

Research Methodology Report: Decision Tree-Based Message Routing

1. Objective

The objective of this study is to develop a decision tree-based model to predict the next hop for message routing in a network, optimizing for efficient path selection using a predefined network topology operating at 868 MHz with SF7, BW 500 kHz, and CR 4/5 for short airtime and high data rate.

2. Network Representation

Defining the Network Topology: The network is modeled as a graph with nodes and their peers. Each node is represented by a dictionary containing a unique `nodeId` and a list of directly connected peer nodes. The network is implemented as an adjacency list for efficient neighbor lookup.

3. Data Generation

Computing Shortest Paths: Breadth-First Search (BFS) is employed to compute the shortest paths from each source node to all possible destination nodes. BFS ensures the shortest path in an unweighted graph, providing the parent node for each destination.

Extracting Paths: For each source-destination pair, the path is reconstructed from the parent dictionary. The next hop (the second node in the path) is extracted as the target variable for training.

Training Data Creation: Training data is generated by iterating over all source-destination pairs (excluding self-loops). Each data point consists of a feature tuple (source, destination) and the corresponding next hop.

4. Data Preprocessing

Feature Encoding: The source and destination node IDs are categorical variables. One-hot encoding is applied using `scikit-learn`'s `OneHotEncoder` to convert these into a binary vector representation suitable for the decision tree model.

5. Model Development

Decision Tree Classifier: A decision tree classifier (`DecisionTreeClassifier` from `scikit-learn`) is trained on the one-hot encoded features (source, destination) to predict the next hop. The decision tree is chosen for its interpretability and ability to handle categorical data effectively.

Training Process: The model is trained on the dataset of source-destination pairs and their corresponding next hops, learning the routing decisions based on the shortest paths derived from BFS.

6. Prediction and Application

Predicting Next Hop: The trained model predicts the next hop for a given source-destination pair. The input pair is one-hot encoded, and the decision tree outputs the predicted next hop, which is used for message routing in the network.

Implementation: The prediction function integrates the encoder and classifier to provide real-time routing decisions, ensuring compatibility with the 868 MHz LoRa configuration (SF7, BW 500 kHz, CR 4/5) for minimal airtime (0.1 s for 255 bytes) and high data rate (21.9 kbps).

7. Tools and Libraries

The methodology leverages Python with the following libraries:

- `numpy` for numerical operations.
- `collections.deque` for efficient BFS queue management.
- `scikit-learn` for one-hot encoding and decision tree classification.

8. Evaluation

The model’s performance is evaluated implicitly through the correctness of predicted next hops, as the training data is derived from BFS, which guarantees optimal paths. Future work may include testing on larger, dynamic networks and evaluating robustness under node failures.