In the Name of God

Journal of
Information Systems & Telecommunication
Vol. 14, No.2, April-June 2026, Serial Number 54

Research Institute for Information and Communication Technology
Iranian Association of Information and Communication Technology
Affiliated with: Academic Center for Education, Culture and Research (ACECR)

Print ISSN: 2322-1437
Online ISSN: 2345-2773
Publication License: 91/13216
Editorial Office Address: No.5, Saeedi Alley, Kalej Intersection., Enghelab Ave., Tehran, Iran,
P.O.Box: 13145-799 Tel: (+9821) 88930150 Fax: (+9821) 88930157
E-mail: info@jist.ir , infojist@gmail.com
URL: jist.acecr.org

Indexed by:
- SCOPUS www. Scopus.com
- Islamic World Science Citation Center (ISC) www.isc.gov.ir
- Directory of open Access Journals (DOAJ) www.Doaj.org
- Scientific Information Database (SID) www.sid.ir
- Regional Information Center for Science and Technology (RICeST) www.ricest.ac.ir
- Magiran www.magiran.com

Publisher:
Iranian Academic Center for Education, Culture and Research **(ACECR)**

This Journal is published under scientific support of
Advanced Information Systems (AIS) Research Group and
Telecommunication Research Group, ICTRC

Acknowledgement

JIST Editorial-Board would like to gratefully appreciate the following distinguished referees for spending their valuable time and expertise in reviewing the manuscripts and their constructive suggestions, which had a great impact on the enhancement of this issue of the JIST Journal.

(A-Z)

- Alaeiyan, Mohammad Hadi, K.N. Toosi University of Technology, Tehran, Iran
- Asghari, Seyed Amir, Kharazmi University, Tehran, Iran
- Azarkasb, Seyed Omid, K.N. Toosi University of Technology, Tehran, Iran
- Alizadeh, Mojtaba, Lorestan University, Khorramabad, Iran
- Al-Qurabat, Ali Kadhum, Al-Mustaqbal University, Iraq
- Behmanesh, Ali, Iran University of Medical Sciences, Tehran, Iran.
- Ebadati, Omid Mahdi, Kharazmi University,Tehran, Iran
- Ebady Manaa, Mahdi, University of Babylon, Hillah, Iraq
- Faili, Heshaam, Tehran University, Tehran, Iran
- Farsi, Hassan, University of Birjand, South Khorasan, Iran
- Kashef, Seyed Sadra, Urmia University, Urmia, Iran
- Kheirkhah, Fatemeh, ACECR, Tehran, Iran
- Kuchaki Rafsanjani, Marjan, Shahid Bahonar University, Kerman, Iran
- Kolahkaj, Maral, Islamic Azad University, Karaj Branch, Iran
- Moradi, Gholamreza, Amirkabir University,Tehran, Iran
- Mohammadi, Mohsen, University in Esfarayen, North Khorasan, Iran
- Razavi, Seyyed Mohammad, University of Birjand, South Khorasan Province, Iran
- Soleimanian Gharehchopogh, Farhad, Islamic Azad University Urmia, Iran
- Tourani, Mahdi, University of Birjand, South Khorasan, Iran
- Vahabie, Abdol-Hossein, Tehran University, Tehran, Iran
- Zahedi, Mohammad Hadi, K. N. Toosi University of Technology, Tehran, Iran

Table of Contents

Journal of Information Systems and Telecommunication
Vol.14, No.1, January-March 2026, 79-98
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

ABC-GAN: An Attention-Based Conditional GAN with DTW Loss for ECG Signal Generation to Address Class Imbalance

Parmida Behain[1*] , Noushin Riahi[1]

[1].Department of Computer Engineering, Alzahra University, Tehran, Iran

Received: 13 Sep 2025/ Revised: 04 Jun 2026/ Accepted: 19 Jul 2026

Abstract

The imbalance between normal and pathological cases in Electrocardiogram (ECG) datasets significantly degrades the performance of automated deep learning-based diagnostic systems, particularly for minority classes. This paper introduces ABC-GAN, a novel Attention-Based Conditional Generative Adversarial Network designed to mitigate this critical data imbalance by generating high-fidelity, class-specific synthetic ECG signals. Our proposed model incorporates two key innovations: an attention mechanism within the generator to focus on critical morphological features of the ECG waveform, and the integration of Dynamic Time Warping (DTW) as a distance metric in the generator's loss function to better preserve essential temporal dynamics. Trained and thoroughly evaluated on the MIT-BIH Arrhythmia dataset across seven different heartbeat classes, ABC-GAN successfully generates highly realistic signals that closely match the original data's distribution. When used to augment the training set, these synthetic signals significantly enhance classifier performance and generalization. A downstream classifier achieved an accuracy of 95%, representing a substantial improvement over the baseline trained on the raw imbalanced data and clearly outperforming both standard oversampling techniques and a vanilla Conditional GAN (cGAN). The overall findings demonstrate that ABC-GAN is a powerful and effective tool for data augmentation, fully capable of improving diagnostic accuracy in cardiac research and practical clinical applications.

Keywords: ECG Signal Synthesis; Generative Adversarial Network (GAN); Attention Mechanism; Conditional GAN; Dynamic Time Warping (DTW); Class Imbalance; Data Augmentation.

1- Introduction

The electrocardiogram (ECG) is a fundamental diagnostic tool that records the heart's electrical activity, enabling the detection of cardiac abnormalities, functional disorders, and arrhythmias. The accurate and automated analysis of ECG signals is therefore critical for proactive cardiac care. However, the performance of deep learning models for this task is heavily reliant on large, balanced datasets. In practice, ECG datasets are notoriously imbalanced, as normal heartbeats vastly outnumber pathological ones, and some arrhythmias are exceptionally rare. This imbalance causes classifiers to be biased toward the majority class, leading to poor generalization and reduced sensitivity for detecting critical cardiac events [1][2].

This challenge of analyzing complex physiological signals is not unique to ECG data. The broader field of medical informatics has seen a significant rise in the application of deep learning to various types of biomedical data. Recent research has explored areas such as the remote monitoring of cardiac conditions [3], deep learning-based analysis of cardiac MRI [4], and stress detection from physiological signals [5]. These advancements motivate further exploration of sophisticated deep learning models, particularly generative models, to address inherent data limitations such as class imbalance in biomedical signal analysis.

Generative Adversarial Networks (GANs) have emerged as a promising solution to this problem, capable of synthesizing realistic synthetic data to augment minority classes [6]. While previous studies have successfully applied GANs for ECG generation [7]–[9], they often rely on standard loss functions like Mean Squared Error (MSE)

✉ **Parmida Behain**
behain.parmida@gmail.com

or adversarial loss alone. These losses can lead to overly smoothed outputs that fail to capture the precise morphological nuances, such as the shape and duration of the P-wave, QRS complex, and T-wave, that are essential for accurate diagnosis. Furthermore, while conditional GANs (cGANs) can generate class-specific signals [9], ensuring the morphological fidelity of these generated signals, especially for rare classes, remains a significant challenge.

In this paper, we propose the Attention-Based Conditional GAN (ABC-GAN), a novel framework designed to address these limitations. Our work makes the following key contributions:

1. We introduce an attention mechanism into the generator of a cGAN, enabling the model to dynamically focus on the most diagnostically relevant segments of the ECG waveform during synthesis.
2. We innovatively incorporate the Dynamic Time Warping (DTW) distance metric into the generator's loss function. DTW is particularly suited for ECG signals as it accounts for temporal misalignments and variations in phase, ensuring the generated signals are morphologically faithful to real heartbeats.
3. We demonstrate that signals generated by ABC-GAN effectively balance the MIT-BIH Arrhythmia dataset. Training a classifier on this augmented dataset leads to a significant improvement in accuracy, particularly for minority classes, outperforming classifiers trained on data augmented by other standard methods.

Unlike conventional ECG signal generation methods that rely solely on convolutional architectures, the proposed ABC-GAN integrates an attention mechanism within a conditional GAN framework to enhance the modeling of long-range temporal dependencies in ECG signals. Existing approaches such as CardioGAN and AC-WGAN-GP primarily focus on adversarial learning and class conditioning but lack explicit mechanisms for selectively emphasizing diagnostically relevant waveform regions.

The proposed method further incorporates Dynamic Time Warping (DTW) loss into the generator objective. While conventional GAN losses focus on distribution matching, DTW provides temporal alignment between generated and real ECG signals, preserving clinically significant morphological characteristics. This is particularly important for arrhythmia classes exhibiting temporal variability.

The main contributions of this study are:

1. Development of an Attention-Based Conditional GAN (ABC-GAN) for ECG signal generation.
2. Integration of an attention module to capture long-range temporal dependencies and emphasize diagnostically relevant waveform segments.
3. Incorporation of DTW loss to improve temporal alignment and waveform fidelity.
4. Generation of synthetic minority-class ECG samples to alleviate class imbalance.
5. Comprehensive evaluation against conventional GAN-based ECG augmentation methods on the MIT-BIH Arrhythmia Database.

The remainder of this paper is organized as follows. Section 2 reviews related work on ECG synthesis. Section 3 provides the necessary background on cGANs and attention mechanisms. Section 4 details the architecture of our proposed ABC-GAN and the experimental setup. Section 5 presents and discusses the results. Finally, Sections 6 and 7 conclude the paper and outline future research directions.

2- Related Works

The use of generative models, particularly Generative Adversarial Networks (GANs) and Variational Autoencoders (VAEs), for synthesizing ECG signals has been extensively explored to address data scarcity and imbalance. Previous research can be categorized based on architectural innovations and application goals.

A significant body of work has focused on adapting established GAN architectures for ECG data. Edmond Adib et al. [2] utilized WGAN-GP and AC-WGAN-GP for class-conditioned heartbeat generation, employing a post-generation filtering threshold. Similarly, Fei Zhu et al. [6] and Yu-He Zhang et al. [7] leveraged recurrent architectures (BiLSTM) in the generator to capture temporal dependencies in single and 12-lead ECG signals, respectively. Others, like Naren Wulana et al. [8], have explored architectures like WaveNet and SpectroGAN for generating specific heartbeat types. These approaches demonstrated the feasibility of GANs for ECG synthesis but often relied on standard loss functions that can blur critical features.

To enhance utility, several studies have integrated generation with downstream classification tasks. Pu Wang et al. [9] and Wenqiang Li et al. [10] proposed frameworks combining ACGAN-based data augmentation with CNN classifiers for anomaly and myocardial infarction detection, showing improved performance by leveraging synthetic data.

Addressing mode collapse and improving diversity has been another research focus. Haixu Yang et al. [11] proposed ProEGAN-MS to generate multi-scale ECG signals with high diversity. Subhrajyoti Dasgupta et al. [12] introduced CardioGAN, which incorporated an attention-based generator and a gradient penalty loss, marking a notable step towards focusing on salient waveform features. This is the closest to our use of attention; however, their work did not explore a specialized loss function like DTW to further enforce morphological accuracy.

Rohan Banerjee et al. [13] presented a GAN-based model for generating ECG signals of Atrial Fibrillation (AF), a type of cardiac arrhythmia. Their GAN architecture includes a combination of GAN pairs to simulate Heart Rate Variability (HRV) patterns and unique signal morphology. Results show significant improvement in AF classification performance with the addition of generated data to both datasets.

Recent advances have incorporated state-of-the-art architectures. Li et al. [14] and E. Brophy et al. [15] employed Transformer-based generators for multivariate time-series synthesis, showcasing their ability to model long-range dependencies. Shota Harada et al. [16] proposed a GAN-based method for time series data generation, with LSTM-based generator and discriminator networks. They demonstrated improved classification performance with the addition of generated data. Jintai Chen et al. [17] presented the ME-GAN model for generating ECG signals, incorporating Mixup normalization to inject disease information appropriately. Their evaluation using rFID metrics shows the model's effectiveness in generating Multi-view ECG signals.

Other efforts have tackled the challenge of generating variable-length signals [18] or using VAEs for feature-space generation [19].

A significant branch of research focuses on improving GAN training dynamics for ECG data. Munia et al. [21] leveraged the Wasserstein GAN (WGAN) to stabilize training on one-dimensional signals. Building on this, Nankani et al. [22] introduced the DCCGAN, which incorporated a 'Twist' parameter in the discriminator to mitigate vanishing gradients and accelerate convergence, specifically for generating minority class samples. Conversely, Golany et al. [20] addressed the issue of label scarcity directly with their Personalized GAN (PGAN), which generates patient-specific signals without relying on labeled arrhythmia data, enhancing performance for individualized models.

Hatamian et al. [23] demonstrated that for ventricular fibrillation classification, GAN-based augmentation (using DCGAN) significantly outperformed conventional oversampling techniques, a finding reinforced by qualitative assessments of morphological similarity.

Further innovations involve the structural decomposition of the generation task. Agarwal et al. [24] proposed a two-module framework (CardiacGen), where separate WGAN-GP models dedicated to generating heart rate variability and morphological features are combined to produce highly realistic signals. In one of the most complex approaches, Adib et al. [25] abandoned the 1D domain entirely, mapping ECG signals to 2D representations (using GASF/GADF/MTF) for processing with a Denoising Diffusion Probabilistic Model (DDPM), before mapping back to 1D; notably, their results indicated that a well-tuned WGAN-GP model could still outperform this sophisticated DDPM-based approach on certain metrics.

Recent advances in ECG synthesis have shifted from traditional GAN-based architectures toward transformer, diffusion, and hybrid self-supervised frameworks that better capture long-range temporal dependencies and improve physiological realism.
Class imbalance remains a critical issue in ECG analysis, particularly for rare arrhythmia classes. To address this, CECG-GAN introduces a conditional generative adversarial framework that augments minority ECG classes using class-conditioned synthesis. By improving the distribution of underrepresented rhythms, the model enhances downstream diagnostic robustness and classification sensitivity for rare cardiac conditions [26].

Incorporating task-specific constraints into generative modeling has also gained attention. The Task-Aware Conditional GAN introduces a multi-objective optimization strategy combining adversarial loss with temporal alignment constraints such as Dynamic Time Warping (DTW), reconstruction loss, and statistical distance measures. This enables the generator to preserve both global distributional characteristics and fine-grained temporal structure, which is particularly important for ECG signals where waveform alignment (e.g., R-peak consistency) is clinically significant [27].

More recently, diffusion-based generative models have demonstrated superior stability and sample quality compared to GAN-based approaches. BioDiffusion extends diffusion modeling to biomedical time-series synthesis, providing improved coverage of complex signal distributions and mitigating mode collapse issues commonly observed in GANs. Its probabilistic denoising

formulation allows more faithful reconstruction of physiological variability in ECG signals [28].

Personalized ECG synthesis has also emerged as a key research direction. The Personalized 12-lead ECG generation framework conditions the generative process on patient-specific attributes, enabling individualized cardiac waveform synthesis. This improves clinical relevance by preserving subject-specific morphology while maintaining consistency across leads [29]. Hybrid self-supervised generative approaches further improve representation learning in low-label scenarios. ECG synthesis models combining self-supervised pretraining with adversarial generation improve feature extraction from limited ECG datasets, leading to more stable training and higher-quality pathological waveform generation [30].

Transformer-based diffusion models such as TransDiffECG integrate attention mechanisms with diffusion processes to enable semantically controllable ECG synthesis. This allows explicit conditioning on cardiac attributes while maintaining high fidelity in waveform generation, demonstrating strong performance in both generation quality and downstream diagnostic tasks [31]. Finally, RDiffGAN-ECG combines diffusion models, adversarial training, and state-space sequence modeling to enhance long-range dependency modeling in ECG signals. By explicitly conditioning on R-peak information and leveraging structured temporal modeling, it produces high-fidelity ECG waveforms with improved morphological consistency and heart-rate realism [32].

2-1- Limitations of Existing Methods

1. **Morphological Smoothing from MSE-based Losses.** Standard GANs and cGANs for ECG generation [6][9] predominantly rely on Mean Squared Error (MSE) or adversarial loss alone. MSE produces overly smooth outputs by averaging multiple plausible reconstructions, leading to loss of sharp morphological features such as precise QRS complex width and ST-segment elevation patterns that are clinically significant [2][12]. ABC-GAN overcomes this by replacing MSE with Dynamic Time Warping (DTW) loss, which preserves sharp transitions by matching temporal patterns rather than pointwise differences.

2. **Temporal Misalignment Insensitivity.**
   Conventional loss functions treat each time point independently, failing to penalize small temporal shifts that preserve overall shape but misalign critical landmarks (P-wave onset, R-peak, T-wave offset). This is particularly problematic for ECG signals where beat-to-beat variability is natural [13][41]. DTW loss explicitly optimizes for optimal temporal alignment, making the generator invariant to benign temporal variations while penalizing morphological distortions.

3. **Insufficient Focus on Diagnostically Relevant Regions.**
   While CardioGAN [12] introduced attention for ECG generation, their attention mechanism operates without temporal alignment guidance, potentially focusing on irrelevant segments. ABC-GAN combines self-attention with DTW loss, where the attention weights are indirectly optimized to focus on regions that minimize temporal alignment cost, directing attention to diagnostically critical waveform components.

4. **Mode Collapse in Minority Classes.**
   Standard GANs often fail to capture the full distribution of rare arrhythmia classes [2]. Our conditional architecture with class-specific attention maps and DTW loss for each class separately ensures that minority class distributions are modeled with higher fidelity.

Most models optimize using losses like MSE or adversarial loss, which are insufficient for capturing complex, non-linear temporal alignments and precise morphological patterns in ECG waveforms. The use of Dynamic Time Warping (DTW), a powerful similarity measure for time series that is invariant to non-linear warps in the time dimension, as a loss function for training ECG GANs remains largely unexplored.

Our work, ABC-GAN, bridges this gap. While building upon the foundation of conditional generation [2][6] and inspired by the use of attention [12], we uniquely integrate a DTW-based loss into the adversarial training regimen. This combination ensures that the generated signals are not only realistic and class-specific but also morphologically precise, addressing the core challenge of fidelity in ECG synthesis for imbalanced learning.

3- Background

This section outlines the fundamental concepts underlying our proposed model: Conditional Generative Adversarial Networks and the Attention Mechanism.

3-1- Conditional GAN

The Conditional Generative Adversarial Network (CGAN), introduced by Mirza and Osindero [33], is an extension of

the original GAN framework [34]. While a standard GAN learns to generate data from a random noise vector z, a CGAN conditions the generation process on additional information y, such as a class label. This allows for the targeted generation of data from specific classes.

The generator G and discriminator D play a two-player minmax game with the following objective function:

$$\min_G \max_D V(D,G) = E_{x\sim p_{data}(x)}[\log D(x|y)] + E_{z\sim p_z(z)}\left[\log\left(1 - D\left(G(z|y)\right)\right)\right] \quad (1)$$

where x is real data, z is the noise vector, and y is the conditional label. The discriminator learns to distinguish between real pairs (x,y) and fake pairs ($G(z|y)$,y), while the generator learns to fool the discriminator by producing outputs that are realistic for the given label y.

In the context of ECG synthesis, CGANs enable the generation of heartbeat signals for a specific arrhythmia class (PVC, APC, LBBB). However, a known limitation of standard CGANs is that they often produce samples that lack fine-grained morphological details, as the standard adversarial and reconstruction losses (MSE) may not sufficiently penalize small temporal misalignments or shape inaccuracies crucial for ECG diagnosis.

Fig. 1 illustrates the process of conditioning the GAN model, which involves providing class labels (C) as auxiliary information along with input noise (Z) to the generator to produce ECG signals.

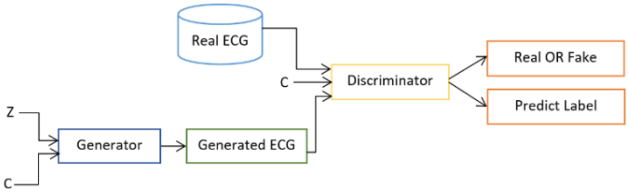


Fig. 1 Conditional GAN

3-2- Attention Mechanism

The attention mechanism, a cornerstone of the Transformer architecture [35] allows a model to dynamically focus on different parts of the input sequence when producing an output. It has since been successfully integrated into GANs (SAGAN [36])to capture long-range dependencies in images.

The core component is the Scaled Dot-Product Attention. It operates on learnable linear projections of the input: Query (Q), Key (K), and Value (V). The output is a weighted sum of the values, where the weight assigned to each value is determined by the compatibility of the query with the corresponding key:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \quad (2)$$

where dk is the dimensionality of the keys. The scaling factor $\frac{1}{\sqrt{d_k}}$ prevents the softmax function from having extremely small gradients.

For ECG generation, self-attention enables the model to learn intra-signal dependencies, focusing on critical waveform regions (P-wave onset, R-wave peak, ST segment) during generation. We hypothesize that this focused generation is crucial for morphological fidelity and works synergistically with DTW loss.

Fig. 2 visualizes the Waveform Attention Mechanism employed in the model architecture.

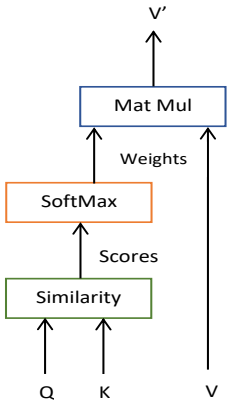


Fig. 2 Illustration of Waveform Attention Mechanism

In our work, we integrate a self-attention block [36] into the generator of the CGAN. For ECG generation, this mechanism allows the model to learn intra-signal dependencies, effectively focusing its "attention" on critical regions of the waveform (the onset of the P-wave, the peak of the R-wave, or the duration of the ST segment) while generating each point of the output signal. We hypothesize that this focused generation is crucial for achieving high morphological fidelity and works synergistically with our DTW-based loss function, which directly optimizes for temporal alignment.

4- Materials and Methods

The proposed method consists of four main steps: preprocessing, GAN training, post-processing, and data classification. Initially, real ECG signals from the dataset are preprocessed to prepare them for GAN training.

