Advanced Deepfake Detection using Machine Learning Algorithms: A Statistical Analysis and Performance Comparison

Conference Paper · March 2024

DOI: 10.1109/ICICT62343.2024.00019

2 authors:

Md Shohel Rana
University of Massachusetts Dartmouth
29 PUBLICATIONS **595** CITATIONS

SEE PROFILE

Andrew H. Sung
University of Southern Mississippi
33 PUBLICATIONS **689** CITATIONS

SEE PROFILE

2024 7th International Conference on Information and Computer Technologies (ICICT) | 979-8-3503-8562-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICICT62343.2024.00019

Advanced Deepfake Detection using Machine Learning Algorithms: A Statistical Analysis and Performance Comparison

Md Shohel Rana*
Computing and Software Engineering
Florida Gulf Coast University
Fort Myers, FL 33965
mrana@fgcu.edu

Andrew H. Sung
Computing Sciences and Computer Engineering
The University of Southern Mississippi
Hattiesburg, MS 39406
andrew.sung@usm.edu

***Abstract*— As techniques and tools for synthetic media and Deepfakes continue to advance, it is increasingly clear that video, audio and images can no longer be relied upon as truthful recordings of reality. Every digital communication channel is now vulnerable to manipulation, and there is widespread use of Deepfakes to propagate misinformation and disinformation, inflame political discord, defame opposition, commit cyber frauds or blackmail individuals. While deep learning (DL) methods have been widely used to identify Deepfakes, this paper demonstrates that classical machine learning (ML) methods can achieve superior performance--comparable with or exceeding state-of-the-art DL methods in detecting Deepfakes. Using the traditional procedures of feature development and selection, training, and testing of ML classifiers for the task actually provides better understandability and interpretability while consuming much less computing resource. In addition, an omnibus test, the Analysis of Variance (ANOVA), is conducted to compare the performance of multiple ML models. We present experiments that achieve 99.84% accuracy on the FaceForecics++ dataset, 99.38% accuracy on the DFDC dataset, 99.66% accuracy on the VDFD dataset, and 99.43% accuracy on the Celeb-DF dataset. Our study thus challenges the notion that DL approaches are the only effective way to detect Deepfakes and demonstrates that judicious use of ML approaches can be highly efficacious and cost-effective.**

Keywords—Deepfakes, Deepfake Detection, Face Manipulation, Machine Learning, Analysis of Variance, Omnibus Test.

I. INTRODUCTION

Deepfake technology has evolved significantly, showcasing notable progress in generative modeling for real-time facial reenactment and image transformation using tools like CycleGAN [1]. The University of Washington's lip synchronization method [2] has further improved the alignment of lip movements in videos with external speech sources. Beyond its initial application in pornography, Deepfake technology has been exploited for deceptive political propaganda, fake news, and scams, posing risks to public trust and national security. Noteworthy incidents, such as the manipulated 2019 video of Nancy Pelosi, underscore the potential for misinformation and its grave consequences. Deepfakes also present threats to personal security, enabling identity theft and fraud through convincing audio clips that mimic individuals' voices. The growing prevalence of Deepfake pornography raises concerns about privacy, consent, and potential harm, as evidenced by a 2019 study [3] revealing thousands of Deepfake videos on top pornographic websites.

In light of the harmful impact of Deepfakes, it is imperative to develop robust detection systems. Existing approaches to Deepfake detection include using forensic analysis, DL techniques, and human verification. However, each of these approaches has its limitations, and more research is needed to improve their accuracy and effectiveness. Many academics and experts have turned to DL-based methods to automatically detect Deepfakes as these methods are directly involved in feature engineering, which lessens the burden on human analysts. However, these methods can also be difficult to interpret and comprehend due to the complexity of the models involved. This can create a trade-off between accuracy and interpretability when assessing any machine learning (ML) model. DL-based approaches are often better at achieving high accuracy than classical ML methods, particularly in complex scenarios with non-linear relationships and interactions between inputs. However, they can also be resource-intensive and require large amounts of data to train effectively. In contrast, classical ML algorithms can be easier to understand and adjust since they involve direct feature engineering. For example, tree-based methods like decision trees and random forests show the decision-making process as a tree, making them more interpretable. However, DL approaches can be difficult to interpret because they are essentially black boxes with no explicit rules for inferring the decision process. As a result, it is a shared challenge for researchers to define what it means to explain the behavior of DL systems and evaluate their ability to explain their decisions. Despite the many DL approaches to detecting Deepfakes, our ongoing research [4-5] has found that classical ML algorithms can be equally effective, with the added benefit of greater interpretability and lower computational requirements.

The purpose of this paper is to advocate classical ML approaches to Deepfake detection, highlighting its strengths and limitations. This paper also presents the results of an analysis of variance (ANOVA) to compare the performance of multiple ML models on several Deepfake datasets and aims to provide insights into the current state of the art in Deepfake detection and identify future directions for research. This can open up new avenues for research in the field of Deepfake detection.

*Work was supported in part by Florida Gulf Coast University.

