.12	0.06
37	4.20	4.08	0.12
38	5.26	5.11	0.15
39	3.06	2.97	0.09
40	2.04	1.98	0.05
41	7.48	7.39	0.09

Table 24: Model PQ - Line Performance on Arbitrary Data

Contingency Performance:

An arbitrary data is taken. Voltage and Line flow predictions are presented as these are the most important parameters to determine violations.

Bus	DNN PQS - V		%Err	Max Limit	Min Limit	Violations
	Predicted	Actual				
1	1.05	1.05	0.02	1.05	0.95	0
2	1.05	1.05	0.02	1.1	0.95	0
3	1.04	1.04	0.02	1.05	0.95	0
4	1.04	1.04	0.02	1.05	0.95	0
5	1.04	1.04	0.00	1.05	0.95	0
6	1.04	1.04	0.03	1.05	0.95	0
7	1.04	1.04	0.01	1.05	0.95	0
8	1.03	1.03	0.08	1.05	0.95	0
9	1.04	1.04	0.02	1.05	0.95	0
10	1.04	1.04	0.02	1.05	0.95	0
11	1.04	1.04	0.02	1.05	0.95	0
12	1.05	1.05	0.02	1.05	0.95	0
13	1.06	1.06	0.02	1.1	0.95	0
14	1.04	1.04	0.01	1.05	0.95	0
15	1.04	1.04	0.01	1.05	0.95	0
16	1.04	1.04	0.03	1.05	0.95	0
17	1.04	1.04	0.02	1.05	0.95	0
18	1.03	1.03	0.01	1.05	0.95	0
19	1.02	1.02	0.02	1.05	0.95	0
20	1.03	1.03	0.01	1.05	0.95	0
21	1.04	1.04	0.03	1.05	0.95	0
22	1.05	1.05	0.03	1.1	0.95	0
23	1.05	1.05	0.00	1.1	0.95	0
24	1.04	1.04	0.04	1.05	0.95	0
25	1.04	1.04	0.00	1.05	0.95	0
26	1.03	1.03	0.00	1.05	0.95	0
27	1.05	1.05	0.01	1.1	0.95	0
28	1.04	1.04	0.02	1.05	0.95	0
29	1.03	1.03	0.04	1.05	0.95	0
30	1.02	1.02	0.10	1.05	0.95	0

Table 25: Model PQS-V Performance on Arbitrary Data

Line	DNN PQS - Line		AE	Line Limit	Violations
	Predicted	Actual			
1	19.36	19.88	0.52	130	0
2	17.72	17.65	0.07	130	0
3	16.52	16.32	0.20	65	0
4	16.14	16.43	0.29	130	0
5	9.53	9.54	0.01	130	0
6	18.49	18.27	0.22	65	0
7	12.91	12.73	0.17	90	0
8	9.47	9.59	0.12	70	0
9	3.26	3.73	0.47	130	0
10	11.50	11.67	0.17	32	0
11	8.95	8.96	0.01	65	0
12	5.11	5.12	0.01	32	0
13	0.00	0.00	0.00	65	0
14	8.96	8.98	0.01	65	0
15	13.50	13.53	0.03	65	0
16	13.78	13.99	0.20	65	0
17	2.67	2.54	0.13	32	0
18	5.66	5.57	0.09	32	0
19	4.44	4.39	0.05	32	0
20	0.21	0.32	0.11	16	0
21	1.15	1.05	0.10	16	0
22	5.47	5.55	0.08	16	0
23	4.77	4.82	0.05	16	0
24	5.55	5.61	0.06	32	0
25	7.79	7.84	0.05	32	0
26	5.92	5.73	0.19	32	0
27	5.86	5.93	0.06	32	0
28	4.54	4.58	0.03	32	0
29	15.13	15.11	0.02	32	0
30	3.30	3.21	0.09	16	0
31	4.06	4.03	0.03	16	0
32	2.16	2.20	0.04	16	0
33	2.79	2.93	0.14	16	0
34	3.78	3.83	0.04	16	0
35	3.19	3.06	0.13	16	0
36	10.54	10.65	0.11	65	0
37	4.79	4.71	0.07	16	0
38	5.60	5.42	0.18	16	0
39	2.85	2.80	0.06	16	0
40	3.16	2.83	0.33	32	0
41	10.26	10.06	0.20	32	0

Table 26: Model PQS - Line Performance on Arbitrary Data

6.2.8. Computational Time Performance

The model's computational time is compared with MATPOWER solver. The execution time for MATPOWER solver is 2.5 seconds whereas the time for DNN model is 0.5 seconds.

CHAPTER SEVEN: CONCLUSION AND FUTURE WORK

This research successfully developed a feedforward Deep Neural Network (DNN) to tackle the challenges of Optimal Power Flow (OPF) problem and security assessment in power systems, particularly under contingencies that are critical. By leveraging data-driven learning, the proposed approach bridges the gap between traditional iterative OPF solvers and the growing need for real-time decision-making in modern power grids. An extensive dataset was generated using MATPOWER under different loading conditions, ranging from 20% to 100% of the base load, and critical contingencies—three critical line outages and two generator outages—were identified based on system violations. This dataset provided a solid foundation for training the DNN models.