Following this, the GAN is trained on the preprocessed data to generate artificial ECG signals. Post-processing is then applied to the generated signals to refine them. Finally, the generated signals are validated to ensure high accuracy and similarity to the original signals, measured using appropriate quantitative metrics (DTW, MSE).
Fig. 3 illustrates the block diagram of the proposed method and its four main steps.

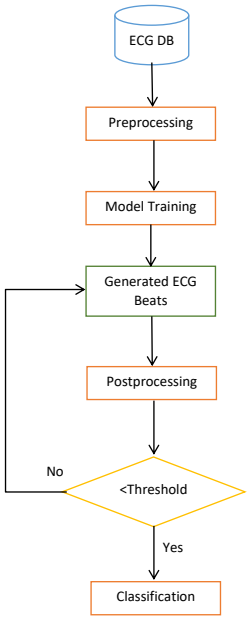


Fig. 3 Block diagram of the proposed method

4-1- Preprocessing

Preprocessing is a crucial step to prepare the dataset for GAN training. From the provided 15-class dataset, a subset of 7 classes is selected. Using the R-peak locations specified in the dataset, intervals to the preceding and succeeding R-peaks are determined. Segments corresponding to 75% of these intervals before and after each R-peak are extracted to define the boundaries of individual heartbeats. Additionally, the sampling frequency is reduced from 360 Hz to 256 Hz to decrease computational load and training time [2].

4-2- Train Attention-based Conditional GAN Model

The Wasserstein Generative Adversarial Network (WGAN) is an extension of the traditional GAN that replaces the discriminator with a critic to estimate the Wasserstein distance between real and generated samples. This modification stabilizes training and makes it less sensitive to model architecture and hyperparameter choices [37].
By conditioning the WGAN on class labels, the generator learns class-specific features of ECG signals, which are then used to generate realistic synthetic signals. The architecture of the proposed model and its loss function are described in the following sections.

1) Model Architecture

The proposed method aims to enhance the conditional WGAN model to generate ECG signals with higher fidelity, thereby improving classification accuracy. To achieve this, three main strategies are adopted:

1. **Attention Mechanism:** Incorporated into the generator to focus on critical temporal and morphological features of ECG signals, enabling the capture of class-specific patterns.
2. **Generator Loss Function Modification:** Different loss functions, including MSE, DTW, and Kullback-Leibler divergence, were evaluated. DTW was ultimately selected for its superior performance in generating signals that improve classification accuracy.
3. **WGAN Architecture:** The discriminator is replaced with a critic, which stabilizes training by estimating the Wasserstein distance between real and generated signals.

Below, the proposed architectures for the generator, critic, and the proposed loss function in the generator are provided.
The architecture of the generator and critic utilizes the blocks specified in Table 1.

Table 1: Blocks in the generator and critic

Layer	*Generator*	*Critic*
1	ConvTranspose1d[1]	ConvTranspose1d[2]
2	BatchNorm1d	InstanceNorm1d
3	ReLU	LeakyReLU

The input noise depicted in Table 2 has a length of 100 with dimensions of 100×1. Additionally, class labels are connected to it through a mapping to dimensions of 100×1. Therefore, the generator receives input of dimensions 200×1. An attention mechanism is incorporated into the architecture to enhance the generator's learning for better ECG signal generation. This mechanism assists the generator in learning more precise dependencies using attention mechanisms, allowing it to better capture the patterns and characteristics specific to each class of ECG signals.
In WGANs, the generator network utilizes CNN layers to transform input noise vectors into synthetic ECG waveforms that closely resemble real ECG signals. These CNN layers allow the generator to capture temporal and

[1] kernel size = 4, stride = 2, padding = 1
[2] kernel size = 4, stride = 2, padding = 1

spatial features inherent in ECG signals, producing realistic synthetic waveforms. The generator's ability to learn from the underlying distribution of ECG signals is crucial for generating accurate and clinically relevant synthetic data.
The architecture of the generator is shown in Table 2:

Table 2: Architecture of the generator

Layer	***Generator***
Input	16 × 200 × 1
1	block[1]
2	block
3	block
4	block
5	ConvTranspose1d[2]
6	FC
7	Tanh
8	Attention layer
Output	16 × 1 × 256

The critic network in WGANs, which replaces the discriminator in traditional GANs, also employs CNNs. The critic evaluates both generated and real ECG signals to estimate the Wasserstein distance between their distributions. By leveraging CNNs, the critic can effectively learn temporal and spatial patterns present in ECG signals, providing more precise assessments of the synthetic signal quality. In the critic, class labels are mapped to dimensions of 1×256. Subsequently, they are connected to either the generated or real heartbeat signal with dimensions of 1×256. The input to the critic is of dimensions 2×256.
The architecture of the critic is shown in Table 3:

Table 3: Architecture of critic

Layer	***Critic***
Input	16 × 2 × 256
1	Conv1D, Leaky ReLU[3]
2	block[4]
3	block
4	block
5	Conv1D[5]
6	FC
7	FC
Output	16× 1 × 1

[1] The architecture of the blocks is presented in **Error! Reference source not found.**

[2] kernel size=4, stride=2, padding=1

[3] kernel size=4, stride=2, padding=1

[4] The architecture of the blocks is presented in **Error! Reference source not found.**

[5] kernel size=4, stride=2, padding=0

2) Proposed Loss Function

The critic's loss function comprises both the Wasserstein loss and the gradient penalty, defined as follows [2]:

$$L_{critic} = -E_{(x,y)\sim p_{data}}[\,critic(x,\,y)\,] + E_{(\tilde{x},y)\sim p_{fake}}[critic(\tilde{x},y)] + \lambda_{GP}\,gp \quad (3)$$

This loss function in Eq.(3) is used to update the critic's parameters. Both the generator and critic loss functions incorporate conditional information and class labels. The term mean(critic(real, labels)) refers to the average output of the critic when evaluating real data. On the other hand, mean(critic(fake, labels)) is the average output of the critic when given fake (generated) data. The difference between these two terms is central to the critic's goal of distinguishing between real and fake samples. The constant λ_{GP} represents the strength of the gradient penalty (GP), which is a regularization term denoted by gp that enforces smooth gradients to stabilize the training process. Together, these terms help the critic model maintain a balance between accurately identifying real and fake data while controlling the gradients for stable training.

The Wasserstein loss function for the critic typically relies on the Wasserstein distance, also known as Earth Mover's Distance (EMD). It measures the dissimilarity between two probability distributions and guides the generator toward producing samples closer to the distribution of real data [37].

$$L_{\text{wasserstein}} = -(\,E[\text{critic(real,labels)}] - E[\text{critic(fake,labels)}]\,) \quad (4)$$

In Eq.(4), critic(real, labels) represents the scores assigned by the critic to real samples, and critic(fake, labels) represents the scores assigned to generated samples. The goal is to maximize the difference between the average scores of real and generated samples, encouraging the critic to assign higher scores to real samples and lower scores to generated samples, guiding the generator toward producing more realistic samples.

The gradient penalty component is a crucial part of the WGAN with Gradient Penalty (WGAN-GP) algorithm. This function helps enforce Lipschitz continuity in the discriminator, which is essential for stable training. Lipschitz continuity is a mathematical property often applied to functions, particularly in the context of WGANs, to ensure training stability. The Lipschitz constraint restricts the rate of change of a function [25].

In the case of WGANs, this constraint is applied to the critic function. Lipschitz constraint ensures that the output of the function changes smoothly, preventing abrupt changes. Applying a Lipschitz constraint to the critic helps stabilize the training process by preventing the critic from assigning very high gradients to the generator, thus avoiding erratic behavior during training. This constraint also aids in reducing mode collapse by softening the critic's response, guiding the generator to cover a broader range of data distribution, and reducing the risk of collapsing into a single mode.

The gradient penalty term is given by:

$$\text{gp} = \lambda\,(|\nabla_{\hat{x}}\text{critic}(\hat{x},\text{labels})|_2 - 1) \tag{5}$$

In Eq.(5), $\hat{x}$ denotes an *interpolated sample*, which is a convex combination of a real data sample x and a generated sample $\tilde{x}$. It is typically computed as $\hat{x} = \alpha\, x + (1-\alpha)\,\tilde{x}$, where $\alpha \in [0,1]$. The coefficient λ, usually set to 1/2, controls the strength of the gradient penalty. $\nabla_{\hat{x}}$ represents the gradient of the critic's output with respect to the interpolated sample, indicating how the critic's score changes with small variations in the input. Finally, $\|\cdot\|_2$ denotes the L_2 norm, which measures the magnitude of this gradient vector.

The proposed generator's loss function is defined as the negative average score assigned by the critic to the generated samples, augmented with the Dynamic Time Warping (DTW) loss to capture the characteristics and waveforms of the generated signals:

$$L_{\text{gen}} = -E[\text{critic}(\text{fake},\text{labels})] + \lambda_{\text{dtw}} \cdot \text{dtw\_loss} \tag{6}$$

Eq.(6), λdtw is the DTW coefficient. This loss function is used to update the generator's parameters.
Minimizing this function aligns with the generator's objective of receiving high scores from the critic, indicating a higher similarity to the distribution of real data. Minimizing this negative average is equivalent to maximizing the positive average, aligning to generate realistic samples. The ultimate aim is to minimize the generator's loss throughout training.
The DTW function measures the dynamic time-warping distance between real and generated samples, quantifying the temporal alignment and dissimilarity between two sequences:

$$\text{dtw\_loss} = \sum_{i=1}^{\text{cur\_batch\_size}} \text{DTW}(\text{real}[i],\text{fake}[i]) \tag{7}$$

This loss, as shown in Eq.(7) is computed for each pair of real and generated samples within the batch.

4-3- Postprocessing

In this step, generated ECG signals that are highly dissimilar to real signals are filtered out, retaining only morphologically similar signals. The similarity is assessed using the Dynamic Time Warping (DTW) distance between each generated signal and the nearest confirmed real signal from the corresponding class. Signals with a DTW distance exceeding a predefined threshold are removed, ensuring that the retained signals closely resemble the real ECG morphology.

4-4- Classification

The filtered generated ECG signals are then incorporated into the training dataset for classification. A one-dimensional implementation of ResNet34, adapted for ECG signal analysis, is employed for classification [38]. ResNet34 consists of 34 layers with residual blocks, each containing two 3×3 convolutional layers with skip connections [39]. The model is trained on a combined dataset of real and generated signals to evaluate the effectiveness of the generated signals in improving classification accuracy.

5- Results and Discussion

In this section, we first describe the dataset used and then present two approaches to evaluate the generated ECG signals by comparing them with real signals. The first approach involves determining cluster centers for each class, and the second involves examining average signals and standard deviations.

5-1- Dataset

The dataset used for this research is the MIT-BIH Arrhythmia Database [40]. This database, collected by the Massachusetts Institute of Technology (MIT), comprises a set of electrocardiogram (ECG) records widely used for research in arrhythmia detection and analysis. It stands as one of the most recognized and authoritative databases in the field of biomedical signal processing and cardiovascular research.
The dataset includes 48 half-hour two-channel ECG records sampled at a frequency of 360 Hz. These records encompass both normal sinus rhythm and various arrhythmias. The MIT-BIH dataset consists of a total of 15 classes, with only 7 classes selected for this research: P (premature atrial contraction), A (atrial premature contraction), L (left bundle branch block beat), N (normal

beat), R (right bundle branch block beat), f (fusion of paced and normal beat), j (junctional beat).

5-2- Determining Cluster Centers for Each Class

For each class, 50 real signals were selected as representative samples. The cluster center was defined as the signal whose maximum DTW distance to all other signals in the class was minimal. This ensures that the selected signal is the most representative of its class.
Fig. 4 shows the cluster centers and the corresponding 50 sample templates:

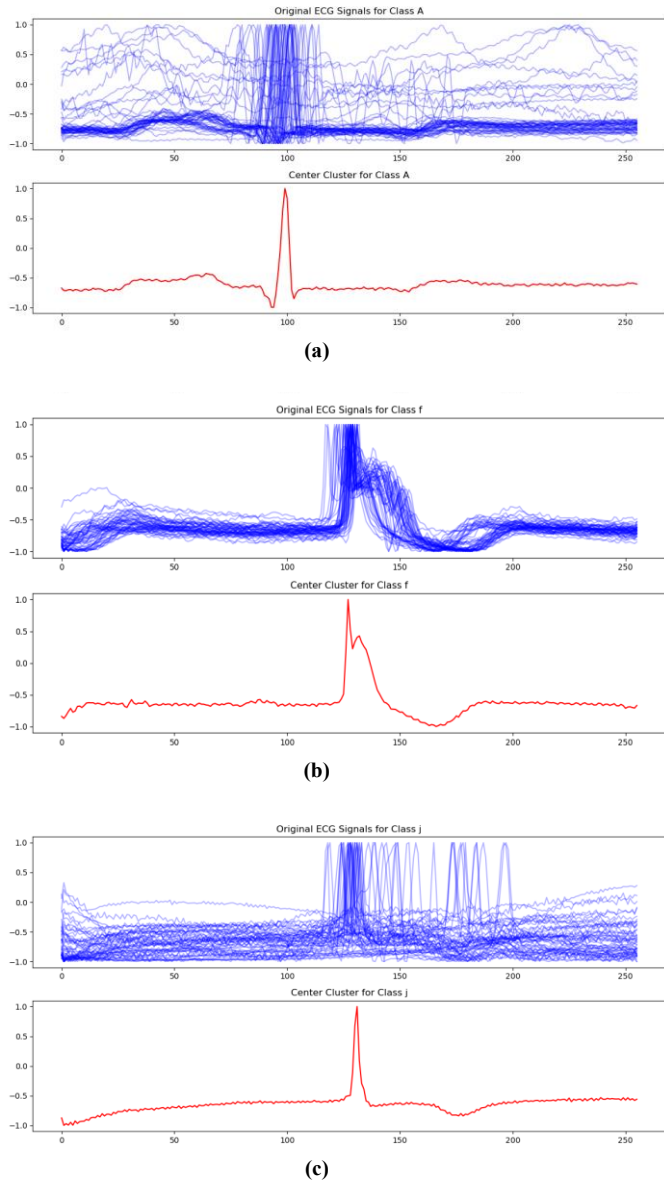
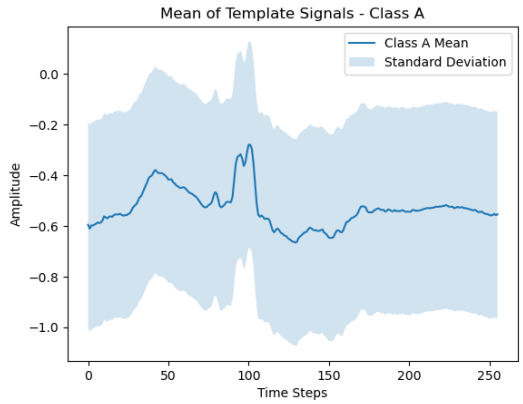


Fig. 4 Cluster centers of each class: (a) Class A (b) Class F (c) Class J (d) Class L (e) Class N (f) Class P (g) Class R. The red-colored signals represent the cluster center obtained for each class. The blue-colored signals are the same 50 real samples selected from the dataset.

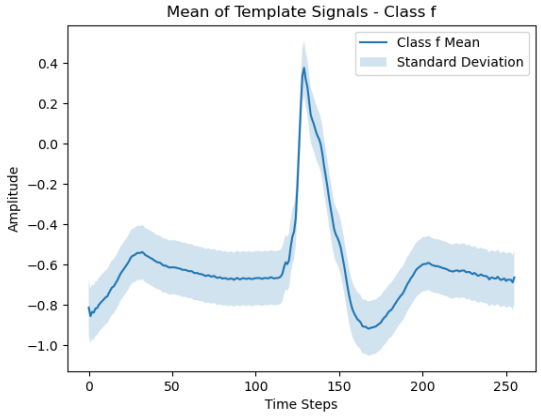
Notably, classes A and j exhibit higher intra-class variability compared to other classes. These cluster centers serve as references for evaluating the similarity of generated signals to real ECG signals.

5-3- Determining Mean Signals and Standard Deviation

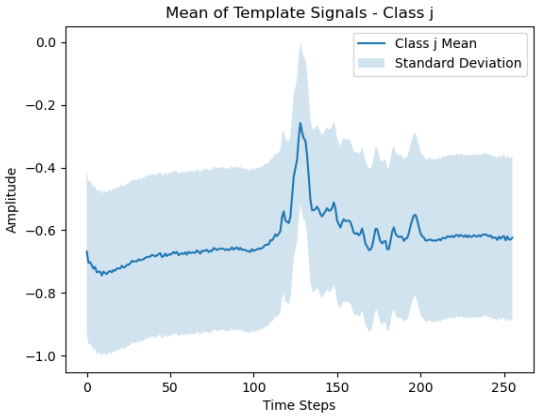
To further characterize the statistical properties of the signals, the mean and standard deviation of the 50 samples in each class were computed (Fig. 5).



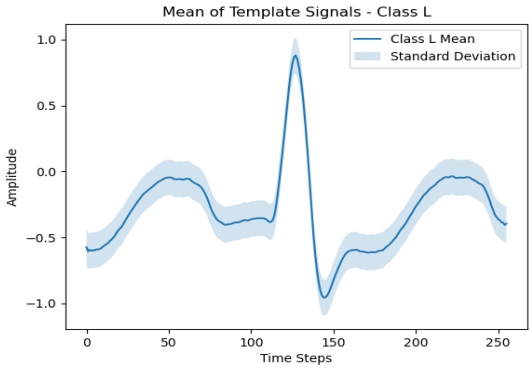
(a)



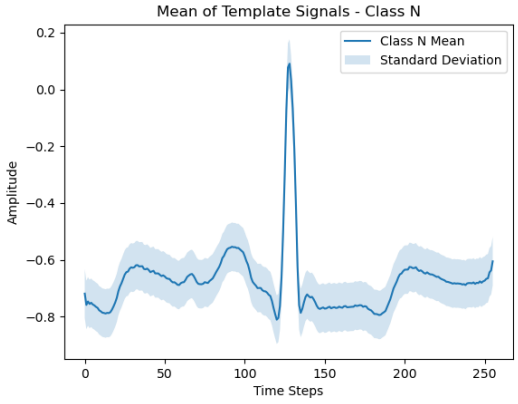
(b)



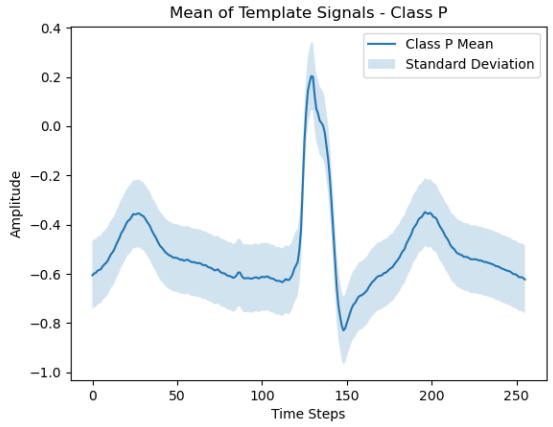
(c)



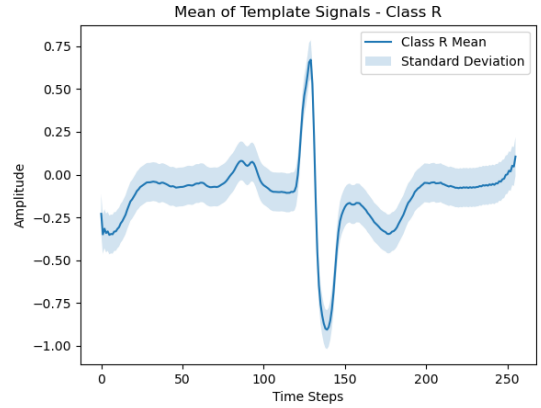
(d)



(e)



(f)



(g)

Fig. 5 Mean signals for each class: (a) Class A (b) Class f (c) Class j (d) Class L (e) Class N (f) Class P (g) Class R. It can also be observed that in classes A and j, a well-defined average signal has not been obtained so the average signal cannot be considered as a template for each class, and there is also a higher standard deviation in these two classes compared to other classes.

While the average signal provides a summary of the class characteristics, it may be insufficient for assessing the similarity of generated signals, particularly in classes with high variability, such as A and j.

5-4- Eliminating outlier signals

As the mean signal may not provide a reliable reference for classes with high variability, we employed Dynamic Time Warping (DTW) to measure the similarity between generated signals and the 50 template signals of each class. For each generated signal, the nearest template based on the minimum DTW distance was identified.

The nearest real signal serves as a robust criterion for evaluating the generated signals and determining outliers. To remove outlier signals, the DTW distance between each generated signal and its closest template was computed. Signals with a DTW distance exceeding a threshold of 3 were considered outliers and excluded from further analysis. This threshold ensures that only generated signals with sufficient morphological similarity to real signals are retained for subsequent classification.

5-5- Evaluating Metrics

We employ two categories of metrics: (1) classification performance metrics to assess downstream task improvement, and (2) similarity metrics to quantify how closely generated signals resemble real ECG signals.

For classification, we use standard metrics including Precision, Recall, Accuracy, and F1-score as defined in standard textbooks [48]. These are computed from the confusion matrix of the ResNet34 classifier on held-out test data.

