ENHANCED ANOMALY DETECTION IN KEYSTROKE DYNAMICS AUTHENTICATION

Presented to
Amith Kamath Belman

TEAM:
Rashmi Sonth
Sakshi Tripathy

WHAT WE WILL TALK ABOUT

- Core concept
- Existing approach
- Algorithm implementation and results
- Conclusion
- Q&A

CORE CONCEPT

Keystroke dynamics authentication systems face significant challenges in detecting sophisticated spoofing attacks and subtle variations in typing behavior. These systems often struggle with high false positive and false negative rates, coupled with limited robustness against advanced spoofing and pattern replication techniques.

- **Objective**: Develop advanced machine learning models for keystroke dynamics authentication to enhance resistance against spoofing attacks.
- **Focus**: Evaluate model robustness against sophisticated spoofing techniques, including GAN-based attacks, adversarial perturbations, and synthetic data augmentation.
- **Goal**: Optimize detection accuracy and improve resilience of keystroke-based user verification systems for secure and adaptive authentication.

Existing Approach

K. S. Killourhy and R. A. Maxion, IEEE/IFIP International Conference on Dependable Systems & Networks, Lisbon, Portugal

- Keystroke Dynamics Analysis: Extracts temporal features such as key press durations (DU), digraph latencies (DD), and transitions (UD, UU) for precise behavioral profiling.
- Feature Engineering: Employs statistical normalization and transformation techniques to enhance feature discriminability.

	Detector	equal-error rate	
1	**Manhattan (scaled)**	**0.096**	**(0.069)**
2	**Nearest Neighbor (Mahalanobis)**	**0.100**	**(0.064)**
3	**Outlier Count (z-score)**	**0.102**	**(0.077)**
4	SVM (one-class)	0.102	(0.065)
5	Mahalanobis	0.110	(0.065)
6	Mahalanobis (normed)	0.110	(0.065)
7	Manhattan (filter)	0.136	(0.083)
8	Manhattan	0.153	(0.092)
9	Neural Network (auto-assoc)	0.161	(0.080)
10	Euclidean	0.171	(0.095)
11	Euclidean (normed)	0.215	(0.119)
12	Fuzzy Logic	0.221	(0.105)
13	k Means	0.372	(0.139)
14	Neural Network (standard)	0.828	(0.148)

	Detector	zero-miss false-alarm rate	
1	**Nearest Neighbor (Mahalanobis)**	**0.468**	**(0.272)**
2	**Mahalanobis**	**0.482**	**(0.273)**
3	**Mahalanobis (normed)**	**0.482**	**(0.273)**
4	**SVM (one-class)**	**0.504**	**(0.316)**
5	Manhattan (scaled)	0.601	(0.337)
6	Manhattan (filter)	0.757	(0.282)
7	Outlier Count (z-score)	0.782	(0.306)
8	Manhattan	0.843	(0.242)
9	Neural Network (auto-assoc)	0.859	(0.220)
10	Euclidean	0.875	(0.200)
11	Euclidean (normed)	0.911	(0.148)
12	Fuzzy Logic	0.935	(0.108)
13	k Means	0.989	(0.040)
14	Neural Network (standard)	1.000	(0.000)

Limitations

Dataset constraints

Small, controlled datasets may not capture real-world keystroke diversity.

Lack of Adaptive Model Design

Static models lack mechanisms like online or continual learning for long-term effectiveness.

Scalability Challenges

Computational overhead limits real-time performance in large-scale systems.

OUR APPROACH

Dataset Description

- free-txt.csv:
  - Contains keystroke timing metrics like dwell time (DU), transitions (DD, UD, UU), and session-based typing data.
- demographics.txt:
  - Provides participant information such as age, gender, handedness, and nationality.

Sample of Cleaned Merged Keystroke Data

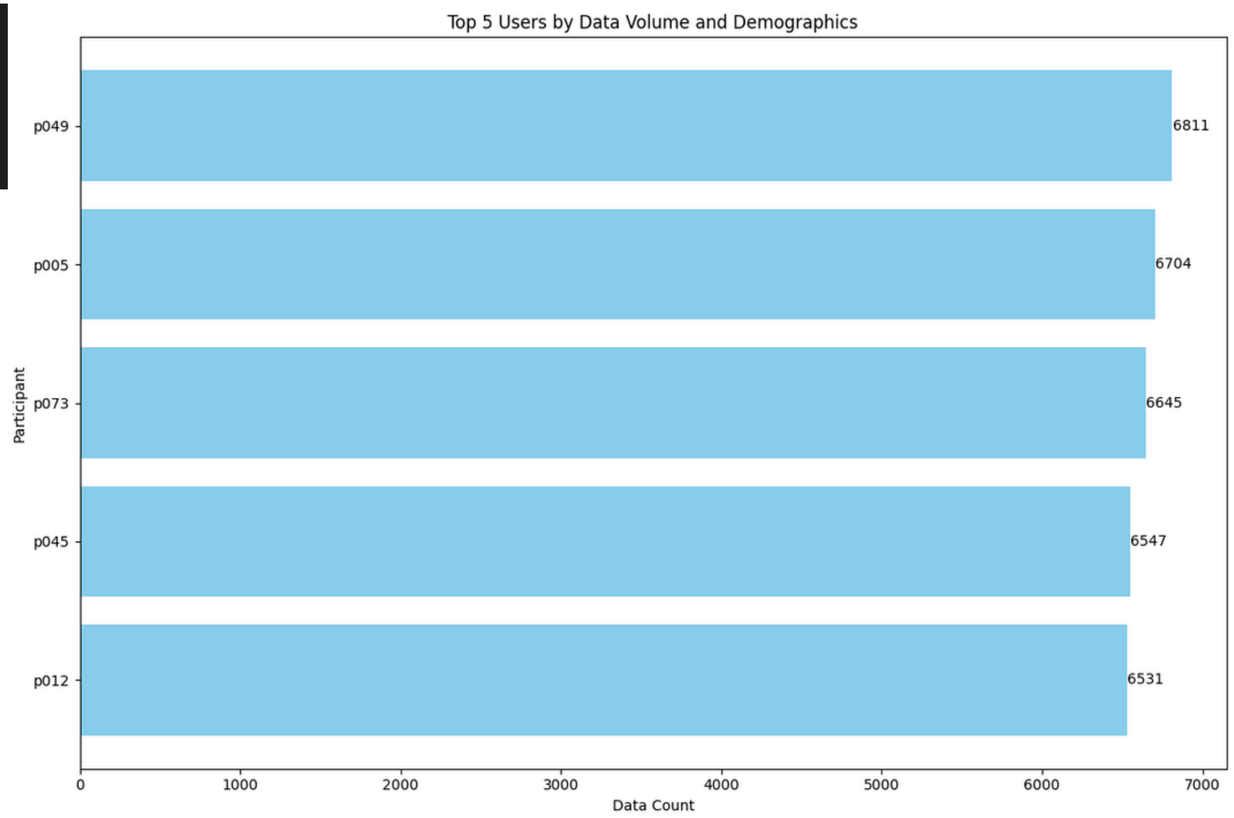
	participant	session	key1	key2	DU.key1.key1	DD.key1.key2	DU.key1.key2	UD.key1.key2	UU.key1.key2	handedness	age	gender	nationality
0	p001	1	W	Shift	0.150	-0.796	0.166	-0.946	0.016	Right-Handed	51	Male	Portugal
1	p001	1	Shift	e	0.962	1.148	1.255	0.186	0.293	Right-Handed	51	Male	Portugal
2	p001	1	e	Space	0.107	0.172	0.252	0.065	0.145	Right-Handed	51	Male	Portugal
3	p001	1	Space	b	0.080	0.200	0.280	0.120	0.200	Right-Handed	51	Male	Portugal
4	p001	1	b	e	0.080	0.320	0.480	0.240	0.400	Right-Handed	51	Male	Portugal

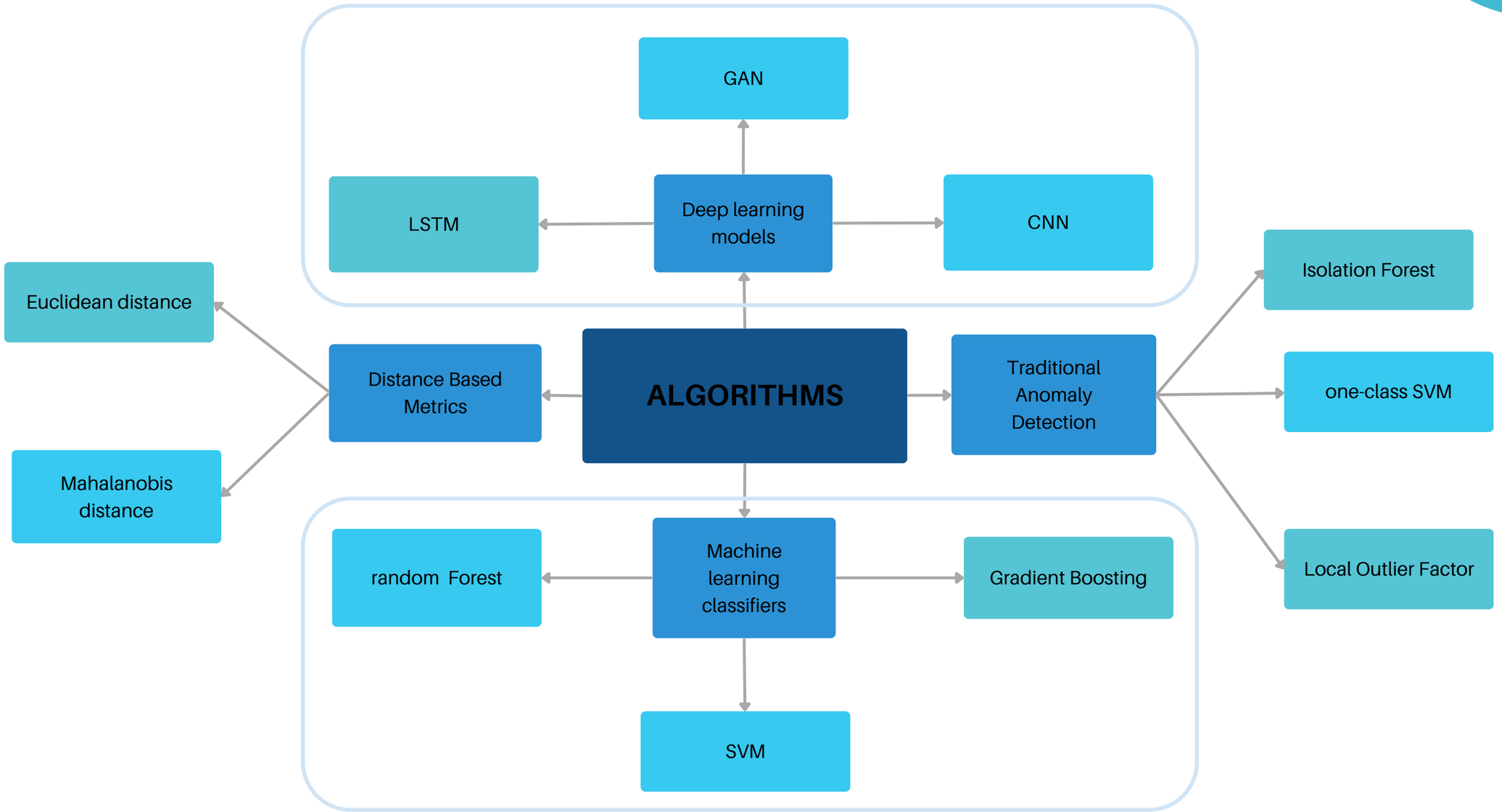
Key features

participant	DU.key1.key1	DD.key1.key2	DU.key1.key2	UD.key1.key2	UU.key1.key2
p001	0.106501	0.313135	0.419588	0.206635	0.313088
p002	0.087517	0.176844	0.264273	0.089326	0.176756
p003	0.184584	0.142621	0.327193	-0.041964	0.142609
p004	0.578485	0.177597	0.795602	-0.400888	0.217117
p005	0.105231	0.318274	0.423303	0.213101	0.318089

