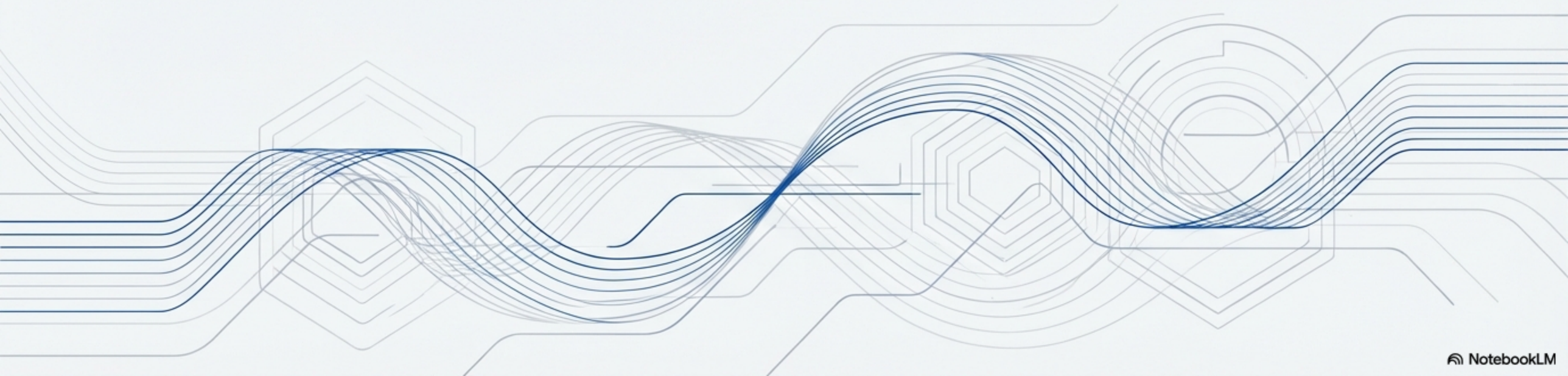
Building a Trustworthy AI for Cybersecurity

A Calibrated & Robust Intrusion Detection System



In Cybersecurity, 99% Accuracy Is Not Enough

The cost of being wrong is asymmetric and extremely high. A standard model's success metrics can be dangerously misleading.

False Positive

A legitimate user or critical process is blocked.

Impact: Disrupts operations, damages user trust, creates support overhead.

False Negative

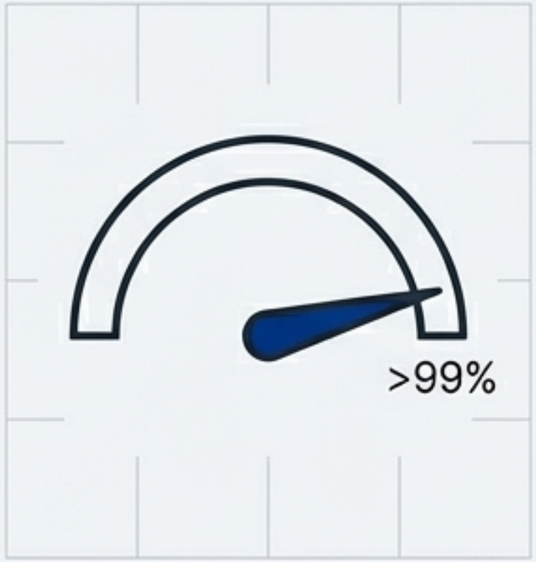
A real attack goes undetected.

Impact: Data breach, system compromise, catastrophic financial and reputational loss.

Our goal must be not just accuracy, but reliability and predictable behavior under pressure.

