UNIVERSITY OF RWANDA
COLLEGE OF SCIENCE AND TECHNOLOGY
SCHOOL OF ICT

DATA PREPROCESSING TECHNIQUES IN PROJECT CONTEXT

TransConnect - CI-Driven Public Transportation Platform

Group 4: TransConnect
Members: MUSAZA PATRICK (223019061), MANZI NSENGA IVAN (223004392)

Module: Computing Intelligence & Applications
Date: October 2025

DATASET OVERVIEW

Source: Bus route information for Kigali public transport

Data Description:
- Tabular data with bus routes, stop names, and coordinates
- Multiple routes with sequential stops
- Mixed data types: text (stop names) and numeric (coordinates)

Initial Data Quality Issues:
- Inconsistent stop naming conventions
- Mixed coordinate formats ($139282 E30.28382)
- Potential missing values and duplicates
- Non-standardized data structure across routes

Nature: Spatial-Temporal Data with Geographic Coordinates

1. DATA CLEANING

Purpose: Handle missing, incorrect, and inconsistent data

Application to TransConnect:
• Standardize stop names: "Mizusho" → "Misaso"
• Clean coordinate formats: Remove '$', 'E' characters
• Convert to standard decimal degrees: -1.39282, 30.28382
• Handle missing stop names or coordinates

2. DATA INTEGRATION & REDUCTION

Data Integration:

- Merge multiple route datasets into unified structure
- Create stops table (unique stops with IDs)
- Create routes table (route sequences referencing stops)

Data Reduction:

- Remove irrelevant columns: "Route Price", "Platinum"
- Focus on core features: stop_id, latitude, longitude
- Deduplicate identical stops across routes

Impact:

- Efficient multi-route management
- Faster processing for route optimization
- Foundation for transfer point identification

3. DATA TRANSFORMATION

Purpose: Convert data into model-ready format
Key Transformations for TransConnect:
• Coordinate validation and normalization
• Feature engineering using Haversine formula:
Distance between consecutive stops
Estimated travel time between stops
• Create sequential ordering of stops

Impact: Raw coordinates → actionable features for AI model

4. DATA DISCRETIZATION & AUGMENTATION

Data Discretization:

- Convert continuous "distance_from_start" into categories:
- Bin 1: "Start" (0-2 km)
- Bin 2: "Early" (2-5 km)
- Bin 3: "Mid" (5-10 km)
- Bin 4: "Late" (10+ km)
  - Enables traffic pattern analysis by route segment

Data Augmentation:

- Generate synthetic trips with varying travel times
- Simulate different conditions: rush hour, weather, events
- Add temporal features: time_of_day, day_of_week

Impact: Enhanced dataset for robust model training

5. VISUALIZATION & IMPACT

Before Preprocessing:

- Messy, overlapping points on map
- Inconsistent stop representations
- Unreliable spatial relationships

After Preprocessing:

- Clean, sequential route visualization
- Accurate stop positioning
- Calculated distances and travel times

Impact on Model Readiness:

- Clean spatial data for route optimization
- Engineered features for arrival prediction
- Structured data for CI algorithms (Genetic Algorithms, ML)
- Foundation for real-time intelligence system

CHALLENGES & LESSONS LEARNED

Challenges Encountered:
- Interpreting mixed coordinate formats
- Standardizing stop names without official references
- Handling incomplete route information
- Ensuring spatial accuracy for distance calculations

Lessons Learned:
- Data preprocessing is crucial (80% of data science work)
- Clean, structured data enables effective CI applications
- Domain knowledge essential for data interpretation
- Quality preprocessing directly impacts model performance

Conclusion: The preprocessed dataset is now ready for:
- Machine learning model training
- Route optimization algorithms
- Real-time arrival prediction systems
- Intelligent transportation analytics

CODE DEMONSTRUCTRATION STRUCTURE

Jupyter Notebook/Python Script Sections:

1. Data Loading & Initial Assessment
2. Data Cleaning Operations
3. Data Integration & Deduplication
4. Feature Engineering (Distance Calculations)
5. Data Transformation & Discretization
6. Data Augmentation for Model Training
7. Visualization of Results

Tools & Libraries:

- Pandas for data manipulation
- NumPy for numerical operations
- Scikit-learn for preprocessing
- Haversine for distance calculations
- Matplotlib/Plotly for visualization
- Folium for map-based displays