https://zenodo.org/records/7886743

Top 5 users




GAN
LSTM
Deep learning models
CNN
Euclidean distance
Distance Based Metrics
ALGORITHMS
Traditional Anomaly Detection
Isolation Forest
one-class SVM
Mahalanobis distance
Local Outlier Factor
random Forest
Machine learning classifiers
Gradient Boosting
SVM

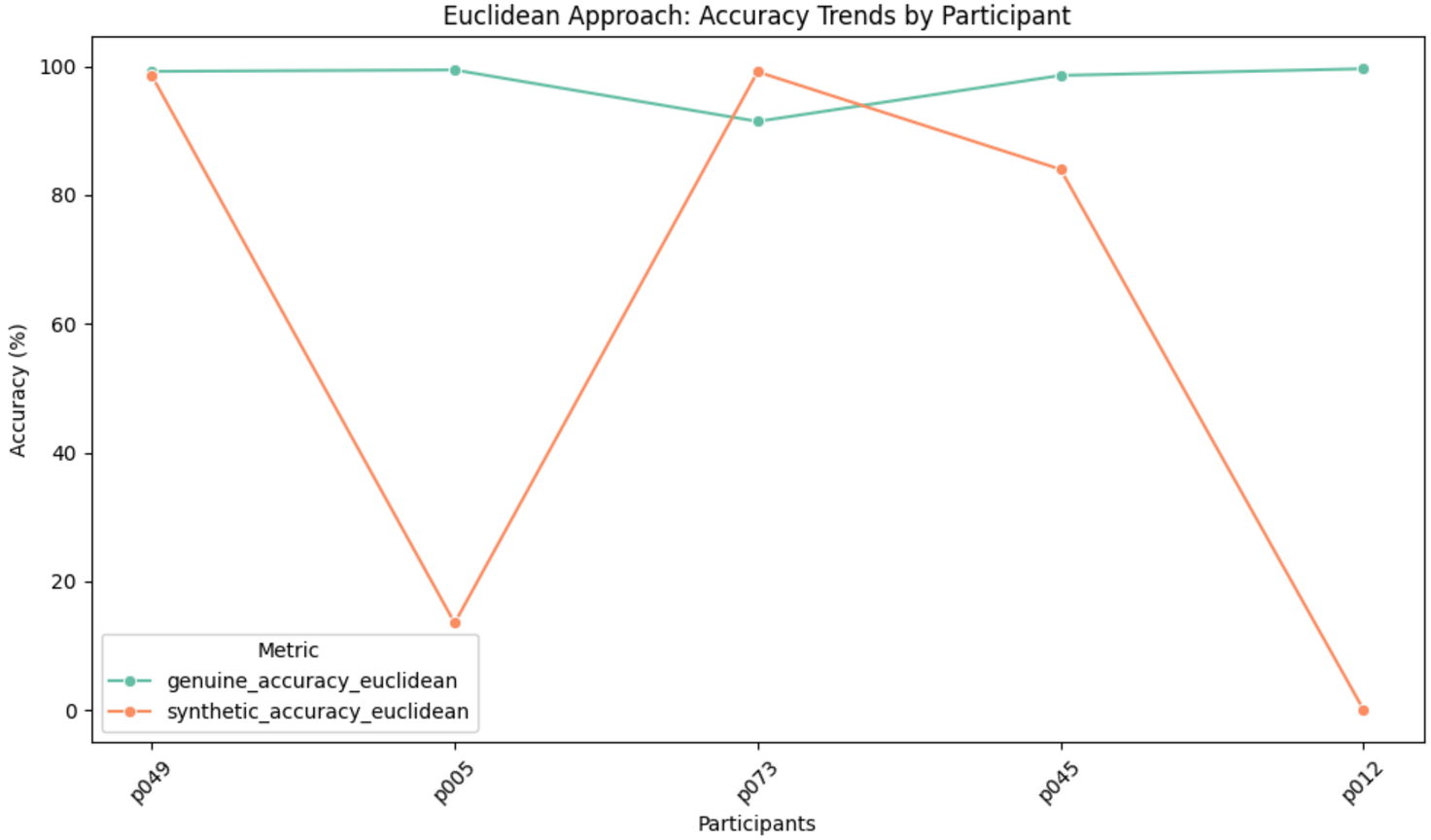
DISTANCE BASED APPROACHES

Euclidean distance

It is a measure of the straight-line distance between two points in an n-dimensional Euclidean space, mathematically defined as: $d(x, y) = \sqrt{\sum_{i=1}^{n}(y_i - x_i)^2}$

Synthetic Data Generation Using Gaussian Distribution

This technique generates synthetic data by sampling from a multivariate Gaussian distribution parameterized by the feature-wise mean vector and standard deviation of the input dataset.