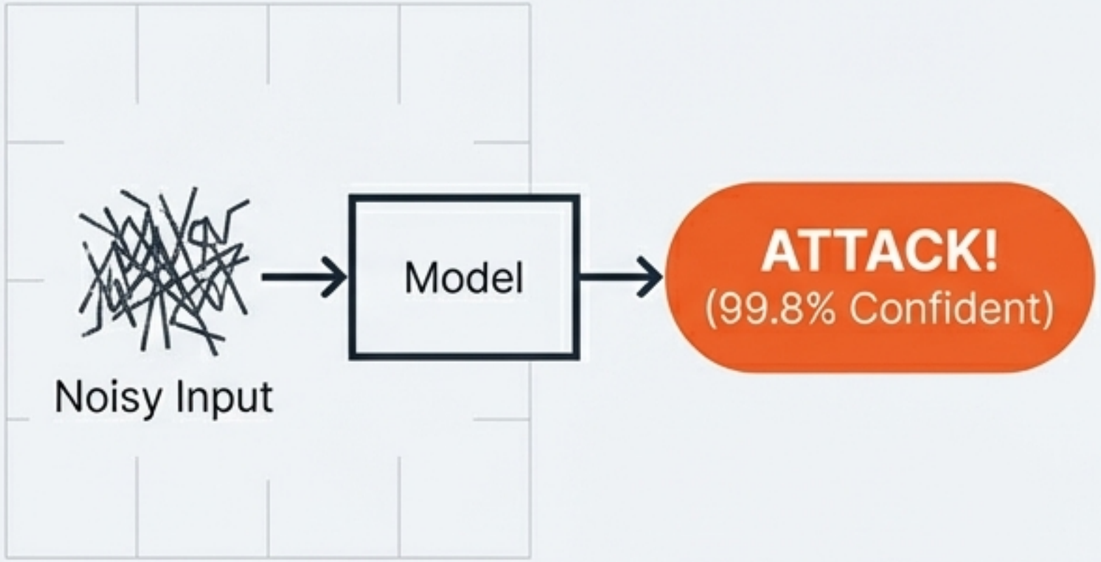
The Two Hidden Dangers of Standard AI Models

Overconfidence



Models often report near-certainty (>99% confidence) even when they are incorrect. Their probability scores are not “calibrated” to reality, making it impossible to set reliable action thresholds.

Brittleness

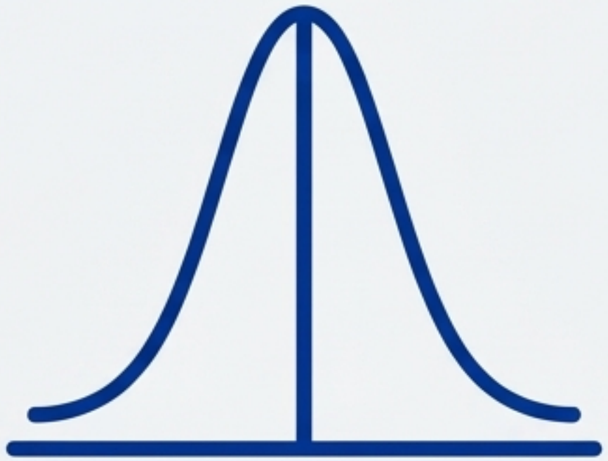


Models can “hallucinate” threats when faced with unexpected or noisy data. This unreliability makes them unfit for production environments where data is never perfect.

Our Mission: An IDS Built on Three Pillars

ACCURACY

Maximize correct classifications of both benign and malicious traffic, effectively minimizing both false positives and false negatives.