Two sets of DNN architectures were developed: a baseline DNN for predicting OPF variables under normal conditions and a contingency-aware DNN that incorporates outage status as inputs to enable real-time security assessment. The baseline DNN demonstrated high accuracy, with R^2 values up to 0.9999 and mean absolute errors (MAE) as low as 0.001 for key variables such as real/active power generations and reactive power generation, bus voltages magnitude, bus voltage angles, and transmission line flows. The contingency-aware DNN, on the other hand, reliably predicted system behavior under critical outages, with voltage and line flow errors remaining within 0.2% and 0.6%, respectively, ensuring no post-contingency violations. This capability is crucial for grid operators to proactively assess risks and maintain system stability.

One of the most significant advantages of the proposed DNN models is their computational efficiency. The models reduced computation time by 80%, taking only 0.5 seconds compared to 2.5 seconds for traditional MATPOWER solvers. This makes the DNN-based approach highly suitable for real-time applications, where speed and accuracy are paramount. The models were rigorously validated on the IEEE 30-bus system, demonstrating their ability to generalize to arbitrary inputs while maintaining alignment with traditional OPF solutions.

For future work, the framework will be extended to larger and more complex grids, such as the Integrated Nepal Power System.

REFERENCES

[1] F. Schafer, J. Menke, M. Braun, ”Contingency ¨analysis of power systems with artificial neural networks,” in 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2018, pp. 1–6

[2] A. Lotfi, M. Pirnia. ”Constraint-guided deep neural network for solving optimal power flow,” in Electric Power Systems Research, vol. 211, pp. 108353, 2022.

[3] X. Pan, ”Deepopf: deep neural networks for optimal power flow,” in Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, 2021, pp. 250–251.

[4] Samek, W., et al. ”Explaining deep neural networks and beyond: A review of methods and applications,” in Proceedings of the IEEE, vol. 109, no. 3, pp. 247–278, 2021.

[5] D. Phan, J. Kalagnanam. ”Some efficient optimization methods for solving the security-constrained optimal power flow problem,” in IEEE Transactions on Power Systems, vol. 29, no. 2, pp. 863–872, 2013.

[6] R. Zimmerman, C. Murillo-Sanchez, R. Thomas. ”MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education,” in IEEE Transactions on power systems, vol. 26, no. 1, pp. 12–19, 2010.

[7] H. Saadat, Power system analysis. McGraw-hill, 1999.

[8] J. Grainger, Power system analysis. McGraw-Hill, 1999.

[9] J. Rahman, C. Feng, J. Zhang, ”Machine learning-aided security constrained optimal power flow,” in 2020 IEEE Power and Energy Society General Meeting (PESGM), 2020, pp. 1–5.

[10] Z. Qiu, G. Deconinck, R. Belmans, ”A literature survey of optimal power flow problems in the electricity market context,” in 2009 IEEE/PES Power Systems Conference and Exposition, 2009, pp. 1–6.

[11] J. Carpentier. "Optimal power flows," in International Journal of Electrical Power & Energy Systems, vol. 1, no. 1, pp. 3–15, 1979.

[12] Cain, M., et al. "History of optimal power flow and formulations," in Federal Energy Regulatory Commission, vol. 1, pp. 1–36, 2012.

[13] G. Misyris, A. Venzke, S. Chatzivasileiadis, "Physics-informed neural networks for power systems," in 2020 IEEE power & energy society general meeting (PESGM), 2020, pp. 1–5.

[14] A. Zamzam, K. Baker, "Learning optimal solutions for extremely fast AC optimal power flow," in 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2020, pp. 1–6.

[15] Venzke, A., et al, "Learning optimal power flow: Worst-case guarantees for neural networks," in 2020 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2020, pp. 1–7.

[16] K. Baker. "A learning-boosted quasi-newton method for ac optimal power flow," in arXiv preprint arXiv:2007.06074, 2020.

[17] M. Aien, A. Hajebrahimi, M. Fotuhi-Firuzabad. "A comprehensive review on uncertainty modeling techniques in power system studies," in Renewable and Sustainable energy reviews, vol. 57, pp. 1077–1089, 2016.

Panas Bhattarai

Final Plagiarism Testing.docx

Tribhuvan University

Document Details

Submission ID
trn:oid:::3117:451390786

Submission Date
Apr 22, 2025, 5:57 PM GMT+5:45

Download Date
Apr 22, 2025, 5:59 PM GMT+5:45

File Name
Final Plagiarism Testing.docx

File Size
4.0 MB

56 Pages

8,010 Words

46,070 Characters

10% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Match Groups

75 Not Cited or Quoted 8%
Matches with neither in-text citation nor quotation marks

4 Missing Quotations 1%
Matches that are still very similar to source material

11 Missing Citation 1%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

6% Internet sources

8% Publications

0% Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.

Match Groups

- **75** Not Cited or Quoted 8%
  Matches with neither in-text citation nor quotation marks
- **4** Missing Quotations 1%
  Matches that are still very similar to source material
- **11** Missing Citation 1%
  Matches that have quotation marks, but no in-text citation
- **0** Cited and Quoted 0%
  Matches with in-text citation present, but no quotation marks

Top Sources

6% Internet sources
8% Publications
0% Submitted works (Student Papers)

Top Sources

The sources with the highest number of matches within the submission. Overlapping sources will not be displayed.