DOI 10.1109/ICICT62343.2024.00019

The main motivation for this study is to address the limitations of DL-based approaches in detecting Deepfakes, which suffer from issues of interpretability, complexity, and resource-intensive training requirements. To overcome these limitations, we sought to evaluate the effectiveness of classical ML methods in detecting Deepfakes. In this study, we first created a unique set of features by combining several popular feature extraction techniques, including Histogram of Oriented Gradients (HOG), Haralick features, Hu Moments, and Color Histogram features. This feature set was designed to capture a range of image characteristics that are typically altered in Deepfakes, such as texture, color, and shape. Next, we split the Deepfake detection task into two stages: object detection and object recognition. In the object detection phase, we scanned the entire image and identified all possible objects, while in the object recognition step, we identified relevant objects. This approach helped to lessen the complexity of the data and make patterns more recognizable. One of the key advantages of this ML-based approach is its greater understandability and interpretability compared to DL approaches. This is because classical ML algorithms involve direct feature engineering and decision-making processes that can be more easily understood and modified by humans. Additionally, our ML approach significantly reduced training time while achieving comparable performance to the most advanced DL algorithms. For instance, while DL models, such as ResNet can take nearly two weeks to train, the ML methods take only between several seconds and a couple of hours to train on datasets such as FaceForensics++ (FF++) [6], DFDC [7], Celeb-DF [8], and VDFD [9] (our newly proposed). To evaluate the effectiveness of ML models, we conducted an omnibus test using various statistical methods. This study includes an analysis of the performance of our ML-based approach on several benchmark datasets, as well as a comparison with other state-of-the-art DL methods. Overall, this study demonstrates that classical ML approaches can be effective in detecting Deepfakes and offer several advantages over DL-based approaches. We believe that the given results have important implications for the development of more interpretable and efficient Deepfake detection methods in the future.

In the rest of the paper, Section II describes a literature review; Section III presents proposed methodology; Section IV presents results and discussion; and Section V gives conclusions and future work.

II. Literature Review

Deepfake technology employs a central approach of manipulating human faces to confound its viewers. This manipulation is approached through various methods. However, the majority of techniques concentrate on altering specific facial regions, such as eye shading or the presence of accessories like earrings, to deceive observers. Such methods, which focus on individual facial features, have limitations in detecting the manipulated area. To address this limitation, the authors in [10] introduced a Deepfake technique that combines a set of these distinctive features. In [11], researchers explore the consistency of biological signals, examining spatial and temporal dimensions [12-14], and connect various facial landmark [15] points (e.g., eyes, nose, mouth, etc.), 3D head pose [16] as unique features for verifying the authenticity of videos or images generated by Generative Adversarial Networks (GANs). It is important to note that facial expressions are often initially associated with head movements.

In the context of detecting Deepfakes in images, many studies have leveraged DL-based methods to identify specific artifacts generated in the creation process. Zhang et al. [17] introduced a GAN simulator that replicates collective GAN-generated image artifacts and inputs them into a classifier to identify Deepfakes. Meanwhile, Zhou et al. [18] proposed a network designed for extracting standard features from RGB data, a similar but generic approach was observed in [19]. For Deepfake video detection, a deep learning-based approach was initially proposed in [20], employing two inception modules, Meso-4 and MesoInception-4, to construct their proposed network. This technique employs Mean Squared Error (MSE) as the loss function for training.

In supervised scenarios, authors in [21] demonstrate that deep Convolutional Neural Networks (CNNs) outperform shallow CNNs. Some methodologies involve the extraction of handcrafted features [22-23], spatiotemporal features [24], common textures [25], 68 facial landmarks [26-27], alongside visual artifacts (e.g., eye, teeth, lip movement, etc.) from video frames. These features are employed as input to detection networks designed to identify Deepfake manipulations. Additional techniques include data augmentation [28], super-resolution reconstruction [29], and the application of Maximum Mean Discrepancy (MMD) loss [30] to identify more general features. Recent innovations encompass the introduction of attention mechanisms [31] and the promising results achieved through the use of capsule networks (CN) [32-33]. These CNs demonstrate an advantage in terms of parameter efficiency compared to very deep networks. In addition, ensemble learning techniques [34-35] have been applied to enhance the performance of such structures, often achieving over 99% accuracy. Numerous approaches have been proposed to improve Deepfake detection by conducting frame-by-frame analysis of videos, tracking facial movements, and achieving more accurate results. For instance, RNN-based networks have been introduced in [36-37] to extract features at various microscopic and macroscopic levels to identify Deepfakes.

III. Methodology

A. Data Preprocessing and Feature Selection

We conducted our experiment and evaluation using the FF++, DFDC, Celeb-DF, and VDFD datasets. To optimize machine learning models, we applied preprocessing and analysis techniques to 400 videos (200 real faces and 200 fake faces). Each video contributed 200 randomly selected frames to minimize computational costs. The 'dlib' Python package was utilized to track and extract faces, serving as input for classifiers. In machine learning, selecting relevant features is crucial; too many can hinder performance. More data helps, but there's a limit. Feature extraction in image processing involves obtaining useful information like color and shape. Feature selection [37] aids in choosing the most critical information, expediting the training process.

TABLE I. CONTRIBUTION OF FEATURE ENGINEERING TECHNIQUE TO THE CONSTRUCTION OF DFF

Technique	#Elements per (DFF) Feature Set					
	109	**117**	**133**	**228**	**717**	**2296**
HMs	7	7	7	7	7	7
HTF	13	13	13	13	13	13
CH	8	16	32	64	256	512
HOG	81	81	81	144	441	1764

Each image is characterized by a feature set that results from applying a range of feature engineering techniques with different feature combinations. Ultimately, we combined these individually extracted features from each dataset to create a distinctive feature vector known as Deepfake Feature (DFF), which was then utilized as input for machine learning classifiers. Table I provides an overview of the feature sets created by combining features extracted through the following feature engineering techniques or descriptors:

- *Haralick Texture Features (HTF)*: A novel global feature descriptor [38] that focuses on texture's role in defining patterns and colors within objects or images. It helps classify objects by considering attributes such as Rough-Smooth, Hard-Soft, and Fine-Coarse. HTF employs the Histogram of Texture Features, providing a quantitative means to characterize image texture. It utilizes the Gray Level Co-occurrence Matrix (GLCM) to assess pixel adjacency, identifying pairs of adjacent pixel values and recording their frequency. This results in 14 textural features rooted in statistical principles. Typically, a 13-dimensional feature vector is employed to maintain computational efficiency.
- *Histogram of Oriented Gradients (HOG)*: A feature descriptor [39-40] is used to extract image features that emphasize the shape and structure of objects. Unlike edge features that only determine if a pixel is an edge, HOG also provides information about the edge direction. It does this by calculating gradients and orientations in localized image regions, breaking down the complete image into smaller segments. Histograms are then generated for each of these regions based on pixel gradient values and orientations. In our experiment, with a 224 x 224 x 3 image size, HOG produces six output feature vectors with lengths of 109, 117, 133, 228, 717, and 2296 (see Table I).
- *Color Histogram (CH)*: A color histogram [41] is one standard method used to measure image similarity.
- *Hu Moments (HMs)*: Moments [42-43] are key features in image analysis, often used in tasks like face recognition and shape retrieval. While central moments offer translation invariance, we require moments that are invariant to translation, scale, and rotation for shape matching. The HMs consist of 7 numbers derived from central moments, ensuring invariance to various image transformations. The first six moments remain consistent across translation, scale, rotation, and reflection, while the sign of the 7th moment changes when the image is reflected, as indicated in Table I.