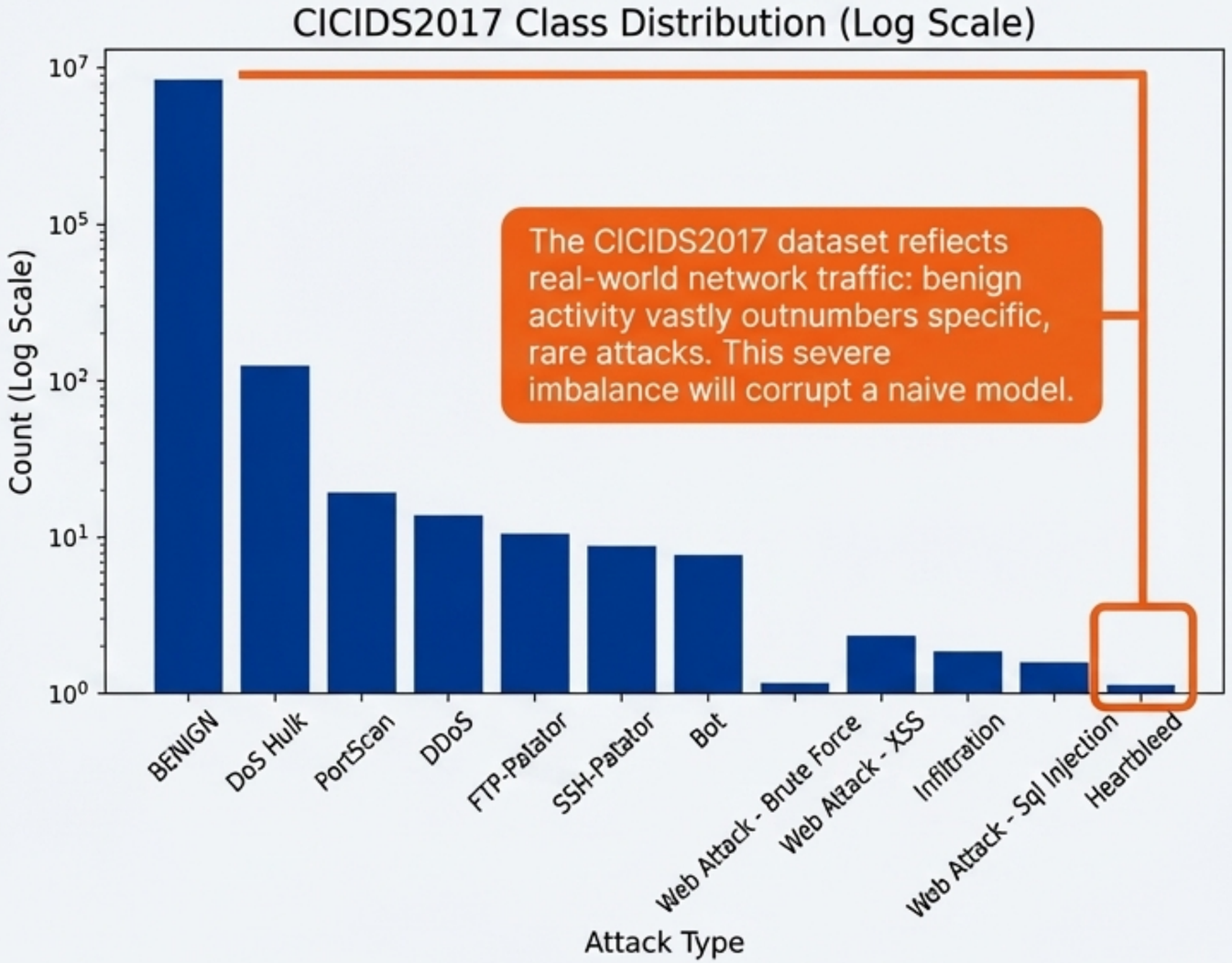
CALIBRATION

Ensure the model's output confidence score is a true reflection of its correctness probability. A 90% confidence should mean it's right 9 times out of 10.

ROBUSTNESS

Build a model that is resilient to noisy data and knows what it doesn't know, preventing it from making confident predictions on garbage input.