1	Internet	
www.mdpi.com		1%
2	Publication	
Umniah H. Jaid, Alia K. Abdulhassan. "Fuzzy-Based Ensemble Feature Selection for...		<1%
3	Publication	
Carlo Mari, Emiliano Mari. "Deep learning based regime-switching models of ener...		<1%
4	Internet	
www.pserc.cornell.edu		<1%
5	Publication	
Park, Byungkwon. "Sparse Tableau Formulation for Power System Networks and ...		<1%
6	Internet	
certs.lbl.gov		<1%
7	Publication	
R. N. V. Jagan Mohan, B. H. V. S. Rama Krishnam Raju, V. Chandra Sekhar, T. V. K. P...		<1%
8	Publication	
Anik Nath, A K M Abdur Rahman Chowdhury, Nur Mohammad. "Optimum Power ...		<1%
9	Internet	
www.telecomtrainer.com		<1%
10	Internet	
automl.github.io		<1%

11 Internet

arxiv.org <1%

12 Publication

Qiang Xu, Chuan He. "Evaluation of Distribution Line Fault Probability and Line R... <1%

13 Publication

Rizwan Ali, Kashif Imran, Syed Ali Abbas Kazmi, Atif Naveed Khan, Abraiz Khattak,... <1%

14 Publication

Yordan Garbatov, C. Guedes Soares. "Innovation in the Analysis and Design of Ma... <1%

15 Publication

Folorunso, Joshua. "Neural Network Approximation of Steady-State-Aware Model ... <1%

16 Internet

www.fastercapital.com <1%

17 Publication

"Proceedings of the International Conference on Systems, Control and Automatio... <1%

18 Internet

ortus0m.rtu.lv <1%

19 Publication

"Proceedings of Trends in Electronics and Health Informatics", Springer Science a... <1%

20 Internet

www.inf.ed.ac.uk <1%

21 Publication

A.R. Moazzeni. "Prediction of Lost Circulation Using Virtual Intelligence in One of ... <1%

22 Publication

Abdulrazzaq, Ali Kareem. "Electro-Thermal Modelling of Photovoltaic Systems", B... <1%

23 Publication

Chia Yu Huat, Danial Jahed Armaghani, Hadi Fattahi, Xuzhen He, Haleh Rasekh, Pi... <1%

24 Internet

vbn.aau.dk <1%

25 Internet
cyseni.com <1%

26 Internet
dspace.lib.cranfield.ac.uk <1%

27 Internet
irep2017.inesctec.pt <1%

28 Internet
psasir.upm.edu.my <1%

29 Internet
limsforum.com <1%

30 Internet
orca.cf.ac.uk <1%

31 Publication
Zimmerman, Ray Daniel, Carlos Edmundo Murillo-Sanchez, and Robert John Thom... <1%

32 Internet
kylo.tv <1%

33 Publication
Arundhati Roy, Sriparna Saha. "Chapter 10 Anomaly Detection in Respiratory Eve... <1%

34 Publication
Congcong Bai, Chengcheng Yang, Donglei Rong, Wentong Guo, Xi Gao, Wenbin Ya... <1%

35 Publication
Feiyun Zhu, Peng Liao, Xinliang Zhu, Jiawen Yao, Junzhou Huang. "Cohesion-drive... <1%

36 Publication
Jie Zhong, Zhenkan Wang. "Chapter 22 Implementing Deep Learning Models for I... <1%

37 Publication
Maria Baldeon Calisto, Susana K. Lai-Yuen. "EMONAS-Net: Efficient multiobjective... <1%

38 Publication
Sandeep Kumar Das, Supriya Sarkar. "Optimal Power Generation, Power Factor a... <1%

39 Publication

Smith, Shannon C. F.. "Characterization of Juvenile Fish Habitats Within Estuarine ... <1%

40 Internet

etheses.dur.ac.uk <1%

41 Internet

www.arxiv-vanity.com <1%

42 Internet

www.cravencountryjamboree.com <1%

43 Internet

www.oldfartsclub.com <1%

44 Publication

Guangchun Ruan, Haiwang Zhong, Guanglun Zhang, Yiliu He, Xuan Wang, Tianjia... <1%

45 Publication

Zeinab Rahimi, Mehrnoush ShamsFard. "A Knowledge-Based Approach for Recog... <1%

46 Publication

Seyed Mahdi Miraftabzadeh, Cristian Giovanni Colombo, Michela Longo, Federica ... <1%

47 Publication

Zuzana Kepesiova, Danica Rosinova, Stefan Kozak. "Comparison of Optimization T... <1%

IOE Graduate Conference
[Placeholder for Publication Information]

Deep Neural Network-Based Optimal Power Flow and Security Assessment Under Critical Contingencies

Panas Bhattarai[a], Anil Panjiyar[b], Mahammad Badrudoza[c]

