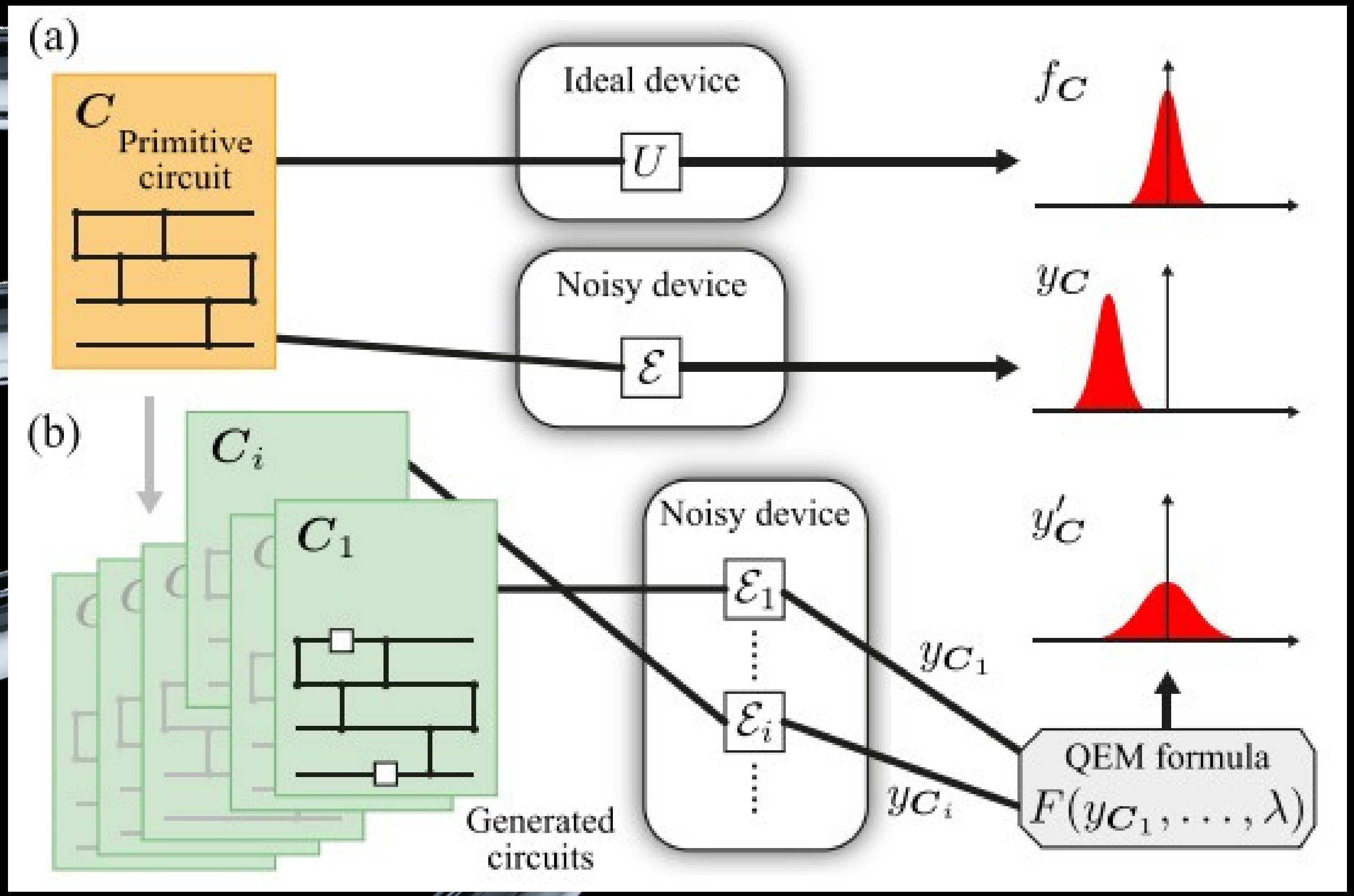
XEON

Quantum Error Mitigation

TEAM 8

AQC

QUANTUM NOISE



THE CHALLENGE

QUBITS ARE HIGHLY SENSITIVE TO DECOHERENCE AND CONTROL IMPERFECTIONS.

THE IMPACT

NOISY MEASUREMENTS BIAS EXPECTATION VALUES AND ALGORITHM PERFORMANCE DEGRADES RAPIDLY WITH DEPTH.

CREDIT: ERROR STATISTICS AND SCALABILITY OF QUANTUM ERROR MITIGATION FORMULAS (NPJ QUANTUM INFORMATION, 2023)