For similarity assessment, we employ three complementary metrics:

1) DTW (Dynamic Time Warping):

A classic approach for measuring the similarity (or dissimilarity) of time series curves is by considering the Dynamic Time Warping (DTW) similarity measure. It calculates the optimal alignment of two given time series with the minimum cost [12]. DTW aligns time series along the time axis to optimally align them and computes the distance between them [41].

$$D_{i,j} = f(x_i, y_j) + \min\{ D_{i-1,j},\ D_{i,j-1},\ D_{i-1,j-1}\} \tag{8}$$

It's particularly useful for comparing time series data where the sequences might have different lengths or where there's some temporal distortion between them. Eq.(8) is the recursive formula used to calculate the cost of aligning two elements, (xi,yj), from two sequences, X and Y, respectively. Di,j represents the cumulative cost of aligning the ith element of sequence X with the jth element of sequence Y. $f(x,y)$ represents the local cost (or distance) between elements xi and yj. The idea is to find the optimal alignment path between the two sequences by minimizing the cumulative cost, considering all possible alignments at each step[12][41].

2) Mean Squared Error (MSE):

The MSE as shown in Eq.(9) measures the average squared difference between corresponding elements of

two signals. For ECG signals X and Y of length N, the MSE is calculated as:

$$MSE(X,Y) = \frac{1}{N}\sum_{i=1}^{N}(x_i - y_i)^2 \tag{9}$$

Where xi is the ith sample of the generated ECG signal. yi is the ith sample of the real ECG signal. N is the length of the signals (number of samples).
The MSE provides a measure of the average squared difference between corresponding samples of the two signals.

3) Kullback-Leibler (KL) Divergence:
The KL divergence in Eq.(10) measures the discrepancy between two probability distributions. For discrete probability distributions P and Q, the KL divergence from Q to P is computed as:

$$D_{KL}(P|Q) = \sum_{i=1}^{N} p_i \log \frac{p_i}{q_i} \tag{10}$$

Where pi and qi are the probabilities of the ith bin in the histograms corresponding to the generated and real ECG signals, respectively. N is the number of bins in the histograms.

These metrics collectively provide a comprehensive evaluation of both classification performance and signal fidelity, ensuring generated ECG signals are accurate and morphologically consistent with real data.

5-6- Training model

To train the model, initially, we need to create two sets: a training set and a testing set, which will be used for both training the generative model and the classification model. 90% of the data is allocated to the training set, and the remaining 10% is assigned to the testing set. Table 4 illustrates the distribution of the number of samples in each dataset.

Table 4: Train and test sets

Class	*Total*	*Training set*	*Test set*
P	7028	6325	702
A	25046	2291	254
L	8075	7267	807
N	75052	67546	7505
R	70259	6533	752
f	982	883	98
j	229	206	22

Only 50% of the samples in each class were used in training because we want the classifier to experience weak training on a highly imbalanced dataset. Due to the insufficient number of samples in small classes, the model is intentionally weakened. Therefore, only half of the training samples, as shown in Table 4 were utilized [2].

The proposed generative model has been trained for 30 epochs. In each epoch, the generator takes a randomly sampled noise from a Gaussian distribution along with class labels as input. The random noise acts as a source of variability, while class labels guide the generator to produce ECG signals belonging to specific classes. Thus, using the defined architecture for the generator, it transforms input noise and labels into artificial ECG signals.

The main task of the critic is to distinguish between real ECG signals from the datasets and artificial signals generated by the generator. The number of critic iterations is a hyperparameter that determines how many updates the critic undergoes for each update of the generator during the training process. The goal of multiple critic iterations is to ensure that the critic is effective in accurately distinguishing between real and fake samples. In this iterative loop, the critic is trained to maximize the Wasserstein distance between the distributions of real and fake samples. By repeating the training of the critic several times before updating the generator, the training process stabilizes, assisting in achieving the Wasserstein GAN objective. Then, the generator is updated once for each batch to minimize the generator loss. This alternating training process continues throughout the epochs.
Therefore, the generator and critic are trained iteratively. The training process involves finding a balance where the generator is neither too weak to be easily detected by the critic nor too strong.

The Loss plot for the generator and critic over 30 epochs is presented in Fig. 6.

Fig. 6 Loss Plot for Generator and Critic

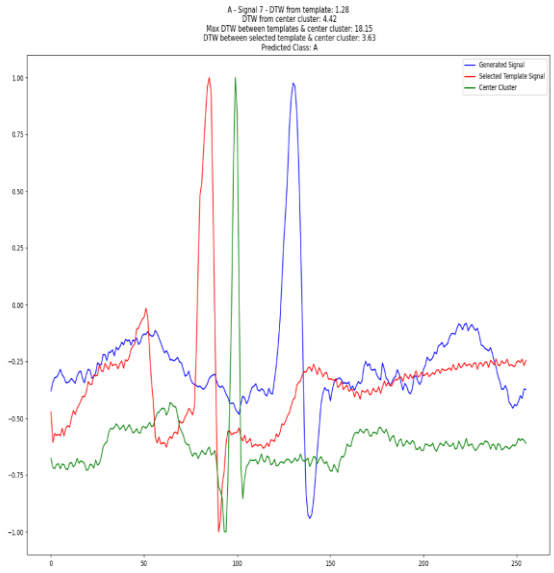
As indicated in Fig. 6, both the generator and critic have reached the desired stability during the training process, showing no significant fluctuations throughout learning. It can be concluded that the model has converged, and

learning has taken place. It should be noted that the generator initially requires some time to adjust its parameters to achieve both the standard GAN objective simultaneously (deceiving the critic) and satisfying the DTW loss. The initial increase in generator loss indicates that the generator is learning to align its generated sequences with the desired patterns and gradually reaches a stable and consistent state.

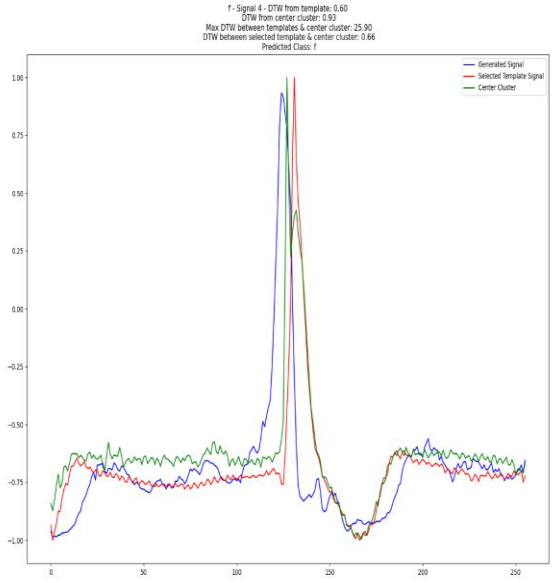
A one-dimensional implementation of ResNet34 is first trained using only real ECG signals. Subsequently, generated signals are added to the training set, and the model is retrained. Performance is finally evaluated on the test set, which contains only real ECG signals, to measure the contribution of the generated data to classification accuracy.

5-7- Evaluate Generated ECG Signals

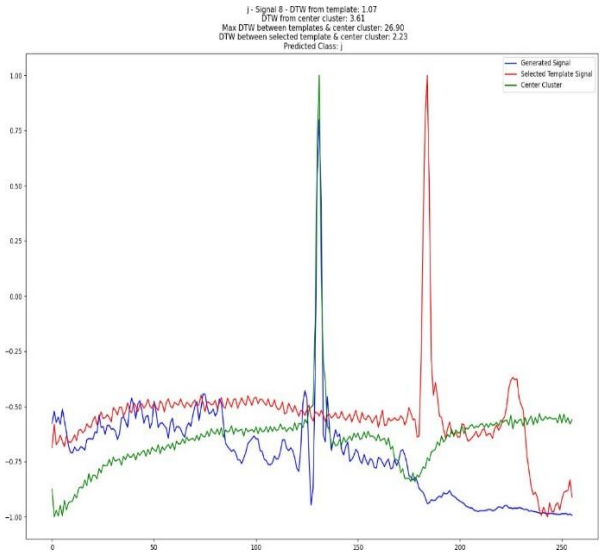
After training the generative model, it was used to synthesize ECG signals for each class. A total of 10,000 samples were generated per class, without applying any threshold at this stage. This initial set of generated signals serves as the basis for further evaluation, including comparison with real signals and outlier removal. Representative generated signals for each class are presented in Fig. 7.



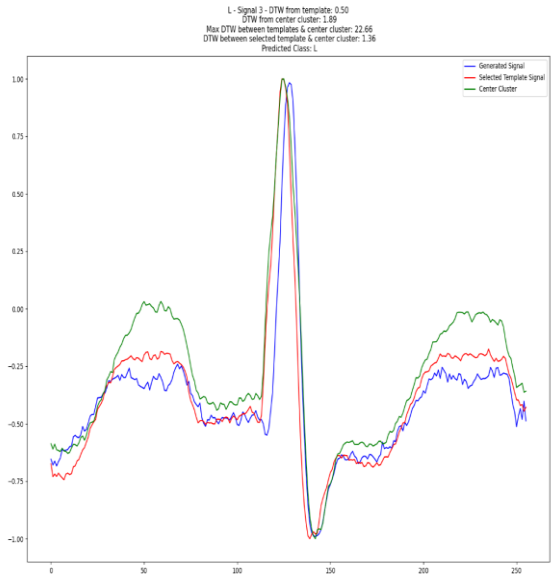
(a)



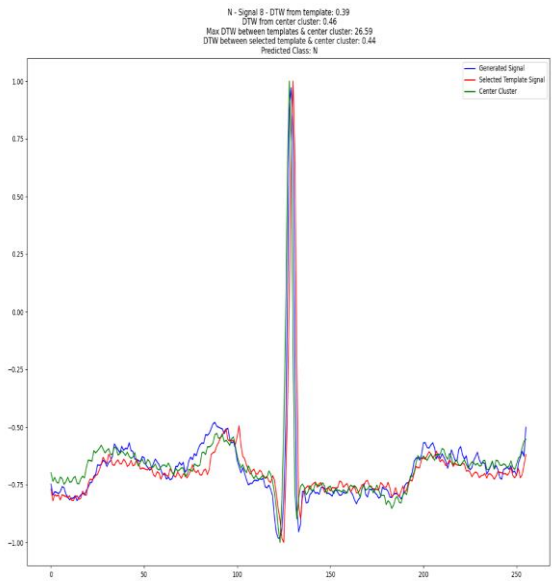
(b)



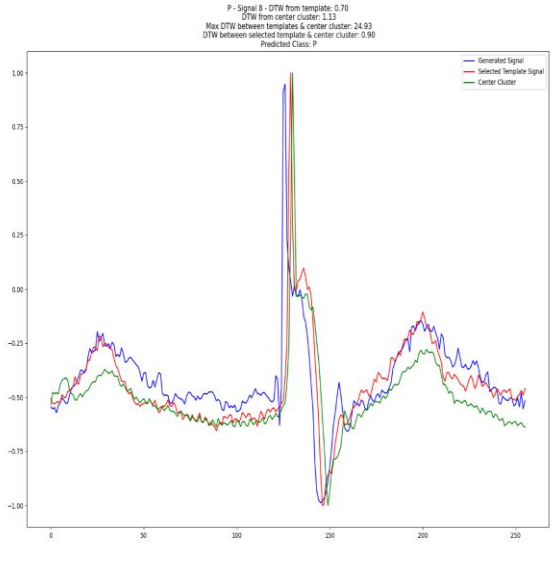
(c)



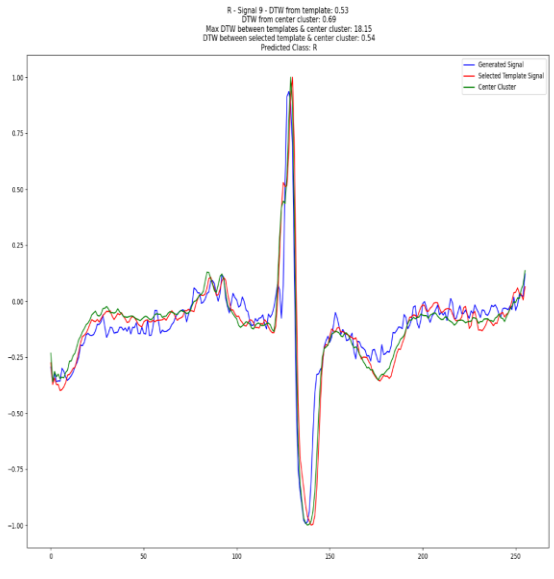
(d)



(e)



(f)

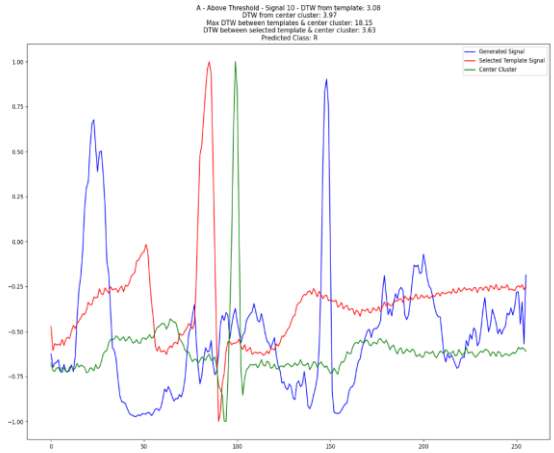


(g)

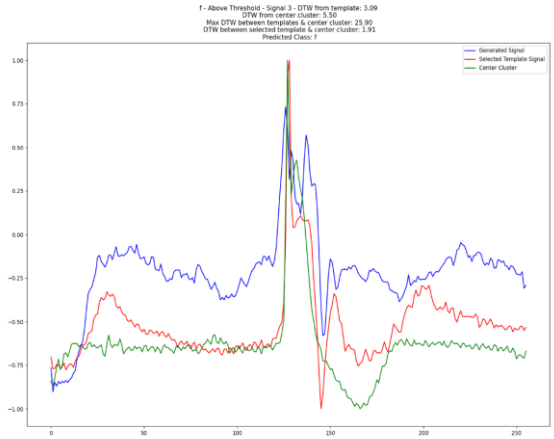
Fig. 7 Generated ECG signals for each class: (a) Class A (b) Class f (c) Class j (d) Class L (e) Class N (f) Class P (g) Class R. The generated signals are indicated in blue, and the nearest real and template sample to each generated signal, based on DTW distance, is shown in red color. Additionally, for a better understanding of the generated signals, the cluster center for each class is also indicated in green along with them. In the figures, the distances of the generated signals from the nearest sample, from the cluster center, the maximum distance between all template samples and the cluster center, and the distance from the cluster center to the nearest real sample are provided. Furthermore, the predicted class by the trained classification model is also given for each generated signal in each class.

As shown in Fig. 7, the generated signals closely resemble the real ECG samples and exhibit the characteristic patterns of their respective classes.

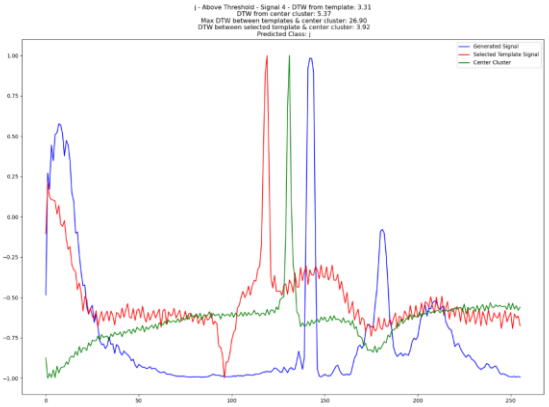
To improve classification accuracy, generated signals with a DTW distance exceeding the defined threshold of 3 from the nearest real sample are considered outliers and removed. Due to computational limitations, only the first 1,000 generated signals per class were evaluated in this step. Representative outlier signals identified for each class are presented in Fig. 8.



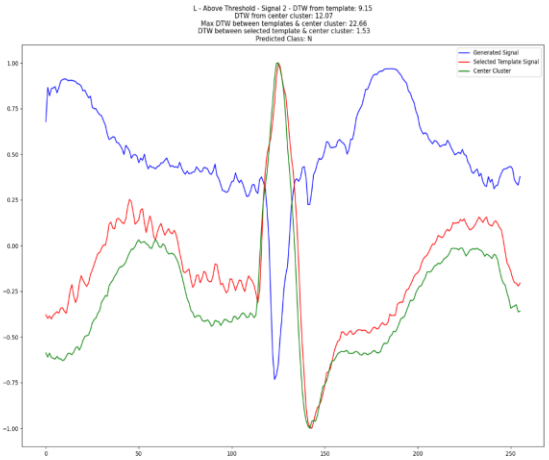
(a)



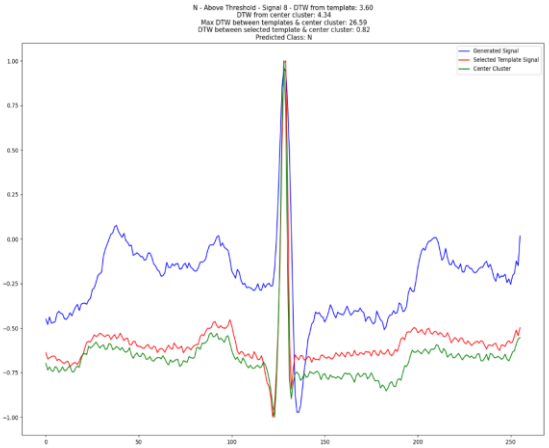
(b)



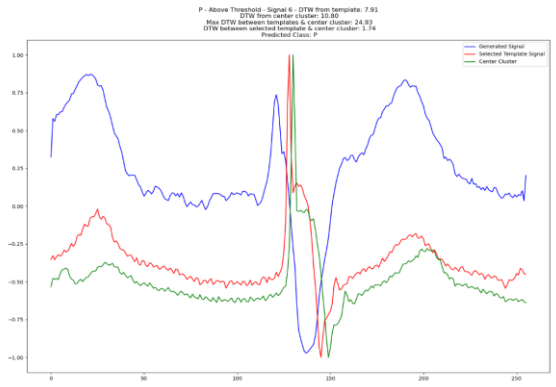
(c)



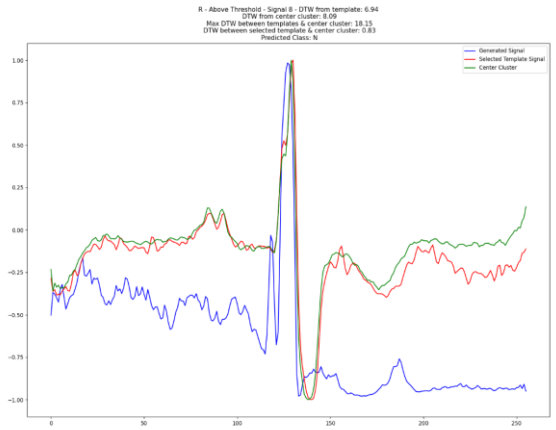
(d)



(e)



(f)



(g)

Fig. 8 Outlier ECG signals for each class: (a) Class A (b) Class f (c) Class j (d) Class L (e) Class N (f) Class P (g) Class R. The generated signals are indicated in blue, and the nearest real and template sample to each generated signal, based on DTW distance, is shown in red color.

As illustrated in Fig. 8, the generated signals show clear dissimilarity from the nearest real signals in each class, which may lead to misclassification. To mitigate this, a DTW threshold of 3 was applied to remove outliers.

Overall, the number of generated outlier samples and the number of good ECG signals, based on only 1000 generated signals from each class, are presented in Table 5.

Table 5: Number of outliers Vs. good signals

Class	*Number of Correct Samples*	*Number of Outlier Samples*
P	463	537
A	985	15
L	821	179
N	670	330
R	792	208
f	895	105
j	993	7

Here, we only examined the number of outlier signals and good signals for the first 1000 generated signals. Therefore, the number of signals in Table 5 is only for the first 1000 signals for each class, not for all generated signals.

5-8- Evaluation using ResNet34 Classifier

In this study, two proposals were examined to investigate the generated ECG signals. One involves modifying the generator's loss function in a way that captures structural differences between the generated signals and real ones. The other proposal involves incorporating an attention mechanism alongside a CNN model in the generator to ensure that the system architecture focuses more on crucial parts of the ECG signals.

A comparison is made regarding the use of other loss functions in the generator loss function. The accuracy and performance of the ResNet34 classifier in classifying generated signals from each of these methods are examined, and the results are presented in Table 6.

Table 6: Comparison of models with different loss functions. In each row, the calculated Precision for each class is displayed, and the last row shows the Accuracy obtained from each method.

Class	*WGAN-GP*	*MSE*	*DTW (Our model)*	*Kullback -Leibler*
P	0.95	0.90	**0.97**	**0.97**
A	0.61	0.53	**0.82**	0.49
L	0.74	0.85	**0.92**	0.65
N	0.98	0.96	**0.99**	0.98
R	0.93	**0.94**	0.88	0.86
f	0.11	0.22	0.34	**0.36**
j	0.12	**0.19**	0.10	0.10
ACC	0.89	0.92	**0.95**	0.89

In total, four evaluation methods were considered. The first method used only the WGAN-GP loss (Eq.(3)) without any additional terms in the generator loss. The second method adds Mean Squared Error (MSE) to encourage similarity to real signals (Eq.(11)).

$$L_{gen} = -E_{x\sim p_{fake}}[D(x,y)] + \lambda_{MSE} \cdot MSE_{loss} \quad (11)$$

The third method, our proposed approach, incorporates DTW to better align generated and real ECG waveforms (Eq.(6)), while the fourth method employs Kullback-Leibler (KL) divergence to promote diversity in the generated signals (Eq.(12)).

$$L_{gen} = -E_{x\sim p_{fake}}[D(x,y)] + \lambda_{KL} \cdot KL_{divergence} \quad (12)$$

As indicated in Table 6, Using MSE increased precision in classes R and j, whereas our DTW-based method achieved higher precision in the remaining classes. The goal of both these approaches is to generate signals as similar as possible to real signals. However, the fourth method, which involves using Kullback-Leibler in the generator's loss function, has a different objective. Its primary goal is to create diversity in the generated signals. As is evident, the proposed method generally has a higher accuracy compared to other methods after adding generated data to the dataset.