B. Statistical Testing for Model Validation

In our examination of multiple ML models, we typically employ a two-step approach. Initially, we conducted an omnibus test to identify variations in classification accuracies; then we proceed to perform pairwise post hoc tests (i.e., *5x2-Fold Cross-Validated Paired T-test* and *Combined 5x2 Cross-Validated F-Test*) to pinpoint specific areas where performance differences exist, all while ensuring appropriate adjustments for multiple comparisons. Omnibus tests, such as Analysis of Variance (ANOVA) [44], serve as statistical tools to determine whether random samples deviate from expected patterns, with ANOVA being a notable example, widely used to assess the significance of suggesting equality in the means of various groups.

C. Experimental Design

In this study, we utilized a range of ML algorithms, including Support Vector Machine (SVM), Random Forest (RF), Extremely Randomized Trees (ERT), Decision Tree (DT), Multilayer Perceptron Network (MLP), Stochastic Gradient Boosting (SGB), and K-Nearest Neighbor (KNN), to train classifiers. We meticulously curated and preprocessed four datasets, namely FF++, DFDC, Celeb-DF, and VDFD, with the aim of unraveling the complexities of Deepfake detection and pinpointing the most effective model-feature set combinations. These datasets provided a diverse range of Deepfake content, enabling a comprehensive evaluation of different machine learning models. To analyze the impact of feature extraction, we employed six distinct feature sets: DFF-109, DFF-117, DFF-133, DFF-228, DFF-717, and DFF-2296, which served as input variables for our models, shedding light on how feature variations influenced overall model performance.

IV. RESULTS AND DISCUSSIONS

A. Results and Key Findings

We conducted nested cross-validation testing to evaluate the performance of classical ML-based classifiers. Our findings, as illustrated in Figure 1 and Table II, particularly when focusing on the FF++ dataset, revealed remarkable insights into Deepfake detection. RF and ERT consistently emerged as the top-performing models, showcasing unexpected accuracy rates exceeding 99%. In a group of their own, RF and ERT proved highly effective in distinguishing between authentic and Deepfake content. SVM, on the other hand, showed comparable performance to RF and ERT, even if exclusively for the DFF-2296 feature set. This performance variation highlighted the effect of feature set size on SVM's accuracy, with a noticeable decline as feature set size was reduced. MLP and KNN, while not achieving the same performance as RF and ERT, however, delivered impressive accuracy rates of 98%. In the meantime, SGB and DT flew around the 95% accuracy, representing their ability to detect such Deepfakes. Shifting our focus to the DFDC dataset, we exposed exciting results. When applying the DFF-2296 feature set, SVM, RF, and ERT consistently reached an exceptional 99% accuracy. However, it was MLP that appeared as the top performer across all feature sets, showcasing its adaptability and reliability in Deepfake detection.