USE CASE LIFE

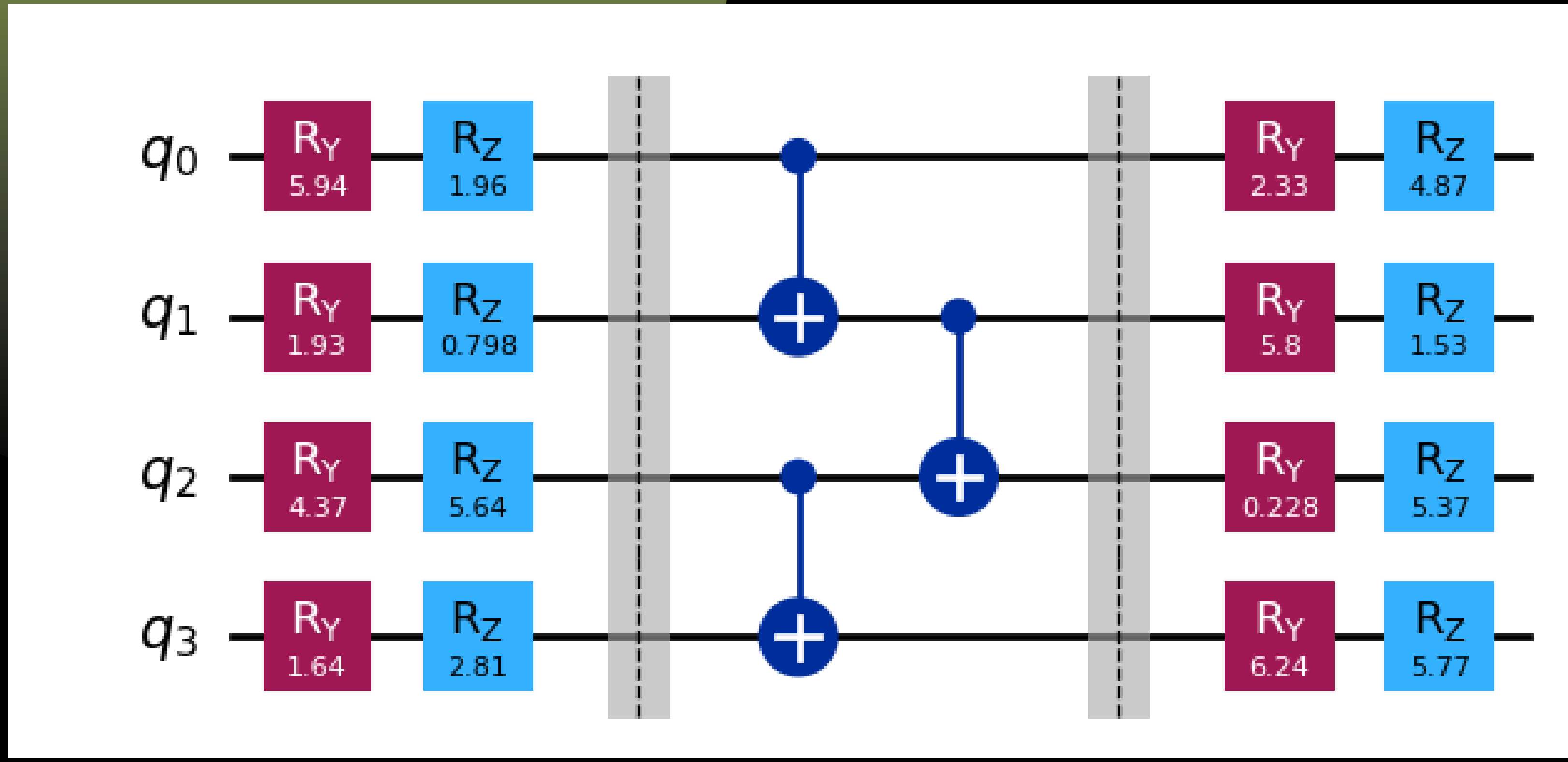
Druge-resistant TB affects 2.5 Million people Africa

TRADITIONAL SIMULATION TAKES 15 YEARS

SIMULATE DISEASE PROTIENS ACCURALTY IN MONTHS

CREDIT: THE NUMBERS AND STATISTICS BY WHO 2023,2024

DATASET GENERATION



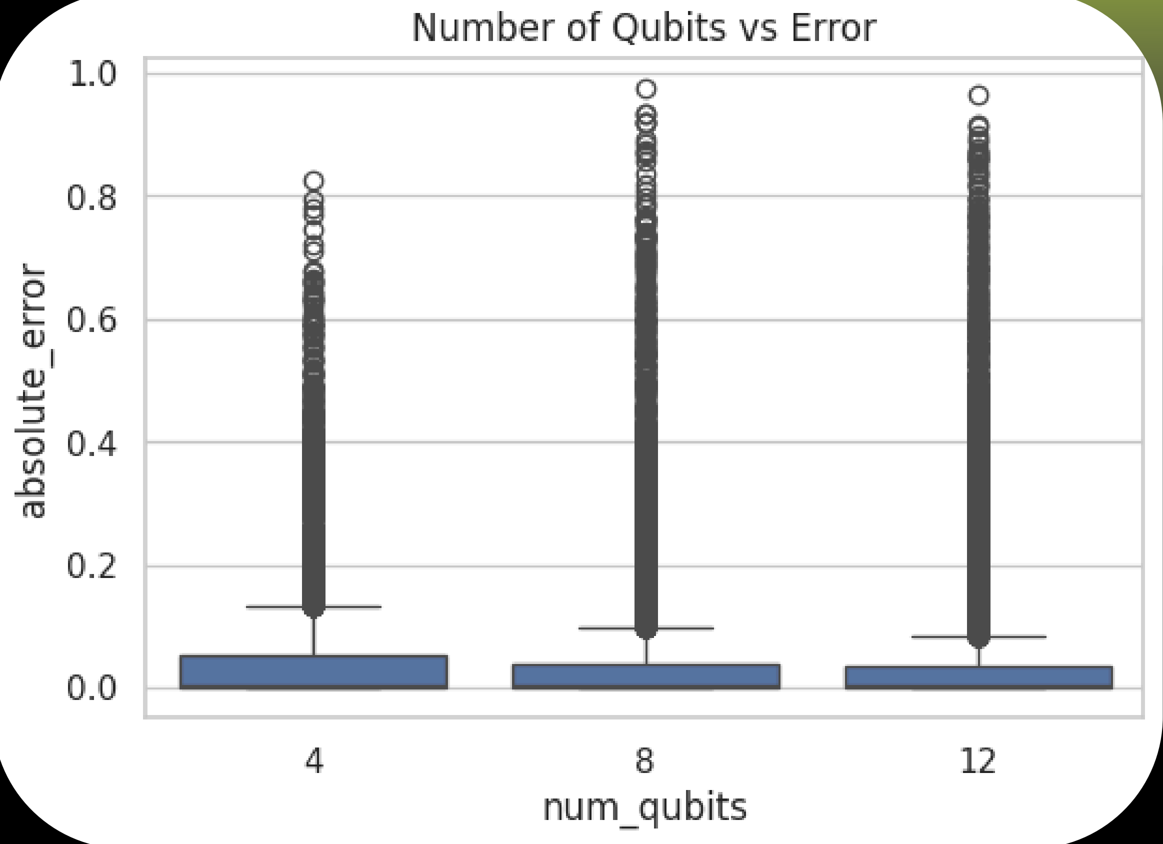
- EfficientSU2 ansatz creates hardware-efficient quantum circuits
- Three entanglement patterns: linear, full, pairwise
- Depth circuit multipliers: 1, 2, 3 repetitions
- Ideal simulations compute noise-free expectation values
- Three noise models: depolarizing, Amplitude damping, readout
- Multiple error rates: 0.001, 0.01, 0.1

