# CSE 472
# Machine Learning
# Project Proposal

# Network Flow Classification & Anomaly Detection

Submitted By:
Sadif Ahmed, 1905058
Abdullah Al Mohaimin, 1905041

# Problem Definition

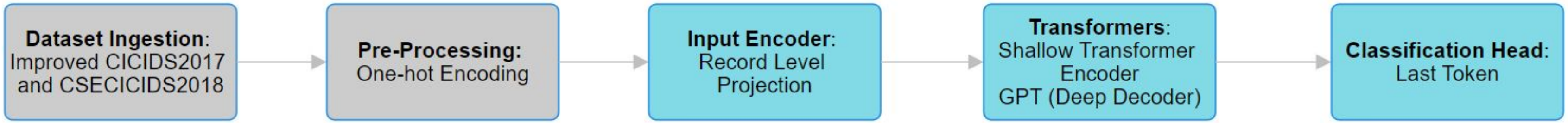
- Network flows: communication between two network endpoints at a specific time interval
- **Network flow classification**: anomalous/malicious traffic is detected and stopped/prevented
- Classifying network flows is an important problem, given that it needs to be fast and accurate
- **We are attempting to perform the classification and detection of flows in a benchmarked dataset**
- **Related Works** (utilizing same dataset)
  - **FlowTransformer: A Transformer Framework for Flow-based Network Intrusion Detection Systems (2024)** - Focuses on general encoder/decoder transformers and specialized models such as GPT, Bert for classification of network flows
  - **Real-Time Intrusion Detection via Machine Learning Approaches (2024)** - Uses traditional ML model of Random Forests, commonly used in network flow classification
  - **Improving Generalization of ML-Based IDS With Lifecycle-Based Dataset, Auto-Learning Features, and Deep Learning (2024):** Among other innovations, this work uses automated feature learning combined with CNN to increase generalizing power of ML/DL based flow classifiers

# Dataset: Improved CICIDS2017 and CSECICIDS2018

- **Source:** **https://doi.org/10.1109/CNS56114.2022.9947235**
- **Description:**
    - **Two similar dataset was rectified, improved and released together in 2022**
    - Organized in CSV files of flows for each day of experimentation (1 and 2 week respectively)
    - Each flow is labelled in detail
- **High Level Information**
    - **Columns: 90 features, 1 target; Rows: On average, each day has over 6 million flows**
    - **Columns: Mostly numerical,** many can be dropped before training ( Will be preprocessed before using)
    - **Flow Labels that we are keeping in consideration:**
        - Benign
        - Web Attacks
        - DoS
        - DDoS
        - Infiltration
        - Botnet
- **Dataset is highly imbalanced as benign network traffic is the most prevalent flow in general**

# Proposed Solution

Transformer Architecture


**Transformer Model: Train & Evaluate**
**Hyperparameters:**

- Number of Attention Heads
- Number of Transformer Layers
- Internal Transformer Size
- Learning Rate
- Sequence Length

# Performance Evaluation

- As we observed in the dataset section, our selected dataset is extremely skewed towards benign which will make accuracy metrics a misleading choice. So, we are planning to use **F1 Score, Precision and Recall** as our performance metrics. The formulation of these metrics are shown below:
- **Precision**: The proportion of positive identifications that are actually correct. This is particularly useful when the cost of false positives is high.
  **Precision = True Positives / (True Positives + False Positives)**
- **Recall (Sensitivity)**: The proportion of actual positives that are correctly identified. This is crucial when the cost of false negatives is high.
  **Recall = True Positives / (True Positives + False Negatives)**
- **F1-Score**: The harmonic mean of precision and recall, balancing the two metrics
  **F1-Score = 2 * (Precision * Recall) / (Precision + Recall)**