Edge AI Deployment in Remote Extractives Locations

Author: Liezl Coetzee

Created with Pi

CONTENTS

1. Practical Implementation

Implementation Guide

Core Architecture: “AI in a Box” Systems

Physical Hardware Solutions

Example Setup: “Community Voice Monitor”

Software Architecture: Offline-First Design

Implementation Scenarios

Scenario 1: Community Grievance Processing

The Setup

How It Works

Technical Details

Scenario 2: Environmental Monitoring

Scenario 3: Resettlement Progress Monitoring

Core Architecture: “AI in a Box” Systems

Physical Hardware Solutions

Ruggedized Edge Computing Units: Designed to withstand harsh environments.

ARM-based Processors: Low power consumption and high efficiency, examples include NVIDIA Jetson and Raspberry Pi Compute modules.

Satellite Communication Modules: Utilizing Iridium, Starlink, or regional satellite networks for reliable connectivity.

Industrial Mini-PCs: Fanless, weatherproof units with an IP65+ rating.

Solar + Battery Systems: Providing 3-7 days of autonomy without sunlight, ensuring continuous operation.

Local Storage: SSDs ranging from 500GB to 2TB for storing model weights and data buffering.

Example Setup: “Community Voice Monitor”

Physical Package (\~shoebox size):

- NVIDIA Jetson AGX Xavier for AI processing.
- 4G/5G modem with satellite backup for internet access.
- Solar panel (100W) coupled with a lithium battery (400Wh) for power.
- Weatherproof enclosure to protect components.
- Local WiFi hotspot for connecting community devices.

Cost: Estimated between $2,000 to $4,000 per unit.

Software Architecture: Offline-First Design

Core Components

- Local AI Models: Implementing compressed and quantized versions of larger models for efficiency.
- Data Pipeline: Automated processing and storage of data collected locally.
- Sync Engine: Facilitating intelligent data transmission when connectivity is available.
- Fallback Systems: Providing human-readable alerts and basic functionality during outages.

Model Compression Techniques

- Quantization: Reducing model size by 4-8 times with minimal accuracy loss.
- Pruning: Removing unnecessary neural network connections to streamline processing.
- Knowledge Distillation: Training smaller "student" models from larger "teacher" models for efficiency.
- Edge-specific Optimization: Utilizing frameworks like TensorRT and OpenVINO for better performance.

Scenario 1: Community Grievance Processing

The Setup

Location

A village located 200km from the nearest town with spotty cell coverage.

Hardware

Solar-powered edge unit equipped with satellite uplink.

Local Interface

A WhatsApp-like application on community smartphones for easy access.

How It Works

1

Community members can record voice complaints in their local language through the smartphone app.

2

Edge AI processes the speech-to-text conversion, translation, and sentiment analysis locally.

3

A classification system determines the urgency of the complaint (emergency, routine, information).

4

Immediate responses are generated, with local storage and automated acknowledgment to the complainant.

5

The system syncs when possible, uploading batched data to a central system during connectivity windows.

Technical Details

```
class GrievanceEdgeProcessor:    def _init_(self):        self.speech_to_text = load_compressed_model('whisper_small')
self.translator = load_model('nllb_distilled')        self.classifier = load_model('grievance_classifier_quantized')
self.urgency_detector = load_model('urgency_model')    def process_complaint(self, audio_file):        /# All
processing happens locally            text = self.speech_to_text(audio_file)        english_text = self.translator(text,
target='en')        category = self.classifier(english_text)        urgency = self.urgency_detector(english_text)            /#
Store locally, sync later            self.store_locally({            'audio': audio_file,            'text': text,            'translation':
english_text,            'category': category,            'urgency': urgency,            'timestamp': time.now()        })            /#
Immediate response to community            return self.generate_acknowledgment(category, urgency)
```

Scenario 2: Environmental Monitoring

The Setup

- Location: Mine site perimeter with no cellular coverage.
- Hardware: Weather-resistant sensor stations integrated with edge AI.
- Sensors: Monitoring air quality, water quality, noise, and vibrations continuously.

How It Works

- Sensors continuously collect environmental data.
- Edge AI analyzes patterns and detects anomalies locally.
- Immediate alerts for threshold breaches are sent (via satellite if critical).
- Daily summaries are uploaded during satellite communication windows.
- Predictive analysis runs locally to forecast potential issues.

Power Management

- Adaptive Processing: Reducing AI computation during low-power periods.
- Intelligent Scheduling: Running complex analyses when solar charging is optimal.
- Hierarchical Alerting: Using local sirens/lights for immediate issues and satellite for more complex analyses.