DATASET STRUCTURE AND FEATURES

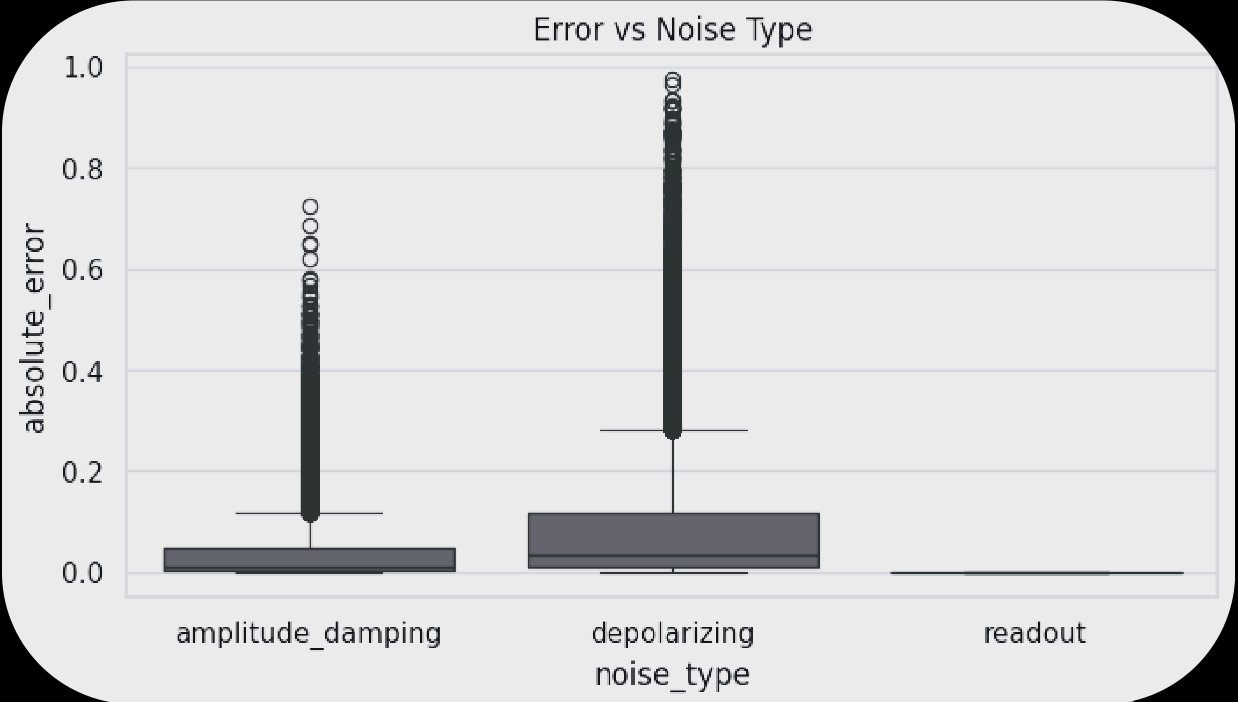
- Multi-scale systems: 4, 8, 12 qubits
- 900+ samples per qubit configuration generated
- Total 3k+ training samples across systems
- Paired data: noisy measurements, ideal targets
- Features: qubits, depth, entanglement, noise, error
- CSV format enables easy ML model training
- Combined dataset merges all qubit counts

