## **NETWORK INTRUSION DETECTION SYSTEM USING DEEP LEARNING**

## **METHODOLOGY**

**1.1 Preprocessing of Dataset(Network Traffic Data Standardization)**

The first step in the preprocessing process is loading raw network traffic data, which includes both malicious and benign traffic patterns. Protocol type (proto), network service (service), and connection status (state) are examples of categorical information that are converted into numerical representations by one-hot encoding. For numerical stability, zeros are used in place of missing values and infinite values. While maintaining the original binary labels (0=normal, 1=attack), labels are encoded using LabelEncoder for multiclass detection (9 attack kinds plus normal).The dataset is divided into training and test sets using stratified sampling (70:30 ratio) to preserve class distribution.

**1.2 Class Imbalance Mitigation**

There is a significant class imbalance in the dataset, with less than 0.5% of samples belonging to uncommon attack types (such as shellcode and worms). A two-pronged strategy fights this: In order to avoid creating synthetic samples for extremely rare classes, adaptive ADASYN oversampling uses dynamic neighbor counts (min=1, max=5) dependent on class population to preferentially increase minority classes. 2) Class-weighted focused loss focuses training on underrepresented classes by employing instance-level weighting with (median\_count/class\_count). Minimum representation is ensured for essential minority classes (Shellcode, Worms) using targeted SMOTE resampling with manual thresholds (1,500 and 500 samples, respectively). Class histograms are used to assess resampled distributions prior to model training.

**1.3 Hybrid CNN-LSTM Architecture**

The model integrates bidirectional LSTMs with attention mechanisms and stacking 1D CNNs. Prior to max-pooling and 30% dropout for noise reduction, two CNN layers (128 filters@kernel5 → 64 filters@kernel3) extract hierarchical spatial patterns from network traffic sequences. Self-attention layers that dynamically weight important timesteps (such as assault initiation points) improve the processing of temporal interdependence in both forward and backward directions using bidirectional LSTMs (64 units). Prior to categorization, a final bidirectional LSTM (32 units) concatenates and processes the attention-LSTM outputs. While focus loss manages any remaining class imbalance, batch normalizing between layers speeds up convergence. Early stopping monitors validation F1-score with 10-epoch patience.

**1.4 Multi/Binary Class Detection**

The model employs softmax output with 11 neurons (9 assaults + normal) trained via categorical focal loss for multiclass detection. Class-specific F1-scores and confusion matrices are used to assess performance. The binary detection variation employs SMOTE-balanced training data (1:1 ratio), binary cross-entropy loss, and sigmoid output (normal vs. attack). The hybrid design at the heart of both models is the same, but the final layers are different. Accuracy curves comparing epoch-wise training/validation performance are used to track training dynamics. To minimize false negatives, binary mode assigns a 3× class weight to critical attack types. The final models have integrated preparation pipelines for distribution and are saved.