TABLE II: ML Classifiers' performances

Feature	Model	Precision				Recall				F1-Score				Accuracy			
		FF++	DFDC	CBDF	VDFD	FF++	DFDC	CBDF	VDFD	FF++	DFDC	CBDF	VDFD	FF++	DFDC	CBDF	VDFD
DFF-109	SVM	94.6	93.25	92.26	93.63	93.99	88.59	90.33	94.13	94.29	90.86	91.28	93.88	94.22	91.46	91.27	93.87
	RF	99.28	99.08	98.95	98.91	99.04	97.94	98.19	98.65	99.16	98.51	98.57	98.78	99.15	98.58	98.56	98.79
	ERT	99.64	99.08	98.95	99.22	99.46	97.85	98.49	99.23	99.55	98.46	98.72	99.23	99.54	98.53	98.70	99.23
	DT	92.57	93.11	92.18	91.62	92.32	92.59	91.53	91.14	92.44	92.85	91.86	91.38	92.33	93.16	91.80	91.40
	SGB	81.41	91.04	81.54	80.29	78.78	80.68	77.43	84.80	80.08	85.54	79.43	82.48	80.1	86.93	79.72	82.01
	MLP	95.97	95.79	95.35	95.88	97.59	97.00	94.91	96.71	96.77	96.39	95.13	96.3	96.69	96.53	95.08	96.29
	KNN	98.42	96.93	96.39	98.27	98.82	96.47	95.85	98.70	98.62	96.7	96.12	98.49	98.59	96.84	96.09	98.51
DFF-117	SVM	94.35	93.27	91.79	93.28	92.76	87.65	88.82	93.54	93.55	90.37	90.28	93.40	93.51	91.05	90.33	93.42
	RF	99.44	99.11	99.10	99.05	99.27	98.06	98.59	98.82	99.36	98.58	98.85	98.94	99.35	98.65	98.84	98.94
	ERT	99.58	99.17	99.07	99.31	99.50	98.44	98.52	99.30	99.54	98.8	98.8	99.28	99.52	98.86	98.78	99.31
	DT	93.51	94.01	92.95	92.71	93.68	94.15	93.34	91.87	93.6	94.08	93.14	92.29	93.47	94.32	93.05	92.33
	SGB	81.96	91.29	83.92	81.07	78.17	81.74	79.24	85.46	80.02	86.25	81.51	83.21	80.19	87.51	81.42	82.78
	MLP	98.34	97.01	97.01	98.59	96.97	97.29	96.94	96.63	97.65	97.15	96.97	97.6	97.63	97.27	96.94	97.63
	KNN	98.41	96.88	96.46	98.42	98.82	96.68	95.53	98.83	98.61	96.78	96.19	98.62	98.59	96.91	96.17	98.63
DFF-133	SVM	93.02	93.02	91.13	92.33	91.20	84.32	88.63	92.03	92.1	88.46	89.86	92.18	92.06	89.46	89.89	92.20
	RF	99.49	99.08	99.19	99.23	99.32	98.32	98.74	99.03	99.41	98.7	98.96	99.13	99.4	98.76	98.95	99.13
	ERT	99.65	99.35	99.24	99.4	99.53	98.59	98.89	99.38	99.59	98.97	99.06	99.37	99.6	99.01	99.08	99.38
	DT	92.83	94.35	93.35	92.53	92.89	94.29	92.85	91.72	92.86	94.32	93.10	92.12	92.76	94.56	93.04	92.17
	SGB	83.66	91.92	83.99	81.89	80.91	83.32	79.82	85.92	82.27	87.41	81.85	83.85	82.29	88.50	82.10	83.48
	MLP	98.83	97.89	91.75	99.54	98.08	98.41	99.17	95.44	98.45	98.15	95.32	97.45	98.44	98.22	95.07	97.52
	KNN	98.54	97.2	96.52	98.39	98.99	96.82	95.98	98.73	98.77	97.01	96.25	98.56	98.74	97.14	96.22	98.58
DFF-228	SVM	98.41	97.02	96.27	97.40	97.50	93.91	95.56	96.94	97.95	95.44	95.91	97.17	97.93	95.70	95.88	97.18
	RF	99.79	99.35	99.69	99.36	99.53	98.71	99.18	99.33	99.66	99.03	99.43	99.35	99.67	99.07	99.43	99.36
	ERT	99.8	99.44	99.56	99.53	99.75	99.26	99.18	99.62	99.78	99.35	99.37	99.58	99.77	99.38	99.36	99.58
	DT	93.22	94.35	93.79	93.95	93.68	94.38	93.42	92.98	93.45	94.37	93.59	93.46	93.34	94.61	93.53	93.51
	SGB	87.34	93.97	86.44	84.23	87.67	85.68	83.02	88.41	87.51	89.63	84.68	86.27	87.31	90.53	84.81	85.95
	MLP	98.99	99.01	99.23	99.66	99.88	99.29	99.27	99.37	99.43	99.15	99.25	99.51	99.42	99.18	99.24	99.52
	KNN	99.63	98.64	97.85	99.18	99.68	98.12	97.47	99.21	99.66	98.38	97.66	99.19	99.65	98.45	97.64	99.21
DFF-717	SVM	97.86	96.60	95.98	98.66	97.30	93.71	95.79	98.4	97.58	95.13	95.89	98.53	97.55	95.41	95.84	98.54
	RF	99.60	98.99	99.06	99.21	99.26	97.76	98.56	98.55	99.43	98.37	98.81	98.88	99.42	98.45	98.82	98.89
	ERT	99.52	99.07	98.88	99.32	99.32	97.53	98.07	99.27	99.42	98.3	98.48	99.29	99.41	98.38	98.46	99.31
	DT	89.29	94.57	92.74	91.21	88.73	93.79	92.1	90.43	89.01	94.18	92.42	90.82	88.9	94.45	92.36	90.89
	SGB	87.46	93.79	87.16	87.89	86.75	86.65	84.74	90.76	87.10	90.08	85.93	89.3	86.97	90.85	85.97	89.16
	MLP	99.56	99.32	99.34	99.72	99.22	98.97	98.64	99.57	99.39	99.15	98.99	99.66	99.38	99.18	98.98	99.65
	KNN	96.75	96.58	94.62	97.01	97.43	96.21	94.57	97.21	97.09	96.39	94.6	97.11	97.04	96.55	94.54	97.12
DFF-2296	SVM	99.42	99.15	98.15	99.71	99.52	99.38	97.81	99.62	99.47	99.27	97.98	99.67	99.46	99.3	97.96	99.66
	RF	99.28	99.22	98.36	99.16	99.42	99.03	97.47	97.55	99.85	99.11	97.86	98.34	99.84	99.14	97.85	98.36
	ERT	99.15	99.23	97.91	98.91	98.92	98.94	97.29	98.68	99.03	99.09	97.62	98.79	99.02	99.15	97.58	98.79
	DT	83.32	94.63	89.18	86.44	84.26	94.88	89.14	84.82	84.04	94.76	89.16	85.62	83.76	94.97	89.07	85.79
	SGB	87.6	95.8	86.76	91.63	90.36	89.32	86.21	93.12	88.96	92.45	86.48	92.35	88.61	93.01	86.37	92.32
	MLP	99.23	99.44	98.85	99.46	98.6	99.76	98.69	99.63	98.9	99.6	98.77	99.55	98.91	99.62	98.78	99.55
	KNN	95.92	97.83	93.92	95.96	96.23	98.06	93.76	96.29	96.08	97.94	93.84	96.12	96.01	98.03	97.79	96.13