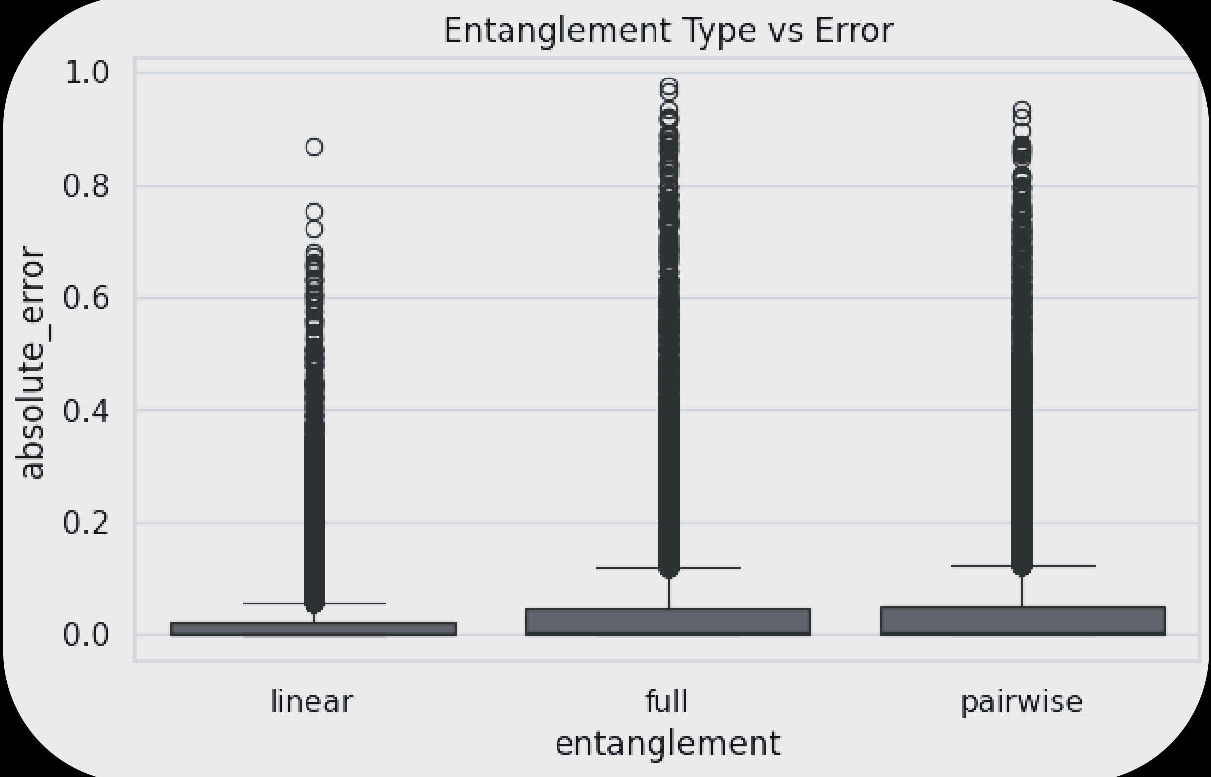
	n_qubits	rep	depth	entanglement	noise_type	error_rate	X_noisy	X_ideal
0	4	1	7	linear	depolarizing	0.100	-0.079067	-0.108459
1	4	1	7	linear	depolarizing	0.010	-0.283841	-0.292529
2	4	1	7	linear	depolarizing	0.001	0.295649	0.296538
3	4	1	7	linear	amplitude_damping	0.100	-0.027949	-0.027949
4	4	1	7	linear	amplitude_damping	0.010	0.002946	0.002946

DATA ANALYSIS



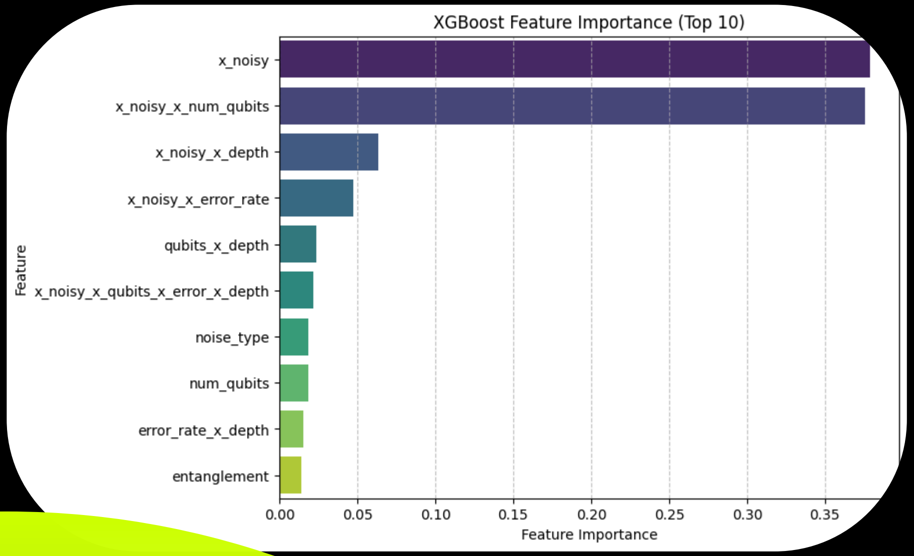
entanglement	absolute_error
full	0.052169
linear	0.030923
pairwise	0.057483