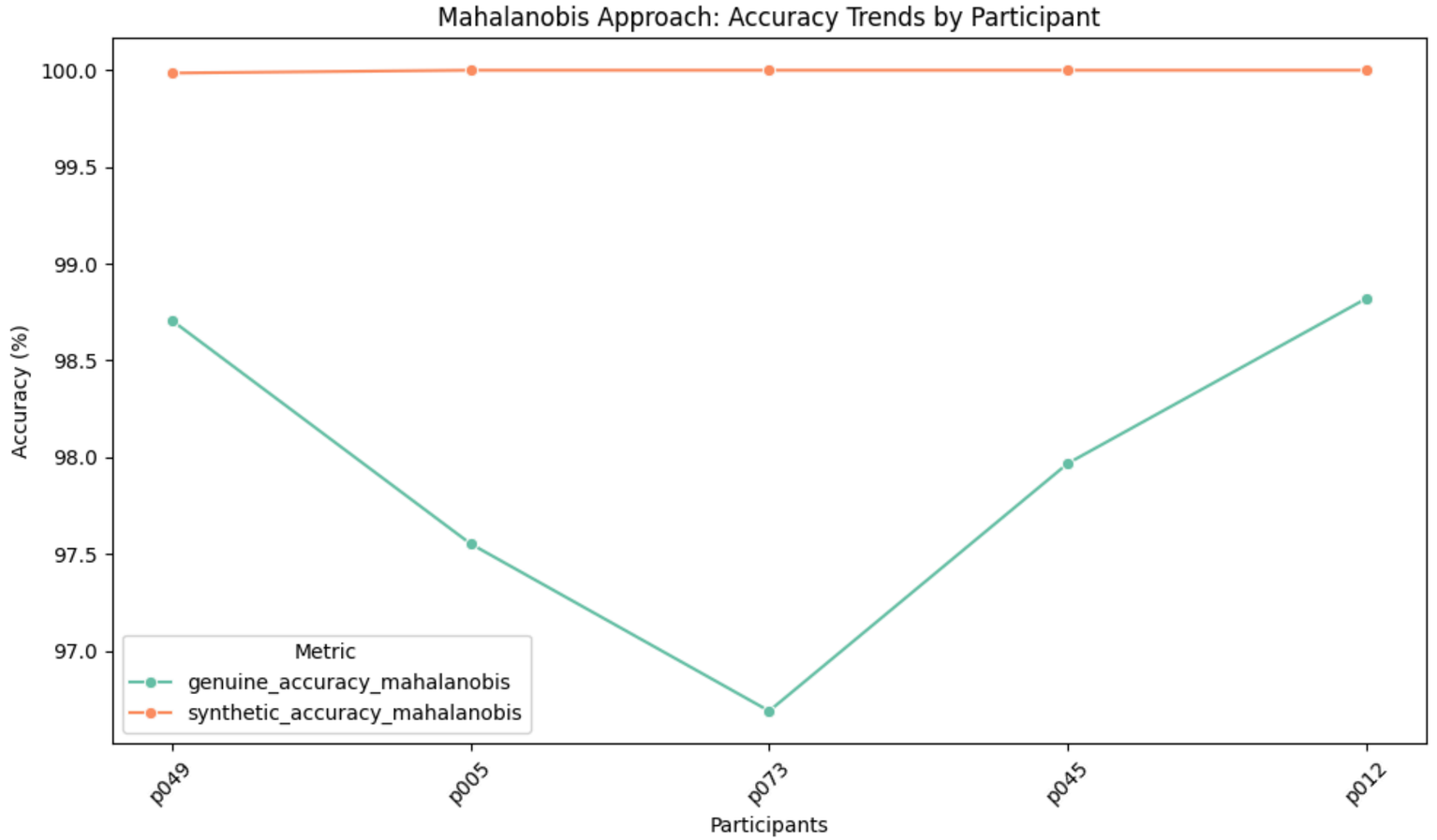


Mahalanobis distance

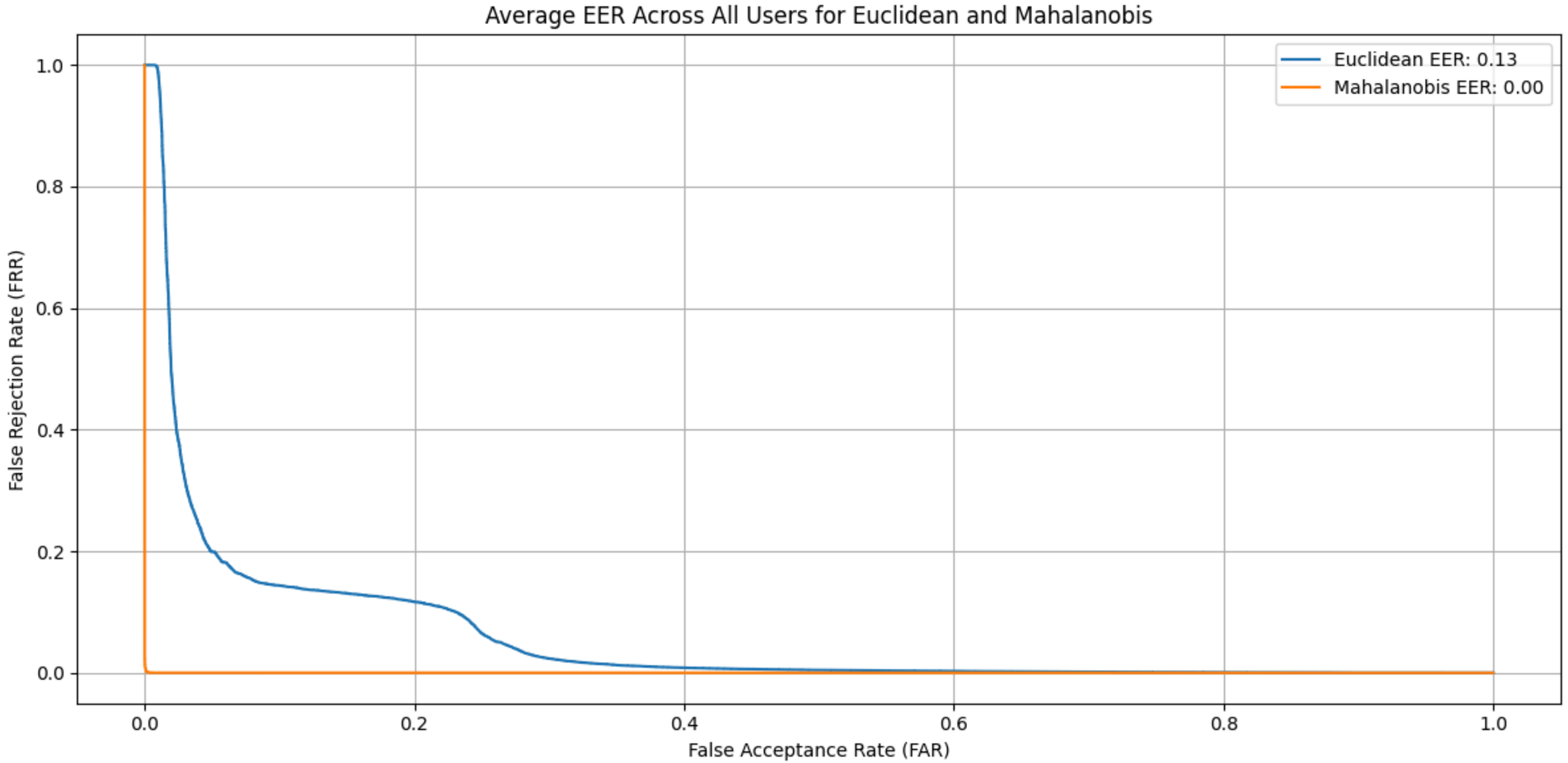
It is a multivariate metric that measures the distance between a point and a distribution, accounting for correlations in the dataset, defined as -

$$M = \sqrt{\frac{(x_1 - \bar{x}_1)^2}{\sigma^2}}$$

Synthetic Data Generation Using Gaussian Distribution



```
Average EER Results Across All Users:
      Approach       EER  Threshold
0    Euclidean  0.134334   0.863764
1  Mahalanobis  0.001549  15.580582
```



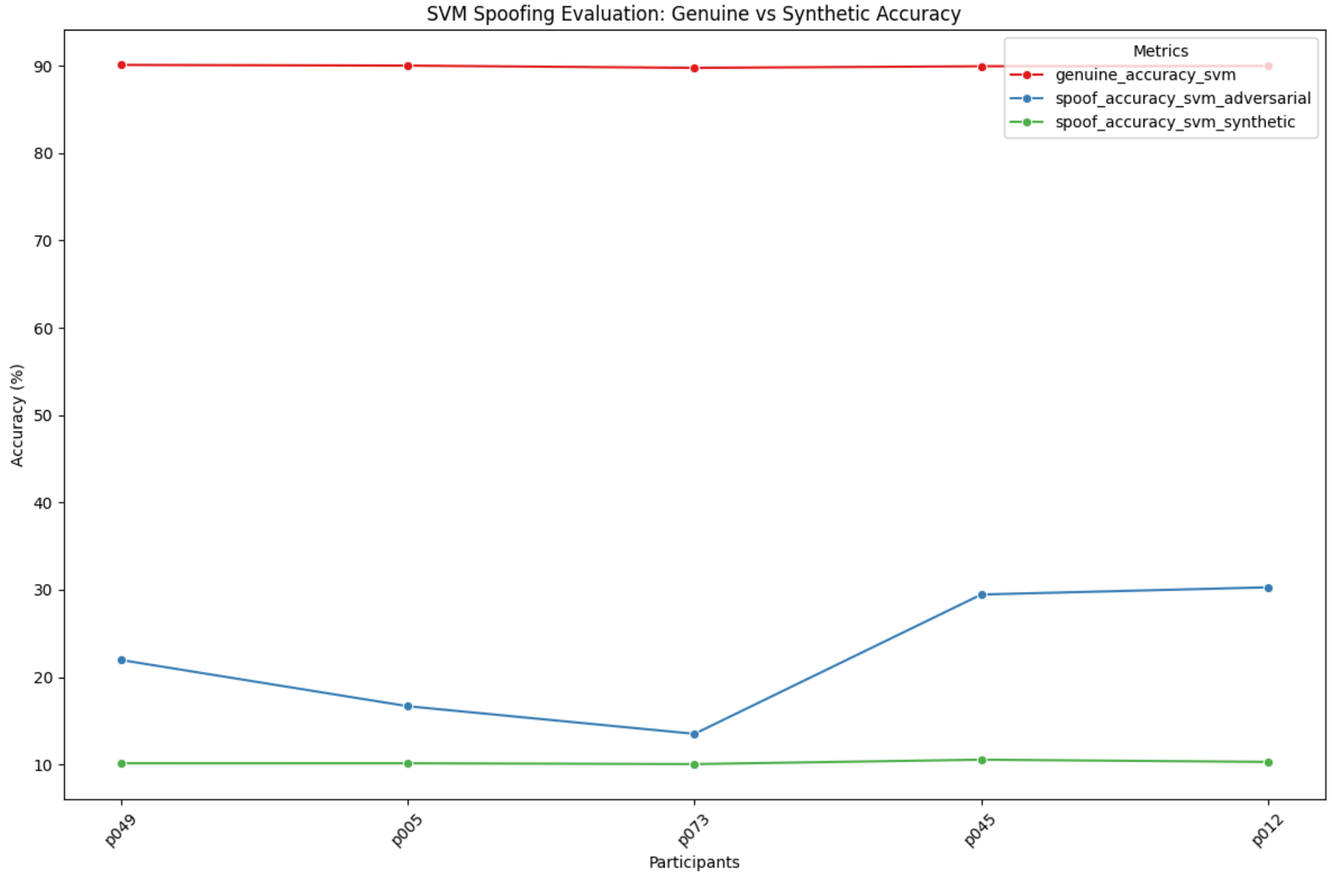
TRADITONAL ANAMOLY DETECTION

One-class SVM

Configured One-Class SVM with an RBF kernel, gamma=0.1 (controls kernel influence), and nu=0.1 (upper bound on anomalies)

Spoofing with Adversarial Perturbation : Introduces deliberate, small changes to data to mimic adversarial attacks.

Spoofing with Data Perturbation : Adds subtle random noise to genuine data to simulate naturally occurring variations.

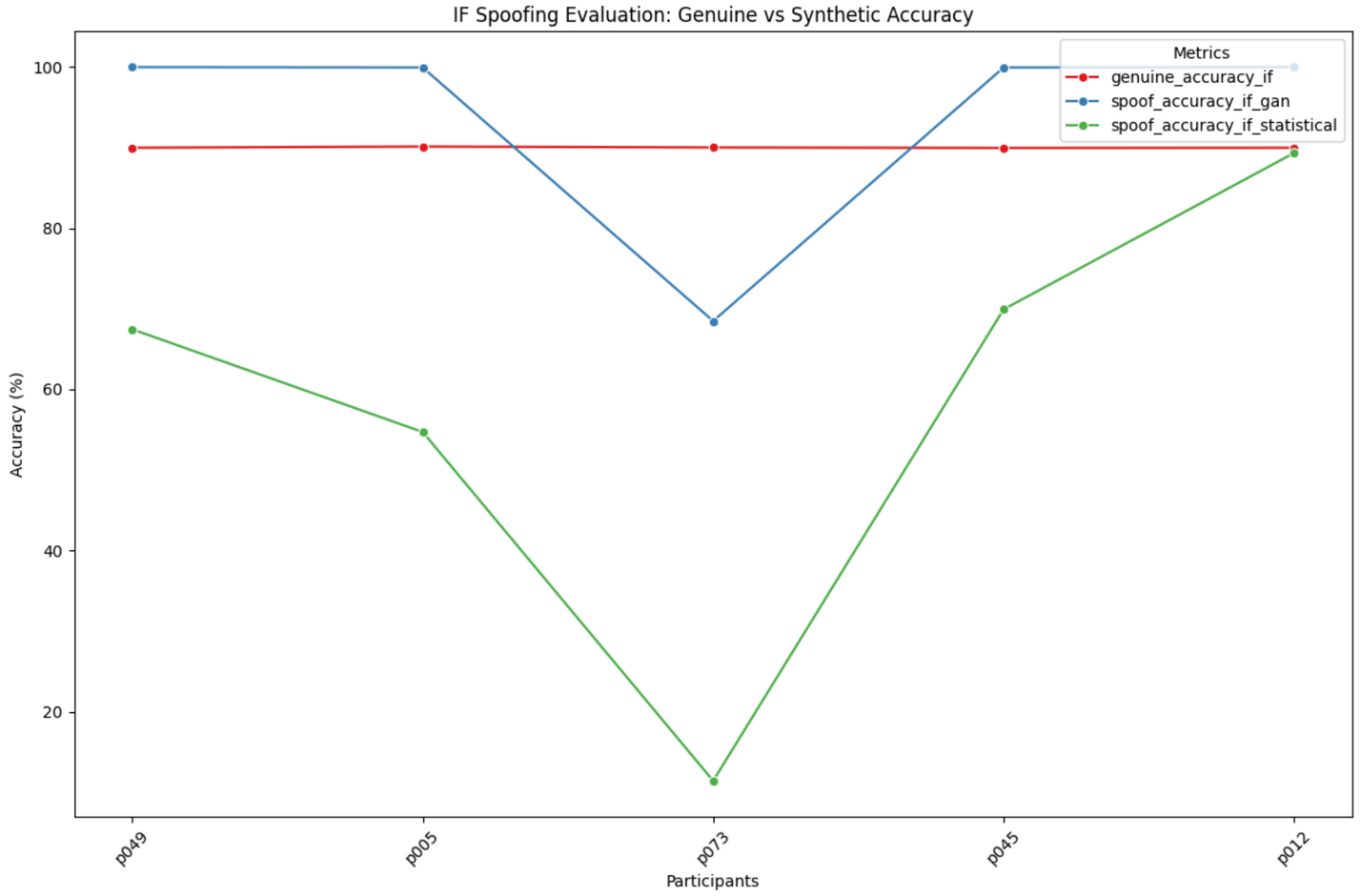


Isolation Forest

Model with a contamination level of 10% to identify outliers in the training data.