The classification results of the proposed model are summarized in Table 7. Compared with the reference model [2] the proposed approach achieved higher overall accuracy (95% vs. 92%) and showed better F1-scores in the main classes (P, L, N, and R). Although the reference model obtained higher recall for the minority classes (f and j), it suffered from lower precision and reduced overall accuracy. These results indicate that the proposed model provides a more balanced and reliable performance, particularly in the clinically important classes related to cardiac signal analysis.

Table 7: Classification Report

Class	*Precision*	*Recall*	*F1-score*	*Support*
P	0.97	0.99	0.98	702
A	0.82	0.91	0.86	254
L	0.92	0.97	0.95	807
N	0.99	0.95	0.97	7504
R	0.88	0.98	0.92	725
f	0.34	0.62	0.44	98
j	0.10	0.36	0.16	22
Accuracy		0.95		10112
Macro avg	0.72	0.83	0.75	10112
Weighted avg	0.96	0.95	0.95	10112

In the next comparison, four generative models were evaluated, as summarized in Table 8. The first model is our proposed DTW-based attention model. The second model is an attention-based version without incorporating DTW in the generator loss. The third model is the baseline WGAN-GP model [2], and the fourth model replaces the critic with a discriminator to assess the impact of this architectural change.

For the baseline model with a discriminator, the generator loss was modified by replacing the Cross-Entropy term with WGAN-GP, ensuring a fair comparison with models using a critic.

All models were evaluated using 10,000 generated signals, without applying an outlier removal threshold, to provide a consistent basis for comparison.

Table 8: Comparison of models in terms of classification accuracy

Class	*Proposed Model*	*Proposed model without DTW*	*Baseline [2]*	*Baseline with Disc*	*No generated signal*
P	**0.97**	0.95	0.95	**0.97**	0.76
A	**0.82**	0.61	0.62	0.49	0.37
L	**0.92**	0.74	0.86	0.63	0.74
N	0.99	0.98	1.00	0.97	0.94
R	0.88	**0.93**	0.76	0.86	0.78
f	0.34	0.11	**0.35**	0.18	0.00
j	0.10	**0.12**	**0.12**	0.02	0.00
ACC	**0.95**	0.89	0.92	0.89	0.89

Based on the results in Table 8, the proposed model outperformed all other evaluated models on the test data.

Table 9 presents a comparison of the proposed generative method with other recently utilized generative approaches in terms of classification accuracy after incorporating generated signals into the dataset using ResNet34.

Table 9: Comparison of the proposed model with related models in terms of classification accuracy

Research	*Year*	*Model*	*Accuracy*
Our	2024	ABC-GAN	**95%**
[2]	2022	ACWGAN-GP	92%
[17]	2022	ME-GAN	90.2%
[25]	2023	Improved DDPM	90%
[26]	2024	CECG-GAN	94%

The proposed method achieves higher accuracy, which can be attributed to its ability to effectively capture class-specific characteristics through the attention mechanism in the generator architecture and the inclusion of the DTW function in the generator loss. By generating signals that more closely resemble real ECG signals within each class, the method improves classification performance when added to the dataset.

5-9- Computational Complexity Analysis

In addition to generation quality, computational complexity is an important consideration for practical deployment. Table 10 compares the computational complexity of ABC-GAN against baseline methods.

Table 10: Comparison of the proposed model with related models in terms of complexity

Method	***Core Mechanism***	***Relative Complexity***	***Training Stability***
CardioGAN [12]	Attention-based GAN with dual discriminators	Low	Medium
AC-WGAN-GP [2]	Wasserstein GAN with gradient penalty	Medium	High
ME-GAN [17]	Multi-view conditional GAN with disease-aware representations	Medium–High	Medium
TTS-GAN [14]	Transformer-based GAN using self-attention	High	Medium
CECG-GAN [26]	Conditional GAN for class-imbalanced ECG synthesis	Medium–High	Medium
Task-Aware CGAN (DTW) [27]	Conditional GAN with DTW + multi-objective temporal loss	Medium–High	Medium
TransDiffECG [31]	Transformer-based diffusion model	High	High
RDiffGAN-ECG [32]	Hybrid diffusion + GAN + state-space modeling for R-peak conditioned ECG	Very High	High
ABC-GAN (Proposed)	**Attention-based conditional GAN with explicit DTW loss in generator**	**Medium–High**	**Medium–High**

5-10- Limitations

In this study, a fixed DTW threshold of 3 was selected for outlier removal across all classes. This choice was primarily motivated by computational constraints and to ensure a uniform filtering criterion. While class-specific thresholds have been proposed in the baseline work, where variability differs significantly between majority and minority classes, our simplified approach applies the same cutoff to all classes. Consequently, certain signals from high-variability classes (A and j) may have been incorrectly discarded, while signals in lower-variability classes might have been retained despite subtle mismatches.

Another limitation arises from evaluating only the first 1,000 generated samples per class when applying the DTW threshold. Although this was necessary due to computational costs, restricting the analysis to a subset of generated signals could bias the assessment of outlier prevalence and overall model fidelity. Future studies should investigate adaptive, class-specific thresholds and conduct a large-scale evaluation across all generated signals to provide a more comprehensive and statistically reliable assessment of synthetic ECG quality.

6- Future Works

Based on the results obtained, combining the DTW and MSE functions in the generator's loss function may enhance classification accuracy across multiple ECG classes. Specifically, in some classes, adding the MSE term to the DTW-based generator loss showed measurable improvements compared to using DTW alone. Future studies could systematically explore this combination and quantify the improvements for each class.

Additionally, ensuring the privacy of generated ECG signals remains an important consideration. Future work could investigate methods to rigorously guarantee patient data privacy, such as differential privacy or data anonymization techniques, when generating synthetic ECG signals.

Finally, alternative evaluation metrics to DTW should be considered. While DTW effectively captures temporal alignment, it is computationally intensive. Exploring other similarity measures, such as Pearson correlation, cosine similarity, or feature-based metrics, could reduce computational time while maintaining evaluation accuracy. Combining multiple evaluation criteria could further enhance both efficiency and reliability in assessing generated signals.

7- Conclusions

This study presents a novel approach for ECG signal generation and classification by integrating an attention mechanism into the generator architecture, employing Dynamic Time Warping (DTW) in the generator loss function, and utilizing a Critic block instead of a traditional discriminator. The proposed method successfully generates realistic ECG signals that, when added to the training dataset, improve classification performance using a ResNet34-based model. The results demonstrate that the proposed approach outperforms recent methods in terms of overall classification accuracy, achieving 95% compared to 92% for the baseline [2]. Key improvements include enhanced precision and F1-scores in major ECG classes (P, L, N, and R), while maintaining balanced performance across minority classes. Although the model incurs a higher computational cost due to DTW calculations, it achieves superior signal fidelity and class-specific learning. For future research, we recommend focusing on optimizing the thresholding process for generated signals, further preserving patient privacy, and

exploring alternative or combined evaluation metrics to reduce computational complexity. Overall, the proposed method provides a robust framework for generating and classifying ECG signals, with significant potential for further enhancements in synthetic biomedical data generation.

References

[1] G. Chen et al., "EmotionalGAN: Generating ECG to Enhance Emotion State Classification," 2019.

[2] E. Adib, F. Afghah, and J. J. Prevost, "Arrhythmia Classification using CGAN-augmented ECG Signals," arXiv preprint arXiv:2201.12345, 2022.

[3] N. P. Singh et al., "Remote Monitoring System of Heart Conditions for Elderly," Journal of Information Systems and Telecommunication (JIST), (accessed online).

[4] A. Sandooghdar and F. Yaghmaee, "Deep Learning Approach for Cardiac MRI Images," Journal of Information Systems and Telecommunication (JIST), vol. 10, no. 37, pp. xx–yy, 2022.

[5] A. Halder Roy, "Integrating Questionnaires and Physiological Signals for Stress Detection," Journal of Information Systems and Telecommunication (JIST), Jul. 2025.

[6] F. Zhu et al., "Electrocardiogram generation with a bidirectional LSTM-CNN generative adversarial network," Scientific Reports, vol. 9, no. 1, p. 107, 2019.

[7] Y.-H. Zhang and S. Babaeizadeh, "Synthesis of standard 12-lead electrocardiograms using two-dimensional generative adversarial networks," Journal of Electrocardiology, vol. 69, pp. 1–9, 2021.

[8] N. Wulana et al., "Generating electrocardiogram signals by deep learning," Neurocomputing, vol. 404, pp. 1–11, 2020.

[9] P. Wang et al., "ECG Arrhythmias Detection Using Auxiliary Classifier Generative Adversarial Network and Residual Network," IEEE Access, vol. 7, pp. 1–10, 2019.

[10] W. Li et al., "SLC-GAN: An automated myocardial infarction detection model based on generative adversarial networks and convolutional neural networks with single-lead electrocardiogram synthesis," Information Sciences, vol. 589, pp. 1–12, 2021.

[11] H. Yang et al., "ProEGAN-MS: A Progressive Growing Generative Adversarial Networks for Electrocardiogram Generation," IEEE Access, vol. 9, pp. 1–10, 2021.

[12] S. Dasgupta, S. Das, and U. Bhattacharya, "CardioGAN: An Attention-based Generative Adversarial Network for Generation of Electrocardiograms," in Proc. 25th ICPR, 2021.

[13] R. Banerjee and A. Ghose, "Synthesis of Realistic ECG Waveforms Using a Composite Generative Adversarial Network for Classification of Atrial Fibrillation," in Proc. 29th EUSIPCO, 2021.

[14] X. Li et al., "TTS-GAN: A Transformer-Based Time-Series Generative Adversarial Network," in Artificial Intelligence in Medicine. AIME 2022. LNCS, vol. 13263, Springer, 2022.

[15] E. Brophy, "Synthesis of Dependent Multichannel ECG using Generative Adversarial Networks," in Proc. 29th ACM CIKM, 2020.

[16] S. Harada, H. Hayashi, and S. Uchida, "Biosignal Data Augmentation Based on Generative Adversarial Networks," in EMBC 2018, 2018.

[17] J. Chen et al., "ME-GAN: Learning Panoptic Electrocardio Representations for Multi-view ECG Synthesis Conditioned on Heart Diseases," in ICML 2022, 2022.

[18] F. Ye et al., "ECG Generation With Sequence Generative Adversarial Nets Optimized by Policy Gradient," IEEE Access, vol. 7, pp. 1–10, 2019.

[19] V. V. Kuznetsov et al., "Interpretable Feature Generation in ECG Using a Variational Autoencoder," Frontiers in Genetics, vol. 12, p. 634, 2021.

[20] T. Golany and K. Radinsky, "PGANs: Personalized Generative Adversarial Networks for ECG Synthesis to Improve Patient-Specific Deep ECG Classification," in AAAI 2019, 2019.

[21] M. S. Munia, M. Nourani, and S. Houari, "Biosignal Oversampling Using Wasserstein Generative Adversarial Network," in ICHI 2020, 2020.

[22] D. Nankani and R. D. Baruah, "Investigating Deep Convolution Conditional GANs," in IJCNN 2020, 2020.

[23] F. N. Hatamian et al., "The Effect of Data Augmentation on Classification of Atrial Fibrillation in Short Single-Lead," in ICASSP 2020, 2020.

[24] T. Agarwal and E. Ertin, "CardiacGen: A Hierarchical Deep Generative Model for ECG Synthesis," 2022.

[25] E. Adib et al., "Synthetic ECG Signal Generation Using Probabilistic Diffusion Models," IEEE Access, vol. 11, pp. 75818–75828, 2023.

[26] X. Li, E. Adib, and A. M. Rahmani, “CECG-GAN: Conditional ECG Generation for Class-Imbalanced Cardiac Diagnosis Using GANs,” Scientific Reports, vol. 14, no. 1, p. 14767, 2024, doi: 10.1038/s41598-024-65619-8.

[27] K. Lang and Y. Li, “Task-Aware Conditional GAN with Multi-Objective Loss for Realistic and Efficient Time-Series Generation,” Journal of Big Data, vol. 12, 2025, Art. no. 206, doi: 10.1186/s40537-025-01266-8.

[28] N. Neifar, A. Ben-Hamadou, A. Mdhaffar, and M. Jmaiel, “DiffECG: A Versatile Probabilistic Diffusion Model for ECG Signal Synthesis,” in Proc. IEEE/ACIS Int. Conf. Software Engineering Research, Management and Applications (SERA), 2024, pp. 182–188, doi: 10.1109/SERA61261.2024.10685651.

[29] Y. Lai, B. Liu, X. Guan, Q. Zhao, and S. Hong, “ECGTwin: Personalized ECG Generation Using Controllable Diffusion Model,” arXiv:2508.02720, 2025. [Online]. Available: https://arxiv.org/abs/2508.02720

[30] L. Simone, D. Bacciu, and V. Gervasi, “ECG Synthesis for Cardiac Arrhythmias: Integrating Self-Supervised Learning and Generative Adversarial Networks,” Artificial Intelligence in Medicine, vol. 167, p. 103162, 2025, doi: 10.1016/j.artmed.2025.103162.

[31] Y. Lin, J. Ma, S. Dong, C. Sun, W. Cong, and K. Wang, “TransDiffECG: Semantically Controllable ECG Synthesis via Transformer-Based Diffusion Modeling,” Journal of Biomedical Informatics, vol. 172, p. 104948, 2025, doi: 10.1016/j.jbi.2025.104948.

[32] P. Li, Y. Zhou, J. Min, Y. Wang, W. Liang, and W. Li, “RDiffGAN-ECG: High-Fidelity R-Peak Conditional ECG Generation Using Diffusion GAN with State Space Models,” Biomedical Signal Processing and Control, vol. 108, p. 108544, 2025, doi: 10.1016/j.bspc.2025.108544.

[33] M. Mirza and S. Osindero, "Conditional Generative Adversarial Nets," arXiv preprint arXiv:1411.1784, 2014.

[34] I. Goodfellow et al., "Generative Adversarial Nets," in Advances in Neural Information Processing Systems (NIPS), 2014, pp. 2672–2680.

[35] A. Vaswani et al., "Attention Is All You Need," in Advances in Neural Information Processing Systems (NIPS), 2017, pp. 5998–6008.

[36] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-Attention Generative Adversarial Networks," in Proc. ICML, 2019, pp. 7354–7363.

[37] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein Generative Adversarial Networks," in Proc. 34th ICML, 2017.

[38] A. Lyashuk, "ECG Classification," [Online]. Available: https://github.com/lxdv/ecgclassification/blob/master/README.md, 2021.

[39] K. He et al., "Deep residual learning for image recognition," in Proc. IEEE Conf. CVPR, 2016, pp. 770–778.

[40] G. B. Moody and R. G. Mark, "The impact of the MIT-BIH Arrhythmia Database," IEEE Engineering in Medicine and Biology Magazine, vol. 20, no. 3, pp. 45–50, 2001.

[41] A. M. Delaney, E. Brophy, and T. E. Ward, "Synthesis of realistic ECG using generative adversarial networks," arXiv preprint arXiv:1903.09633, 2019.

[42] S. Harada, H. Hayashi, and S. Uchida, "Biosignal Generation and Latent Variable Analysis With Recurrent Generative Adversarial Networks," IEEE Access, vol. 7, pp. 1–10, 2019.

[43] K. F. Hossain et al., "ECG-Adv-GAN: Detecting ECG Adversarial Examples with Conditional Generative Adversarial Networks," in Proc. 20th IEEE ICMLA, 2021.

[44] A. M. Shaker et al., "Generalization of Convolutional Neural Networks for ECG Classification Using Generative Adversarial Networks," IEEE Access, vol. 8, pp. 1–10, 2020.

[45] T. Golany et al., "Improving ECG Classification Using Generative Adversarial Networks," in IAAI-20, 2020.

[46] A. Furdui et al., "AC-WGAN-GP: Augmenting ECG and GSR Signals using Conditional Generative Models for Arousal Classification," 2021.

[47] Y.-Y. Jo et al., "ECGT2T: Towards Synthesizing Twelve-Lead Electrocardiograms from Two Asynchronous Leads," in ICASSP 2023, pp. 1–5.

[48] T. Lan et al., "Arrhythmias Classification Using Short-Time Fourier Transform," in EMBC 2020, 2020.

[49] A. Goldberger et al., "Components of a new research resource for complex physiologic signals: PhysioBank, PhysioToolkit, and PhysioNet," Circulation, vol. 101, no. 23, pp. E215–E220, 2000.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 99-116
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Optimization of Hyperparameters of BERT Model in Sentiment Analysis Using Genetic Algorithm

Shahla Sadeghani [1], Mohammad Jafar Tarokh [2*] , Mohammad Ali Afshar Kazemi [3]

[1]. Department of Information Technology Management, SR.C.,Islamic Azad University, Tehran, Iran
[2]. Department of Industrial Engineering, K.N.,Toosi University of Technology, Tehran, Iran
[3]. Department of Industrial Management, CT.C., Islamic Azad University, Tehran, Iran

Received: 19 Sep 2025/ Revised: 04 Jun 2025/ Accepted: 20 Jul 2026

Abstract

Sentiment analysis, a vital branch of natural language processing (NLP), is progressing rapidly, and advanced models such as BERT are used to improve the accuracy of such analyses. However, tuning BERT's hyperparameters remains a major challenge, directly influencing its performance. The methods used in previous research to optimize hyperparameters often face limitations in accuracy and efficiency. This study addresses an important research knowledge gap by using genetic algorithm (GA) to present a new approach to optimize the hyperparameters of BERT model in sentiment analysis with the main goal to get better results. In this regard, this research is designed analytically and experimentally. Data were collected through Twitter API and preprocessed using NLP techniques including noise removal, tokenization, and text normalization before being analyzed. The BERT model's hyperparameter optimization was performed using GA and the optimized models were evaluated based on criteria such as accuracy, precision, recall and F-1 score. The proposed algorithm was compared with other common methods such as vanilla BERT, Long Short-Term Memory (LSTM) and convolutional neural network (CNN). The results showed that the proposed model has significantly better performance than other methods, achieving 98.1% accuracy, 98.2% precision, 98.1% recall, 98.15% F-1 score, and an area under the curve (AUC) of 98.5%. In comparison, vanilla BERT model, LSTM and CNN scored 92.8%, 91.3% and 90.5% in accuracy, respectively. These results indicate that the use of GA in optimizing the hyperparameters of BERT model can effectively increase the accuracy and efficiency of sentiment analysis. This approach not only has applicability in various fields of text data analysis, but also can be used as an effective solution for future research in optimizing deep learning models.

Keywords: Sentiment Analysis; BERT; Genetic Algorithm; Hyperparameter Optimization.

1- Introduction

Sentiment analysis has attracted the attention of many researchers and organizations in recent years as a tool for understanding public sentiment and opinions [1], which provides the ability to extract feelings and emotions from texts. This ability is especially important in social platforms such as Twitter, which publish millions of comments and views daily [2]. With the increase in the extent of text data, the use of more advanced models such as BERT for sentiment analysis has become a necessity [3]. BERT is one of the most advanced NLP models [4], which is able to significantly increase the accuracy of sentiment analysis by using a Bidirectional Pre-Training Approach [3]. Fine-tuning the hyperparameters of this model can significantly affect its final performance [4]. However, optimizing the hyperparameters of this model to achieve the best performance is a big challenge [3]. The potential consequences of not addressing this issue include reduced accuracy in sentiment analysis, higher computational resource consumption, and the inability to improve existing models. This problem can have a negative impact, especially in industrial and commercial applications where model accuracy and efficiency are of great importance. Hence, there is a strong need for a novel and efficient method for automatically optimizing hyperparameters of complex models such as BERT.

Most previous research has focused on manual optimization of hyperparameters, which is a time-consuming and computationally intensive process. In addition, conventional optimization methods such as grid search and random

✉ **Mohammad Jafar Tarokh**
mjtarokh@kntu.ac.ir

search are not efficient for complex models because they cannot automatically find optimal parameter combinations in a large search space [5]. This shortcoming not only leads to a decrease in the accuracy and performance of the models, but also inefficient optimization leads to a greater consumption of time and resources. Therefore, there is a major gap in the current research field to develop more efficient and automated optimization approaches for the hyperparameters of BERT model.

In recent years, researchers have been looking for new methods for optimizing hyperparameters. One of these methods is the use of GA, which use Darwinian Evolutionary Principles to find the best hyperparameter settings. Studies such as the research conducted by Jones et al. [6] have shown that the use of GA can effectively improve the accuracy of deep learning models. In addition, another study points to the successful application of GA in optimizing NLP models [7]. These studies indicate the importance of using advanced optimization methods to achieve higher accuracy and efficiency in complex models such as BERT.

This research seeks to solve the problem of optimizing the hyperparameters of BERT model in sentiment analysis by using GA. The main objective of this study is to present an approach based on GA to optimize the hyperparameters of BERT model and improve its performance in sentiment analysis. The main goal of the present research is to create a method that would be capable of enhancing the performance of machine learning models through automatic hyperparameter selection and the more precise optimization of the chosen ones. The current paper suggests the use of a Genetic Algorithms (GA) to face the optimization of BERT hyperparameters in sentiment analysis. The primary aim of this investigation is to develop an approach based on GA that can maximize the potential of BERT in the sentiment analysis task through hyperparameter tuning and consequently achieve better performance. The proposed approach aims to improve model performance by enabling automatic selection and more precise optimization of the model hyperparameters. In order to provide proof of the proposed method's effectiveness, we will be comparing the GA-optimized BERT model's performance to that of the baseline techniques that are often used, such as vanilla BERT (standard pre-trained model) [8], Convolutional Neural Networks (CNNs) [8], and Long Short-Term Memory networks (LSTMs) [9]. These kinds of comparisons make it possible to form an overall assessment of the strengths and weaknesses of each approach to the methods. For other deep learning models in hyperparameter optimization and help address existing knowledge gaps in this domain.