TABLE III: Confidence interval

Dataset	Model	DFF-109	DFF-117	DFF-133	DFF-228	DFF-717	DFF-2296
FF++	SVM	94.22% ± 0.34%	93.51% ± 0.37%	92.06% ± 0.36%	97.93% ± 0.21%	97.55% ± 0.23%	99.46% ± 0.12%
	RF	99.15% ± 0.61%	99.35% ± 0.63%	99.4% ± 0.64%	99.67% ± 0.44%	99.42% ± 0.48%	99.84% ± 0.47%
	ERT	99.54% ± 0.12%	99.52% ± 0.12%	99.6% ± 0.11%	99.77% ± 0.08%	99.41% ± 0.1%	99.02% ± 0.16%
	DT	92.33% ± 0.09%	93.47% ± 0.11%	92.76% ± 0.1%	93.34% ± 0.08%	88.9% ± 0.11%	83.76% ± 0.15%
	SGB	80.1% ± 0.39%	80.19% ± 0.38%	82.29% ± 0.38%	87.3% ± 0.33%	86.97% ± 0.45%	88.61% ± 0.61%
	MLP	96.69% ± 0.31%	97.63% ± 0.24%	98.44% ± 0.18%	99.42% ± 0.09%	99.38% ± 0.11%	98.91% ± 0.15%
	KNN	98.59% ± 0.17%	98.59% ± 0.16%	98.74% ± 0.18%	99.65% ± 0.08%	97.04% ± 0.25%	96.01% ± 0.29%
DFDC	SVM	91.46% ± 0.6%	91.05% ± 0.73%	89.46% ± 0.78%	95.7% ± 0.45%	95.41% ± 0.48%	99.3% ± 0.16%
	RF	98.58% ± 0.79%	98.65% ± 0.78%	98.76% ± 0.76%	99.07% ± 0.73%	98.45% ± 0.64%	99.14% ± 0.53%
	ERT	98.53% ± 0.26%	98.86% ± 0.27%	99.01% ± 0.28%	99.38% ± 0.21%	98.38% ± 0.28%	99.15% ± 0.25%
	DT	93.16% ± 0.27%	94.32% ± 0.25%	94.56% ± 0.24%	94.6% ± 0.17%	94.45% ± 0.28%	94.97% ± 0.21%
	SGB	86.93% ± 0.53%	87.51% ± 0.54%	88.5% ± 0.5%	90.5% ± 0.51%	90.85% ± 0.51%	93.01% ± 0.49%
	MLP	96.53% ± 0.39%	97.27% ± 0.32%	98.22% ± 0.3%	99.18% ± 0.2%	99.18% ± 0.21%	99.62% ± 0.21%
	KNN	96.84% ± 0.45%	96.91% ± 0.39%	97.14% ± 0.39%	98.45% ± 0.29%	96.55% ± 0.4%	98.03% ± 0.29%
Celeb-DF	SVM	91.27% ± 0.42%	90.33% ± 0.49%	89.89% ± 0.43%	95.88% ± 0.3%	95.84% ± 0.31%	97.96% ± 0.22%
	RF	98.56% ± 0.67%	98.84% ± 0.51%	98.95% ± 0.62%	99.43% ± 0.63%	98.82% ± 0.46%	97.85% ± 0.52%
	ERT	98.7% ± 0.19%	98.78% ± 0.15%	99.08% ± 0.15%	99.36% ± 0.12%	98.46% ± 0.17%	97.58% ± 0.24%
	DT	91.8% ± 0.17%	93.05% ± 0.17%	93.04% ± 0.17%	93.53% ± 0.12%	92.36% ± 0.17%	89.07% ± 0.23%
	SGB	79.72% ± 0.4%	81.42% ± 0.46%	82.1% ± 0.41%	84.81% ± 0.37%	85.97% ± 0.36%	86.37% ± 0.46%
	MLP	95.08% ± 0.31%	96.94% ± 0.32%	95.07% ± 0.25%	99.24% ± 0.21%	98.98% ± 0.12%	98.78% ± 0.15%
	KNN	96.09% ± 0.28%	96.17% ± 0.26%	96.22% ± 0.29%	97.64% ± 0.23%	94.54% ± 0.37%	97.79% ± 0.32%
VDFD	SVM	93.87% ± 0.35%	93.42% ± 0.36%	92.2% ± 0.42%	97.18% ± 0.25%	98.54% ± 0.19%	99.66% ± 0.09%
	RF	98.79% ± 0.52%	98.94% ± 0.54%	99.13% ± 0.59%	99.36% ± 0.49%	98.89% ± 0.44%	98.36% ± 0.38%
	ERT	99.23% ± 0.17%	99.31% ± 0.14%	99.38% ± 0.14%	99.58% ± 0.11%	99.31% ± 0.16%	98.79% ± 0.18%
	DT	91.4% ± 0.14%	92.33% ± 0.13%	92.17% ± 0.11%	93.51% ± 0.09%	90.89% ± 0.14%	85.79% ± 0.17%
	SGB	82.01% ± 0.47%	82.78% ± 0.44%	83.48% ± 0.39%	85.95% ± 0.38%	89.16% ± 0.4%	92.32% ± 0.49%
	MLP	96.29% ± 0.27%	97.63% ± 0.27%	97.52% ± 0.24%	99.52% ± 0.11%	99.65% ± 0.09%	99.55% ± 0.13%
	KNN	98.51% ± 0.18%	98.63% ± 0.17%	98.58% ± 0.17%	99.2% ± 0.17%	97.12% ± 0.27%	96.13% ± 0.31%

To assess model performance and reliability, we delved into the confidence intervals (depicted in Figure 2) to highlight the inherent uncertainties within the metrics found in Table III. Notably, datasets like Celeb-DF and VDFD pose more substantial challenges, often leading to diminished model performance. This underscores the crucial need to weigh accuracy alongside associated confidence levels when making optimal model selections. Also, our investigations uncovered a consistent pattern: the performance of SVM decreased when utilizing smaller feature sets. This trend persisted when examining the Celeb-DF dataset, underlining the consistent relationship between SVM's performance and the dimensions of the feature set. In the VDFD dataset-based experiments, MLP exhibited consistent excellence by surpassing RF and ERT, achieving an amazing accuracy rate exceeding 99%. Also, our study observed a noteworthy improvement in KNN's performance for specific feature sets, such as DFF-117 and DFF-228, when utilizing the VDFD dataset.