GAN-Based Spoofing : Generates spoofed data using a GAN generator to mimic the distribution of genuine data.

Statistical Sampling : Sampling from a normal distribution parameterized by the genuine data's mean and standard deviation**.**

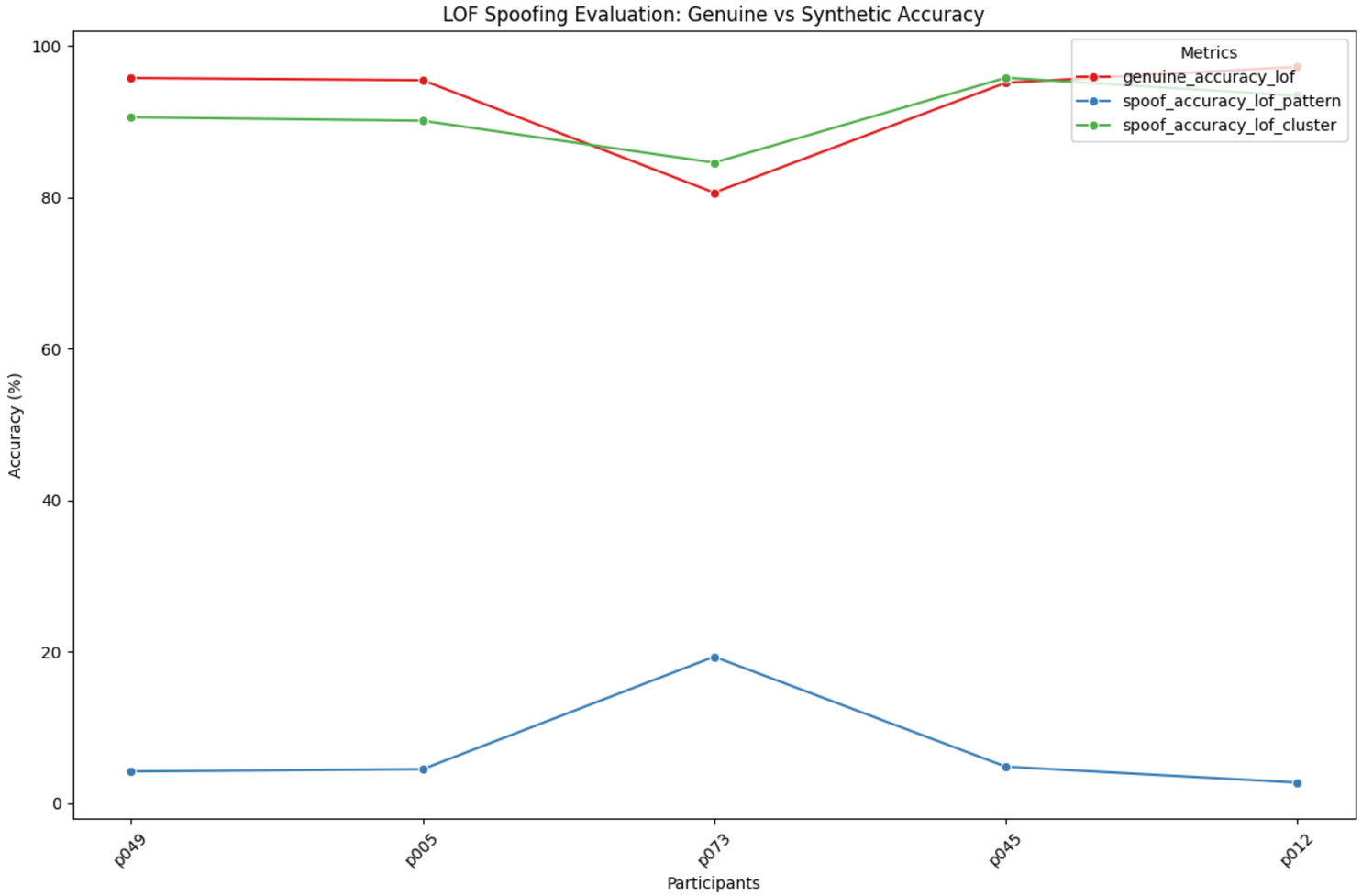


Local Outlier Factor (LOF)

Fit a LOF model with 20 neighbors and novelty detection enabled to identify outliers in the training data.

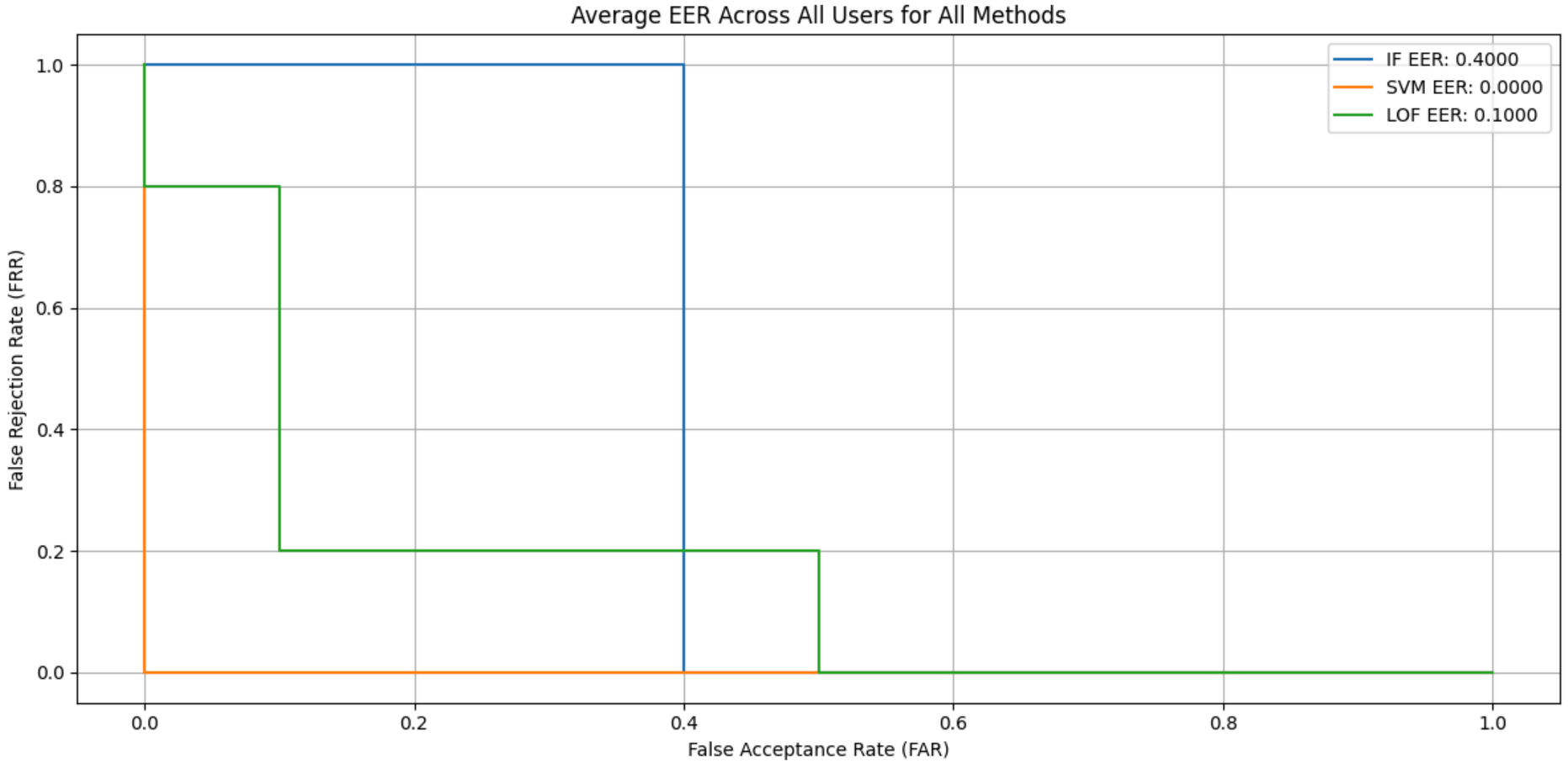
Pattern Mimicking: Rearranges genuine data to mimic realistic yet spoofed patterns.

Cluster Shifting: Introduces small shifts in data clusters to simulate subtle anomalies.



Average EER Results Across All Users:

	Approach	EER	Threshold
0	IF	0.4	3
1	SVM	0.0	2
2	LOF	0.1	3



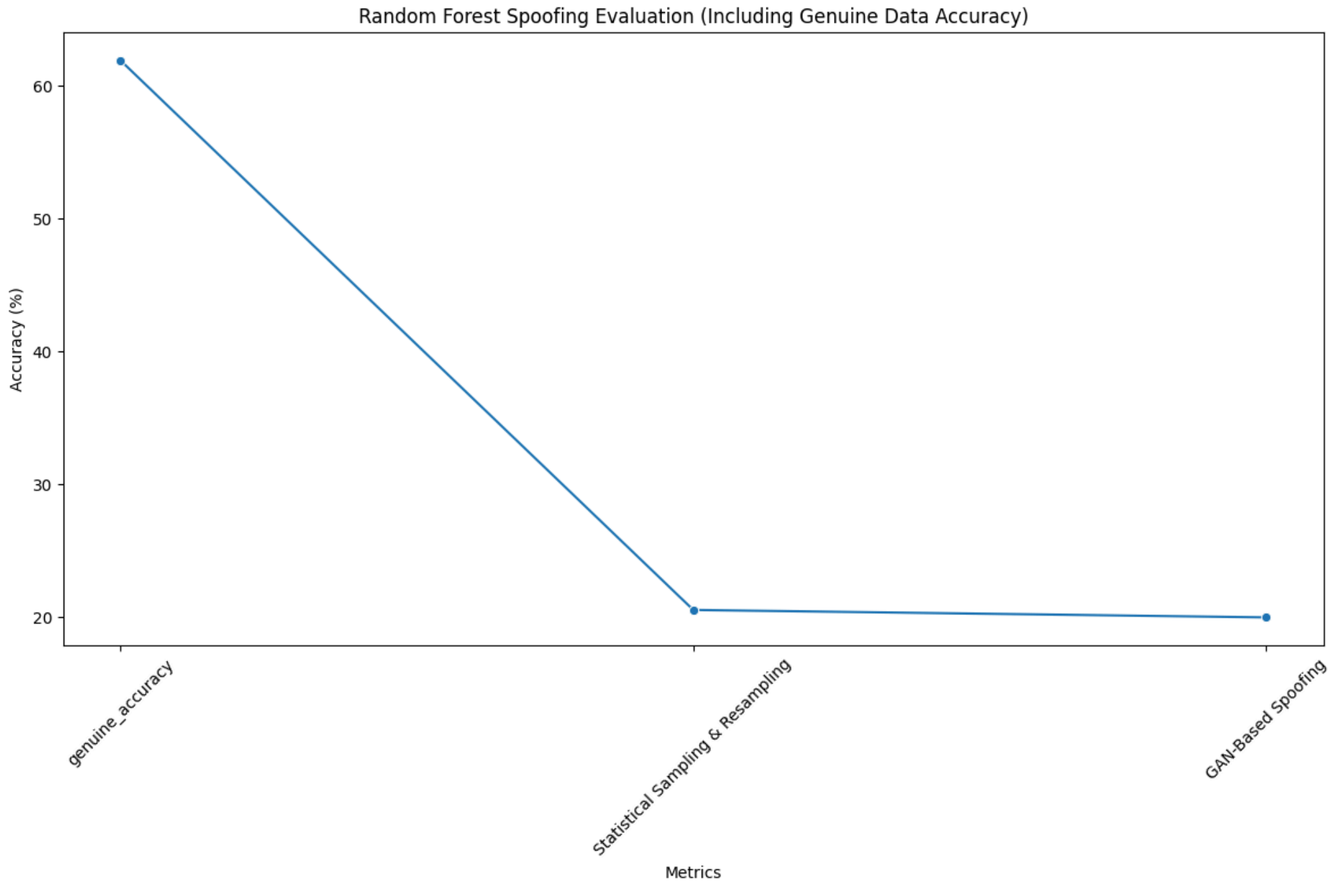
MACHINE LEARNING CLASSIFIERS

Random Forest

Configured with 100 decision trees, a random state of 42

Spoofing with Statistical Sampling & Resampling : Introduces deliberate, small changes to data to mimic adversarial attacks.

GAN-Based Spoofing : Creates synthetic spoofed data using a GAN generator to mimic genuine data patterns.

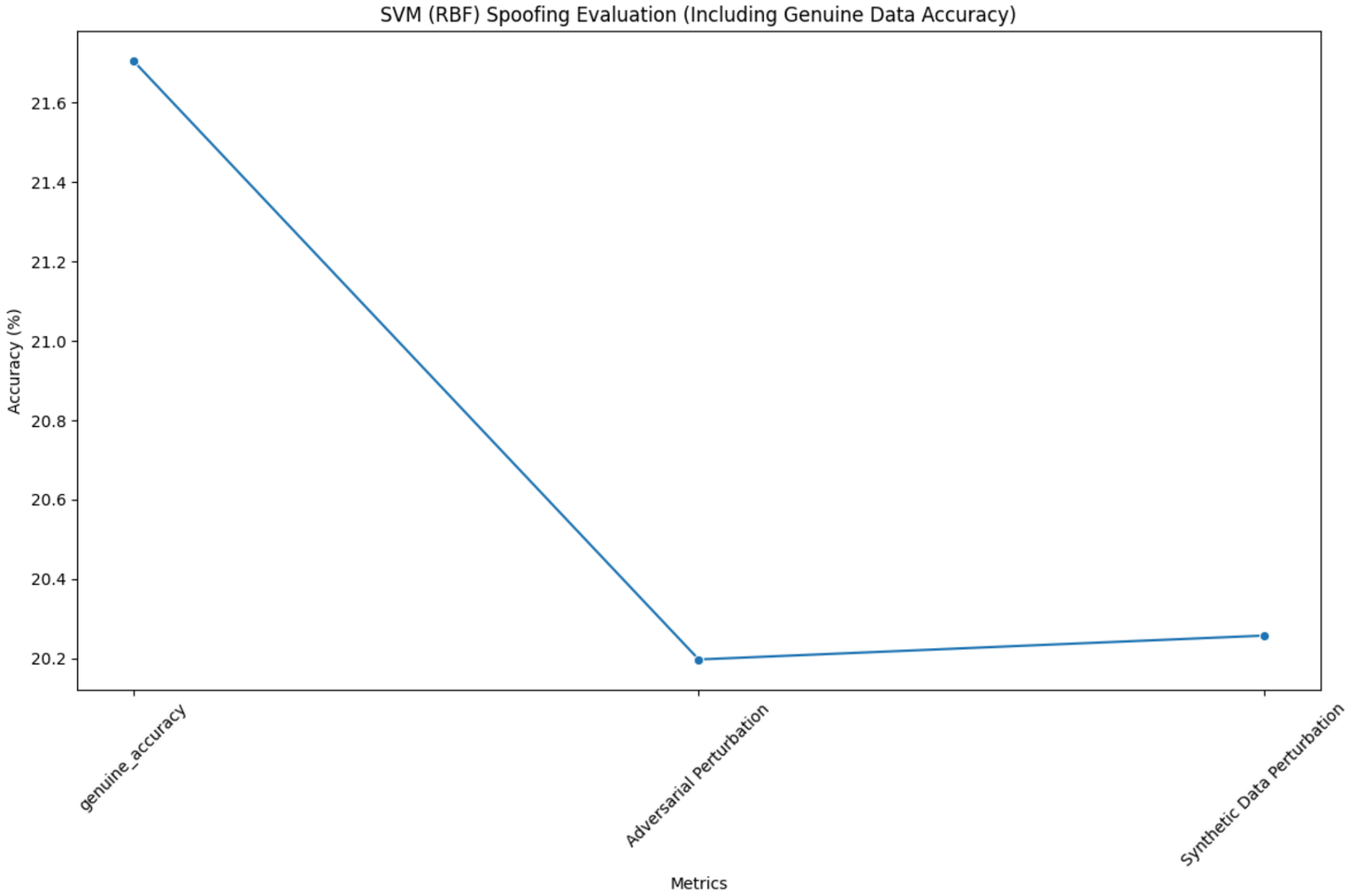


SVM

SVM (RBF) is configured with C=1.0, gamma='scale'

Adversarial Perturbation Spoofing : Adds deliberate small perturbations to genuine data to simulate adversarial attacks.

Spoofing with Data Perturbation : Adds subtle random noise to genuine data to simulate naturally occurring variations.