[a,b,c] *Department of Electrical Engineering, Pulchowk Campus, Institute of Engineering, Tribhuvan University, Nepal*
[a] acpanasbhattarai@gmail.com

Abstract
The increasing complexity of modern power systems necessitates advanced tools for efficient operation and security assessment. This paper proposes a Deep Neural Network (DNN)-based framework to solve the Optimal Power Flow (OPF) problem and assess power system security under critical contingencies. A comprehensive dataset is generated under varying load conditions, serving as the foundation for training and validation. Critical contingencies are identified by simulating line and generator outages, with the three most severe line outages and two most severe generator outages selected based on their impact on system violations.
Two sets of DNNs are developed: the first set comprises five DNNs that has bus real and reactive loading as input and are trained to predict key system variables—real and reactive power generation, bus voltages, bus angles, and line loadings—under normal operating conditions. The second set includes five DNNs that incorporate the status of critical lines and generators as additional inputs, enabling accurate OPF solutions and security assessments during contingency scenarios. The proposed DNN-based approach offers a computationally efficient alternative to traditional OPF solvers, facilitating real-time decision-making and enhancing system resilience.
The results demonstrate the effectiveness of the DNN models in providing reliable OPF solutions and identifying security risks under both normal and contingency conditions. This work highlights the potential of machine learning techniques in advancing power system operation and planning, paving the way for smarter and more adaptive grid management strategies.

Keywords
Deep Neural Network, Power Flow, Optimal Power Flow, Security Assessment, Contingency Analysis

1. Introduction

The increasing complexity of modern power systems, driven by growing demand and integration of renewable energy sources, necessitates advanced techniques for ensuring optimal power flow (OPF) and system security. Traditional OPF methods, such as interior-point algorithms and Newton-Raphson techniques, efficiently solve optimization problems but often become computationally expensive when dealing with large-scale networks and real-time security assessments. Furthermore, power system contingencies, such as transmission line or generator outages, can significantly impact system stability, making it essential to develop fast and reliable methods for security assessment under critical contingencies.

Deep learning has emerged as a powerful tool for modeling complex, high-dimensional systems, offering real-time approximations of computationally intensive problems. In recent years, Deep Neural Networks (DNNs) have shown promising results in power system applications, including load forecasting, fault detection, and OPF solutions. By leveraging data-driven learning, DNNs can approximate the nonlinear relationships between power system variables with high accuracy while significantly reducing computation time. However, limited research has been conducted on integrating deep learning into OPF solutions that account for critical contingencies.

This paper proposes a novel DNN-based framework for OPF and security assessment under critical contingencies. The approach involves generating a large dataset of OPF solutions using the MATPOWER toolbox, incorporating various system loading conditions. The critical line and generator outages that result in the most severe system violations are

identified, and separate DNN models are trained to predict key power system parameters under both normal and contingency conditions. Specifically, two sets of DNN architectures are developed:

A baseline DNN model that predicts real and reactive power generation, bus voltages, bus angles, and line loadings based on load conditions. An extended DNN model that includes contingency conditions (status of critical line and generator outages) as additional inputs, enabling real-time security assessment. The proposed method enhances computational efficiency by bypassing iterative OPF solvers while maintaining high accuracy. It provides a rapid assessment of system security, enabling grid operators to make proactive decisions for maintaining stability under critical contingencies. The effectiveness of the model is validated using extensive simulations, demonstrating its potential for real-time power system operation and contingency management.

1.1 Optimal Power Flow

Mathematically, the optimal power flow problem can be formulated as follows.

$$\min_x f(x) \tag{1}$$

subject to

$$g(x) = 0 \tag{2}$$

$$h(x) \leq 0 \tag{3}$$

$$x_{\min} \leq x \leq x_{\max} \tag{4}$$

The objective function *f(x)* consists of the polynomial cost of generator injections, the equality constraints *g(x)* are the power balance equations, the inequality constraints *h(x)* are the branch flow limits, and the x_{min} and x_{max} bounds include reference bus angles, voltage magnitudes (for AC) and generator injections.

1.1.1 Standard AC-OPF

The optimization vector x for the standard AC OPF problem consists of the $n_b \times 1$ vectors of voltage angles Θ and magnitudes V_m and the $n_g \times 1$ vectors of generator real and reactive power injections P_g and Q_g..

$$x = \begin{bmatrix} \Theta \\ V_m \\ P_g \\ Q_g \end{bmatrix} \tag{5}$$

The objective function $f(x)$ in (1) is simply a summation of individual polynomial cost functions of real power injections, for each generator:

$$f(P_g) = \sum_{i=1}^{n_g} A_i P_{gi}^2 + B_i P_{gi} \tag{6}$$

The terms A_i and B_i are cost coefficients in quadratic fuel cost function of the generators. The equality constraints in (2) are simply the full set of $2 \cdot n_b$ nonlinear real and reactive power balance equations from (7) and (8).

$$g_P(\Theta, V_m, P_g) = P_{\text{bus}}(\Theta, V_m) + P_d - C_g P_g = 0 \tag{7}$$

$$g_Q(\Theta, V_m, Q_g) = Q_{\text{bus}}(\Theta, V_m) + Q_d - C_g Q_g = 0 \tag{8}$$