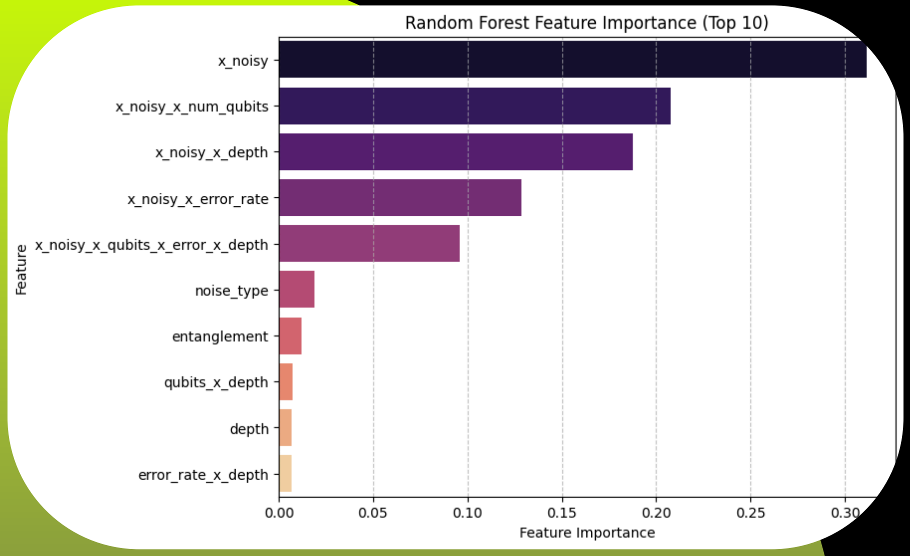


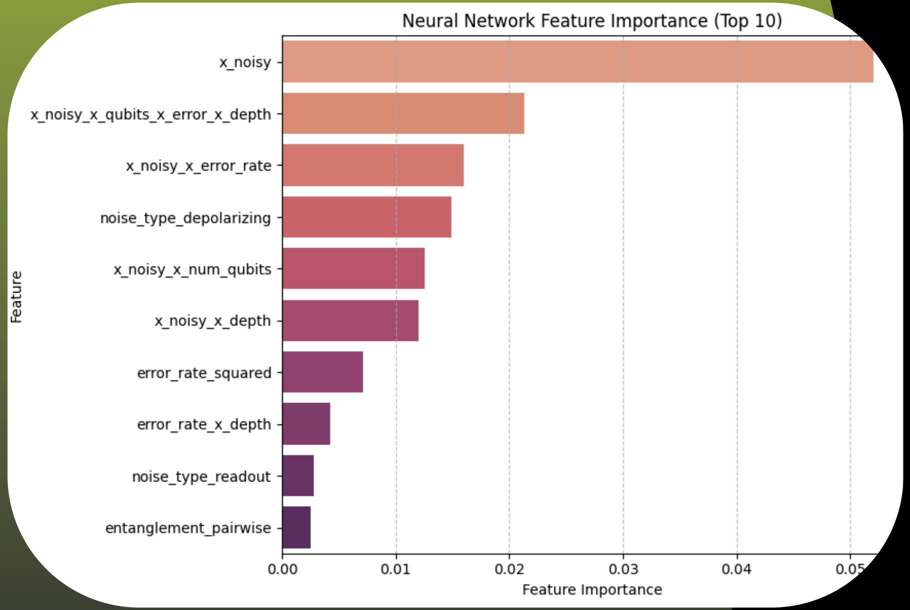


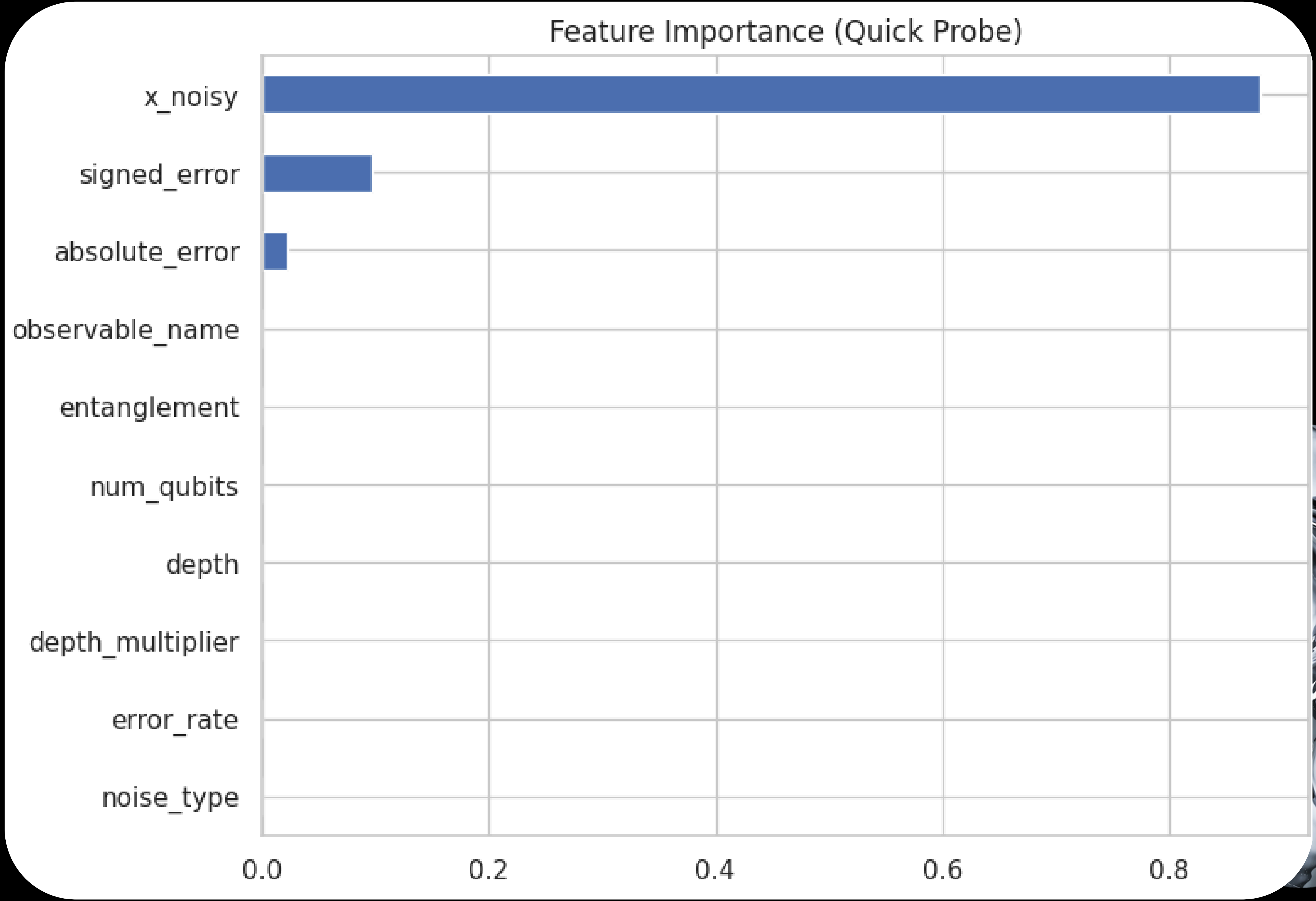
XEON

FEATURE IMPROTANCE & FEATURE ENGENEERING









QEM MODEL: DATA-DRIVEN MITIGATION ARCHITECTURE

Core Idea:

- **Data-Driven: Learns to correct noisy (x_noisy) to ideal (x_ideal) quantum measurements.**
- **Stacking Ensemble: Combines diverse base models for robust mitigation.**
- **Architecture:**
- **Base Models (GPU-Accelerated):**
    - **XGBoost: Powerful gradient boosting.**
    - **Neural Network (PyTorch): Custom deep learning with LSTM, Batch Norm, Dropout for complex feature interactions.**
    - **Random Forest: Ensemble of decision trees for robustness.**
- **Meta-Learner:**
    - **Ridge Regression: Optimally weights base model predictions for the final mitigated output.**

- Features: x_noisy, error_rate, depth, num_qubits, noise_type, entanglement, observable_name, plus engineered interaction terms.
- Target: x_ideal (ideal, noiseless measurement).

Experimental Design:

- Data Split: Training (60%), Validation (20%), Test (20%) for robust evaluation.
  - Train (base models): 21,870 samples
  - Validation (meta-learner): 7,290 samples
  - Test (final eval): 7,290 samples

Comparison Groups:

- No Mitigation (Baseline): Raw x_noisy as prediction.
- Individual Base QEM Models: XGBoost, Neural Network, Random Forest.

```
# Prepare X and y
X = df[num_features + cat_features]
y = df["x_ideal"]

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=42
)

# Further split train into train and validation for stacking
X_train_base, X_val_base, y_train_base, y_val_base = train_test_split(
    X_train, y_train, test_size=0.25, random_state=42
)

print(f"\nBase train size: {len(X_train_base)}")
print(f"Validation size: {len(X_val_base)}")
print(f"Test size: {len(X_test)}")
```

```
Base train size: 21870
Validation size: 7290
Test size: 7290
```

