IMPLEMENTATION OF A DEEP LEARNING-BASED PREDICTIVE MAINTENANCE SYSTEM FOR TRANSFORMERS

BY

ORUGUN DAVID MODUPE

18CK024251

USER Manual

JUNE 2024

Transformer Failure Prediction System User Manual

Introduction

Welcome to the Transformer Failure Prediction System user manual. This document provides comprehensive guidance on setting up, using, and maintaining the system. The system leverages Convolutional Neural Networks (CNN) and Continuous Wavelet Transform (CWT) analysis to predict real-time transformer failures, enhancing power grids' reliability and stability.

Table of Contents

7. Contact Information

System Requirements

1. **Hardware**
    i. A computer with a minimum of 8GB RAM and a modern multi-core processor.
    ii. A reliable internet connection for updates and cloud-based processing (if applicable).
    iii. Power transformers equipped with current measurement devices.
    iv. A data acquisition system is used to capture phase current waveforms.

2. Software

i. Operating System: Windows, macOS, or Linux.
ii. Python 3.7 or higher.
iii. Python libraries, such as Streamlit, TensorFlow, Keras, NumPy, and Matplotlib, are required.

Setting up the System

Hardware Setup

1. Current Measurement Devices:

i. Install current transformers (CTs) on the power transformers to measure the phase currents.
ii. Ensure CTs are securely connected to the power lines and adequately rated for the current levels**.**

2. Data Acquisition System:

i. Connect the outputs of the CTs to a data acquisition (DAQ) system.
ii. Configure the DAQ system to sample the phase current waveforms at an appropriate rate (e.g., 10 kHz or higher).
iii. Ensure the DAQ system is calibrated correctly and synchronised with the power system's operational frequency.

3. Signal Processing Unit:

i. A signal processing unit or a computer with a digital signal processing (DSP) capability can process the captured waveforms.
ii. Implement a Continuous Wavelet Transform (CWT) algorithm to convert the time-domain waveforms into scalogram images.

4. Data Storage and Transfer:

i. Store the generated scalogram images on a local or cloud-based storage system.
ii. Ensure the images are correctly labelled and organised for easy access and analysis**.**

Software Installation

i. Install Python from the official website.
ii. Open a terminal or command prompt and install the required libraries using the following command: Pip install streamlit tensorflow keras numpy matplotlib
iii. Clone the project repository from the provided source.

Using the Web Interface

Streamlit Configuration

i. Open the terminal or command prompt.
ii. Navigate to the project directory.
iii. Run the Streamlit application using the command: streamlit run app.py

Loading and Encoding Images

i. Background and other necessary images are loaded and encoded in base64 format for embedding into the HTML/CSS of the application**.**

User Authentication

i. A login mechanism is implemented in the sidebar where users must input valid credentials to access the system.

Model Selection and Loading

i. Users can select from multiple models available for analysis.
ii. Load the selected model and its corresponding label files.

Uploading Scalogram Images

i. Users can upload multiple scalogram images (generated from transformer current waveforms) for analysis.

Running the Analysis

i. After uploading the images and selecting the model, click the "Run Analysis" button to initiate the process.
ii. The system will process the images and display the results.

Displaying Results

i. The combined scalogram image and prediction results will be displayed on the interface.

Maintenance and Troubleshooting

Regular Maintenance

i. Ensure the software and libraries are up-to-date.
ii. Regularly check hardware connections and replace any faulty components.
iii. Periodically clean the system to avoid dust accumulation, which may cause overheating.

Common Issues and Solutions

1. Issue: The web interface is not loading.

Solution: Check the terminal for errors, ensure all dependencies are installed, and verify internet connectivity.

2. Issue Incorrect predictions**.**

Solution: Verify the quality and format of the uploaded scalogram images. Ensure the model is trained correctly.

Recommendations for Best Practices

i. Enhance data augmentation techniques to increase training data diversity.
ii. Explore different CNN architectures and hyperparameters for optimisation.
iii. Implement a continuous learning framework for the model**.**

iv. Ensure seamless and scalable deployment, potentially using cloud-based solutions.
v. Develop an intuitive user interface for easy interaction.
vi. Implement robust security measures for data protection.

Contact Information

For further assistance, please contact our support team at [orugundavid6@gmail.com].