

> EdgePurify: Real-Time Dataset Cleaner for Edge AI



Do you know why your test results of Neural Network gone wrong? may be your dataset?!

SOLUTION OVERVIEW

- **Solution:** Our project is a WEB Solution , where you upload your train and test datasets, we use our real-time edge AI dataset purifier that uses a custom feature-scoring neural network and Image filters to analyze and suggests datasets faults and improvements by assessing image quality based on test dataset itself!
(know why model went wrong?)
- **Problem Addressed:** This tool addresses the challenge of noisy and uncertain data in image datasets, which is crucial for creating accurate models, especially for applications on edge devices like Raspberry Pi.

Innovation and Uniqueness

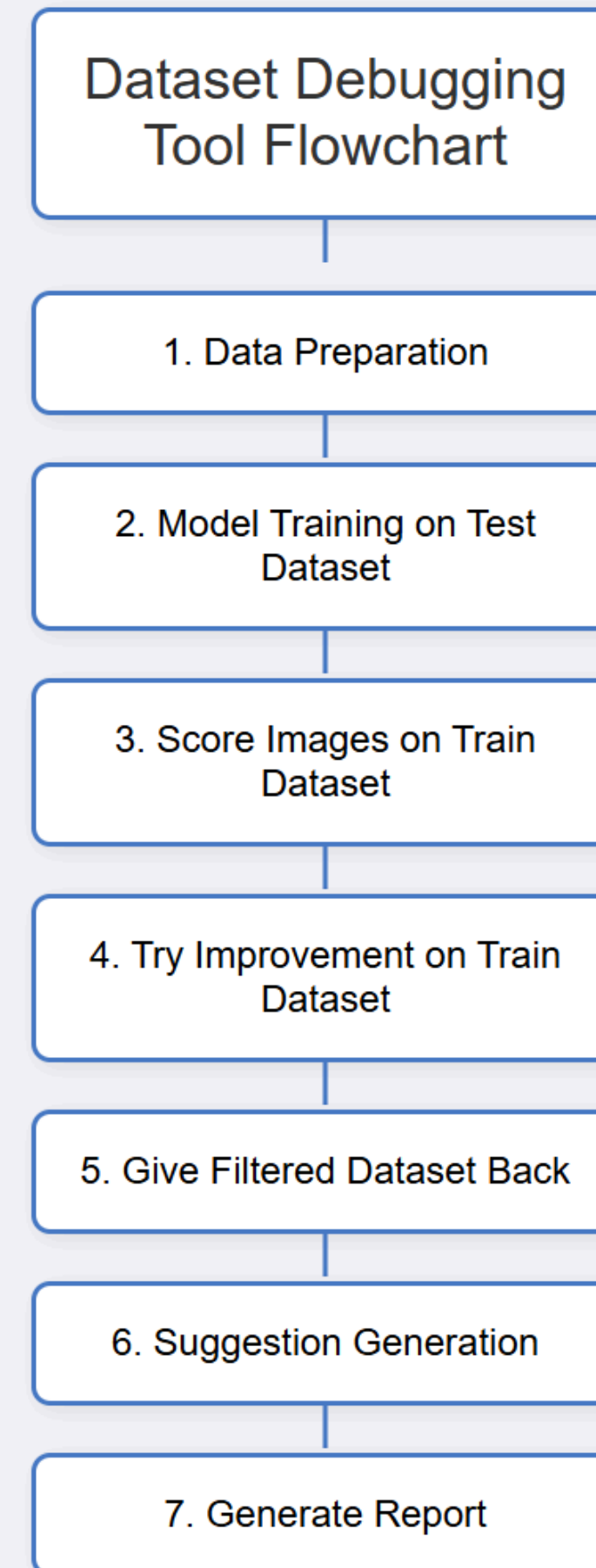
- Model-Aware Dataset Analysis
- Training on Test Dataset for Error Diagnosis
- Actionable Dataset Improvement Suggestions
- CUSTOM NEURAL NETWORK

Key Features and Benefits

- Root Cause Analysis of Accuracy Drops
- Performance-Linked Data Insights
- Targeted Dataset Augmentation Recommendations
- Scalable and Modular Architecture
- Plug-n-play and web solution too

TECHNICAL ARCHITECTURE

- **Programming language:** Python
- **Deep Learning Framework:** TensorFlow
- **Data Processing:**
 - NumPy:** Numerical computations
 - Pandas:** Data manipulation and analysis
 - OpenCV:** Image processing
- **Model Training & Evaluation:**
- Keras (with TensorFlow) for building and training models
- Scikit-learn for metrics and data splitting
- Influence Analysis Tools:
- **TensorFlow :** For gradient calculations
- Deep Explain or custom implementations for influence functions
- **Visualization:**
 - Matplotlib:** For static visualizations
 - Seaborn:** For statistical data visualization
- **FrontEnd :** HMTL,CSS
- **Flask :** For web interface
- **Version Control:**
- Git
- **Google Colab (for prototyping)**



SCALABILITY AND FUTURE SCOPE

Scalability:

- **Cloud Integration:** Future iterations could offload processing to cloud services when handling large datasets
- **Load Balancing:** For larger implementations, the model could be containerized (Docker) and scaled using microservices architecture.

Future Functionalities:

- **Advanced Filtering Options:** Allow users to set their quality thresholds.
- **Extended Compatibility:** Compatibility with other edge devices for expanded applicability
- **Feedback-Driven Model Tuning:** A self-adapting threshold mechanism for continuous improvement.

FEASIBILITY

Challenges and Risks:

- **Limited Generalization (Self-Betrayal)**
- **Bias Introduction**
- **Noise in Input Data:** Risk of incorrectly filtering out valid images.

Mitigation Strategies:

- **Quantization:** Reduce model size without sacrificing accuracy
- **Balanced removal**
- **Image filtering**



>Team Details



Dontu Kowshik
22BCE9556

Team Lead



Durgam Moksha sree
22BCE9979

Team Member



Jyesta Satya priya
22BCE8445

Team Member



**HACKTOBER
FEST**



Thank You