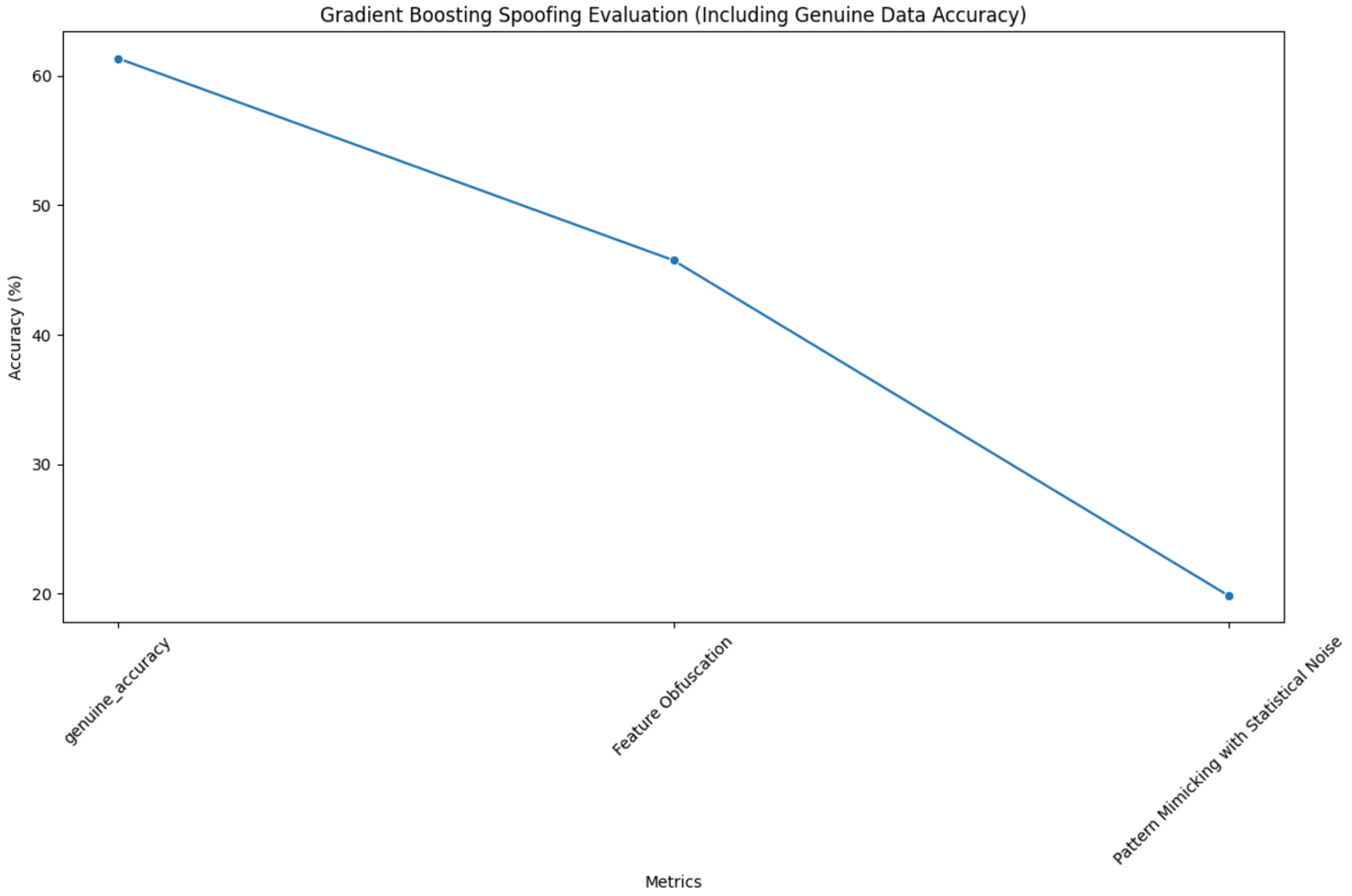


Gradient Boosting

Configured with 100 estimators, a learning rate of 0.1

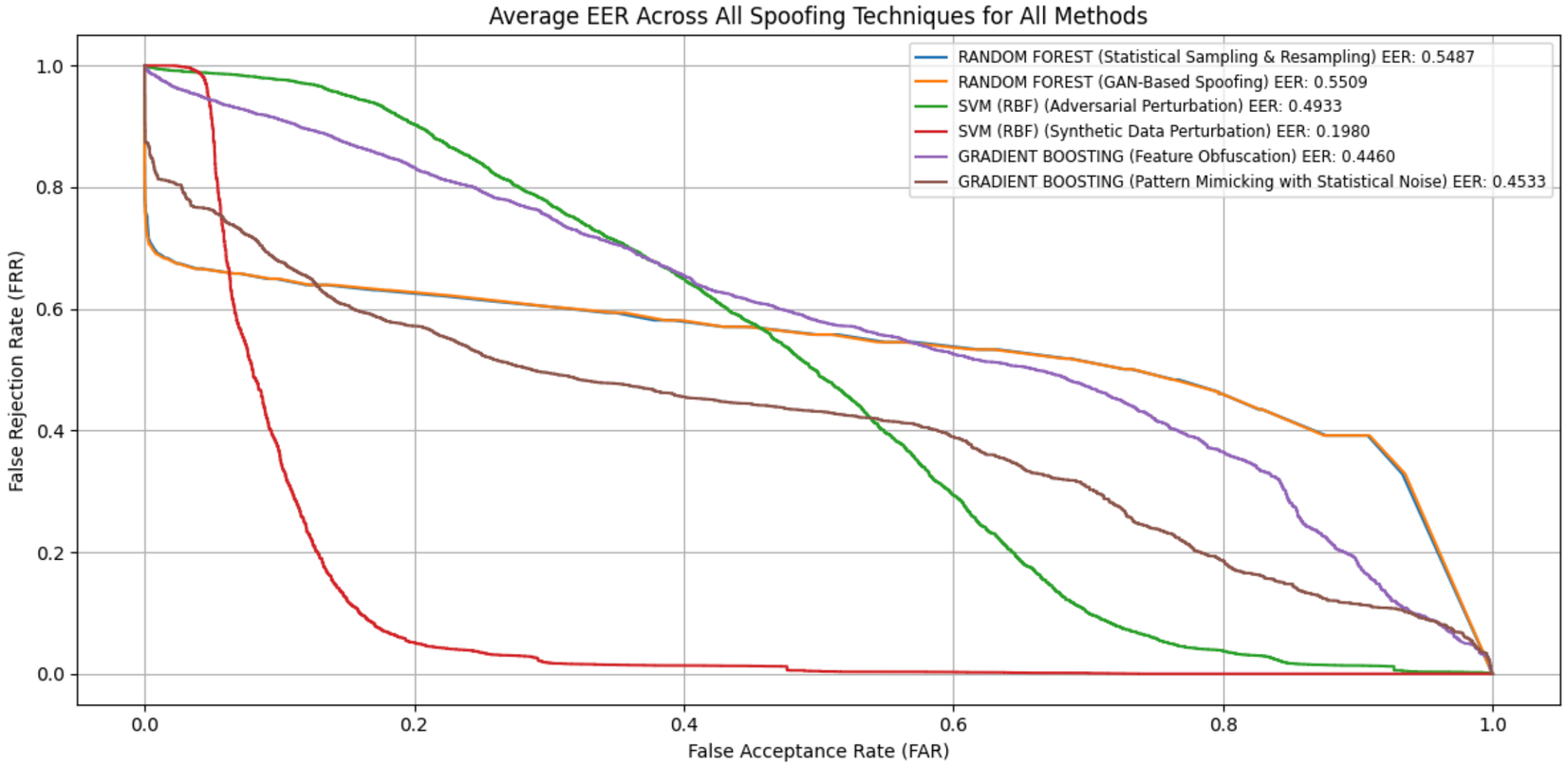
Feature Obfuscation Spoofing : Obscures a fraction of features to simulate pattern-mimicking spoofing.

Pattern Mimicking: Rearranges genuine data to mimic realistic yet spoofed patterns.



```
Average EER Results Across All Methods:
              Method                        Spoofing Technique       EER  \
0      RANDOM FOREST         Statistical Sampling & Resampling  0.548726
1      RANDOM FOREST                       GAN-Based Spoofing  0.550892
2          SVM (RBF)                  Adversarial Perturbation  0.493276
3          SVM (RBF)               Synthetic Data Perturbation  0.197975
4  GRADIENT BOOSTING                        Feature Obfuscation  0.446042
5  GRADIENT BOOSTING  Pattern Mimicking with Statistical Noise  0.453261

   Threshold
0      292.2
1      292.2
2     3114.6
3     1934.8
4     3039.6
5     2640.8
```



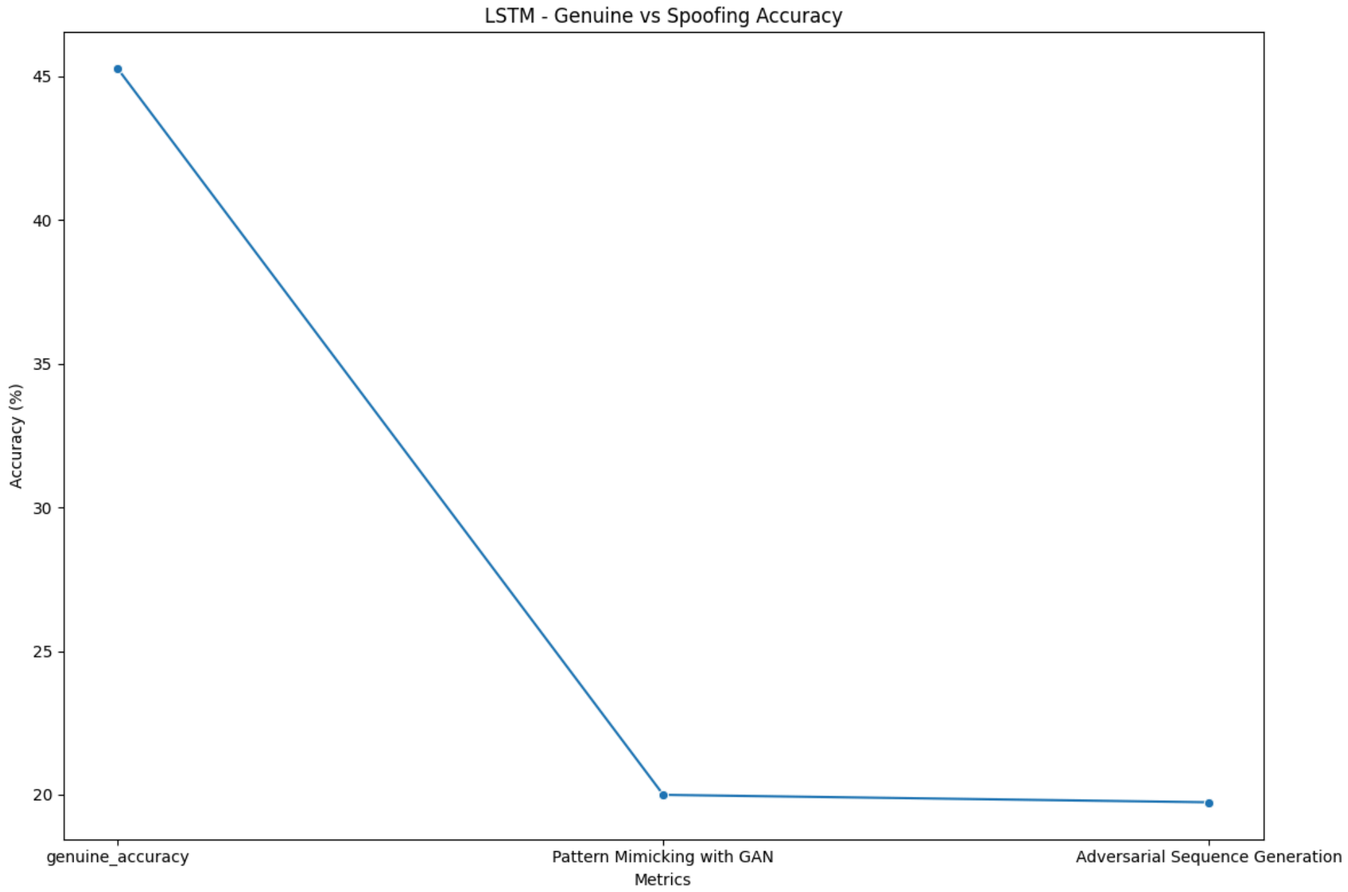
DEEP LEARNING MODELS

Long Short-Term Memory (LSTM)

2 LSTM layers (64 and 32 units), trained using Adam optimizer

Pattern Mimicking : Generates synthetic data by mimicking genuine patterns using a Gaussian distribution.

Adversarial Sequence Generation : Simulates adversarial attacks by generating plausible sequence-based spoofed data.