This paper is structured as follows: the "Background and Description" section provides basic information about deep learning models and GA optimization methods for BERT model. The "Literature Review" section surveys previous research in the field of sentiment analysis and hyperparameter optimization methods, providing readers with theoretical foundations and insights from existing work. Then, the "Proposed System" section provides details about the proposed research framework, methods, datasets, and how to implement the optimized model. In the "Results Analysis" section, the empirical findings of the research are analyzed and evaluated, and a comprehensive comparison between the proposed approach and existing methods is made. Finally, the paper ends with the "Conclusion" section, where the main implications of the research are discussed. This structure helps the reader to move steadily from theoretical foundations to empirical findings and practical applications.

2- Background and Explanation

2-1- BERT Algorithm

BERT is a deep learning model introduced by Google researchers in 2018 [10]. It is based on a transformer architecture and is used for NLP. Unlike previous unidirectional models, BERT's main goal is to develop language models that understand the meaning of words in the context of sentences in a bidirectional manner [11]. Before BERT, models such as Word2Vec and GloVe for words representation provided fixed representations of words without regard to the overall meaning of the sentence [12]. These models were unable to understand complex semantic dependencies. Models such as ChatGPT, although they made significant improvements, still only analyzed dependencies in one direction (left to right) [13]. BERT overcomes these limitations by using a bidirectional approach. The transformative architecture that underlies BERT consists of two main parts: an encoder and a decoder. The encoder was used to analyze the inputs and generate semantic models. This part consists of multiple layers of attention and feedforward neural networks, whose self-attention mechanism allows the model to evaluate the relationship of each word to other words in the sentence and to detect long-range semantic dependencies [10]. Self-attention uses three main components: query(Q), key(K), and value(V), which weight words based on their importance in the text.

BERT uses a set of encoder layers that include tokens, positional embeddings, and representations of input segment embedding. Its architecture has two main versions, BERT-base and BER-large, which have 12 and 24 encoder layers, respectively. BERT’s pre-training uses two main tasks: Masked Language Modeling (MLM) and Next Sentence Prediction (NSP) [8]. In MLM, 15% of the tokens are masked and the model tries to predict them, which enhances the text representations. In NSP, BERT

predicts logical relationships between sentences, which helps to understand inter-sentence dependencies.

BERT is used in many NLP tasks, including sentiment analysis, question answering, machine translation, nominal entity recognition, and text classification. The advantages of BERT include bidirectional representation, portability, and high performance in various tasks. However, its limitations include the need for high computational resources and sensitivity to tokenizers [10], [12]. Advanced research based on BERT includes models such as RoBERTa to improve pre-training methods and an optimized version of the model (DistilBERT) as a lighter version with a reduced number of parameters (ALBERT). BERT is a revolutionary model in NLP that has achieved impressive results in many NLP tasks by using bidirectional word representation and a transformer architecture, and has paved the way for the development of other advanced models. The architecture of BERT algorithm is shown in Figure 1.

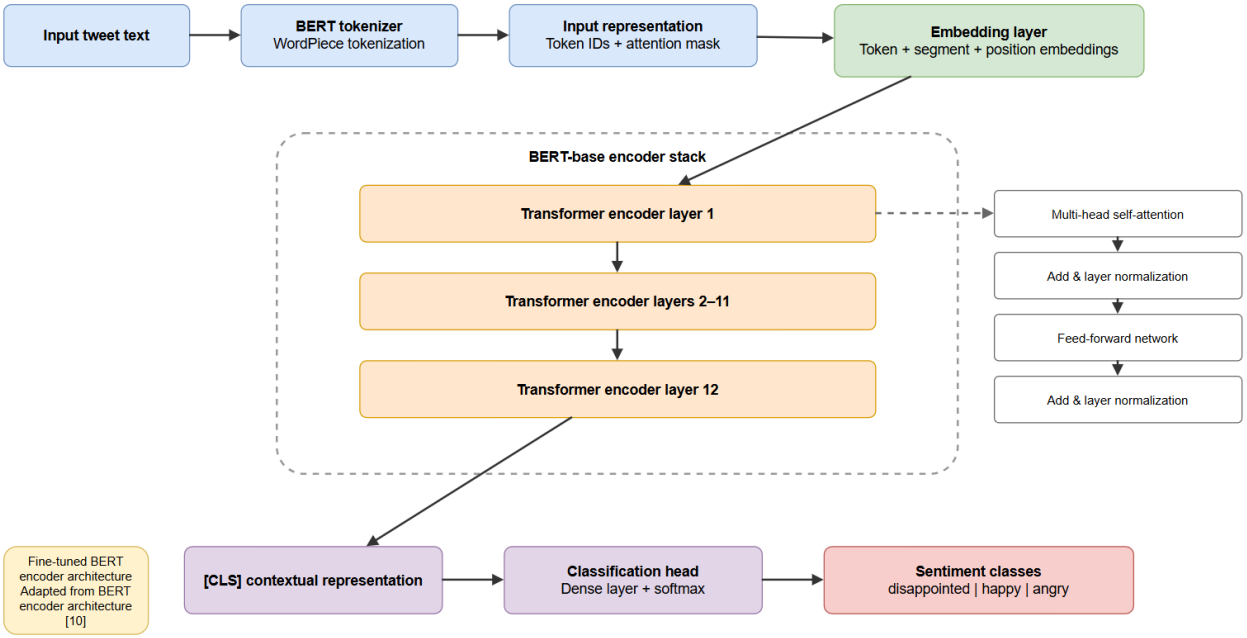


Fig. 1. Architecture of BERT algorithm [10]

2-2- Hyperparameters and the Importance of Optimization

Hyperparameters are variables that are set before the model training process begins and directly affect the model performance [14]. Examples of hyperparameters include the number of epochs, learning rate, and optimizer type. Unlike the internal weights of the network, which are optimized during the training process, hyperparameters are tuned manually or using specific optimization methods. Optimizing these variables is of particular importance, because their incorrect tuning can lead to reduced accuracy or instability in the learning process. Using intelligent methods for optimization, such as genetic algorithms, can significantly improve model performance [19].

For example, an appropriate learning rate can increase the convergence speed of the model, while an excessively high learning rate may lead to model instability. Also, choosing the appropriate number of epochs ensures that the model is well trained and prevents overfitting.

2-2-1- Hyperparameters of BERT Model

In BERT model optimization process, three key hyperparameters are set, each of which has a direct impact on the final performance of the model. Table 1 shows the hyperparameters of BERT model and their descriptions.

Table 1: Hyperparameters of BERT model

Hyperparameters	Description
Number of Epochs	Number of epochs refers to the total times a model is fully trained on the training dataset. Selecting an appropriate number of epochs enables the model to learn effectively while avoiding overfitting
Learning rate	Learning rate determines the size of weight updates in each training step. Properly tuning this parameter can improve model convergence rate
Optimizer type	Optimizer type refers to the optimization algorithm used to update model weights, such as AdamW or SGD. This parameter influences both the learning efficiency and performance of the model

The choice of these three hyperparameters (epoch, learning rate, and optimizer type) was primarily influenced by their impact on the performance of BERT during fine-tuning for sentiment analysis reported in the literatures. Along with other hyperparameters such as batch size, dropout rate, and number of hidden layers which definitely play a role, it has been confirmed through research that these three parameters significantly influence the model convergence and final performance [5] [4].The batch size was consistently set to 16 because it is considered standard when fine-tuning BERT on datasets that are about the same size [10]. Dropout rate (0.1) and the number of hidden layers (12 for BERT-base) are the default settings that have been successful in various NLP tasks and thus were not changed [10]. The optimization using a genetic algorithm was therefore carried out on the most influential hyperparameters in order to achieve a trade-off between computational efficiency and performance gains.

2-3- Genetic Algorithm

2-3-1- Introduction to Genetic Algorithm

The genetic algorithm is a method for optimization based on an evolution-like process, which features nature's way of selection, crossover, and mutation, extensively used to tackle complicated problems with a great deal of solutions to be searched through [15][14]. The algorithm starts by generating a population of random chromosomes (candidate solutions) and then each chromosome gets evaluated using a fitness function. The fittest chromosomes are then chosen to pass on their genes to the next generation as well as to produce new offspring using combinations of crossover and mutation [17], [18]. This loop of performing the same task repetitively continues until a terminating condition of either a predetermined number of generations reached or the optimum solution obtained is satisfied. In this study, GA is used to optimize the three most important parameters of the BERT model in evaluating sentiments: number of epochs, learning rate, and optimizer type. Through this technique, the importance of genetic algorithms in the optimization of the most crucial parameters becomes clear due to their ability to efficiently search through very large search spaces. The GA flowchart is shown in Figure 2.

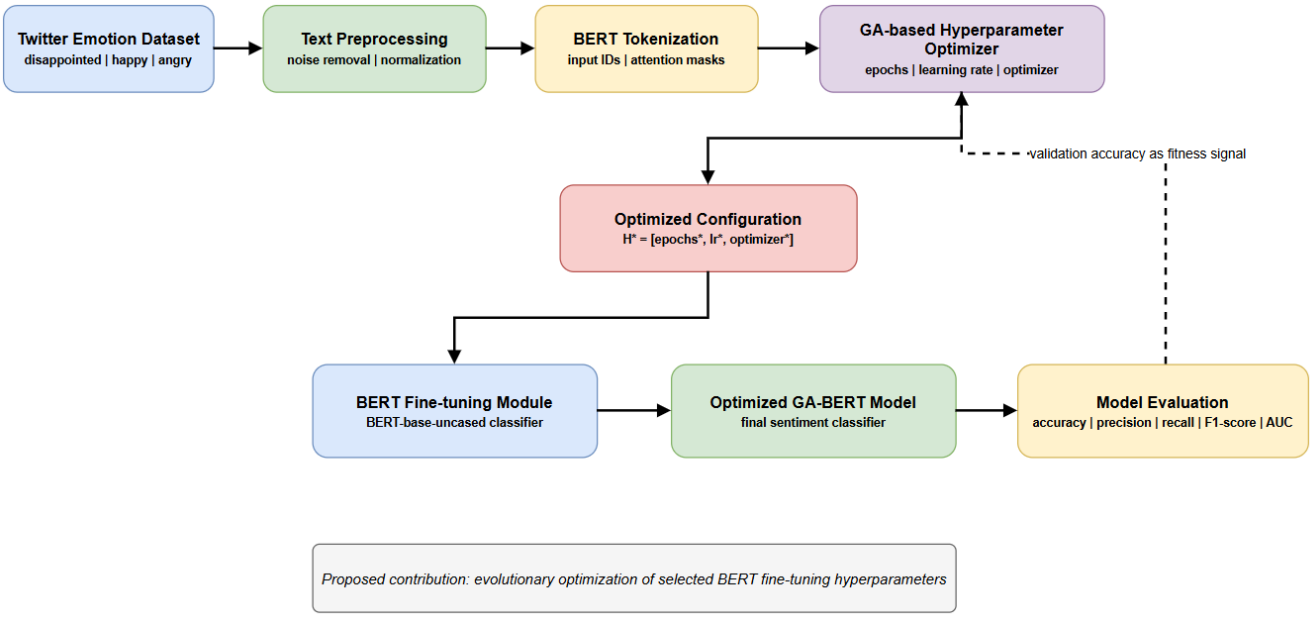


Fig. 2 Flowchart of the Genetic Algorithm [14]

2-3-2- Genetic Algorithm Parameters

he genetic algorithm adopts a predefined set of parameters, with each of them being crucial to the step of searching and optimizing. The important parameters of the GA are described in Table 2.

Table 2: Key Parameters of the Genetic Algorithm

parameters	Description
Population size	The number of individuals per generation in the algorithm, representing the number of hyperparameters combinations evaluated in each generation
Crossover rate	The probability of combining two individuals (solutions) to create a new individual inheriting features from its parents
Mutation rate	The probability of altering one or more genes (hyperparameters) in an individual to sustain diversity and prevent local optima convergence
Number of Generations	Number of iterations to produce new generations and assess hyperparameter combinations

The size of the population is what determines the number of different combinations of hyperparameters being assessed in every generation. Crossover and mutation rates control the frequency of generating new combinations through parental recombination and random alterations, respectively. The total generations represent the entire search process formed by the closure or best discovery found in the search iterations. The particular values for these parameters applied in the suggested model are stated in the Proposed Model section.

3- Literature Review

Over the past few years, sentiment analysis has become a widely recognized and frequently used method for public opinion and sentiment detection, especially through the medium of social media where Twitter posts alone are in millions every day[1][2]. With the advancement of deep learning models, the use of advanced architectures such as BERT for sentiment analysis has grown considerably due to their high accuracy in processing large-scale data [19].
This literature review examines previous studies on optimizing BERT models for sentiment analysis to provide a comprehensive view of existing research, identify knowledge gaps, and justify the importance of the present study. Recent research on BERT-based sentiment analysis can be categorized into three main approaches: hybrid model architectures, preprocessing optimization, and domain-specific adaptations. A detailed comparison of these studies is presented in Table 3, which summarizes their objectives, methodologies, key findings, strengths, and limitations. Hybrid Model Architectures Several studies have explored combining BERT with other deep learning architectures to enhance sentiment analysis performance. Bello et al. [8] integrated BERT with CNNs, RNNs, and Bi-LSTM, achieving 93% accuracy and 95% F1-score across six datasets, demonstrating that architectural combinations can outperform word-to-vector conversion methods. Similarly, Talaat [20] developed a hybrid system combining BiGRU and BiLSTM layers with RoBERTa and DistilBERT, showing improved accuracy through effective preprocessing and layer integration. Sirisha and Chandana [9] proposed a RoBERTa-LSTM hybrid model for aspect-based sentiment and emotion analysis, achieving 94.7% accuracy on Ukraine-Russia war tweets. While these hybrid approaches demonstrated performance improvements, they did not comprehensively investigate hyperparameter optimization, which represents a significant research gap. Preprocessing and Multilingual Optimization Other research has focused on optimizing preprocessing techniques for BERT models. Pota et al. [21] conducted a comprehensive evaluation of preprocessing methods for multilingual BERT models using Twitter datasets in English, Italian, and Spanish, demonstrating that appropriate preprocessing can significantly improve model accuracy. However, this study did not address hyperparameter optimization or examine combinations of different preprocessing methods, limiting its scope for performance enhancement. Domain-Specific BERT Applications Research has also adapted BERT for specific domains and applications. Lin and Moh [22] presented Covid-Twitter-BERT using the auxiliary sentence technique for COVID-19 tweet analytics, thereby enhancing accuracy through the modification of the sentence pair. Palani et al. [23] developed T-BERT, combining Latent Dirichlet Allocation with BERT for microblog sentiment analysis, achieving 90.81% accuracy. Chiorrini et al. [24] applied BERT to sentiment analysis and emotion detection in tweets, with their accuracies being 92% and 90% respectively. Anggrainingsih et al. [25]did rumor detection on Twitter using BERT and showed the system's ability to perform automated feature extraction. Wu et al. [26] in their experiment ranking BERT through optimization techniques mainly for sentiment analysis reported that although the optimization strategies achieved promising results after fine-tuning, the cost in terms of computational resources was substantial. Research Gaps and Study Justification the gaps revealed by the reviewed literature are presented and summarized in Table 3. The systematic hyperparameter optimization through evolutionary algorithms is an area that has been very slightly touched in the past while the whole research trend has focused on combining BERT with other methods and looking at preprocessing techniques. Most studies relied on default hyperparameter settings or manual tuning, which restricts model performance potential. Furthermore, none of the reviewed studies employed genetic algorithms for automated hyperparameter optimization of BERT models in sentiment analysis tasks.
The present study proposes an integration of BERT with hyperparameters optimized via genetic algorithms in order to overcome these shortcomings with a systematic methodology to find an appropriate combination of

parameters of the number of epochs, learning rate, and optimizer. The method is more accurate than manual tuning methods and also solves the problems that were pointed out in the previous studies conducted. Table 3 provides a detailed comparison of the reviewed studies, highlighting their respective methodologies, findings, strengths, and limitations.

Table 3: Comparison of reviewed articles

Article Title	Study Objective	Methodology	Key Findings	Strengths	Weaknesses
Multilingual Evaluation of Pre-Processing for BERT-based Sentiment Analysis of Tweets[17]	Multilingual Preprocessing Evaluation to Enhance BERT-Based Sentiment Analysis on Tweets	Evaluating the Impact of Different Text Preprocessing Techniques on BERT Performance in Multiple Languages	Techniques of appropriate preprocessing were found to substantially enhance the aptness of the BERT model in performing multi-lingual sentiment analysis tasks in English, Italian, and Spanish datasets in languages other than English, though the exact figures were not reported.	Emphasizing on multilingual analysis and the importance of preprocessing in improving model performance	Lack of examining the combination of different preprocessing methods and their impact on model performance
Aspect-Based Sentiment and Emotion Analysis with ROBERTa, LSTM[9]	Aspect-Based Sentiment and Emotion Analysis Using Hybrid RoBERTa and LSTM Models	Using RoBERTa-LSTM Hybrid Model for Aspect-Based Sentiment and Emotion Analysis of Ukraine-Russia War Tweets	The RoBERTa-LSTM hybrid model scored 94.7% accuracy and exceeded the existing methods in aspect-based sentiment and emotion analysis.	Combining advanced models and using deep learning methods for more accurate sentiment analysis	Lack of comprehensive examination of the impact of different hyperparameter settings and different model applications in different environments
Sentiment Analysis Classification System Using Hybrid BERT Models[20]	A hybrid BERT-based system for sentiment classification to improve accuracy	Combination of BiLSTM and BiGRU with BERT is applied to sentiment classification on multi-purpose data, with and without emojis	The hybrid model which combines BiGRU & BiLSTM with RoBERTa and DistilBERT performed better in terms of accuracy compared to the baseline S-BERT models on the three different datasets of tweets. The preprocessing analyses indicated the effect of the removal of emojis in the dataset.	Using multi-purpose data to examine the impact of preprocessing on model performance	Limited to three datasets and lack of examination of the applications of the proposed models in other environments
A BERT Framework to Sentiment Analysis of Tweets[8]	Developing an advanced framework for Twitter sentiment analysis using BERT and different types of deep neural networks	Combining BERT model with CNNs and RNNs to improve the accuracy of sentiment analysis	The combination of BERT with CNNs, RNNs, and BiLSTM led to obtaining 93% accuracy and 95% F1-score on six datasets, thereby significantly outdoing word-to-vector conversion methods.	Using a combination of different deep learning models to improve model accuracy and performance	Reliance on internet-sourced data and restricted to basic sentiment analysis
Topic-BERT (T-BERT) Model for Sentiment Analysis of Micro-blogs[23]	Investigating the effectiveness of a hybrid T-BERT model for sentiment analysis using Twitter datasets	Utilizing a hybrid T-BERT model for improving sentiment analysis accuracy	The T-BERT hybrid model that integrated LDA topic modeling with BERT has achieved 90.81% accuracy on a 42,000-tweet dataset, thus proving its effectiveness in the sentiment analysis of microblogs.	Integrating topic models with BERT to improve sentiment analysis performance	Limited scalability to complex data and high computational requirements

Article Title	Study Objective	Methodology	Key Findings	Strengths	Weaknesses
Research on the Application of Deep Learning-based BERT Model in Sentiment Analysis[26]	Investigating the application of BERT models in sentiment analysis and improving its performance through model fine-tuning	Applying BERT to optimize performance through experimental evaluation	BERT demonstrated effective sentiment analysis performance with significant improvements after optimization and fine-tuning, though specific quantitative metrics were not reported in the reviewed study.	Optimizing the performance of BERT model and using it in high-accuracy sentiment analysis	Requires high computational resources for large-scale and time-consuming data processing
Emotion and Sentiment Analysis of Tweets using BERT[24]	Exploring the use of BERT models to identify emotions and sentiments in tweets	Utilizing BERT Models for High-Accuracy Sentiment Analysis and Emotion Recognition	BERT achieved 92% accuracy in sentiment analysis and 90% accuracy in emotion identification on Twitter data, demonstrating strong performance in both tasks.	High accuracy in sentiment and emotion recognition in tweets	Requires more data for improved performance and has limitations in analyzing complex or multilingual texts
BERT based classification system for detecting rumors on Twitter[25]	Rumor detection on Twitter using BERT model	Rumor detection on Twitter using BERT model with extracted tweet content features	The BERT-based rumor detection delivered better performance and faster processing time than the conventional techniques through the automatic feature extraction, therefore, without the need for manual feature engineering (exact accuracy figures not disclosed).	Improving time efficiency with BERT by removing manual feature extraction	Requires processing large datasets and high-volume tweets
Sentiment Analysis on COVID Tweets Using COVID-Twitter-BERT with Auxiliary Sentence Approach [22]	COVID tweet sentiment analysis using COVID-Twitter-BERT and the auxiliary sentence method	Using Covid-Twitter-BERT model and the auxiliary sentence method to analyze the sentiment of Covid-19 tweets	The Covid-Twitter-BERT with auxiliary sentence strategy realized higher accuracy and F1-score in the COVID-19 tweet sentiment analysis than that of the standard classification methods (exact percentage improvements given in the original study).	Combining COVID-Twitter-BERT and auxiliary sentences enhances model accuracy	Improving model performance requires larger and more diverse data

A literature review showed that previous research in sentiment analysis has only focused on combining BERT models with classical methods and has not considered the issue of hyperparameter optimization and the use of evolutionary algorithms in combination with advanced models such as BERT. Considering the limitations observed in the reviewed papers, this study, by combining the BERT model and GA, presents a proposed framework that improves the performance of the model, which was one of the major challenges in previous papers. This approach, by utilizing evolutionary algorithms, provides more accurate optimization of the model hyperparameters and effectively addresses the gaps identified in previous studies.

