

User Manual

Telecom Churn Prediction System

1. System Overview

This manual guides users through installation, setup, and operation of the Telecom Churn Prediction System — a machine learning and Flask-based web application to predict customer churn in the telecom sector.

2. Prerequisites

- Python 3.10 or above
- Conda or Virtual Environment setup
- Internet connection for installing dependencies

3. Installation Steps

Step 1: Clone the Repository

```
git clone https://github.com/Devanshpratapsingh28/Telecom-Churn-SE  
cd Telecom-Churn-SE
```

Step 2: Create a Virtual Environment

```
conda create -p venv python==3.10 -y
```

Step 3: Activate the Environment

```
conda activate venv/
```

Step 4: Install Dependencies

```
pip install -r requirements.txt
```

4. Running the Application

Start the Flask Server:

```
python app.py
```

Access the Web App:

- Open the displayed URL (e.g., `http://127.0.0.1:5000/`)
- To train the model, visit `/train` endpoint: `http://127.0.0.1:5000/train`
- Return to home page and enter input values to predict churn.

5. Directory Outputs

- `artifacts/` – Stores trained model and related outputs.
- `logs/` – Execution logs for debugging and tracking.
- `data/` – Contains dataset files.
- `templates/` – HTML files for web interface.
- `static/` – CSS styling for the front-end.

6. Troubleshooting

Issue	Possible Cause	Solution
Flask app not starting	Port already in use / Missing dependencies	Stop old process or reinstall requirements
Dataset error	Incorrect schema	Verify and update schema.yaml file
Model not training	Missing configuration files	Check config.yaml and data paths

7. Expected Outputs

- Trained model metrics and accuracy scores
- Log entries of training and prediction
- Web interface displaying churn prediction

8. Notes

- Modify hyperparameters via `params.yaml` for tuning.
- Clear `artifacts/` before retraining to avoid conflicts.
- Logs are auto-generated for each run for traceability.