Foundation: Taming the CICIDS2017 Dataset

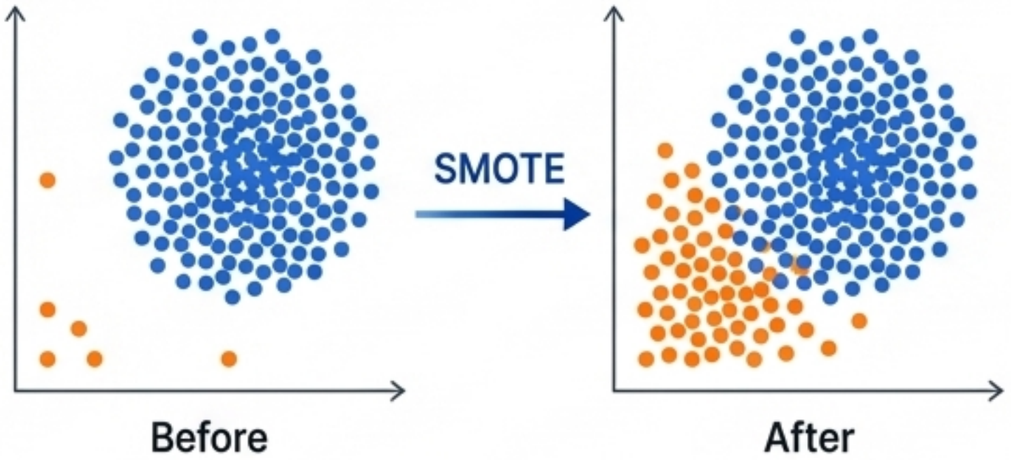


A Disciplined Preparation Pipeline

- **Robust Scaling**
  We used RobustScaler, which scales features using the Interquartile Range (IQR). Unlike StandardScaler, it is not skewed by the massive outliers and volume spikes inherent in network traffic data.
- **Correlation Analysis**
  Removed features with >95% correlation to reduce multicollinearity and model complexity.
- **Intelligent Feature Selection**
  A Random Forest model was trained to identify the top 40 most predictive features, focusing the model's attention on the strongest signals.

A Two-Pronged Strategy to Conquer Class Imbalance

1. Balancing the Data with SMOTE



We applied the Synthetic Minority Oversampling Technique (SMOTE) to the training data. This creates new, plausible examples of rare attack types, ensuring the model has sufficient data to learn their patterns. Our strategy guaranteed a minimum of 5,000 samples for each class.

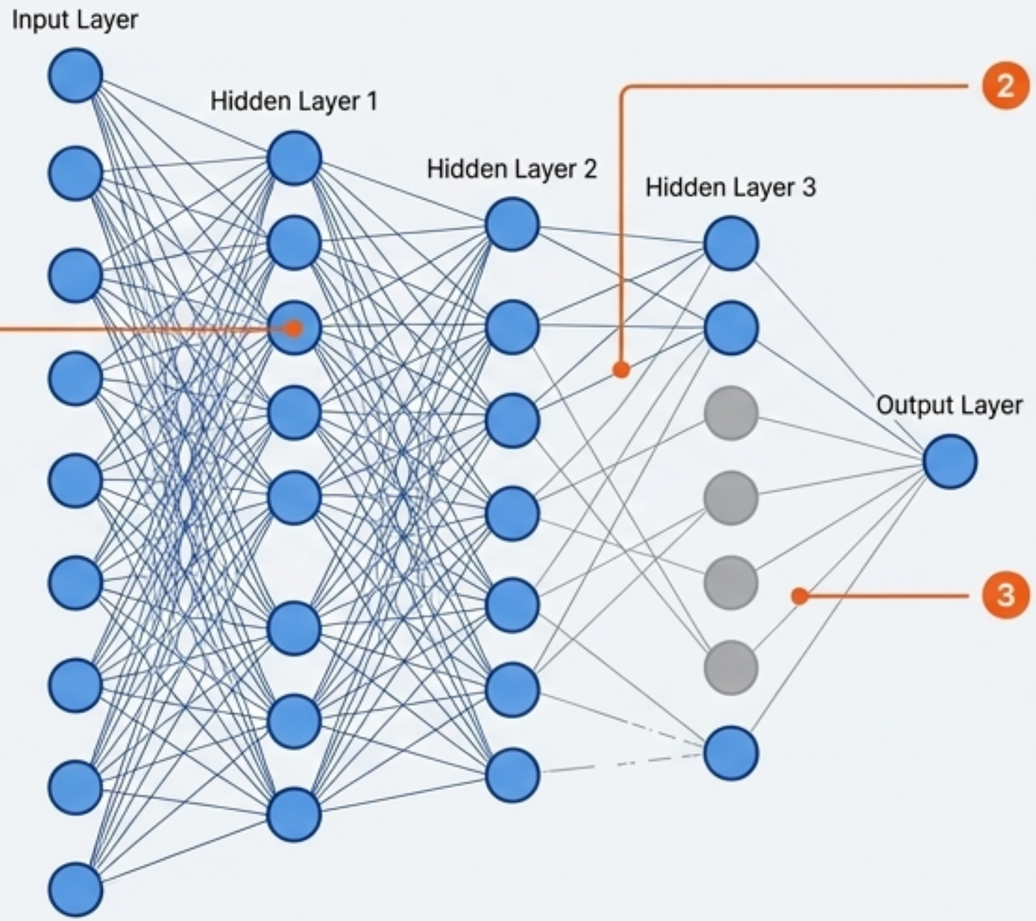
2. Focusing the Training with Focal Loss

We replaced the standard Cross-Entropy loss function with Focal Loss. This function dynamically down-weights the loss for easy-to-classify, common examples (like 'BENIGN'), forcing the model to pay significantly more attention to learning the features of rare and difficult attack classes.