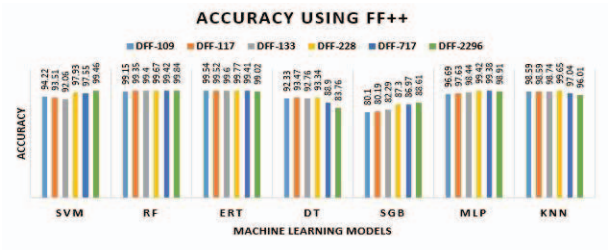
B. General Efficiency of Classifiers

In this comprehensive analysis, we delve into the efficacy of diverse Deepfake detection strategies, drawing insights from experimental results across **precision, recall, F1-Score**, and **accuracy** metrics shown in Table IV. The minimum precision of 80.29% indicates that even less optimal strategies maintained a relatively high precision, while the maximum precision of 99.80% underscores the capability of certain strategies to achieve near-perfect precision. With an average precision of 95.47% and a standard deviation of 4.83%, there is a notable consistency in correctly identifying true positives among predicted positives across the evaluated strategies. Similarly, the recall metrics reveal a robust performance, with a minimum recall of 77.43%, a maximum of 99.88%, an average of 94.75%, and a standard deviation of 5.50%. This indicates a strong ability to identify genuine content among actual positives.

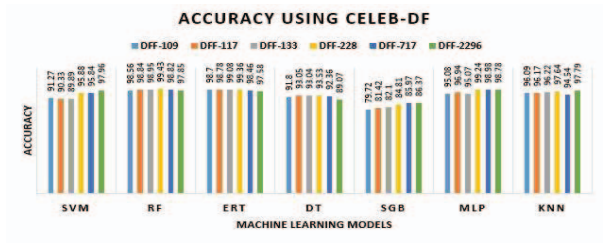
TABLE IV. STATISTICAL SUMMARY OF DETECTION METHODS' PERFORMANCES

Metrics	MIN	MAX	AVG	MED	STD
Precision	80.29	99.80	95.47	97.30	4.83
Recall	77.43	99.88	94.75	97.28	5.50
F1-Score	79.43	99.85	95.10	97.16	5.07
Accuracy	79.72	99.84	95.17	97.40	5.04

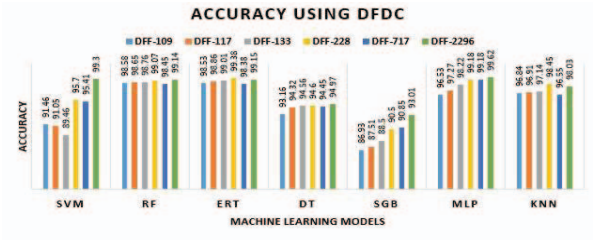
The F1-Score metrics showcase a balanced performance, with a minimum of 79.43%, a maximum of 99.85%, an average of 95.10%, and a standard deviation of 5.07%. The balanced assessment of model performance is further emphasized by the median F1-Score of 97.16%. Lastly, the accuracy metrics demonstrate high correctness levels, with a minimum accuracy of 79.72%, a maximum of 99.84%, an average of 95.17%, and a standard deviation of 5.04%. The median accuracy of 97.40% affirms consistent high-level correct classifications. Overall, these findings collectively underscore the general efficiency of current Deepfake detection strategies in effectively distinguishing between authentic and manipulated content, providing valuable insights for ongoing advancements in this critical domain.



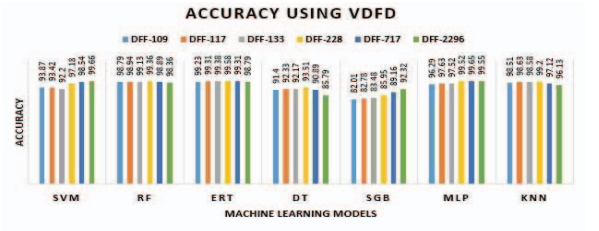
(a)



(b)



(c)



(d)

Fig. 1. Performance evaluation of machine learning classifiers with different feature sets on these four different datasets: (a) FF++, b) Celeb-DF, (c) DFDC, and (d) VDFD.

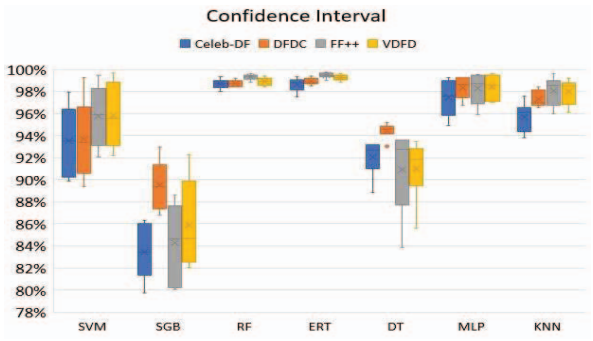


Fig. 2. Confidence interval base on datasets.

C. Dataset-Specific Insights

A more in-depth examination of these datasets reveals the complex relationship between machine learning models, feature sets, and the distinctive characteristics of each dataset.