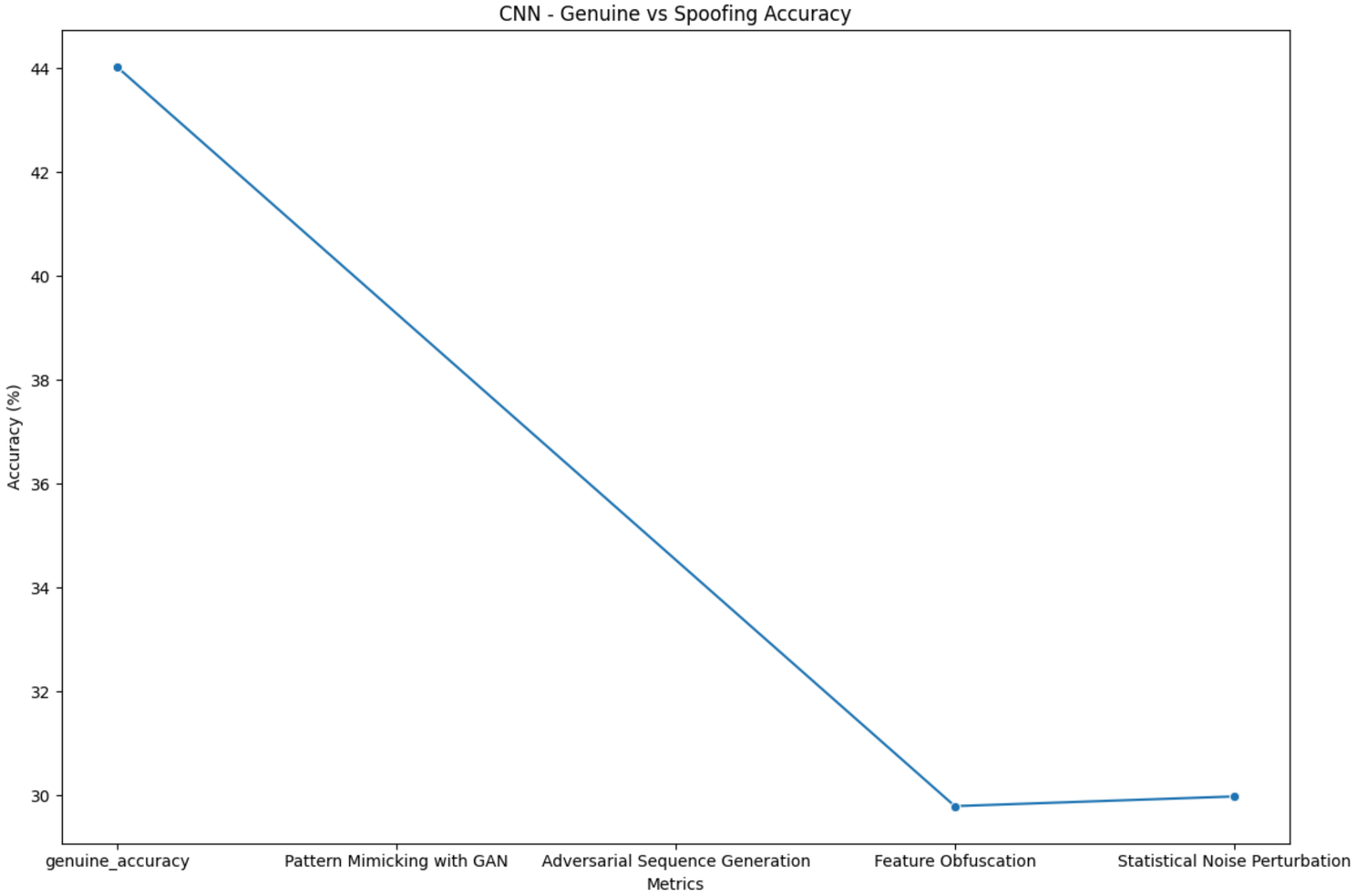


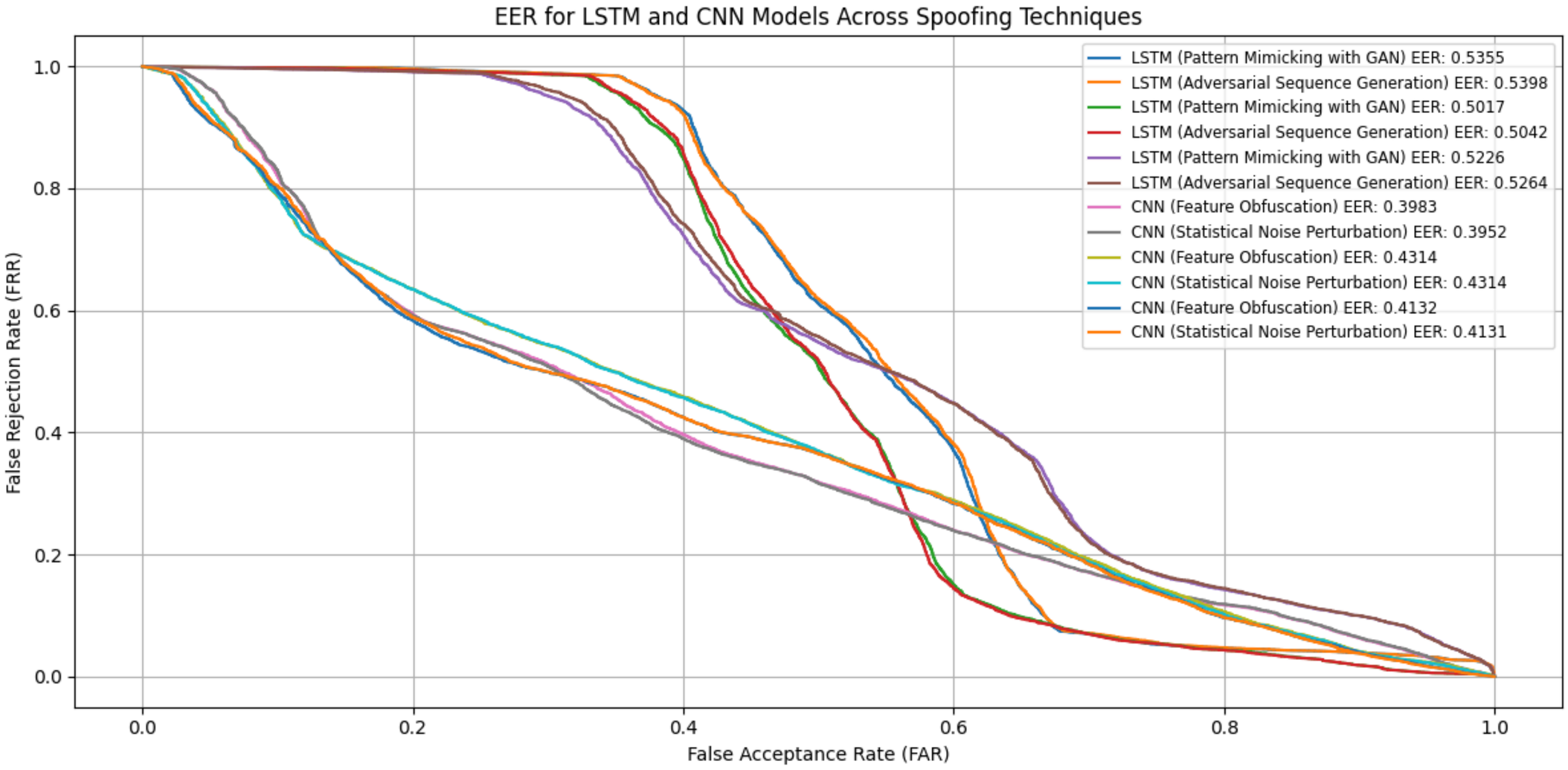
Convolutional Neural Networks (CNN)

2 Conv1D layers (32 and 64 filters), followed by pooling layers, trained using Adam optimizer

Feature Obfuscation: Hides a fraction of input features to simulate pattern-mimicking spoofing.

Statistical Noise Perturbation : Introduces noise into data to create plausible but spoofed variations.

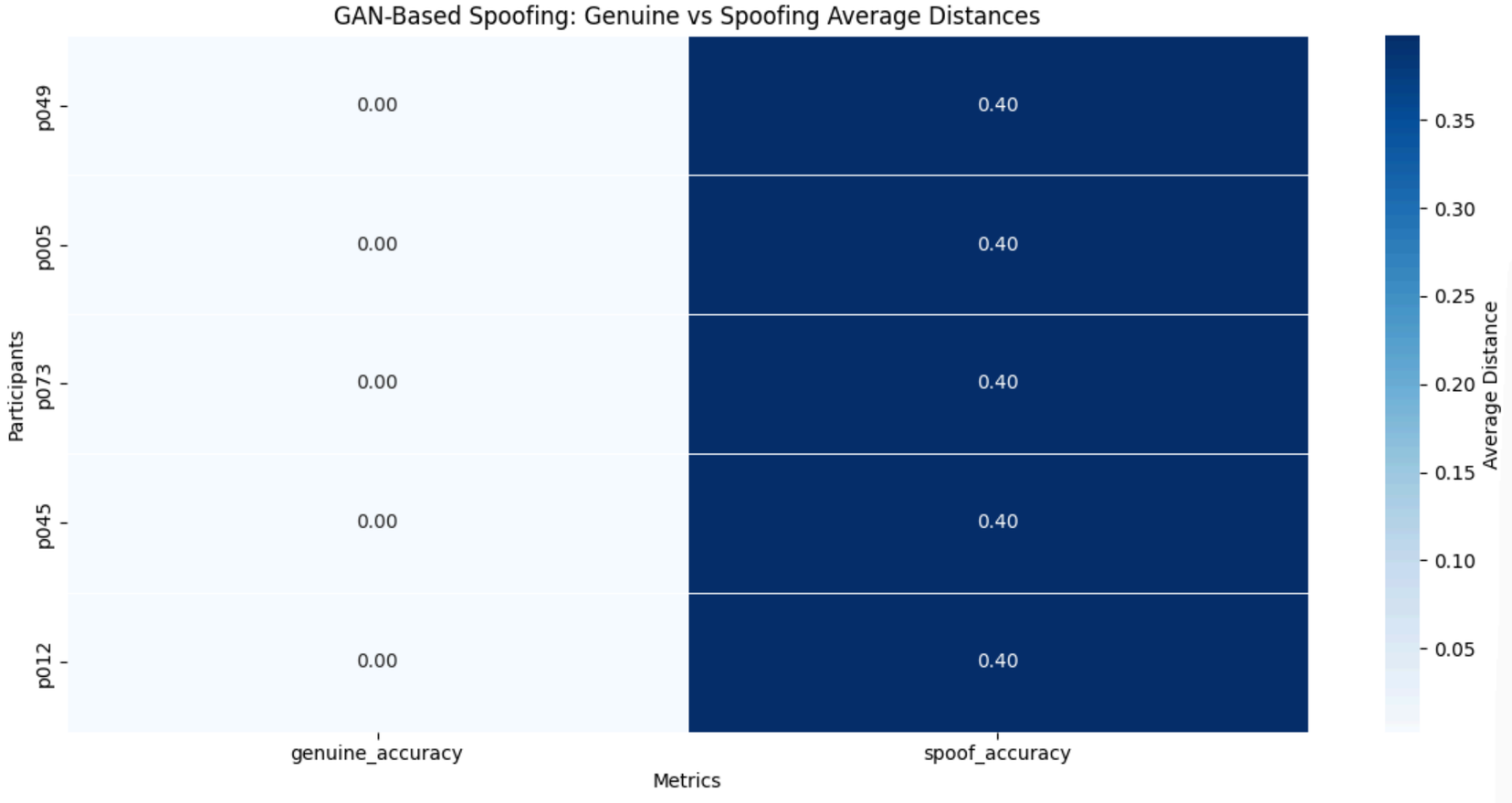



EER for LSTM and CNN Models Across Spoofing Techniques
False Rejection Rate (FRR)
False Acceptance Rate (FAR)
0.0
0.2
0.4
0.6
0.8
1.0
LSTM (Pattern Mimicking with GAN) EER: 0.5355
LSTM (Adversarial Sequence Generation) EER: 0.5398
LSTM (Pattern Mimicking with GAN) EER: 0.5017
LSTM (Adversarial Sequence Generation) EER: 0.5042
LSTM (Pattern Mimicking with GAN) EER: 0.5226
LSTM (Adversarial Sequence Generation) EER: 0.5264
CNN (Feature Obfuscation) EER: 0.3983
CNN (Statistical Noise Perturbation) EER: 0.3952
CNN (Feature Obfuscation) EER: 0.4314
CNN (Statistical Noise Perturbation) EER: 0.4314
CNN (Feature Obfuscation) EER: 0.4132
CNN (Statistical Noise Perturbation) EER: 0.4131

Generative Adversarial Network (GAN)