The Engine: A Heavily Regularized Neural Network

L2 Regularization 1

Applied to dense layers to penalize large weight values, preventing the model from relying too heavily on any single feature and promoting simpler, more generalizable patterns.



2 Batch Normalization

Stabilizes and accelerates training by normalizing the inputs to each layer, reducing internal covariate shift.

3 Aggressive Dropout (up to 40%)

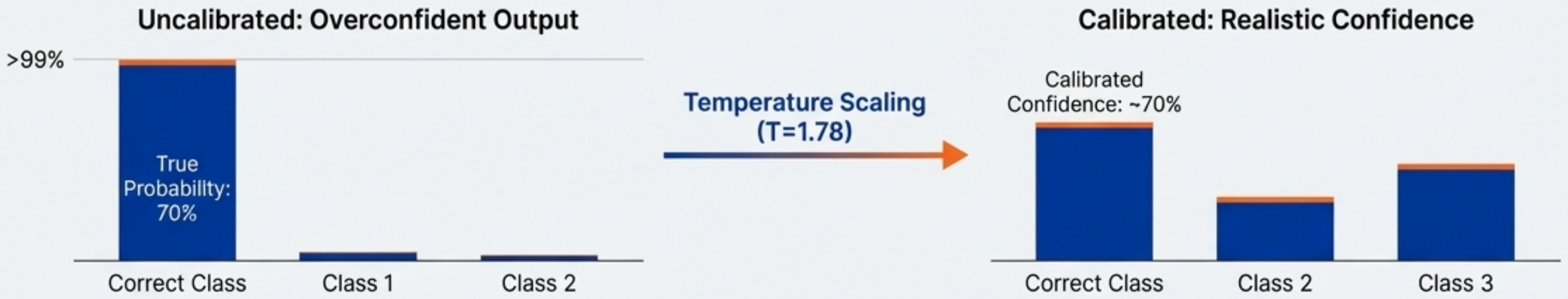
During training, randomly deactivates a significant portion of neurons. This forces the network to learn redundant representations and prevents complex co-adaptations, making it more robust.

Utilized the Adam optimizer with a controlled learning rate of 0.0001 for stable convergence.

Innovation #1: Achieving True Confidence with Temperature Scaling

Even a highly accurate model can be dangerously overconfident. We implemented Temperature Scaling to systematically correct this, making its confidence scores genuinely meaningful.

Before vs. After



How It Works

We calculated an optimal 'temperature' (T=1.78) on a validation set. By dividing the model's pre-softmax outputs (logits) by this temperature before the final activation, we soften the probabilities to reflect the model's true uncertainty.

> **"In security, a false positive is expensive.** I needed the model's confidence to be accurate so we can set a **threshold (e.g., only block if calibrated confidence > 90%)**. Temperature scaling makes this possible."

Innovation #2: The Sanity Check—Proving Robustness with a ‘Noise Test’

The Experiment

To test for brittleness, we asked a critical question: What does the model do when it sees pure garbage? We fed the calibrated model 10 samples of pure Gaussian noise.

Expected Outcome

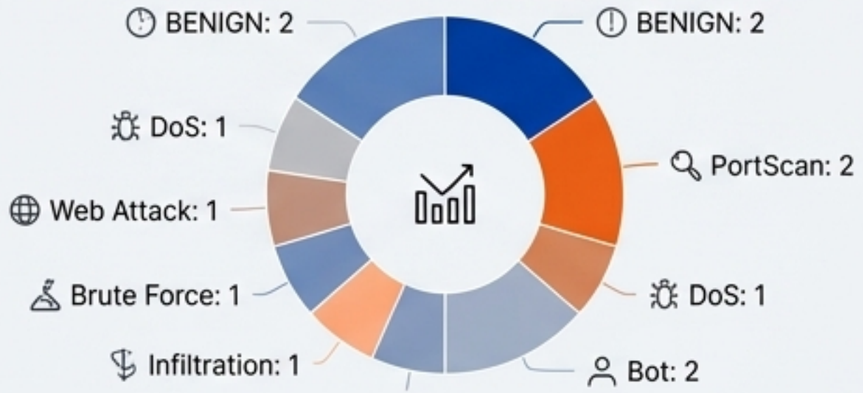
A robust, well-behaved model should not “hallucinate“ an attack. It should show low confidence and its predictions should be distributed, not fixated on one class.