The inequality constraints (3) consist of two sets of n_l branch flow limits as non-linear functions of the bus voltage angles and magnitudes, one for the *from* end and one for the *to* end of each branch:

$$h_f(\Theta, V_m) = \left|F_f(\Theta, V_m)\right| - F_{\max} \leq 0 \tag{9}$$

$$h_t(\Theta, V_m) = \left|F_t(\Theta, V_m)\right| - F_{\max} \leq 0. \tag{10}$$

The flows are typically apparent power flows expressed in MVA, but can be real power flows (in MW) or currents, yielding the following three possible forms for the flow constraints:

$$F_f(\Theta, V_m) = \begin{cases} S_f(\Theta, V_m), & \text{apparent power} \\ P_f(\Theta, V_m), & \text{real power} \\ I_f(\Theta, V_m), & \text{current} \end{cases} \tag{11}$$

The variable limits (4) include an equality constraint on any reference bus angle and upper and lower limits on all bus voltage magnitudes and real and reactive generator injections:

$$\theta_i^{\text{ref}} \leq \theta_i \leq \theta_i^{\text{ref}}, \qquad i \in \mathscr{I}_{\text{ref}} \tag{12}$$

$$v_m^{i,\min} \leq v_m^i \leq v_m^{i,\max}, \qquad i = 1 \ldots n_b \tag{13}$$

$$p_g^{i,\min} \leq p_g^i \leq p_g^{i,\max}, \qquad i = 1 \ldots n_g \tag{14}$$

$$q_g^{i,\min} \leq q_g^i \leq q_g^{i,\max}, \qquad i = 1 \ldots n_g. \tag{15}$$

1.2 Deep Neural Network

A Deep Neural Network (DNN) is a type of artificial neural network (ANN) composed of multiple hidden layers that learn complex nonlinear mappings between inputs and outputs. A simple DNN architecture is shown in Figure 1. Mathematically, a DNN can be represented as a function $\mathbf{F} : \mathbb{R}^n \to \mathbb{R}^m$, where n is the number of input features, and m is the number of output variables.

A DNN consists of multiple layers, each performing a weighted transformation of the input followed by a nonlinear activation function. The transformation at each layer can be expressed as follows:

$$z^{(l)} = W^{(l)} a^{(l-1)} + b^{(l)} \tag{16}$$

$$a^{(l)} = \sigma(z^{(l)}) \tag{17}$$

where:

- $z^{(l)}$ is the weighted sum of inputs at layer l,
- $W^{(l)}$ is the weight matrix for layer l,
- $b^{(l)}$ is the bias vector for layer l,
- $a^{(l)}$ is the activation output of layer l,
- $\sigma(\cdot)$ is the activation function.

The final layer produces the predicted output $\hat{y}$:

$$\hat{y} = F(x;\theta) = W^{(L)} a^{(L-1)} + b^{(L)} \tag{18}$$

where L is the number of layers, and $\theta = \{W^{(l)}, b^{(l)}\}$ represents the set of learnable parameters.

1.2.1 Loss Function and Training

The DNN is trained by minimizing a loss function $\mathscr{L}$, which measures the difference between the predicted values $\hat{y}$ and the actual values y. A common choice for regression problems in OPF prediction is the **Mean Squared Error (MSE)**:

$$\mathscr{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} \|y_i - \hat{y}_i\|^2 \tag{19}$$

where N is the number of training samples. The optimization is performed using **gradient-based methods** such as the Adam optimizer:

$$\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_\theta \mathscr{L}(\theta) \tag{20}$$

where η is the learning rate, and $\nabla_\theta \mathscr{L}$ is the gradient of the loss function with respect to the network parameters.

1.2.2 Activation Functions

The activation function introduces nonlinearity to the network, enabling it to learn complex patterns. Common activation functions used in DNN include:

- **ReLU (Rectified Linear Unit):**

$$\sigma(z) = \max(0, z) \tag{21}$$

- **Sigmoid:**

$$\sigma(z) = \frac{1}{1 + e^{-z}} \tag{22}$$

- **Hyperbolic Tangent (Tanh):**

$$\sigma(z) = \frac{e^{z} - e^{-z}}{e^{z} + e^{-z}} \tag{23}$$

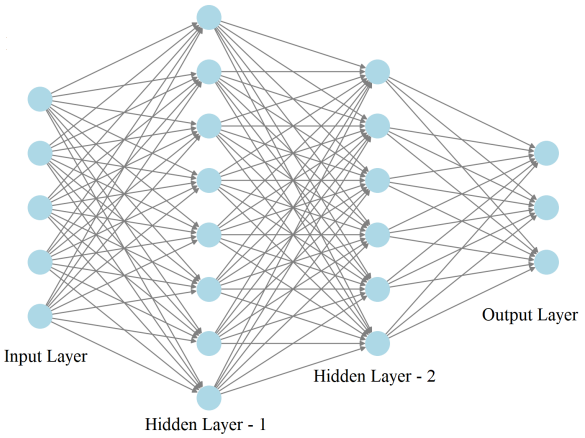

Figure 1: Deep Neural Network Architecture

2. Methodology

The methodology of this research is summarized on Figure 2 and consists of the following key steps:

2.1 Data-set Generation

To train and validate the proposed DNN models, an extensive dataset of OPF solutions is generated using MATPOWER, a MATLAB-based power system simulation tool.