BENCHMARKING SETUP: COMPREHENSIVE COMPARISON

EVALUATION METRICS: QUANTIFYING PERFORMANCE

Primary Metrics:

- Mean Absolute Error (MAE): Average absolute prediction error (lower is better).
- Root Mean Squared Error (RMSE): Measures magnitude of errors, penalizes large errors more.
- R-squared (R^2): Proportion of variance explained by the model (higher is better).

Improvement Quantification:

- Improvement (%) over Baseline: ((Baseline MAE - Model MAE) / Baseline MAE) * 100.
- Mitigation Factor: Baseline MAE / Model MAE (how many times error is reduced).

Generalization to Unseen Circuits:

- All final metrics reported on a completely unseen Test Set (20% of data).
- Ensures reliable performance estimate for new quantum circuits and noise conditions.

```
# 1. Retrieve the Baseline MAE
baseline_mae = baseline_mae # This variable is already defined in the notebook

# 2. Retrieve the Stacked Ensemble MAE
stacked_ensemble_mae = results_df[results_df['Model'] == 'Stacked Ensemble']['MAE'].values[0]

# 3. Calculate the absolute improvement
absolute_improvement = baseline_mae - stacked_ensemble_mae

# 4. Calculate the percentage improvement
percentage_improvement = (absolute_improvement / baseline_mae) * 100

# 5. Print the results
print(f"\nBaseline MAE: {baseline_mae:.6f}")
print(f"Stacked Ensemble MAE: {stacked_ensemble_mae:.6f}")
print(f"Absolute Reduction in MAE: {absolute_improvement:.6f}")
print(f"Percentage Improvement over Baseline: {percentage_improvement:.2f}%")
```

```
================================================================================
CLARIFYING MODEL PERFORMANCE IMPROVEMENT
================================================================================

Baseline MAE: 0.046858
Stacked Ensemble MAE: 0.030826
Absolute Reduction in MAE: 0.016032
Percentage Improvement over Baseline: 34.21%
```

RESULTS: STACKED ENSEMBLE PERFORMANCE SUMMARY

Mitigated vs Unmitigated Performance:

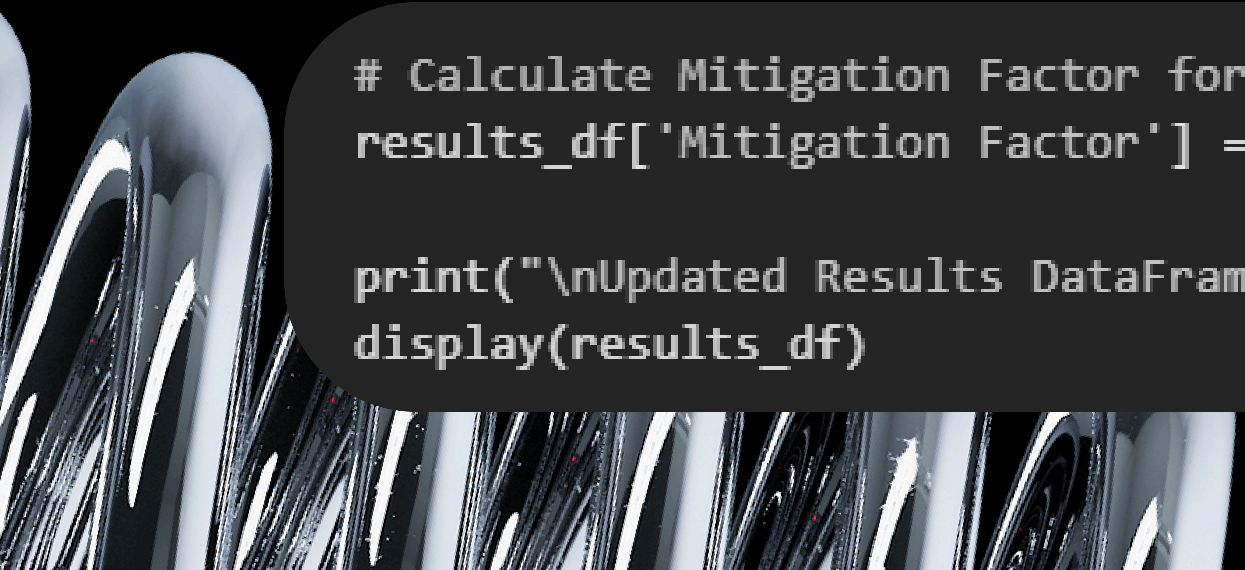
- Baseline (No Mitigation) MAE: 0.046858
- Stacked Ensemble MAE: 0.030826
- Absolute MAE Reduction: 0.016032
- Percentage Improvement: 34.21%

Detailed Performance Comparison (Test Set):

```
================================================================================
CALCULATING MITIGATION FACTORS
================================================================================

Updated Results DataFrame with Mitigation Factors:
```

	Model	MAE	RMSE	R²	Improvement (%)	Training Time (s)	Mitigation Factor
0	Baseline (No Mitigation)	0.046858	0.112830	0.877145	0.000000	0.000000	1.000000
1	XGBoost	0.035601	0.072544	0.949214	24.023159	2.087147	1.316191
2	Neural Network	0.042761	0.080968	0.936734	8.744417	31.861636	1.095823
3	Random Forest	0.035196	0.073790	0.947454	24.887696	18.012011	1.331340
4	Stacked Ensemble	0.030826	0.066567	0.957237	34.213471	51.962981	1.52006



```
# Calculate Mitigation Factor for each model
results_df['Mitigation Factor'] = baseline_mae / results_df['MAE']

print("\nUpdated Results DataFrame with Mitigation Factors:")
display(results_df)
```

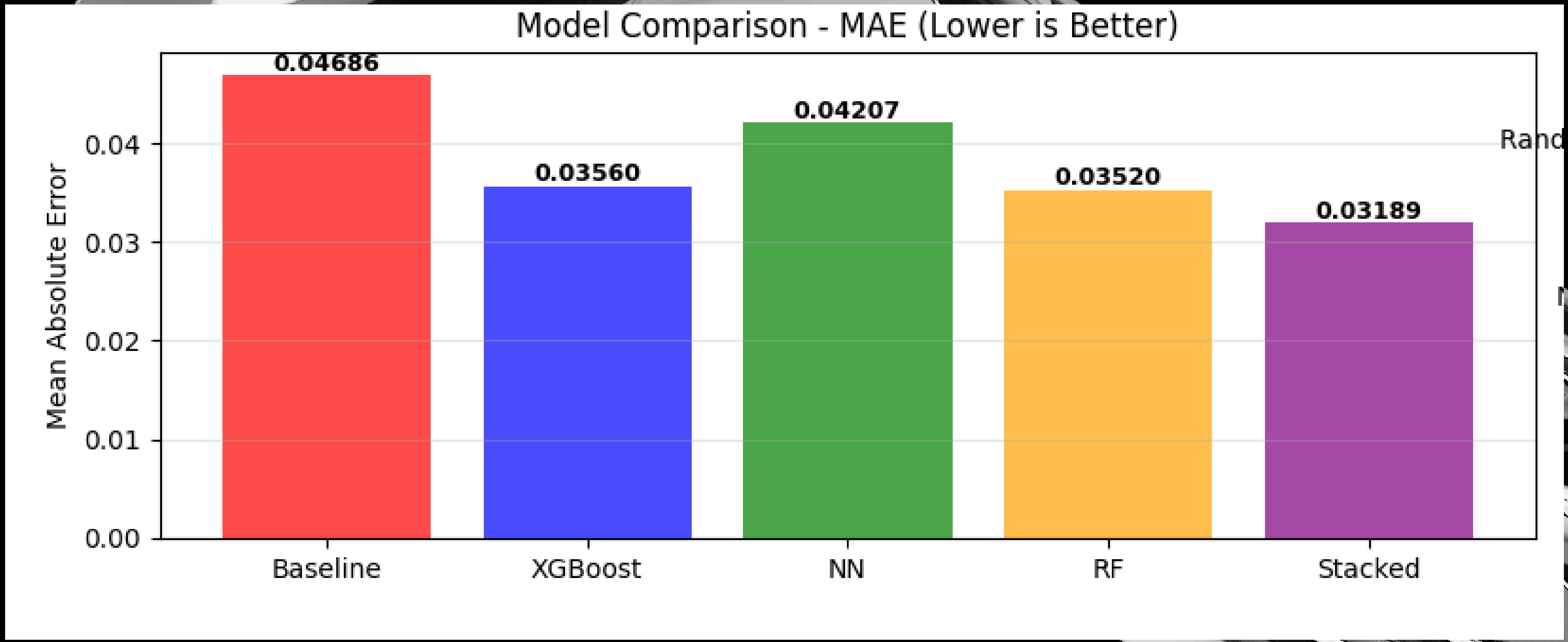
Key Findings:

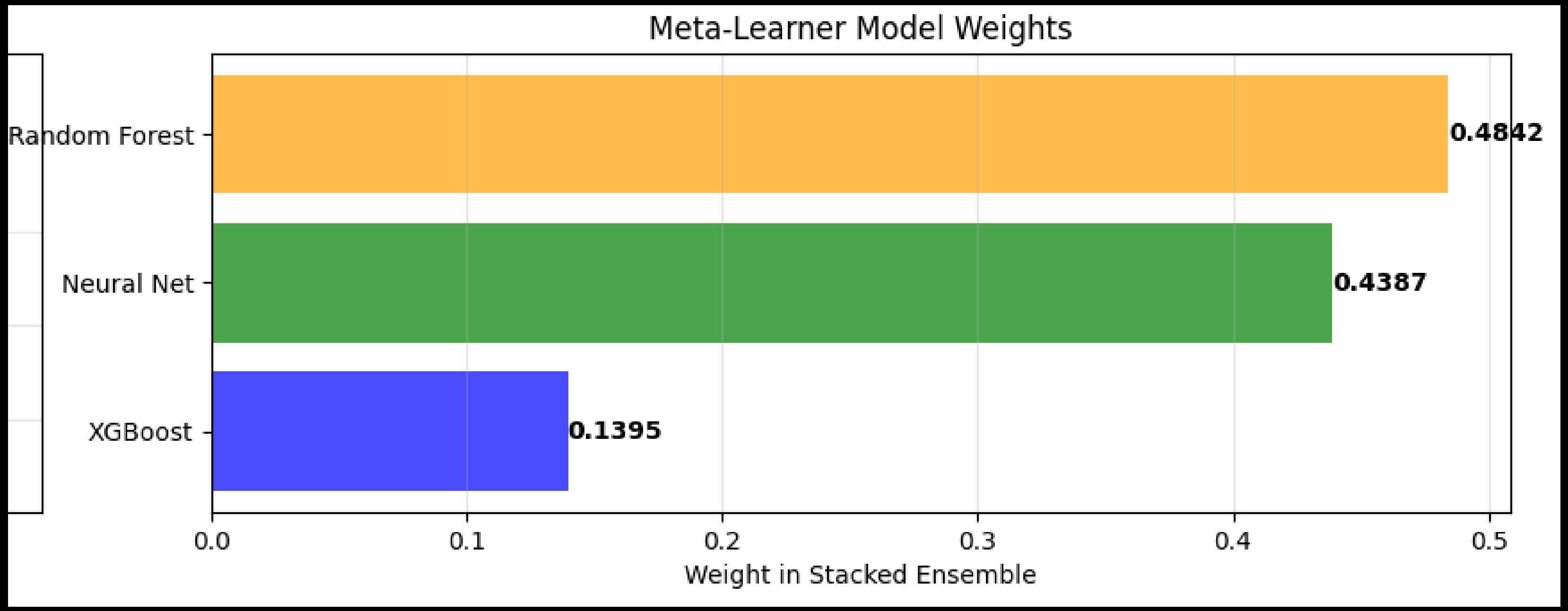
- Stacked Ensemble achieves lowest MAE and RMSE, signifying superior accuracy.
- Highest R^2 value, indicating best explanation of ideal outcomes.
- 34.21% improvement in MAE demonstrates significant error reduction.

QEM - STACKED ENSEMBL ANALYSIS

Model Robustness with Increasing Noise:

- The models are trained using error_rate as a feature, alongside x_noisy and various interaction terms (e.g., error_rate_x_depth, x_noisy_x_error_rate).
- This allows the models to implicitly learn relationships with different noise strengths present in the dataset.
- The strong generalization performance observed on the unseen test set suggests a good level of robustness across the range of noise conditions used in the quantum circuit simulations.





RESULTS: VISUALIZATIONS & META-LEARNER INSIGHT

RESULTS: MODEL ROBUSTNESS

Model Robustness with Increasing Noise:

- The models are trained using error_rate as a feature, alongside x_noisy and various interaction terms (e.g., error_rate_x_depth, x_noisy_x_error_rate).
- This allows the models to implicitly learn relationships with different noise strengths present in the dataset.
- The strong generalization performance observed on the unseen test set suggests a good level of robustness across the range of noise conditions used in the quantum circuit simulations.

THANK YOU

what we are tackling is not merely a technical problem—it is a bridge between our most complex technologies and our most pressing human needs. We are not just correcting errors in qubits… we are restoring hope.

Mentor

Hamza Benkadour

TEAM MEMBERS:

Team 8

Hossam Mohamed
Khalil Ullah
Ola Tarig

-

Mohamed Abdalfattah
Dr. Mayur Kumar Chhipa

AQC