The Results

34.7%

Low Average Confidence

The model’s average confidence on pure noise. This proves it is appropriately uncertain when faced with data that contains no real patterns.



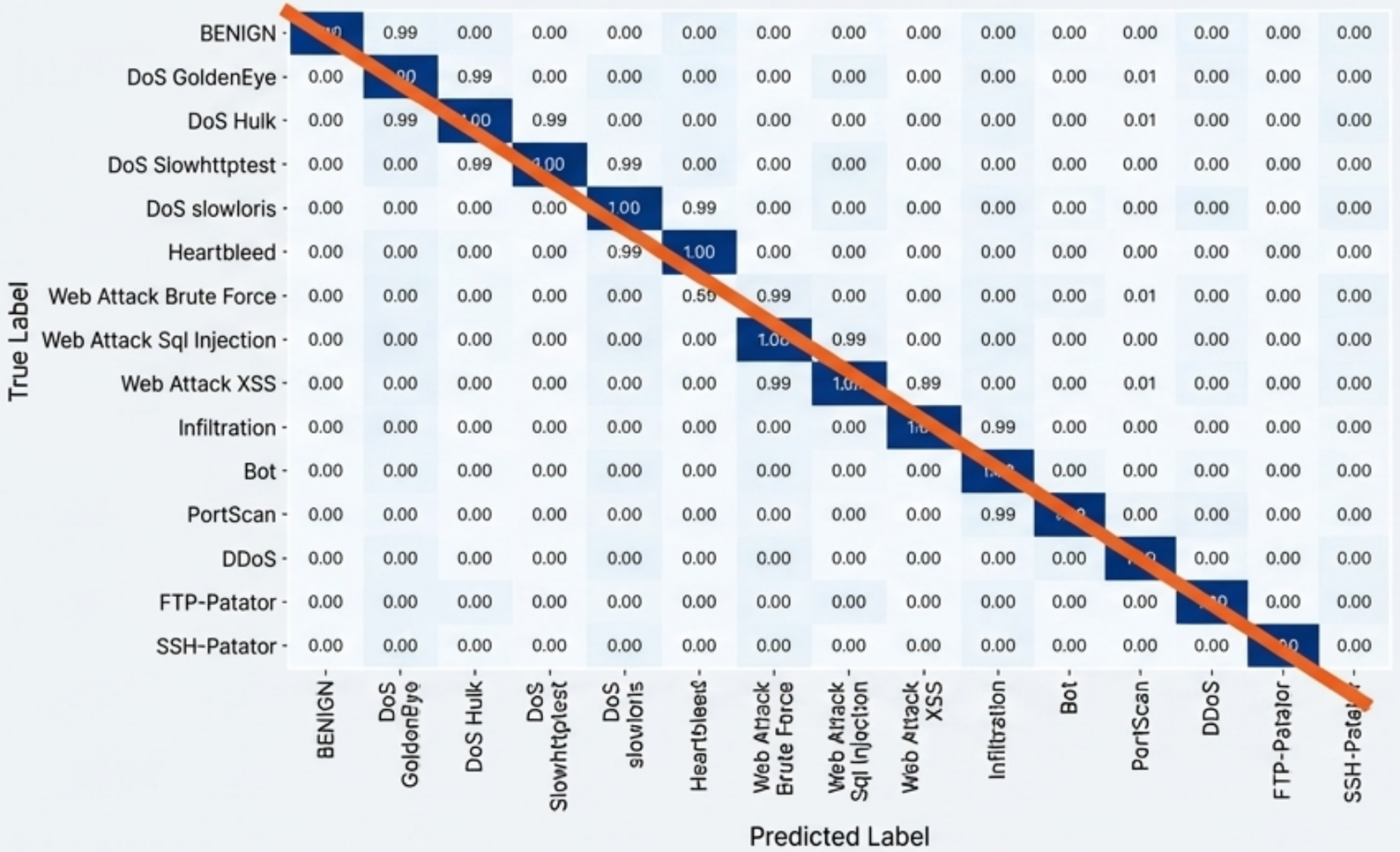
Distributed Predictions

The 10 noise predictions were spread across 7 different classes. The model isn't defaulting to a single prediction; it’s correctly showing confusion.