Scenario 3: Resettlement Progress Monitoring

The Setup

- Location: A new resettlement village with intermittent connectivity.
- Hardware: Community kiosk equipped with edge AI capabilities.
- Interface: Tablet-based surveys and photo documentation.

How It Works

- Community liaisons conduct household surveys using tablets.
- Edge AI processes responses, photos, and GPS coordinates locally.
- Quality checks are conducted immediately for missing data or inconsistencies.
- Progress tracking updates are displayed on local dashboards.
- Batch sync uploads processed data whenever connectivity allows.

2. Connectivity Solutions

Satellite Options

- Iridium: Offers global coverage with low bandwidth (\~2.4 kbps) and high reliability.
- Starlink: Provides higher bandwidth (\~100 Mbps) with expanding coverage.
- Regional Satellites: Options like VSAT, Thuraya, and Inmarsat depending on specific location needs.

Terrestrial Backup

- 4G/5G: Used as the primary connection where available.
- Long-range WiFi: Establishing point-to-point links to distant communication towers.
- Mesh Networks: Multiple units creating a local communication network for enhanced connectivity.

Sync Strategies

- Differential Sync: Only uploading changes rather than full datasets to save bandwidth.
- Compression: Implementing aggressive data compression before transmission.
- Prioritization: Ensuring critical alerts are sent first, followed by routine data.
- Scheduling: Syncing during optimal satellite communication windows for efficiency.

3. Data Management: Challenges & Solutions

1 Storage Constraints

- Local Databases: Utilizing SQLite or similar lightweight options for data management.
- Data Lifecycle: Implementing automatic cleanup processes for old, synced data.
- Critical Data Protection: Ensuring multiple backups of essential information for security.

2 Model Updates

- Over-the-Air Updates: Downloading new model versions whenever connectivity allows.
- Staged Rollouts: Testing new models on a subset of devices before full deployment.
- Fallback Mechanisms: Keeping the previous model version as a backup to ensure continuity.

3 Security Considerations

- Local Encryption: Ensuring all data is encrypted at rest for security.
- Secure Boot: Preventing tampering with edge units during operation.
- VPN Connections: Utilizing encrypted data transmission for enhanced security.
- Physical Security: Implementing tamper-evident enclosures to protect hardware.

4. Implementation Roadmap

1

Phase 1: Proof of Concept (3-6 months)

- Deploy 3-5 edge units in a single location.
- Test core functionality including grievance processing and basic monitoring.
- Validate power systems and connectivity solutions.
- Gather user feedback for iterative improvements.

2

Phase 2: Pilot Deployment (6-12 months)

- Scale up to 20-50 units across the project area.
- Implement full multi-agent coordination for efficient operations.
- Develop a central monitoring dashboard for oversight.
- Train local staff on the maintenance and operation of the systems.

3

Phase 3: Production Rollout (12+ months)

- Deploy across multiple project sites to maximize impact.
- Establish a supply chain for hardware procurement to ensure availability.
- Create standardized deployment procedures for consistency.
- Develop a partner ecosystem for ongoing support and collaboration.

5. Cost Considerations

Hardware Costs

- Basic Unit: Costs between $2,000 to $4,000 per location.
- Advanced Unit: Ranges from $5,000 to $10,000 per location depending on features.
- Satellite Connectivity: Costs vary from $100 to $500 per month per unit.
- Maintenance: Estimated at $500 to $1,000 per year per unit.

Software Development

- Initial Development: Costs between $200,000 to $500,000.
- Ongoing Maintenance: Ranges from $50,000 to $100,000 per year.
- Model Training/Updates: Estimated costs between $20,000 to $50,000 per year.

ROI Drivers

- Reduced Site Visits: Savings of $10,000 to $50,000 per year per location.
- Faster Issue Resolution: Improved community relations and reduced disputes.
- Proactive Monitoring: Prevention of costly environmental incidents.
- Regulatory Compliance: Reduced fines and delays through effective monitoring.

6 Technical Challenges & Solutions

Limited Processing Power

Solution

Implementing model optimization and hierarchical processing to enhance efficiency.

Fallback

Utilizing simple rule-based systems for critical functions when necessary.

Harsh Environmental Conditions

Solution

Deploying industrial-grade hardware with redundant systems to ensure reliability.

Fallback

Establishing automatic unit replacement protocols to maintain operations.

Theft/Vandalism

Solution

Implementing camouflaged installations and engaging the community for safety.

Fallback

Using GPS tracking and remote wipe capabilities for enhanced security.

Thank You