2.1.1 Base-case OPF Data-set

- OPF is solved for various load scenarios to generate dataset samples
- The dataset with real and reactive power loads, optimal real and reactive power generations, bus voltages, angles, and transmission line flows is used to train base-case DNN model.

2.1.2 Contingency Simulation

1. **Line Outages:**
   - For each transmission line outage, the PF is solved to identify restrictions violations.
   - Critical line outages are selected by ranking simulated outages by total constraint violation index (sum of power flow exceedance and bus voltage limit breaches). Highest-ranking three line outages are chosen
2. **Generator Outages:**
   - For each generator outage (except for a slack generator), the PF is solved to analyze security violations.
   - The two most critical generator outages are selected by ranking the simulated outages according to the total constraint violation index (sum of power flow exceedance and bus voltage limit breaches).

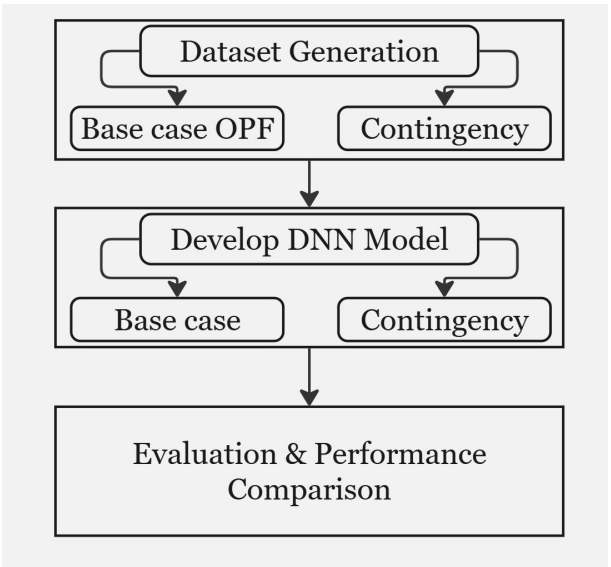

Figure 2: Methodology Summary

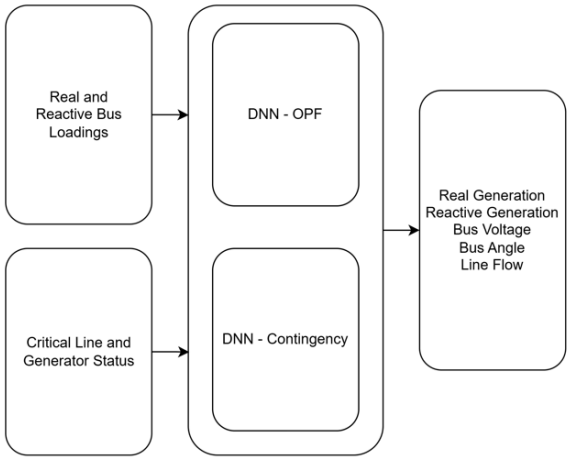

Figure 3: DNN Model

2.2 Development of DNN Models

Two sets of Deep Neural Network (DNN) models are designed for the prediction of OPF and the assessment of security under critical conditions.

2.2.1 Baseline DNN for OPF Prediction

Input: Real and reactive power demands at each bus.
Output:

- Real and reactive power generation at each generator.
- Voltage magnitudes and angles in all buses.
- Power flow in transmission lines.

Architecture:

- Hyperparameter tuning is done via grid search over hidden layers (1–3) and activations (tanh, ReLU). Two hidden layers with tanh gave best validation MSE.
- One input layer (size = number of buses $\times$ 2).
- Two hidden layers with Tanh activation.
- One output layer (size = number of predicted OPF variables).

Training:

- Loss function: **Mean Squared Error (MSE)**.
- Optimizer: **Adam** with a learning rate of **0.001**.

2.2.2 Contingency-Aware DNN for Security Assessment

Input:

- Real and reactive power demands at all buses.
- Binary indicators for critical line outages (0: in service, 1: out of service).
- Binary indicators for critical generator outages (0: in service, 1: out of service).

Output: Same as baseline OPF model.

Architecture:

- Similar to the baseline DNN, but with additional inputs for contingency status.

Purpose:

- Enables real-time security assessment by predicting power system parameters under critical outages without solving PF.

2.3 Model Training and Validation

Dataset Split:

- 80% training, 20% testing.

Performance Metrics:

- Mean Absolute Error (MAE), Mean Squared Error (MSE) and Coefficient of Determination.
- Computation time comparison with traditional OPF solvers.

3. Result

The study for this research is conducted on IEEE 30-bus test system, which is a well-established benchmark for power system optimization and security assessment studies. The system consists of 41 transmission lines, 6 generators, and 30 total buses, providing a realistic testbed for analyzing optimal power flow (OPF) and contingency scenarios.