4- Proposed GA-BERT Optimization Model

This section presents the proposed GA-BERT model for sentiment classification. The main innovation of this research is the integration of the fine-tuning process of BERT model with a hyperparameter optimization mechanism based on genetic algorithms. The model is designed to automatically identify the appropriate combination of three hyperparameters that influence the fine-tuning of BERT: number of training epochs, learning rate, and optimizer type.

Unlike the standard BERT training process, the proposed model introduces an evolutionary search process in which different combinations of hyperparameters are evaluated

and iteratively improved based on their performance on the validation set.

4-1- Overview of the Proposed Model

The proposed model consists of two interconnected parts. The first part is a BERT-based sentiment classification module that receives preprocessed tweets and classifies them into three sentiment categories: Disappointed, Happy, and Angry.
The second part is a GA-based optimization module that searches the hyperparameter search space and determines the best configuration for fine-tuning BERT model.

In this paper, text preprocessing, tokenization, data partitioning, BERT model training, and model evaluation are considered as standard implementation steps. These steps are necessary to ensure the reproducibility of the research, but do not constitute the main contribution and algorithmic innovation of this study.
The main and innovative component of this research is the GA-based optimization mechanism that controls the fine-tuning process of BERT model and determines the optimal values of the number of training epochs, learning rate, and optimizer type. The overall architecture of the proposed model is shown in Figure 3.

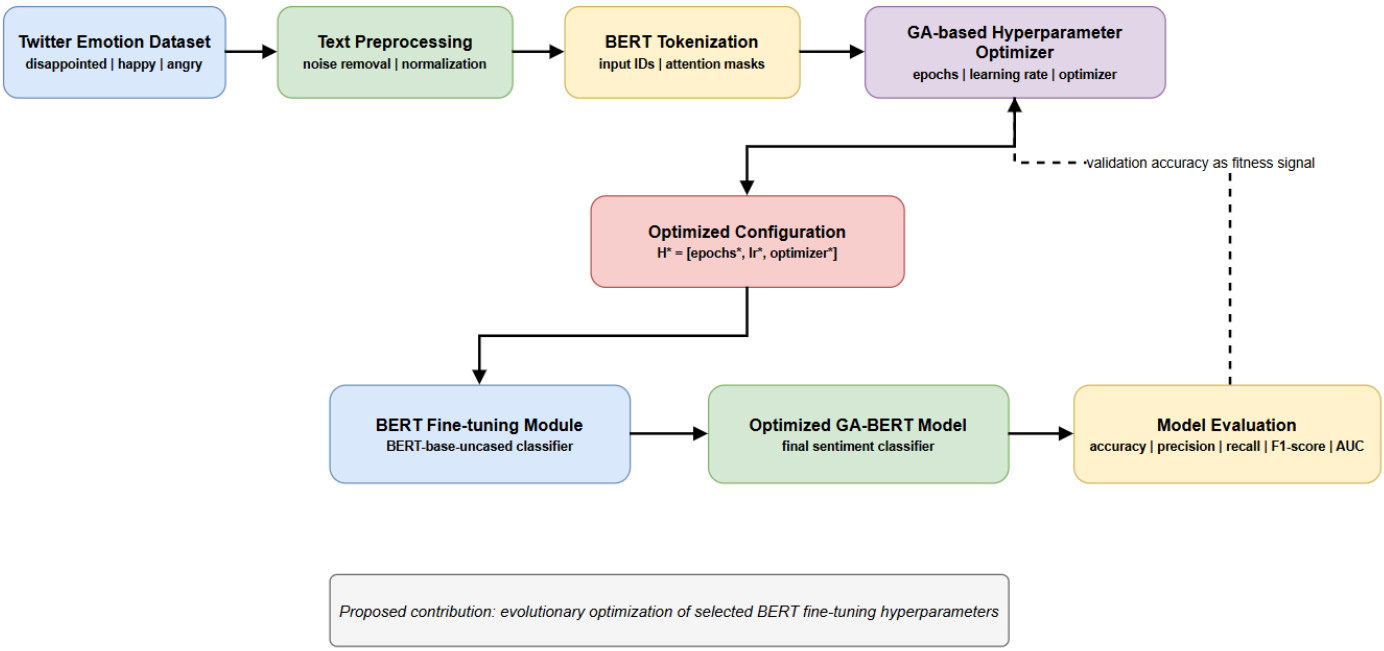


Fig. 3 Architecture of the Proposed GA-BERT Optimization Model

4-2- Main Components of the Proposed Model

The proposed model includes the following main components:

1. BERT-based sentiment classifier: In this section, BERT-base-uncased model is used as the core of the transformer architecture-based classifier. This model receives tokenized tweets and provides sentiment predictions for three target classes. BERT fine-tuning process is performed using a combination of hyperparameters selected by the genetic algorithm.
2. Hyperparameter Search Space: The search space consists of three main hyperparameters in BERT fine-tuning process: the number of training epochs, the learning rate, and the type of optimizer. Each solution in the genetic algorithm represents a possible combination of these hyperparameters.
3. GA-based optimizer: The genetic algorithm first generates an initial population of different combinations of hyperparameters. It then evaluates each combination based on the accuracy obtained from the validation of the tuned BERT model and gradually improves the population through selection, crossover and mutation operators.
4. Fitness Evaluation Module: For each solution, BERT model is fine-tuned using the training data and then evaluated with the validation data. The fitness value is calculated based on the relationship (1 - Accuracy); therefore, the optimization problem is formulated as a minimization problem.

5. Optimized GA-BERT Model: After the evolutionary search process is completed, the best combination of hyperparameters is selected and used for the final training of BERT-based sentiment classifier. . Finally, the model performance is evaluated using the criteria of Accuracy, Precision, Recall, F1 score and Area Under the ROC Curve (AUC).

4-3- GA-based Optimization Workflow

The optimization process begins by encoding each combination of hyperparameters as a chromosome. Each chromosome contains three genes: the number of training epochs, the learning rate, and the optimizer type. An initial population is then randomly generated within predetermined ranges for the search. Each chromosome is transformed into a configuration for fine-tuning BERT model, and BERT model is trained and validated against that configuration. The accuracy obtained from the validation is then used to calculate a fitness value.

After evaluating the fitness, the genetic algorithm selects the chromosomes that perform better as parents. The crossover operator combines the parent chromosomes together to create new configurations, while the mutation operator maintains the population diversity and reduces the likelihood of premature convergence by making random changes to the selected genes.

This evolutionary process continues until a predetermined number of generations are reached. Finally, the chromosome with the lowest fitness value is selected as the optimal configuration of hyperparameters. The optimization process based on the proposed genetic algorithm is presented in Figure 4.

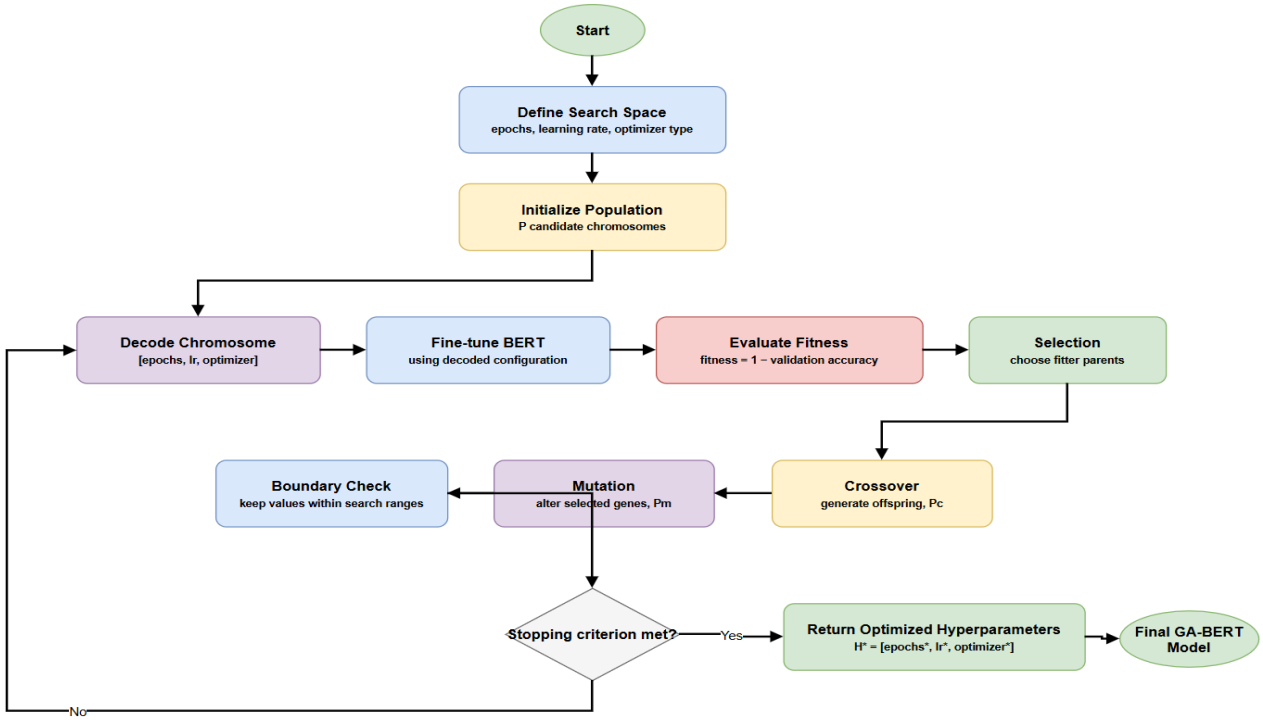


Fig. 4 GA-based Hyperparameter Optimization Workflow

4-4- Assumptions and Scope of the Proposed Model

The proposed model is designed based on two main assumptions. First, it is assumed that the computational environment, including hardware and software settings, remains constant during the training and evaluation stages to ensure the repeatability of the experimental results. Second, it is assumed that the selected hyperparameters, including the number of training epochs, learning rate, and optimizer type, have a direct impact on the fine-tuning performance of BERT model.

Other parameters, including the batch size, dropout rate, and number of hidden layers of BERT, were kept constant based on the conventional settings of BERT-base model to reduce the computational complexity and focus the optimization process on the selected search space.

These assumptions determine the scope of the proposed model. Therefore, the innovation and main contribution of

this research should be interpreted as the optimization of the selected hyperparameters in BERT fine-tuning process using a genetic algorithm, not as a change in the internal architecture of BERT model.

4-5- Experimental Dataset

In this study, we used the public dataset "Cleaned Emotion Extraction Dataset from Twitter" available on the Kaggle platform. This dataset was originally collected from Twitter and contains 916,575 tweets categorized into three emotion classes: disappointment, happiness, and anger. The distribution of classes is approximately balanced, with 313,714 tweets in the disappointment class (34.2%), 301,871 tweets in the happiness class (32.9%), and 300,990 tweets in the anger class (32.9%).

This balanced distribution reduces the risk of bias towards a particular class in the model training process and justifies the use of the accuracy criterion as a fitness criterion in the GA-based optimization process.

The emotion labels were already present in the original dataset and were used without any manual relabeling. The dataset consists of three main columns: sentiment tags, cleaned tweet content, and original tweet content.

The cleaned content was used as the main input to BERT model, while the original tweet content was retained to maintain transparency and allow for verification of the accuracy of the preprocessing steps.

This is an even spread of cases across the categories of sentiment, which means that the training of the model will not be affected by any one emotion in particular. The balanced distribution across the categories of sentiment, which means that the training of the model will not be affected by any one emotion in particular.

Data preprocessing for this study was conducted through the use of automated Python scripts. The original dataset had a cleaned version of tweets already, but further preprocessing steps were applied in a planned manner so that the data could be input to the BERT model. Programmatic removal of URLs using regular expressions, user mentions (@) and hashtags (#) elimination, conversion to lowercase, and punctuation marks removal were among the steps taken. All the preprocessing operations were carried out in the same way throughout the whole dataset by using Python libraries like re (regular expressions), pandas, and NLTK. The data has been categorized in three major columns such as emotions, content, and original content. The emotions column includes the sentiment labels that are connected to every tweet. The content column has the preprocessed and cleaned version of tweets that are ready for the deep learning model to process. In the original content column, the raw tweets collected from Twitter are kept, which consist of emojis, hashtags, and symbols, and this column is the main source for preprocessing. All tweets were anonymized to ensure user privacy protection by removing user identifiers, and only publicly available content was used which is in keeping with the ethical research standards and data privacy regulations. The researchers followed ethical research practices while working with social media data. The data was all collected from tweets that were publicly available and this was done according to Twitter's terms of service and data usage policies. Regarding potential biases, we acknowledge that Twitter data may reflect demographic and geographic biases inherent to the platform's user base. The balanced class distribution reduces the risk of model bias toward any particular sentiment category. However, the applicability of findings to other social media platforms or offline contexts might be restricted and this should be considered when drawing conclusions. Table 4 provides a summary of the dataset structure and representative samples from each sentiment category.

Table 4: Dataset Content and Data Sample

Column	Description	Sample
Emotions	Refers to the emotion linked to the tweet text (e.g., "disappointed," "happy," "angry"), used as sentiment labels for data analysis	disappointed: 313,714; happy: 301,871; angry: 300,990
Content	Preprocessed tweet text, cleaned of noise and irrelevant elements, prepared for deep learning model processing and analysis	imagine if that reaction guy that called jj kfc saw this my man would ve started crying lmao
Original Content	The original and unprocessed tweets containing all essential details, including emojis, hashtags, and symbols, used as the primary data source	"@KSIOlajidebt imagine if that reaction guy that called JJ KFC saw this. my man would've started crying lmao"

4-6- Implementation Process

In this section, the implementation process used to train, validate, and optimize the proposed GA-BERT model is described. The implementation process includes the steps of data collection, data preprocessing, data partitioning, data tokenization using BERT, generating data set in PyTorch, model fine-tuning, model performance evaluation, and GA-based hyperparameter optimization. The general implementation steps are presented to ensure the reproducibility of the research, while the algorithmic innovation of this research specifically lies in the GA-based hyperparameter optimization process, which is described in Algorithm 1.

4-6-1- Step 1: Dataset Selection and Acquisition

The selection of an appropriate dataset for sentiment analysis is the basis of this research. This study used the Cleaned Emotion Extraction Dataset from Twitter, which is found on Kaggle. The selection of the data was determined by some robust criteria, which included the quality, balanced class distribution, relevance to the research, and the availability of pre-labeled sentiment categories. The data consists of a cumulative total of 916,575 tweets, which fall under the following three categories: Disappointed – 313,714 (34.2%), Happy – 301,871 (32.9%), and Angry – 300,990 (32.9%). The almost equal distribution was a major reason for the selection because it reduces the problems of class imbalance and allows the model to recognize the patterns in all sentiment categories equally. The dataset was first acquired by Kosweet using the Twitter API and it was done following Twitter's Developer Agreement and Policy, hence the data collection was done ethically. The dataset structure includes three columns: emotions (sentiment labels), content (cleaned text), and original content (raw tweets). The choice of having both raw and pre-cleaned versions gave the opportunity to implement custom preprocessing steps according to the BERT model requirements. Moreover, the dataset being publicly accessible on Kaggle is an assurance of this research's reproducibility, other researchers will thus be able to confirm and further develop the findings. Detailed dataset characteristics are presented in the Dataset section (Section 4.4).

4-6-2- Step 2: Data preprocessing

The text preprocessing was performed by means of automated Python scripts in order to keep the whole dataset uniform and repeatable. The dataset in question was one that had previously assigned sentiment labels (disappointed, happy, angry) that were used in the experiment without any changes. These labels were checked through the original data set and no further manual coding was needed. The pipeline was also automated via the Python libraries used to carry out regular expressions for pattern matching, pandas for manipulation of data, and NLTK for text processing utilities. The preprocessing operations on tweet text included extracting URLs through the use of regular expressions, deleting user mentions (the ones beginning with @) and hashtags (the ones beginning with #), making all text lowercase to treat it uniformly, and getting rid of punctuation signs while content of the tweets still being conveyed. The labels of the sentiments were converted to a numerical form through the application of scikit-learn's LabelEncoder. The labels included emotions of "disappointed," "happy," "angry" and were converted to integer labels "0," "1," "2" that were ideal for the learning process. This process of encoding was done systematically all over the dataset with the encoder fitted on the training data to keep proper data handling practices. All the preprocessing steps were done programmatically without any manual intervention, thus ensuring scalability and reproducibility of data preparation process.

4-6-3- Step 3: Data Division

The preprocessed data was split into two parts - training (80%) and evaluation (20%) which helped in the determination of the generalizability of the model and thus, overfitting was avoided.

4-6-4- Step 4: Data Tokenization

The BERT-base-uncased tokenizer was used to perform the text tokenization that transforms the text into sequences of tokens. The process of tokenization entails the use of the truncation process and padding that gives uniform length of sequences and aids uniform input to the model. This version is designed to process English texts and considers all letters in lowercase. This tokenizer was designed to maintain a balance between accuracy and efficiency and is widely used in natural language processing research. The tokenization process involves converting each text into a sequence of tokens and then applying complementary operations such as truncation and padding to ensure that the sequences are of equal length. These operations help the model process input texts in a consistent and homogeneous format and avoid defects that can arise due to differences in text length.

4-6-5- Step 5: Creating PyTorch Dataset Objects

The data after tokenization was transformed into objects of PyTorch Dataset to allow processing by batches in an efficient manner during model training and evaluation. The implementation of a custom dataset class was done, which inherits the torch.utils.data. Dataset base class, providing standardized methods for data access and manipulation. The custom Sentiment Dataset class contained the tokenized encodings along with the respective sentiment values that facilitated the definition of the necessary __getitem__ method supporting the return of individual samples as tensors along with the definition of the __len__ method providing the dataset's length. This allowed PyTorch's DataLoader module to perform the operations of batch generation, shuffling of data, along with the iteration over the dataset efficiently while the model was being trained. The training and validation split dataset objects were instantiated independently with the help of the generated encodings and labels of the tokens of the training and validation splits, in turn, creating the necessary data separation in the development pipeline.

4-6-6- Step 6: Model Training

The BERT-base-uncased model was loaded from the Hugging Face transformers library and configured for sequence classification with three output classes corresponding to the sentiment categories. Model training was executed on NVIDIA GeForce RTX 3090 GPU hardware to accelerate computation, with all tensors automatically transferred to CUDA device during processing. Training was performed using the AdamW optimizer with the learning rate determined by the genetic algorithm optimization process (detailed in Step 8). The model was trained using PyTorch DataLoader objects with a batch size of 16, iterating through the specified number of epochs as optimized by the GA. For each training batch, the following operations were executed: optimizer gradients were reset using zero_grad(), input tensors (input_ids, attention_mask) and labels were moved to GPU memory, a forward pass computed the model outputs and classification logits, cross-entropy loss was calculated between predictions and true labels, backward propagation computed gradients via loss.backward(), and the optimizer updated model parameters through optimizer.step(). Training progress was monitored through loss values across batches and epochs to ensure proper convergence. The model weights were preserved after training for subsequent evaluation and hyperparameter optimization iterations.

4-6-7- Step 7: Model Evaluation

The performance of the BERT model was evaluated on the validation dataset, which was the model's assessment to unseen data. The BERT model was set to the evaluation mode with the help of model.eval() function. Hence, dropouts were turned off and predictions were made consistently. In the process of evaluation, no gradients were calculated (by means of the torch.no_grad() context) and model weights were kept unaltered. For each validation batch, input tensors were transferred to GPU memory and passed through the model to generate predictions. The predicted classes were found out by applying the argmax function to the output logits of the model. The predictions were eventually accumulated for all validation batches, together with the actual classes, for comprehensive model assessment. Model performance was quantified using five key evaluation metrics: accuracy (overall percentage of correct predictions), precision (ratio of true positive predictions to all positive predictions), recall (ratio of true positives to all actual positives), F1-score (harmonic mean of precision and recall), and area under the ROC curve (AUC, measuring the model's ability to discriminate between classes). The above-mentioned calculation of the metrics has been performed by using the classification metrics provided by the Scikit-learn library. The evaluation results of this phase were considered for model convergence assessment and for taking decisions regarding the genetic algorithm optimization process in the subsequent iterations.

4-6-8- Step 8: Hyperparameter Optimization

Hyperparameter optimization using GA is used as a new approach to improve the performance of BERT model. Genetic algorithm, as an evolutionary optimization method, imitates natural selection processes to find an optimal combination of hyperparameters such as epochs, learning rate, and optimizer type. This optimization is based on iterated evaluations of the model's performance under different conditions, so that each time a different combination of parameters is tested and the results are measured using accuracy criteria. The optimization process is designed in such a way that its ultimate goal is to reduce the mistake and increase the model's accuracy at the same time. This approach based on genetic evolution is very effective and efficient due to its ability to search large and complex parameter spaces, especially in deep learning models such as BERT. Here, important parameters of the genetic algorithm such as population size, crossover rate, and mutation rate are finely tuned to optimize the search in the parameter space. The exact values of the genetic algorithm parameters used in this optimization step are given in Table 5. The fitness function was defined as (1 - accuracy) to minimize error and maximize accuracy. Although F1-score is often preferred for imbalanced datasets, our dataset largely has a balanced distribution of classes (disappointed: 313,714; happy: 301,871; angry: 300,990), with a maximum class imbalance ratio of approximately 1.04:1. For the balanced environment that was created, accuracy was proven to serve as an apt and efficiently calculated optimization function. The end result of the entire evaluation of the model would consider different aspects of the dataset including precision, recall, and the F1-score (as seen in Tables 9 and 10) to provide a comprehensive assessment of model performance across all classes.

Table 5: Parameter values of the genetic algorithm used in optimization

Parameter	Value Used
Population Size	10
Crossover Rate	0.8
Mutation Rate	0.2
Number of Generations	2

In the optimization process of BERT model, key hyperparameters including epochs, learning rate, and optimizer type are set. The permissible range of these hyperparameters is presented in Table 6.