Our model knows what it doesn’t know. It will not confidently flag random data as a threat.

The Verdict: Exceptional Performance on Unseen Data

Normalized Confusion Matrix (Test Set)



The normalized confusion matrix shows near-perfect prediction across all classes on the test set. The strong diagonal indicates that the true labels (Y-axis) consistently match the predicted labels (X-axis).

Weighted Averages on Test Set

99.86%
Accuracy

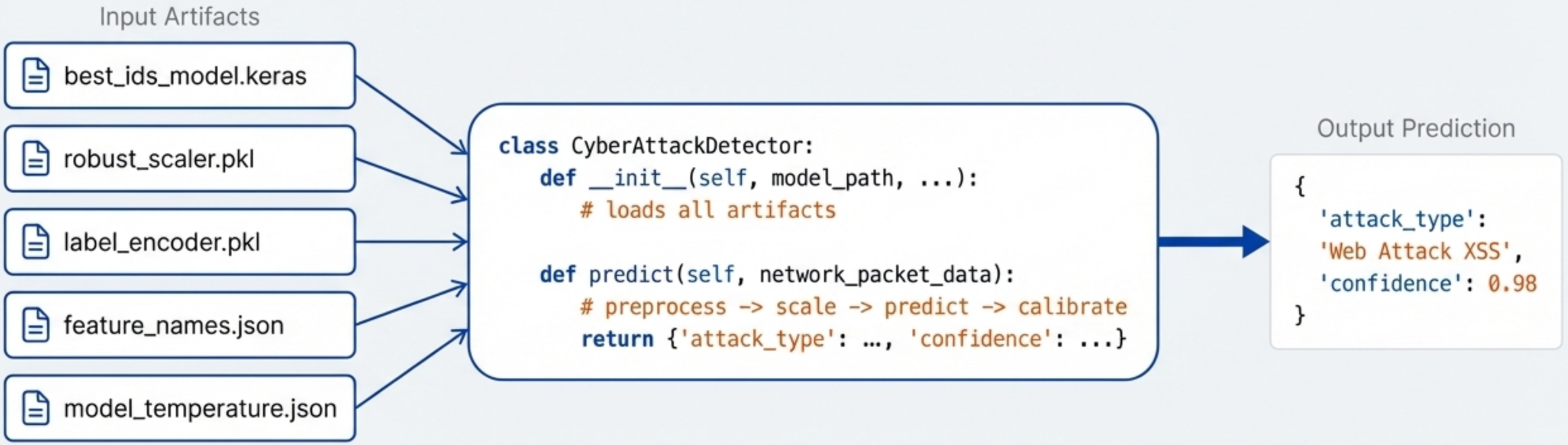
99.86%
Precision
("Are we raising false alarms? No.")

99.86%
Recall
("Are we catching all attacks? Yes.")

99.86%
F1-Score
("The optimal balance.")

Beyond the Notebook: A Production-Ready Inference System

A model is only useful if it can be deployed. We packaged the entire logic—from preprocessing to calibrated prediction—into a single, portable class.



This solution is not a research experiment; it is an engineering asset ready for integration.

A Truly Defensible Intrusion Detection System

We successfully built an IDS that meets all three of our core mission objectives. This is a system built for the complexities of the real world.

ACCURACY ✓

Achieved **99.86% F1-Score** on unseen test data through rigorous feature engineering, imbalance handling, and a regularized model.

CALIBRATION ✓

Implemented **Temperature Scaling** to produce honest probability scores, enabling reliable, threshold-based decision-making.

ROBUSTNESS ✓

Passed a **'Noise Injection Test'**, proving the model is not brittle and does not hallucinate threats when presented with garbage data.

Next Steps

The `CyberAttackDetector` class is ready for the next phase: deployment as a real-time stream processing API or integration into a security dashboard like Kibana for live threat monitoring.