3.1 Dataset Overview

For base case OPF, 10,000 data are generated using MATPOWER. The load is varied from minimum 20% to 100% of standard loading. The power factor for each load is varied from minimum 0.75 lag to unity. While actual load profiles can improve realism for system-specific studies, the IEEE 30-bus network is a synthetic benchmark whose primary purpose is algorithm validation. Random variation of loads between 20–100% is an established practice in OPF literature to uniformly sample the full feasible operating space and ensure model generalization across all loading conditions . Indeed, prior works solving OPF with DNNs on IEEE test systems have employed similar random sampling to comprehensively assess performance (e.g., Lotfi & Pirnia 2022; Pan 2021). Therefore, the approach appropriately captures the range of operating points needed to evaluate model accuracy on this benchmark

For contingency case, 1000 data are generated for each three most critical line outage and two most critical generator outage. Severe line and generator outage are given in Table (1).

Type	ID	Total Violations
Line Outage	Line 8	1025
Line Outage	Line 10	418
Line Outage	Line 38	348
Generator Outage	Gen 2	403
Generator Outage	Gen 4	388

Table 1: Line and Generator for critical contingency

3.2 DNN Model Training Performance

Each DNN model is trained for 500 epochs, with 20% of the training data held out for validation. A ModelCheckpoint callback recorded and saved the weights corresponding to the epoch with the lowest validation MSE. The resulting validation MSE, MAE, and R^2 for each best model are summarized in Table 2

3.3 Model Prediction Accuracy

Mean Squared Error (MSE), Mean Absolute Error (MAE) and Coefficient of Determination R^2 between DNN and MATPOWER test data is calculated for real power generation, reactive power generation, bus voltage, bus angle and line flows. The results are presented in Table 3. The prediction errors are minimal compared to typical system uncertainties,

DNN Model	MAE	MSE	R^2
BaseOPF-Pg	0.0924	0.0178	0.99
BaseOPF-Qg	0.1822	0.0807	0.9986
BaseOPF-V	0.0016	4e-6	0.95
BaseOPF-A	0.0182	7e-4	0.9994
BaseOPF-S	0.0268	0.114	0.9992
Contingent-Pg	0.1303	0.0365	0.99
Contingent-Qg	0.17	0.0498	0.9994
Contingent-V	0.0016	4e-6	0.9697
Contingent-A	0.0316	0.002	0.9987
Contingent-S	0.2057	0.0787	0.9985

Table 2: Accuracy and Loss for Validation Data

DNN Model	MSE	MAE	R^2
Base OPF - Pg	0.012	0.047	0.9982
Base OPF - Qg	0.012	0.047	0.9999
Base OPF - V	2e-6	0.001	0.9782
Base OPF - A	5e-4	0.02	0.9996
Base OPF - S	0.03	0.111	0.9992
Contingent - Pg	0.04	0.13	0.9944
Contingent - Qg	0.04	0.1421	0.9994
Contingent - V	4e-6	0.002	0.965
Contingent - A	0.002	0.03	0.999
Contingent - S	0.08	0.2	0.9985

Table 3: MSE and RMSE between Predicted and Actual Data

with key parameters predicted within acceptable operational tolerances. Thus, the low MAE/MSE values support their practical use in real-time decision-making.

3.4 Computation Time Comparison

The time to compute MATPOWER-based OPF and DNN-based OPF is shown in Table 4. The results shows that the DNN can give approximations to the Optimal Power Flow problem five times faster than MATPOWER. MATPOWER uses Matlab Internal Point Solver to solve the optimization problem which takes multiple iteration to obtain accurate solutions.