- In the **FF++ dataset**, two standout performers are RF and ERT. These models consistently achieve notable accuracy, often increasing above the desired 99% of accuracy. This high level of accuracy is a piece of evidence of their robustness in classifying between authentic and Deepfake content. MLP deserves special notice for its amazing accuracy and outstanding performance in recall rates. This characteristic makes MLP an essential component in scenarios where minimizing false negatives is crucial. While SVM demonstrates good overall performance, it exhibits some sensitivity to feature set variations, leading to variations in accuracy.
- The **DFDC dataset** showcases a trio of high-performing models: RF, ERT, and MLP. These models maintain a consistent track record of excellent performance. They exhibit remarkable accuracy, precision, recall, and F1 scores across various feature sets, underscoring their reliability in Deepfake detection. SVM, although competitive, falls slightly behind in terms of overall accuracy. Remarkably, the choice of feature set plays a significant role in determining the model's success, indicating the importance of feature engineering and selection.
- In the **Celeb-DF dataset**, RF and ERT appear as leaders in terms of accuracy. These models consistently achieve high accuracy, highlighting their competence in classifying between genuine and Deepfake content within this dataset. SVM, although not reaching the same level of performance as RF and ERT, still produces competitive results. However, a notable observation arises: SVM's performance is significantly impacted by the size of the feature set. Smaller feature sets tend to decrease SVM's accuracy, indicating the need for meticulous feature engineering when tackling this dataset.
- Within the **VDFD dataset**, the feature set DFF-228 consistently stands out as a reliable choice. This feature set consistently produces excellent results for all models. SVM and RF, in particular, achieve notable accuracy levels when paired with this feature set, demonstrating their ability to detect Deepfake content. Additionally, MLP exhibits exceptional recall rates, making it a standout choice when sensitivity to false negatives is a key consideration.

While RF and ERT often take the lead, MLP shows strengths in recall, and SVM's performance varies with the dimensions of the feature set. These insights provide valuable guidance for selecting the most suitable model and feature set combination based on the specific requirements and nuances of each dataset in the domain of Deepfake detection.

D. Discussion and Implications

The consequences of these findings are significant, as they provide key insights for the development of Deepfake detection systems and highlight the need for adaptable approaches customized for the specific attributes of datasets and feature sets:

- RF and ERT are recommended as primary models for Deepfake detection due to their consistently high accuracy.
- SVM can be competitive but requires thorough feature engineering and is sensitive to feature set size.
- SGB offers a balanced performance profile, making it a valuable alternative.
- MLP should be considered when recall is of the highest importance or when working with larger feature sets.
- Feature engineering and selection are paramount. Experiments with diverse feature sets are crucial to identify the optimal configuration for a given dataset.
- The AUC values validate the models' ability to detect Deepfake content, strengthening their efficiency.

V. Conclusion

While DL-based approaches have been widely used for Deepfake detection, our study demonstrates that classical ML methods are capable of outperforming or matching state-of-the-art DL techniques in Deepfake detection. By applying established procedures of feature engineering, selection, and classifier training, it not only offers improved understandability and interpretability but also reduces computational resource requirements. Our experiments obtained remarkable results, with accuracy of 99.84% on the FF++ dataset, 99.38% on the DFDC dataset, 99.66% on the VDFD dataset, and 99.43% on the Celeb-DF dataset. Importantly, statistical analysis revealed the consistency of these high accuracy figures across diverse datasets. The study thus challenges the prevailing notion that DL approaches are the only effective means of detecting Deepfakes and provides assurance that judicious applications of ML techniques can deliver highly accurate, cost-effective, and more sustainable solutions for Deepfake detection.

Our ongoing and future research endeavors focus on investigating feature development and selection for the most challenging Deepfake detection problems, as well as detecting deepfaked audio and other types of multimedia produced by generative AI techniques.

References

[1] J. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," IEEE International Conference on Computer Vision, 2017, pp. 2242–2251.

[2] S. Suwajanakorn, S. M. Seitz, and I. K. Shlizerman, "Synthesizing Obama: learning lip sync from audio," ACM Transactions on Graphics (TOG), 36(4), 2017.

[3] G. Patrini, F. Cavalli, and H. Ajder, "The state of Deepfakes: reality under attack," Annual Report v.2.3. 2018.

[4] M. S. Rana, B. Murali and A. H. Sung, "Deepfake Detection Using Machine Learning Algorithms," 10th International Congress on Advanced Applied Informatics (IIAI-AAI), Niigata, Japan, 2021, pp. 458–463.

[5] M. S. Rana, M. N. Nobi, B. Murali and A. H. Sung, "Deepfake Detection: A Systematic Literature Review," in IEEE Access, vol. 10, 2022, pp. 25494–25513.

[6] A. Rossler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Nießner, "FaceForensics++: Learning to detect manipulated facial images," in International Conference on Computer Vision (ICCV), 2019.

[7] B. Dolhansky, et al., "The Deepfake Detection Challenge (DFDC) Preview Dataset," arXiv preprint arXiv:1910.08854, 2019.

[8] Y. Li, X. Yang, P. Sun, H. Qi and S. Lyu, "Celeb-DF: A Large-Scale Challenging Dataset for DeepFake Forensics," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), USA, 2020, pp. 3204-3213.

[9] M. S. Rana, "Analyzing and Detecting Android Malware and Deepfake", Dissertations, The University of Southern Mississippi, 2021. https://aquila.usm.edu/dissertations/1948

[10] F. Matern, C. Riess, and M. Stamminger, "Exploiting Visual Artifacts to Expose Deepfakes and Face Manipulations," IEEE Winter Applications of Computer Vision Workshops (WACVW), HI, USA, 2019, pp. 83–92.

[11] U. A. Ciftci, I. Demir and L. Yin, "FakeCatcher: Detection of Synthetic Portrait Videos using Biological Signals," in IEEE Transactions on Pattern Analysis and Machine Intelligence.

[12] X. Li, Y. Lang, Y. Chen, X. Mao, Y. He, S. Wang, H. Xue, and Q. Lu, "Sharp Multiple Instance Learning for DeepFake Video Detection," arXiv:2008.04585, 2020.

[13] L. Guarnera, O. Giudice, and S. Battiato, "Fighting Deepfake by Exposing the Convolutional Traces on Images," arXiv:2008.04095, 2020.

[14] M. Bonomi, C. Pasquini, and G. Boato, "Dynamic texture analysis for detecting fake faces in video sequences," arXiv:2007.15271, 2020.

[15] G. Wang, J. Zhou, and Y. Wu, "Exposing Deep-faked Videos by Anomalous Co-motion Pattern Detection," arXiv:2008.04095, 2020.