Table 6: The permissible ranges of BERT model hyperparameters used in optimization

Hyperparameter	Range of Values
Number of Epochs	[10, 100]
Learning Rate	[1e-6, 1e-5]
Optimizer Type	[0 (AdamW), 1 (SGD)]

Here, each solution in the GA is considered as a specific combination of hyperparameters including epochs, learning rate,

and optimizer type. The algorithm searches for the optimal combination by evaluating the performance of each combination through the accuracy of the model. The goal of this optimization process is to maximize the accuracy of the model by minimizing the error (defined as 1 minus the accuracy).

The hyperparameters allowed would be set according to the conventional best practices defined in the BERT fine-tuning literature. The epoch range of 10-100 corresponds to suggestions for transformer-based models, where fewer than 10 epochs usually lead to underfitting, and more than 100 epochs seldom produce any performance improvements and rather increase the risk of overfitting [5][4]. The learning rate range of 1e-5to 1e-6 is in line with the generally accepted range for BERT fine-tuning since learning rates outside this range are likely to result in either unstable training or slow convergence [10].

The hyperparameter values selected for BERT model after optimization using the GA are given in Table 7. These optimized values have provided the best performance for the model in sentiment analysis.

Table 7: Optimized hyperparameter values of BERT Model after genetic algorithm optimization

Hyperparameter	Selected Value	Description
Number of epochs	90	Setting the number of epochs to 90 improves model learning, particularly for training on diverse datasets
Learning Rate	1e-5	A learning rate of 1e-5 is selected as an optimal learning rate to accelerate model convergence
Optimizer Type	AdamW (0)	The AdamW optimizer is chosen for its efficiency and enhanced performance in many deep learning models

The number of epochs is set to 90 so that the model can be well optimized on the training dataset without overfitting, which is suitable for models that do not require more iterations for convergence. The learning rate is chosen to be 1e-5, which is a relatively small value and allows the model to gradually update its weights and avoid large and sudden changes. Also, the AdamW optimizer is chosen as the model optimizer due to its high efficiency in adjusting the weights and its ability to avoid convergence problems, which allow the model to achieve more accurate results in sentiment prediction.

Ultimately, this optimization method increases the accuracy of BERT model and improves the hyperparameter tuning process in an efficient and effective manner. This improvement allows researchers to create models with better performance and higher generalizability, which can be useful in various fields of text data analysis and natural language processing applications. The mechanism of GA-based optimization, which constitutes the main algorithmic innovation of this research, is summarized in Algorithm 1.

4-6-9- Step 9: Final Training Using Optimal Hyperparameters

After the GA-based optimization process identified the best combination of hyperparameters, BERT model was fine-tuned using the resulting optimized configuration. The performance of the final model was then evaluated using the criteria of Accuracy, Precision, Recall, F1 score, and Area Under the ROC Curve (AUC) on the validation dataset.

The results were used to compare with the baseline models, including standard BERT (Vanilla BERT), LSTM, and CNN. The algorithmic innovation of this research is summarized in Algorithm 1, which specifically focuses on the GA-based hyperparameter optimization process.

The innovation of this research in the general sentiment classification pipeline does not include dataset preparation, preprocessing, tokenization, BERT model training, or model evaluation. These operations are standard implementation steps required for training transformer-based classifiers.

The innovative part of this research is the GA-based optimization mechanism that automatically searches and determines the best combination of hyperparameters used in fine-tuning BERT model. Therefore, the pseudocode presented in Algorithm 1 is limited only to the proposed process of GA-based hyperparameter optimization and includes chromosome representation, fitness evaluation, selection, crossover, mutation, and extraction of the optimal configuration of BERT model.

Table 8. Pseudocode of the Proposed algorithm for GA-based Hyperparameter Optimization

Algorithm 1. GA-based Hyperparameter Optimization for BERT Sentiment Classification

Input:

- Training dataset D_train
- Validation dataset D_val
- Search space H = {epochs, learning rate, optimizer type}
- Population size P
- Crossover rate Pc
- Mutation rate Pm
- Maximum number of generations G

Output:

- Optimized hyperparameter set H*
- Best validation accuracy Acc_best
- Final GA-optimized BERT model

1. Define the chromosome representation:
chromosome_i = [epochs_i, learning_rate_i, optimizer_i]
2. Initialize a population Pop = {chromosome_1, chromosome_2, ..., chromosome_P}
by randomly sampling each chromosome from the predefined hyperparameter search space H.
3. For each chromosome_i in Pop:
a. Decode chromosome_i into:
epochs_i
learning_rate_i

optimizer_i
b. Fine-tune BERT on D_train using the decoded hyperparameters.
c. Evaluate the fine-tuned BERT model on D_val.
d. Compute the fitness value:
fitness_i = 1 - Accuracy_i
4. Set the best chromosome as the chromosome with the minimum fitness value.
5. For generation = 1 to G do:
a. Select parent chromosomes from Pop based on their fitness values.
b. Apply crossover with probability Pc to generate offspring chromosomes.
c. Apply mutation with probability Pm to selected genes in the offspring chromosomes.
d. Ensure that all generated hyperparameter values remain within the predefined search space H.
e. For each offspring chromosome:
i. Decode the chromosome into BERT hyperparameters.
ii. Fine-tune BERT using the decoded configuration.
iii. Evaluate the model on D_val.
iv. Compute fitness = 1 - Accuracy.
f. Update the population using the newly evaluated offspring.
g. Update the best chromosome if a lower fitness value is obtained.
6. Return the best chromosome as the optimized hyperparameter set H*.
7. Fine-tune the final BERT sentiment classifier using H*.
8. Report the final performance metrics, including accuracy, precision, recall, F1-score, and AUC.

In Algorithm 1, the proposed part of the research is the GA-based optimization and search process used to fine-tune BERT model. The standard implementation steps, including data set loading, text preprocessing, data partitioning, tokenization, generating data set in PyTorch, model training, and model performance evaluation, are intentionally omitted from the pseudocode because these steps are not considered algorithmic innovations.
These steps are described separately in the implementation process section to ensure the reproducibility of the research, while Algorithm 1 focuses solely on the proposed optimization mechanism.

4-7- Analysis Methods

To evaluate the performance of the proposed hybrid algorithm, the developed model was compared with the CNN algorithms, LSTM, and Vanilla BERT. This comparison was made based on the main criteria for evaluating machine learning and deep learning models, which include precision, accuracy, recall, F-1 score, Receiver Operating Characteristic (ROC) curve, and the Area Under the Curve (AUC) [22, 23, 24].

- Accuracy: Accuracy is one of the main criteria for evaluating the performance of classification models and indicates the percentage of samples that are correctly classified. Accuracy is generally a suitable metric for data that has a balanced distribution between classes. Accuracy is calculated as follows [27],[30]:

$$\text{Accuracy}=\frac{\text{True Positive}+\text{True Negative}}{\text{True Positive}+\text{True Negative}+\text{False Positive}+\text{False Negative}} \quad (1)$$

In this equation, the number of correct predictions includes the number of samples that the model correctly classified into the corresponding classes. This metric is particularly suitable for multi-class classification problems where the data distribution is balanced.
- Precision: Precision is the percentage of correctly predicted positive samples. In other words, this metric shows how many of all samples predicted as positive are actually positive. This metric is especially important in situations where the cost of false positive predictions is high (e.g., in fraud detection). The equation for calculating precision is as follows [28],[30]:

$$\text{Precision}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Positive}} \quad (2)$$

where, the number of true positive predictions refers to the examples that are truly positive and correctly predicted by the model.
- Recall: Recall represents the percentage of true positives that are correctly identified by the model. In other words, this metric looks at the problem from the perspective of true positives and determines how many true positives the model correctly identified. Recall is important in situations where the cost of missing positives is high (e.g., in cancer diagnosis). Recall is calculated as follows [29]:

$$\text{Recall}=\frac{\text{True Positive}}{\text{True Positive}+\text{False Negative}} \quad (3)$$

here, the number of true positives includes all samples that actually belong to the positive category, whether correctly predicted or incorrectly.
- F-1 Score: A composite metric that measures the trade-off between precision and recall. This metric is particularly useful when there is a need to strike a balance between precision and recall, and we do not want either metric to be overly influential. The F-1 score is defined as the harmonic mean of precision and recall, and it is expressed as follows [28]:

$$\text{F1 Score}=\frac{2\times\text{Precision}\times\text{Recall}}{\text{Precision}+\text{Recall}} \quad (4)$$

This metric will be high when both precision and recall are high. Hence, the F-1 score is used as a comprehensive and useful metric to evaluate the performance of models in unbalanced classification problems.

- Calculating the area under the curve (AUC): The area under the curve is a numerical measure for evaluating the performance of binary classifier models that calculates the area under Receiver Operating Characteristic (ROC) curve. The value of the area under the curve varies between 0.5

and 1, where a value of 0.5 indicates random performance and a value of 1 indicates excellent model performance. In general, the higher the value of the area under the curve, the more capable the model is in distinguishing between classes. The area under the curve is a comprehensive indicator because it considers all possible thresholds and takes into account the fluctuations caused by threshold changes, so it can be an effective and accurate measure for evaluating the overall performance of classifier models.

5- Analysis of Results

The accuracy, precision, recall, F1-score, and AUC metrics were used to evaluate the proposed GA-optimized BERT model on both training and testing datasets. The performance results of the model are reported in Table 9.

Table 9: Evaluation results of the hybrid model using five key metrics

Evaluation Metric	Proposed Model (Training)	Proposed Model (Testing)
Accuracy	98.2%	98.1%
Precision	98.4%	98.2%
Recall	98.3%	98.1%
F1-Score	98.35%	98.15%
AUC	98.8%	98.5%

The purpose of Table 9 is to show model performance in both training and testing datasets which is needed to evaluate the generalization capability and to identify potential overfitting. The minimal performance gap between training and testing phases (98.2% vs 98.1% accuracy) clearly shows that the GA-optimal BERT model generalizes extremely well on new examples without any overfitting on the training examples.

This small difference of 0.1% in accuracy, along with similarly minimal variations in other metrics (precision: 0.2%, recall: 0.2%, F1-score: 0.2%, AUC: 0.3%), confirms that the hyperparameter optimization successfully balanced model complexity with generalization capability. Performance of the highest level on all examples of both steps validates the efficiency of genetic algorithm optimization approach. The training phase results (98.2% accuracy, 98.4% precision, 98.3% recall, 98.35% F1-score, 98.8% AUC) demonstrate the model's learning capacity, while the testing phase results (98.1% accuracy, 98.2% precision, 98.1% recall, 98.15% F1-score, 98.5% AUC) confirm robust performance on unseen data. These findings indicate that optimized hyperparameters allowed the model to extract significant patterns from the training data and at the same time to maintain strong predictive power against new samples.

Next, the performance of the proposed hybrid BERT model with genetic optimization was compared with other standard sentiment analysis algorithms, including vanilla BERT, LSTM, and CNN. This comparison is performed to demonstrate the superiority of the proposed model and accurately evaluate its performance against other well-known models. The results of this comparison are shown in Table 10.

Table 10: Comparative results between the proposed model and other algorithms

Evaluation metric	Proposed Model	BERT	LSTM	CNN
Accuracy	98.1%	92.8%	91.3%	90.5%
Precision	98.2%	91.2%	91.1%	90.2%
Recall	98.1%	92.0%	90.7%	90.1%
F1-Score	98.15%	91.6%	90.9%	90.0%
AUC	98.5%	93.2%	91.0%	89.6%

As shown in Table 10, the accuracy of the model indicates the percentage of test data samples that are correctly classified. The proposed model, with an accuracy of 98.1%, significantly outperform the other models. In comparison, the accuracies of Vanilla BERT, LSTM, and CNN models are 92.8%, 91.3%, and 90.5%, respectively. This significant difference in accuracy indicates that genetic optimization had a positive impact on the performance of BERT model. The precision indicates how many of all samples that are predicted as a specific positive class are correctly classified. The precision of the proposed model is 98.2%, which is higher than that of Vanilla BERT (91.2%), LSTM (91.1%), and CNN (90.2%). This indicates the proposed model is able to identify positive samples more accurately and has fewer false positive errors. The recall or sensitivity of the model indicates what percent of all true positive samples are correctly identified. The proposed model achieved 98.1% recall, outperforming BERT (92.0%), LSTM (90.7%), and CNN (90.1%), indicating its superior detection of true positives. The F-1 score provides a balance between precision and recall. This measure is especially useful when the data is unbalanced. The proposed model's F1-score of 98.15% outperforms BERT (91.6%), LSTM (90.9%), and CNN (90.0%), demonstrating its strength in detecting true positives while reducing false positives.

The area under the curve is a more comprehensive measure of the model's performance, which indicates its ability to discriminate between different classes. The proposed model with an area under the curve of 98.5% had the best performance compared to other models (BERT 93.2%, LSTM 91.0%, CNN 89.6%). The high area under the curve means the proposed model has a high ability to discriminate between samples and generally performs better in all classification scenarios. This comparison demonstrates that the proposed model significantly outperforms other models in precision, recall, F1-score, and especially in AUC. This optimal performance is the result of improving the model parameters through the use of genetic algorithm to fine-tune hyperparameters, which led to the development of a model with very high accuracy and efficiency in sentiment analysis.

One of the important tools for measuring the accuracy and precision of models in sentiment analysis is the confusion matrix. The results of examining the performance of different models in sentiment analysis using this tool are shown in Figure 6. This figure clearly shows how each model placed the samples in the correct or incorrect categories and reveals the difference in performance between them.

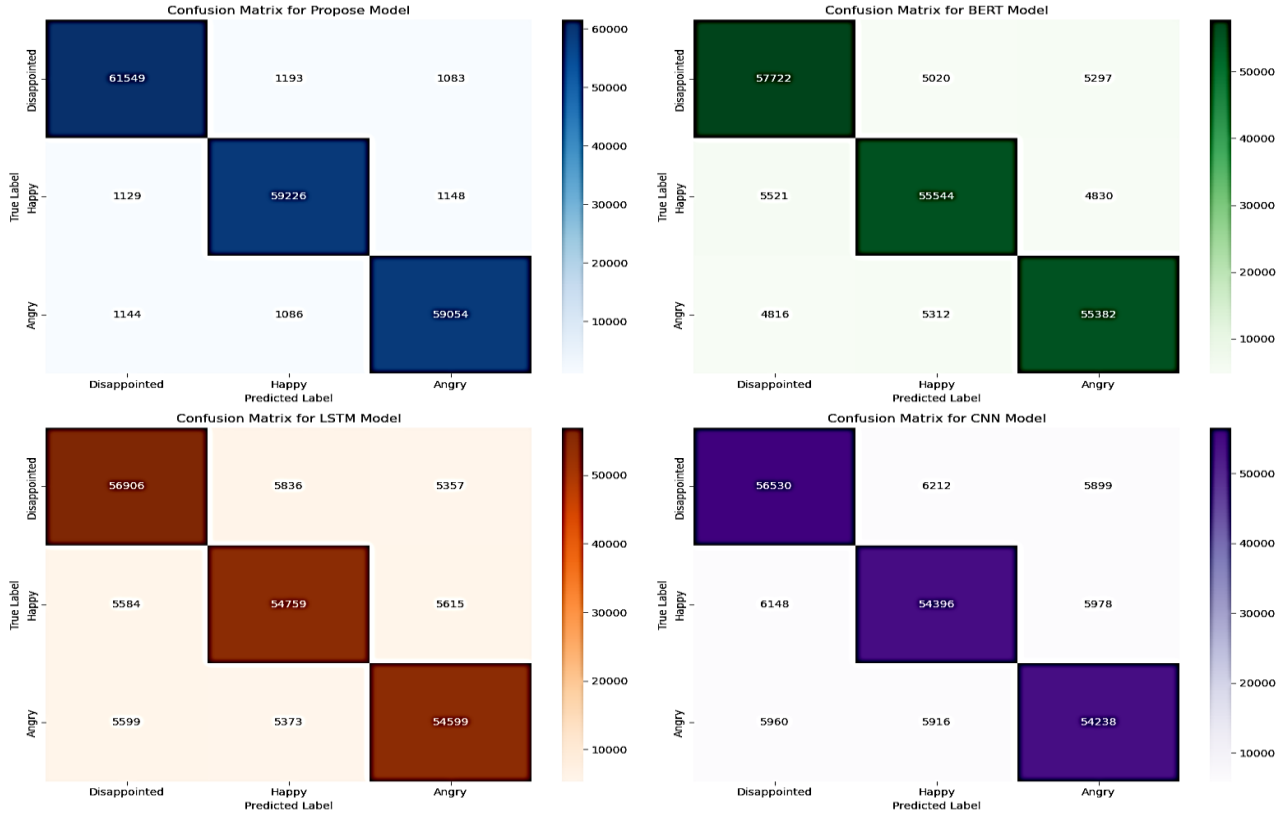


Fig. 6 Confusion matrices of four different models for sentiment analysis

The optimized BERT model (proposed model) using hyperparameter optimization showed excellent performance in all metrics including accuracy, precision, recall, and F-1 score. This optimized model was able to effectively reduce the type I and type II error rates, especially in the classes labeled "disappointed" and "angry", which indicates the high accuracy of this model in distinguishing between these emotions. In comparison, BERT model without optimization has more errors in recognizing the "happy" classes, which indicates the weakness of this model in the optimal tuning of hyperparameters. These differences demonstrate the importance of fine-tuning hyperparameters in improving the performance of deep learning models.

LSTM model had a lower performance than BERT models. It is particularly weak in detecting the "disappointed" class, with a higher type II error rate in this class. This may be due to the limitations of LSTM model in learning complex long-term dependencies and semantic differences between various classes. CNN model performs similarly to LSTM model, but performs slightly better in detecting the "happy" class. However, it had a higher error rate in detecting the "angry" class, which indicates the limitations of CNN in understanding long-term dependencies in text data.

Overall, these results show that the optimized BERT model using metaheuristic algorithms such as genetic algorithm can achieve significant performance improvements over more conventional models such as LSTM and CNNs, and even vanilla BERT. These findings confirm that fine-tuning hyperparameters can have a major impact on the accuracy and efficiency of deep learning models in sentiment analysis and can be used as an effective strategy to improve other deep learning models in this field.

6- Conclusion

This study addresses the problem of hyperparameters optimization of BERT model for sentiment classification

in Twitter texts using a genetic algorithm. In many previous studies, default values of hyperparameters or their manual tuning have been used, which can limit the performance of transformer-based sentiment classifiers. In order to overcome this limitation, this study proposed a GA-BERT optimization model, in which the genetic algorithm automatically searches for an effective combination of hyperparameters used in fine-tuning BERT model, including the number of training epochs, learning rate, and optimizer type.

The experimental results showed that BERT model optimized with the genetic algorithm performed very well in sentiment classification. The model achieved a precision of 98.1%, a precision of 98.2%, a recall of 98.1%, an F1 score of 98.15%, and an area under the ROC curve (AUC) of 98.5%.

Also, compared with standard BERT (Vanilla BERT), CNN, and LSTM models, the proposed model outperformed all the main performance evaluation criteria. These findings indicate that the GA-based hyperparameter optimization can improve the effectiveness of BERT model fine-tuning process in sentiment classification based on Twitter texts.

Despite the results obtained, this study has several limitations. First, the tests were conducted on a Twitter-based dataset consisting of three emotion classes, namely disappointment, happiness, and anger; therefore, the generalizability of the obtained optimal configuration to other platforms, application domains, languages, or emotion classification systems requires further investigation.

Second, the optimization process focused on only three selected hyperparameters, while other influential parameters, including Batch Size, Dropout Rate, Weight Decay, Maximum Sequence Length, and Learning-Rate Scheduling, were considered constant.

Third, the GA-based optimization requires repeated execution of BERT model fine-tuning process, which increases the computational cost and training time.

Future research should extend the proposed GA-BERT optimization model in several directions. First, the hyperparameter search space can be expanded to include batch size, dropout rate, weight decay, maximum sequence length, and learning-rate scheduling strategies. Second, the proposed GA-based optimization strategy should be compared with other automated hyperparameter optimization methods, such as Bayesian optimization, particle swarm optimization, random search, grid search, and differential evolution, to evaluate its relative performance and computational efficiency. Third, the generalizability of the optimized model should be assessed using cross-domain, multilingual, and non-Twitter sentiment datasets. Fourth, future studies should investigate computationally efficient variants of the proposed approach, including early stopping, parallel fitness evaluation, lightweight transformer models such as DistilBERT or ALBERT, and surrogate-assisted optimization. Finally, alternative fitness functions, including macro-F1, weighted-F1, AUC, and multi-objective criteria, should be examined, particularly for imbalanced or heterogeneous sentiment datasets.

References

[1] M. Li, L. Chen, J. Zhao, and Q. Li, “Sentiment analysis of Chinese stock reviews based on BERT model,” Applied Intelligence, vol. 51, no. 7, 2021, doi: 10.1007/s10489-020-02101-8.

[2] K. Naithani and Y. P. Raiwani, “Realization of natural language processing and machine learning approaches for text-based sentiment analysis,” Expert Syst, vol. 40, no. 5, 2023, doi: 10.1111/exsy.13114.

[3] A. Mewada and R. K. Dewang, “SA-ASBA: a hybrid model for aspect-based sentiment analysis using synthetic attention in pre-trained language BERT model with extreme gradient boosting,” Journal of Supercomputing, vol. 79, no. 5, 2023, doi: 10.1007/s11227-022-04881-x.

[4] X. Li, X. Wang, and H. Liu, “Research on fine-tuning strategy of sentiment analysis model based on BERT,” in 2021 IEEE 3rd International Conference on Communications, Information System and Computer Engineering, CISCE 2021, 2021. doi: 10.1109/CISCE52179.2021.9445882.