Solver	MATPOWER	DNN
Execution Time	2.5 secs	0.5 secs

Table 4: Execution time for MATPOWER and DNN

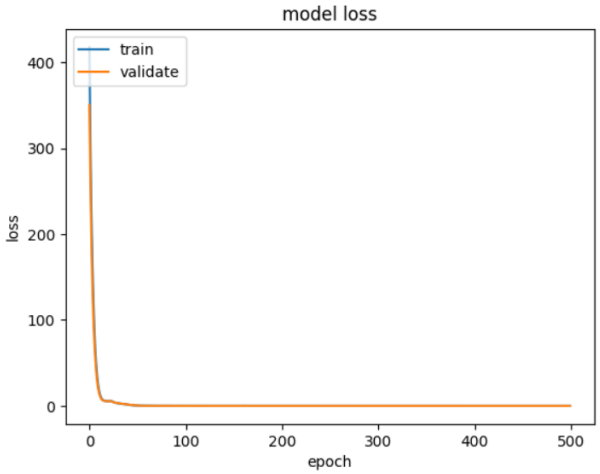

Figure 4: Training and Validation Loss Curve for PQ-Pg DNN

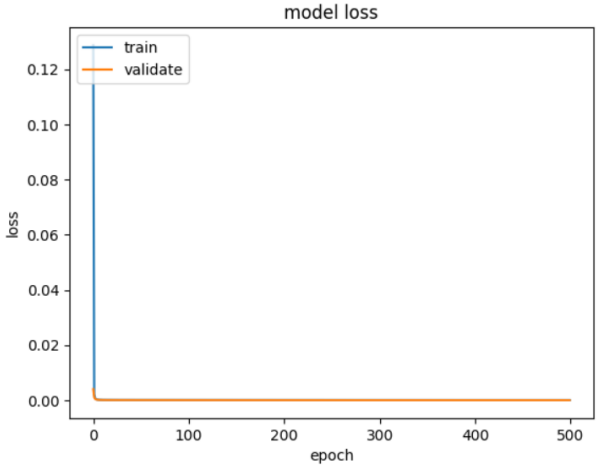

Figure 5: Training and Validation Loss Curve for PQ-V DNN under Contingency

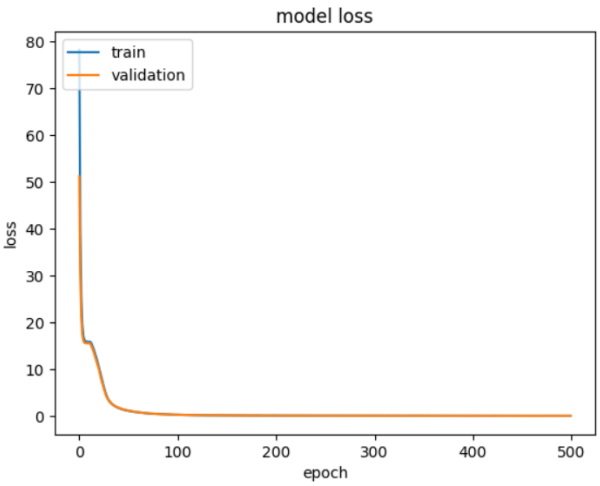

Figure 6: Training and Validation Loss Curve for PQ-S DNN under Contingency

4. Conclusion

This paper presented a Deep Neural Network (DNN)-based framework for solving the Optimal Power Flow (OPF) problem and assessing power system security under critical contingencies. By leveraging a large dataset generated through MATPOWER simulations, two sets of DNN models were developed: one for standard OPF prediction and another for contingency-aware security assessment. The results demonstrated that the proposed DNN approach provides approximately accurate OPF solutions while significantly reducing computation time compared to traditional solvers.

The ability of the DNN models to capture nonlinear relationships between system variables allows for fast and reliable contingency analysis, making them a viable alternative for real-time power system operation. The contingency-aware model effectively predicted system states under severe line and generator outages.

References

[1] F. Schäfer, J. Menke, M. Braun, ”Contingency analysis of power systems with artificial neural networks,” in 2018 IEEE International Conference on Communications, Control, and Computing Technologies for Smart Grids (SmartGridComm), 2018, pp. 1–6.

[2] A. Lotfi, M. Pirnia. ”Constraint-guided deep neural network for solving optimal power flow,” in Electric Power Systems Research, vol. 211, pp. 108353, 2022.

[3] X. Pan, ”Deepopf: deep neural networks for optimal power flow,” in Proceedings of the 8th ACM International Conference on Systems for Energy-Efficient Buildings, Cities, and Transportation, 2021, pp. 250–251.

[4] Samek, W., et al. ”Explaining deep neural networks and beyond: A review of methods and applications,” in Proceedings of the IEEE, vol. 109, no. 3, pp. 247–278, 2021.

[5] D. Phan, J. Kalagnanam. ”Some efficient optimization methods for solving the security-constrained optimal power flow problem,” in IEEE Transactions on Power Systems, vol. 29, no. 2, pp. 863–872, 2013.

[6] R. Zimmerman, C. Murillo-Sanchez, R. Thomas. ”MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education,” in IEEE Transactions on power systems, vol. 26, no. 1, pp. 12–19, 2010.

[7] H. Saadat, undefined. others, Power system analysis. McGraw-hill, 1999.

[8] J. Grainger, Power system analysis. McGraw-Hill, 1999.

[9] J. Rahman, C. Feng, J. Zhang, ”Machine learning-aided security constrained optimal power flow,” in 2020 IEEE Power and Energy Society General Meeting (PESGM), 2020, pp. 1–5.

[10] Z. Qiu, G. Deconinck, R. Belmans, ”A literature survey of optimal power flow problems in the electricity market context,” in 2009 IEEE/PES Power Systems Conference and Exposition, 2009, pp. 1–6.

Gmail **Panas Bhattarai <acpanasbhattarai@gmail.com>**

[IOEGC16] Editor Decision

1 message

Suwarna Lingden <conference-noreply@ioe.edu.np> Wed, Apr 2, 2025 at 12:16 AM
To: Panas Bhattarai <acpanasbhattarai@gmail.com>, Anil Panjiyar <anil.panjiyar@pcampus.edu.np>, Mahammad Badrudoza <mbadrudoza@ioe.edu.np>

Panas Bhattarai, Anil Panjiyar, Mahammad Badrudoza:

We are pleased to inform you that your manuscript titled "Deep Neural Network-Based Optimal Power Flow and Security Assessment Under Critical Contingencies" submitted to 16th IOE Graduate Conference is **Accepted** for presentation in the Conference as well as inclusion in the Peer-Reviewed Proceedings. Please note that inclusion in hard copy proceedings is contingent upon your timely response to further edits, if any, during the publication process.

With Warm Regards,
IOEGC-16 Editorial Team