[16] X. Yang, Y. Li, and S. Lyu, "Exposing Deep Fakes Using Inconsistent Head Poses," in IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), United Kingdom, 2019, pp. 8261–8265.

[17] X. Zhang, S. Karaman, and S. F. Chang, "Detecting and simulating artifacts in GAN fake images," in IEEE Workshop on Information Forensics and Security (WIFS), 2019.

[18] P. Zhou, X. Han, V. I. Morariu and L. S. Davis, "Learning Rich Features for Image Manipulation Detection," IEEE/CVF Conference on Computer Vision and Pattern Recognition, UT, 2018, pp. 1053–1061.

[19] K. Chugh, P. Gupta, A. Dhall, and R. Subramanian, "Not made for each other- Audio-Visual Dissonance-based Deepfake Detection and Localization," arXiv:2005.14405, 2020.

[20] D. Afchar, V. Nozick, J. Yamagishi, and I. Echizen, "Mesonet: a compact facial video forgery detection network," in IEEE International Workshop on Information Forensics and Security, 2018, pp. 1–7.

[21] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1800–1807.

[22] Y. Li, M.-C. Chang, and S. Lyu, "In Ictu Oculi: Exposing AI created fake videos by detecting eye," in IEEE Workshop on Information Forensics and Security, 2018.

[23] Y. Li and S. Lyu, "Exposing Deepfake videos by detecting face warping artifacts," in IEEE CVPR Workshops, 2019.

[24] I. Ganiyusufoglu, L. M. Ngô, N. Savov, S. Karaoglu, and T. Gevers, "Spatio-temporal Features for Generalized Detection of Deepfake Videos," arXiv preprint arXiv: arXiv:2010.11844, 2020.

[25] X. Wang, T. Yao, S. Ding, and L. Ma, "Face Manipulation Detection via Auxiliary Supervision," In: Yang, H., Pasupa, K., Leung, A.CS., Kwok, J.T., Chan, J.H., King, I. (eds) Neural Information Processing. ICONIP 2020. Lecture Notes in Computer Science(), vol 12532, pp. 313–324.

[26] M. T. Jafar, M. Ababneh, M. Al-Zoube, and A. Elhassan, "Forensics and Analysis of Deepfake Videos," 11th International Conference on Information and Communication Systems, Jordan, 2020, pp. 053–058.

[27] X. Dong, J. Bao, D. Chen, W. Zhang, N. Yu, D. Chen, F. Wen, and B. Guo, "Identity-Driven DeepFake Detection," arXiv preprint arXiv:2012.03930, 2020.

[28] L. Bondi, E. D. Cannas, P. Bestagini, and S. Tubaro, "Training Strategies and Data Augmentations in CNN-based DeepFake Video Detection," arXiv preprint arXiv:2011.07792, 2020.

[29] Z. Hongmeng, Z. Zhiqiang, S. Lei, M. Xiuqing, and W. Yuehan, "A Detection Method for DeepFake Hard Compressed Videos based on Super-resolution Reconstruction Using CNN," Proceedings of the 4th High-Performance Computing and Cluster Technologies Conference & 3rd International Conference on Big Data and Artificial Intelligence, Association for Computing Machinery, New York, USA, pp. 98–103.

[30] J. Han, and T. Gevers, "MMD Based Discriminative Learning for Face Forgery Detection," 15th Asian Conference on Computer Vision, Kyoto, Japan, 2020, pp. 121–136.

[31] H. Dang, et al., "On the Detection of Digital Face Manipulation," IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, WA, USA, 2020, pp. 5780–5789.

[32] H. H. Nguyen, J. Yamagishi, and I. Echizen, "Capsule-forensics: Using Capsule Networks to Detect Forged Images and Videos," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Brighton, United Kingdom, 2019, pp. 2307–2311.

[33] H. Nguyen, J. Yamagishi, and I. Echizen, "Use of a Capsule Network to Detect Fake Images and Videos," arXiv preprint arXiv:1910.12467, 2019.

[34] N. Bonettini, E. D. Cannas, S. Mandelli, L. Bondi, P. Bestagini, and S. Tubaro, "Video Face Manipulation Detection Through Ensemble of CNNs," Computing Research Repository (CoRR), abs/2004.07676, 2020.

[35] M. S. Rana, and A. H. Sung, "DeepfakeStack: A Deep Ensemble-based Learning Technique for Deepfake Detection," 7th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud), New York, USA, 2020, pp. 70–75.

[36] D. Guera, and E. Delp, "Deepfake video detection using recurrent neural networks," in IEEE International Conference on Advanced Video and Signal Based Surveillance, 2018.

[37] M. N. Murti and V. S. Devi, "Feature Extraction and Feature Selection, Introduction to Pattern Recognition and Machine Learning," IISc Lecture Notes Series, June 2015, pp. 75-110

[38] R. M. Haralick, "Statistical and structural approaches to texture," in Proceedings of the IEEE, vol. 67, no. 5, pp. 786-804, May 1979.

[39] L. Weng, "Object Detection for Dummies Part 1: Gradient Vector, HOG, and SS," https://bit.ly/3oRFtxJ, last accessed: 2023/10/2.

[40] R. Ahmed, "A Take on HOG Feature Descriptor," https://bit.ly/2LutMyU, last accessed: 2023/10/2.

[41] F. Alamdar and M. R. Keyvanpour, "A New Color Feature Extraction Method Based on QuadHistogram," Procedia Environmental Sciences, Volume 10, 2011, pp. 777-783.

[42] J. Zunic, K. Hirota and P. L. Rosin, "A Hu moment invariant as a shape circularity measure, Pattern Recognition," Volume 43, Issue 1, 2010, pp. 47-57.

[43] Z. Huang and J. Leng, "Analysis of Hu's moment invariants on image scaling and rotation," 2nd International Conference on Computer Engineering and Technology, Chengdu, 2010.

[44] S. Raschka, "Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning," arXiv: 1811.12808, 2020.