Discriminator: A neural network that evaluates and distinguishes between real data and synthetic data produced by the generator.
2 Layers (64 + 128) with LeakyReLU activation and 1 Layers with tanh activation

Generator: A neural network that learns to produce realistic synthetic data by mapping random noise to the data distribution.
2 Layers (128 + 64) with LeakyReLU activation and 1 Layer with sigmoid activation



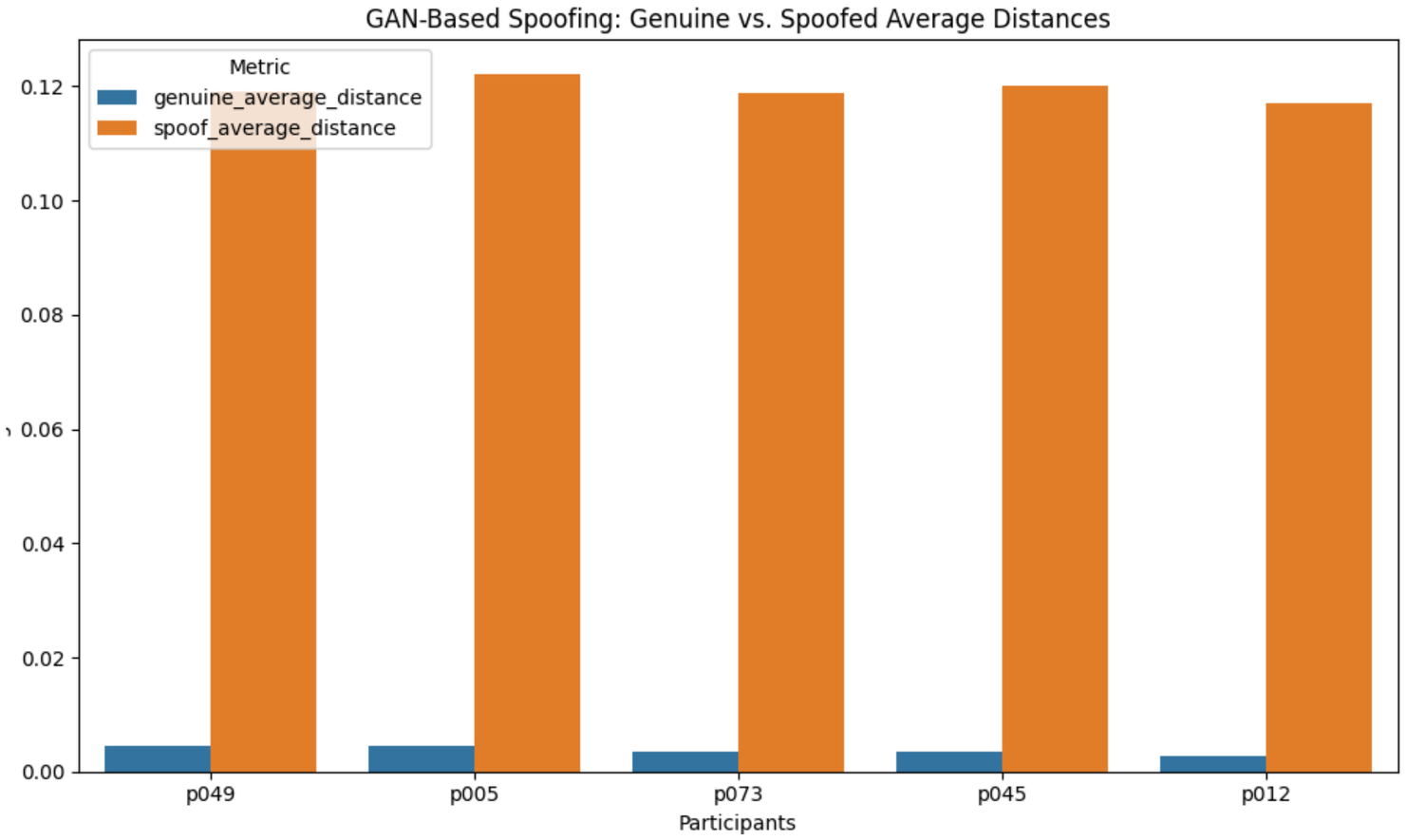
Trained with 500 Epochs and Adam Optimizer

Generative Adversarial Network (GAN)

1000 Epochs with Adam Optimizer with Learning rate 0.0002 and 0.0001 for Generator and Discriminator respectively

Discriminator : 2 Layers (64 + 128) with LeakyReLU activation and 1 Layers with tanh activation

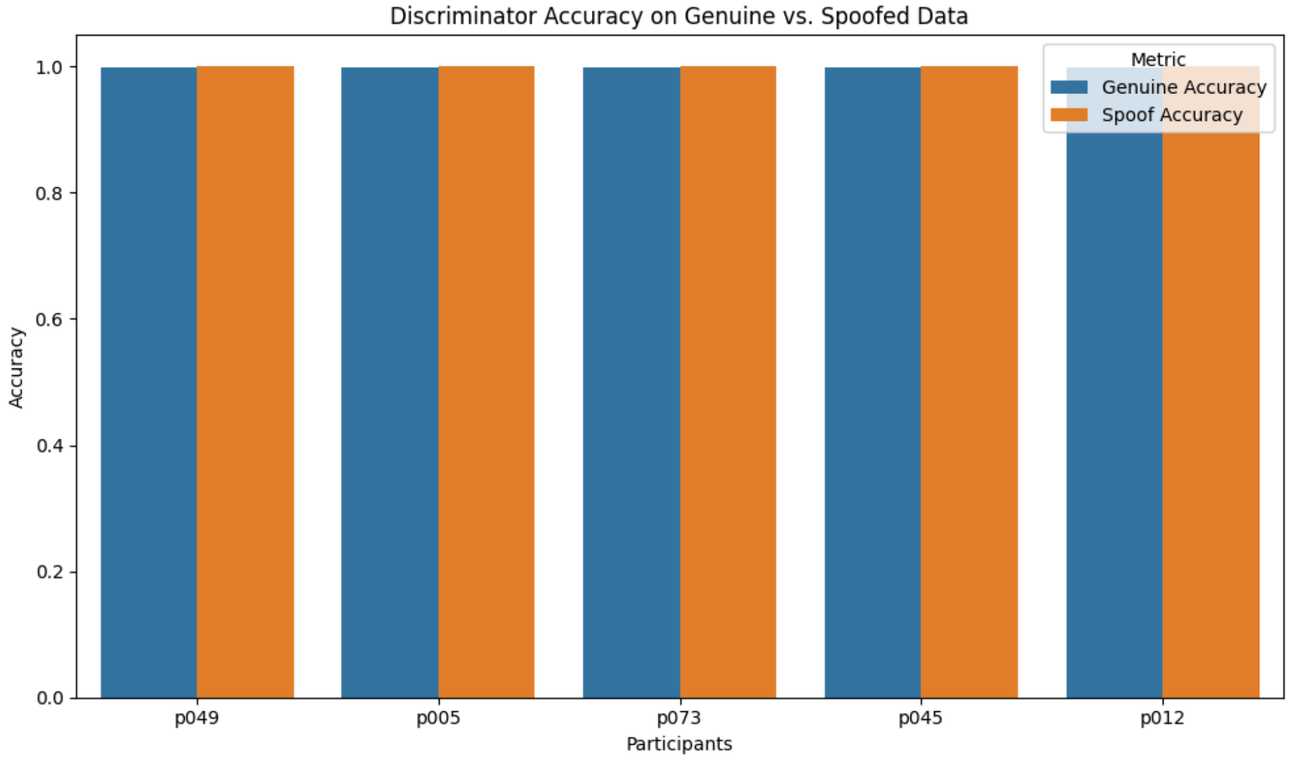
Generator: 2 Layers (128 + 64) with LeakyReLU activation and 1 Layer with sigmoid activation

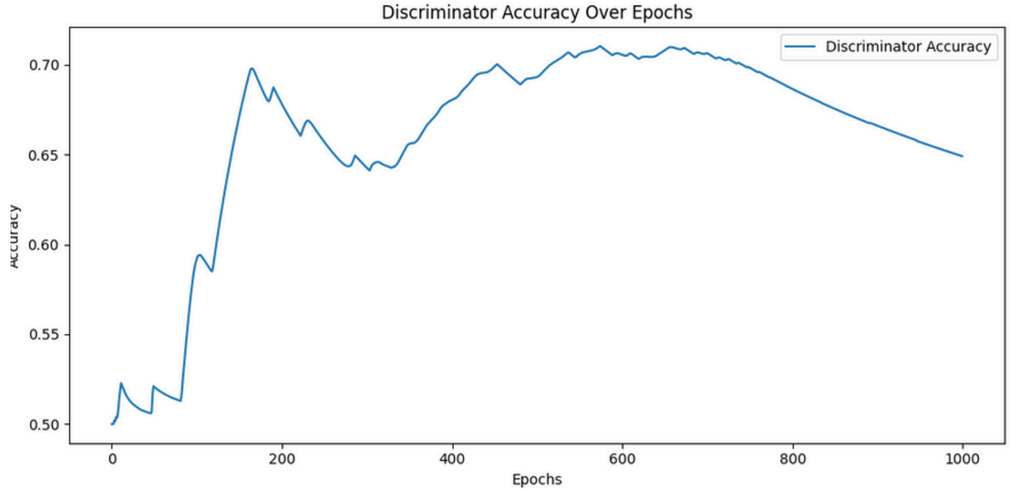


Generative Adversarial Network (GAN)

Discriminator : 3 Layers (256+512+256) with LeakyReLU activation and 1 Layer with sigmoid activation

Generator : 3 Layers (256+512+256) with LeakyReLU activation and 1 Layer with tanh activation





CONCLUSION

Model/Metric	Genuine Accuracy	Spoofing Accuracy	Recommendations
Gradient Boosting	95-99%	Low (~5%)	Best Performer
LSTM	~28%	Low (~21%)	Weak Resilience
CNN	~27%	Moderate (~26%)	Needs Improvement
GAN (Discriminator)	98-99%	Moderate (~28%)	Strong Against Spoofing
Euclidean Distance	98-99%	High (60-90%)	Moderate Performance
Isolation Forest	~90%	High (50-70%)	Poor Against Spoofing

RECOMMENDATIONS

- **Gradient Boosting:** Best performer with high genuine accuracy and robust spoofing detection; ideal for real-world applications.
- **LSTM:** Moderate spoofing resilience; requires optimization for improved genuine accuracy and robustness.
- **CNN:** Needs architectural enhancements for better spoofing resistance; suitable for targeted improvements.
- **GAN Discriminator:** Effective in adversarial scenarios; complements Gradient Boosting for spoofing detection.
- **Euclidean Distance & Isolation Forest:** Poor spoofing resistance; unsuitable for critical systems, viable for low-security use cases.

THANK YOU!

Any questions?