[5] R. Qasim, W. H. Bangyal, M. A. Alqarni, and A. Ali Almazroi, “A Fine-Tuned BERT-Based Transfer Learning Approach for Text Classification,” J Healthc Eng, vol. 2022, 2022, doi: 10.1155/2022/3498123.

[6] J. Martinez-Gil, “Optimizing readability using genetic algorithms,” Knowl Based Syst, vol. 284, 2024, doi: 10.1016/j.knosys.2023.111273.

[7] D. Choudhury and T. Acharjee, “A novel approach to fake news detection in social networks using genetic algorithm applying machine learning classifiers,” Multimed Tools Appl, vol. 82, no. 6, 2023, doi: 10.1007/s11042-022-12788-1.

[8] A. Bello, S. C. Ng, and M. F. Leung, “A BERT Framework to Sentiment Analysis of Tweets,” Sensors, vol. 23, no. 1, 2023, doi: 10.3390/s23010506.

[9] U. Sirisha and B. S. Chandana, “Aspect based Sentiment and Emotion Analysis with ROBERTa, LSTM,” International Journal of Advanced Computer Science and Applications, vol. 13, no. 11, 2022, doi: 10.14569/IJACSA.2022.0131189.

[10] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “Bert: Bidirectional encoder representations from transformers,” arXiv preprint arXiv:1810.04805, p. 15, 2018.

[11] H. Liu et al., “Use of BERT (Bidirectional Encoder Representations from Transformers)-Based Deep Learning Method for Extracting Evidences in Chinese Radiology Reports: Development of a Computer-Aided Liver Cancer Diagnosis Framework,” J Med Internet Res, vol. 23, no. 1, 2021, doi: 10.2196/19689.

[12] L. B. Cesar, M. Á. Manso-Callejo, and C. I. Cira, “BERT (Bidirectional Encoder Representations from Transformers) for Missing Data Imputation in Solar Irradiance Time Series †,” Engineering Proceedings, vol. 39, no. 1, 2023, doi: 10.3390/engproc2023039026.

[13] A. Areshey and H. Mathkour, "Transfer Learning for Sentiment Classification Using Bidirectional Encoder Representations from Transformers (BERT) Model," Sensors, vol. 23, no. 11, 2023, doi: 10.3390/s23115232.

[14] L. Yang and A. Shami, "On hyperparameter optimization of machine learning algorithms: Theory and practice," Neurocomputing, vol. 415, 2020, doi: 10.1016/j.neucom.2020.07.061.

[15] A. Pfob, S. C. Lu, and C. Sidey-Gibbons, "Machine learning in medicine: a practical introduction to techniques for data pre-processing, hyperparameter tuning, and model comparison," BMC Med Res Methodol, vol. 22, no. 1, 2022, doi: 10.1186/s12874-022-01758-8.

[16] S. Katoch, S. S. Chauhan, and V. Kumar, "A review on genetic algorithm: past, present, and future," Multimed Tools Appl, vol. 80, no. 5, 2021, doi: 10.1007/s11042-020-10139-6.

[17] M. Dehghani, E. Trojovská, and P. Trojovský, "A new human-based metaheuristic algorithm for solving optimization problems on the base of simulation of driving training process," Sci Rep, vol. 12, no. 1, 2022, doi: 10.1038/s41598-022-14225-7.

[18] H. Su et al., "RIME: A physics-based optimization," Neurocomputing, vol. 532, 2023, doi: 10.1016/j.neucom.2023.02.010.

[19] M. Pota, M. Ventura, R. Catelli, and M. Esposito, "An effective bert-based pipeline for twitter sentiment analysis: A case study in Italian," Sensors (Switzerland), vol. 21, no. 1, 2021, doi: 10.3390/s21010133.

[20] A. S. Talaat, "Sentiment analysis classification system using hybrid BERT models," J Big Data, vol. 10, no. 1, 2023, doi: 10.1186/s40537-023-00781-w.

[21] M. Pota, M. Ventura, H. Fujita, and M. Esposito, "Multilingual evaluation of pre-processing for BERT-based sentiment analysis of tweets," Expert Syst Appl, vol. 181, 2021, doi: 10.1016/j.eswa.2021.115119.

[22] H. Y. Lin and T. S. Moh, "Sentiment analysis on COVID tweets using COVID-Twitter-BERT with auxiliary sentence approach," in Proceedings of the 2021 ACMSE Conference - ACMSE 2021: The Annual ACM Southeast Conference, 2021. doi: 10.1145/3409334.3452074.

[23] S. Palani, P. Rajagopal, and S. Pancholi, "T-BERT--Model for Sentiment Analysis of Micro-blogs Integrating Topic Model and BERT," arXiv preprint arXiv:2106.01097, 2021.

[24] A. Chiorrini, C. Diamantini, A. Mircoli, and D. Potena, "Emotion and sentiment analysis of tweets using BERT," in CEUR Workshop Proceedings, 2021.

[25] R. Anggrainingsih, G. M. Hassan, and A. Datta, "BERT based classification system for detecting rumours on Twitter," IEEE Access, vol. 11, pp. 80207-80217, 2023., 2021, doi: https://doi.org/10.1109/ACCESS.2023.3299858.

[26] Y. Wu, Z. Jin, C. Shi, P. Liang, and T. Zhan, "Research on the Application of Deep Learning-based BERT Model in Sentiment Analysis," arXiv preprint arXiv:2403.08217, 2024.

[27] H. A. Samir, L. Abd-Elmegid, and M. Marie, "Sentiment analysis model for Airline customers' feedback using deep learning techniques," 2023. doi: 10.1177/18479790231206019.

[28] S. Kardakis, I. Perikos, F. Grivokostopoulou, and I. Hatzilygeroudis, "Examining attention mechanisms in deep learning models for sentiment analysis," Applied Sciences (Switzerland), vol. 11, no. 9, 2021, doi: 10.3390/app11093883.

[29] S. Seo, C. Kim, H. Kim, K. Mo, and P. Kang, "Comparative Study of Deep Learning-Based Sentiment Classification," IEEE Access, vol. 8, 2020, doi: 10.1109/ACCESS.2019.2963426.

[30] M. Sedighimanesh, A. Sedighimanesh, and H. Zandhessami, "Optimizing Hyperparameters for Customer Churn Prediction with PSO-Enhanced Composite Deep Learning Techniques," Journal of Information Systems and Telecommunication (JIST), vol. 13, no. 50, pp. 91–110, 2025, doi: 10.61882/jist.48088.13.50.91.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 117-128
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Improving the Accuracy of Motor Imagery in BCI for the Movement of Artificial Prostheses used for Disabled People

Isaac Jahanbakhshi[1*], Shaghayegh Niavarani[2] , Fatemeh Shahbazi[3], Farahnaz Abdi[4]

[1].Department of Computer science, Central Tehran Branch, Islamic Azad University, Tehran, Iran
[2] Department of Computer science, Science and Research Branch, Islamic Azad University, Tehran, Iran
[3] Department of Computer science, Kermanshah Branch, Jahad University, Kermanshah, Iran
[4] Department of Computer science and Software Engineering, University of Western Australia, Perth, Australia

Received: 05 Oct 2024/ Revised: 04 Jul 2026/ Accepted: 18 Aug 2026

Abstract
A brain-computer interface (BCI) allows users to communicate directly with an external device, such as a computer, using brain signals. These systems involve signal acquisition, temporal and spatial filtering, feature engineering, and classification before transmitting the control signal to an external device. Brain-computer interfaces based on motor imagery (MI-BCI) hold significant potential for applications in motor enhancement and rehabilitation. However, the control capabilities of MI-BCI can vary among individuals. In this article, we present a motor imagery model that leverages the power of deep learning for feature extraction and optimization methods for feature selection, based on individuals' electroencephalogram (EEG) signals. For this purpose, we employed the convolutional neural network (CNN), the golden eagle optimization (GEO) method, and three hybrid classification models to achieve high accuracy in classifying EEG signals. The innovative classification method presented in the third phase of the proposed method has high flexibility and significant accuracy, which is presented for the first time. This method can be used in all matters of prediction and detection of phenomena to increase accuracy and high scalability. The experimental results demonstrate that the proposed model significantly outperforms baseline approaches across all key performance indicators. In terms of precision, the method achieves an improvement of approximately 12–18% over standard CNN and nearly 20% compared to conventional classifiers. For recall, it yields 10–15% higher performance than CNN and 18–22% higher than classical machine learning methods. Finally, the F-measure results indicate a 12–15% improvement over CNN and about 20% over traditional classifiers. These consistent enhancements confirm the robustness and scalability of the proposed approach for motor imagery-based EEG classification and highlight its potential for practical applications in rehabilitation and prosthetic control systems.

Keywords: Brain-Computer Interface; Motor Imagery; CNN; GEO.

1- Introduction

An electroencephalogram (EEG) is a signal that records brain activity and provides a wealth of physiological information to physicians. The data extracted from the EEG signal is of significant importance in clinical research and medical care. One of the most effective and promising applications of the electroencephalogram is its use in brain-computer interface (BCI) technology, which has the potential to greatly contribute to technological advancement and shape the future. A brain-computer interface is a device that serves as an intermediary between the brain and another object, transmitting brain information to a computer and then to an external device. In essence, a BCI system comprises a set of sensors and a signal processing unit that forms a direct connection between the brain and an external interface, converting an individual's brain activity into a series of communication or control signals [1, 2].

In motor imagery-based BCI (MI-BCI), users can control or move virtual or real objects without physically touching a mouse, keyboard, or other objects, using only their mind in conjunction with the computer, which acts as the interface between the brain and the object. These interfaces come in various types and are used in different ways. To operate such a system, electrical activity and brain waves must first be recorded using brain wave recording devices. Electroencephalography is commonly chosen for this purpose due to its high accuracy, low cost,

✉ **Isaac Jahanbakhshi**
isaac.jahanbakhshi@iau.ac.ir

and ease of use [3]. Signal recording with an EEG device relies on a non-invasive method, which stands in contrast to other signal recording techniques that require implanting a chip or electrode inside the human brain. In the non-invasive approach, EEG electrodes are placed on the surface of the scalp to measure the electric field produced by neuronal activity. Subsequently, these waves are analyzed to extract relevant features, which can be used to infer the user's intended activity.

Although the speed and accuracy of these systems have not yet reached an optimal level, and BCI-related equipment requires further research, EEG signals are prone to errors due to various types of noise. One of the most common sources of artifacts is the electrocardiography (ECG) signal and noise from the power supply of the equipment. To achieve high accuracy in diagnosing and processing EEG signals, it is essential to generate signals that are free from noise and artifacts. Therefore, their removal is of great importance [4]. Motion imagery (MI) is a mental representation of motor behavior and a widely used paradigm in electroencephalogram-based brain-computer interface (BCI) systems. Currently, due to the extensive use of BCI systems that rely on movement perception to assist patients with physical disabilities, rehabilitate injured individuals, control virtual reality environments, play computer games, and more, enhancing the performance of these systems is crucial and necessary [5].

Various conditions can cause damage to the neuromuscular system, which enables the brain to communicate with and control the external environment. Diseases such as amyotrophic lateral sclerosis (ALS), stroke, brain and spinal cord injuries, cerebral palsy, muscular dystrophy, and multiple sclerosis are examples where the neural pathways for muscle control are impaired. In severe stages of these diseases, the affected individual may lose all voluntary movements. Even involuntary actions such as eye movements and breathing may become impossible. Patients experiencing this are referred to as having locked-in syndrome. In the absence of methods to physiologically repair the damage caused by these diseases, there are three approaches to restore normal function in patients:

1. Enhancing the capabilities of the remaining neuromuscular pathways
2. Restoring lost function by bypassing the damaged areas in the neural pathways
3. Providing a new, non-muscular communication route for the brain, through which it can directly send control signals and instructions to the external environment.

The feature that distinguishes these brain-computer interfaces from other communication methods is the elimination of the need for any overt bodily movement to transmit information. Ideally, a person should be able to sit still and convey their intentions by focusing on specific thoughts and generating appropriate brain waves [6]. The performance of motor imagery is easily compromised by noise and extraneous information present in multichannel EEG recordings. To address this issue, numerous channel selection methods based on temporal and spatial features have been developed. However, these temporal and spatial characteristics do not accurately capture changes in oscillatory EEG power.

Variations in electroencephalographic (EEG) patterns among individuals necessitate the collection of extensive amounts of labeled data for each person to train models, which lengthens the calibration process. Another significant challenge in enhancing brain-computer interface systems is reducing the number of features extracted from brain signals. Decreasing the number of features can diminish noisy data, lower the risk of overfitting, reduce the memory needed for data storage, enhance classifier performance, provide faster models, and decrease computational complexity [7], [8]. To achieve this objective, various feature selection methods have been employed, which involves addressing current challenges and devising solutions.

In this paper, we will introduce a method designed to improve the accuracy of EEG classification for amputees or individuals with physical disabilities using brain-computer interface systems. This method utilizes a feature extraction mechanism based on deep learning and optimization techniques. The proposed study introduces several methodological innovations that collectively enhance the accuracy, robustness, and practicality of motor imagery-based brain–computer interfaces (MI-BCIs) for prosthetic control in individuals with movement impairments.

- **Hybrid Entropy-Based Preprocessing for Outlier Reduction**: Unlike many previous works that primarily rely on raw EEG preprocessing, this study employs a dual entropy–variance strategy to identify and eliminate outlier EEG segments. This innovation reduces noise introduced by abnormal recordings or measurement errors, thereby improving the quality of training data and ensuring more reliable model convergence.
- **Intermediate Feature Extraction via CNN**: Instead of limiting CNNs to final classification tasks, this work leverages intermediate representations obtained from convolution and pooling layers. These features retain both temporal and spatial dependencies of EEG signals, reducing the reliance on handcrafted features. This approach combines the advantages of deep automatic feature learning with interpretability and flexibility in downstream classification.

- **Golden Eagle Optimization (GEO) for Feature Selection**: To address the challenges of high-dimensional and imbalanced EEG data, a novel application of the Golden Eagle Optimization algorithm is introduced. GEO enables efficient selection of discriminative features by balancing accuracy with feature set compactness. This not only reduces computational complexity but also ensures better generalization, outperforming conventional feature selection and evolutionary optimization techniques.
- **Multi-Classifier Fusion with Semi-Supervised Learning**: A unique three-branch classification framework is proposed:
    - Fuzzy Clustering-Based Semi-Supervised Model, which estimates class probabilities and mitigates uncertainty in EEG classification.
    - Hybrid Voting Classifier, combining Decision Tree, Naïve Bayes, and XGBoost to achieve a balance between interpretability, robustness, and high accuracy.
    - Instance-Based KNN Classifier, which complements the ensemble by capturing local feature similarities.

This study is guided by the overarching research question: "How can the integration of entropy-based preprocessing, CNN-driven intermediate feature extraction, optimization via the Golden Eagle Optimization (GEO) algorithm, and multi-classifier fusion improve the accuracy, robustness, and real-time applicability of motor imagery-based brain–computer interfaces (MI-BCIs) for prosthetic control in individuals with movement impairments?" To address this question, the following hypotheses are formulated:

- H1: Entropy- and entropy variance–based preprocessing will reduce the impact of noisy and outlier EEG segments, thereby improving the quality of input data and enhancing classification stability.
- H2: Intermediate feature extraction using convolutional neural networks will capture spatio-temporal dependencies in EEG signals more effectively than handcrafted features, leading to superior discriminative power.
- H3: The application of the Golden Eagle Optimization (GEO) algorithm for feature selection will outperform conventional feature selection methods by achieving higher accuracy with a reduced feature set, while also lowering computational complexity.
- H4: A multi-classifier fusion framework—combining fuzzy clustering-based semi-supervised classification, a hybrid ensemble of Decision Tree, Naïve Bayes, and XGBoost, and an instance-based KNN classifier—will achieve higher accuracy and robustness compared to any single classifier.

The remainder of the paper is organized as follows: Section 2 discusses related work, Section 3 outlines a proposed strategy to enhance the accuracy of BCI systems based on motor imagery, Section 4 presents an analysis, and Section 5 offers conclusions.

2- Related Works

Arpaia et al. [9] have developed a technique for enhancing the selection of EEG signals in brain-computer interfaces that utilize motion imagery. This advancement aims to improve the real-time functionality and adaptability of BCI systems, while also increasing user comfort. The study further seeks to diminish the variability and interference commonly encountered in MI-BCI due to the extensive array of EEG channels. In another study [10], efforts were made to construct a robust classifier with a high generalization capacity for BCIs that are based on motor imagery (MI). This study introduced a framework known as cluster decomposition-based ensemble learning (CDECL) specifically for classifying MI-based BCIs. Ensemble learning was approached as a multi-objective optimization challenge, leading to the development of a fruit fly-inspired multi-objective binary optimization algorithm utilizing stochastic fractal search to address the problem.

In research presented by [11], a novel methodology termed Knowledge-Based Support Matrix Machine (KL-SMM) is described. This method aims to enhance classification accuracy when dealing with a limited quantity of labeled EEG data in the target domain. The strength of KL-SMM lies in its matrix learning machine capability, which directly processes the structural data from EEGs in matrix form. Furthermore, KL-SMM efficiently leverages the minimal labeled EEG data from the target domain while incorporating pre-existing model knowledge from the source domain during its learning phase. Bispectral analysis is an established approach for extracting complex, non-linear, and non-Gaussian information from signals of similar nature. In light of this, Jin et al. [12] put forward a bispectral channel selection (BCS) strategy tailored for BCIs based on motion perception. Their proposed technique employs sum of logarithmic amplitudes (SLA) and first-order spectral moment (FOSM) derived from bispectral analysis to pinpoint relevant EEG channels without relying on supplementary data.

In an exploration detailed in [13], researchers scrutinized the performance of pianists interacting with an MI-based BCI system, contrasting it with that of a non-pianist control group. To process the EEG signals, various machine learning techniques such as common spatial

patterns (CSP) and linear discriminant analysis (LDA) were utilized for feature extraction and classification. The findings indicated that pianists exhibited superior BCI control through movement imagination, achieving a higher score in the final test (74.69%) compared to the control group (63.13%).

Idowu et al. [14] investigated the efficiency of nature-inspired algorithms including Ant Colony Optimization (ACO), Genetic Algorithm (GA), Cuckoo Search Algorithm (CSA), and Modified Particle Swarm Optimization (M-PSO) in analyzing EEG and ECoG datasets. Their findings demonstrated that M-PSO, particularly with a minimal set of selected features (SF), displayed impressive efficiency and converged rapidly within a low number of iterations.

The Common Spatial Pattern (CSP) and its extension into a filter bank have been demonstrated to be effective in extracting spatial and spectral features from EEG signals that are associated with the perception of movement. To enhance the specificity of CSP-derived features, Liu et al. [15] introduced a framework known as the Distinct Spatial-Spectral Feature Learning Neural Network (DSSFLNN) tailored for brain-computer interfaces that are based on motion perception. The initial phase within this framework involves the extraction of Filter Bank Common Spatial Pattern (FBCSP) features from the raw EEG data. Subsequently, modules for pressure and excitation are employed to refine the CSP attributes. Following this, a parallel convolutional neural network module is utilized to discern and learn unique spatio-spectral characteristics. These features are then processed through a fully connected layer for the purpose of classification.

In research carried out by Dhiman [16], a feature generation approach for EEG signals in Motor Imagery-based Brain-Computer Interface (MI-BCI) systems employs wavelet packet decomposition (WPD) and Approximate Entropy (ApEn). Due to the presence of redundant features within the extracted EEG signal data, a feature selection process is implemented using an enhanced version of binary gray wolf optimization, termed modified binary gray wolf optimization (MBGWO). Classification is subsequently conducted using the K-nearest Neighbors (KNN) algorithm. The findings indicate that this proposed methodology significantly enhances classification accuracy, particularly in BCI applications designed to detect MI activities in individuals with physical impairments.

Signal Decomposition (SD) serves as a fundamental technique for isolating components or intrinsic mode functions from EEG signals, which is crucial for the advancement of brain-computer interface systems. Yu et al. [17] have applied Empirical Fourier Decomposition (EFD) along with its refined iteration, Improved EFD (IEFD), to process EEG signals for mental and motor imagery (MeI). This approach incorporates multiscale principal component analysis (MSPCA) to reduce noise in EEG data. During the classification stage, features from both time and frequency domains are extracted and categorized using a feed-forward neural network (FFNN).

In the context of controlling robotic arms via brain-computer interfaces, Ak et al. [18] have put forth a novel classification strategy for EEG signals that align with motor imagery (MI) tasks in BCI frameworks. This technique involves transforming EEG data into spectrographic images which are then processed through deep learning techniques, feeding 400 images per task into GoogLeNet. After training the model, the system's efficacy was evaluated based on its ability to accurately execute imagined movements such as up, down, left, and right, to control a robotic arm. The outcomes demonstrated that the robotic arm was able to perform these movements with a high degree of precision, reaching an accuracy rate of 90%.

Khademi et al. [19] introduced a trio of composite models for categorizing EEG signals within MI-oriented BCI systems. These composite models integrate convolutional neural networks (CNNs) with long short-term memory (LSTM) networks. To address the challenge of small data sets, these models employ a transfer learning approach and data augmentation techniques, utilizing two robust pre-trained CNNs, namely ResNet-50 and Inception-v3. MI-BCIs rely on ML algorithms that necessitate comprehensive signal processing and feature engineering to identify variations in sensorimotor rhythms (SMR). Recent studies have focused on deep learning (DL) approaches for EEG classification due to their ability to autonomously extract spatio-temporal features from the signals. The disparity in BCI effectiveness among various user groups, particularly those who are less efficient, is problematic; this refers to some users' difficulties in producing the requisite SMR patterns for the BCI classifier to recognize.

Tibrewal et al. [20] examined the capacity of DL models to discern MI features in users who struggle with efficiency. Within the ML framework, Common Spatial Patterns were utilized for feature extraction, followed by the application of a Linear Diagnostic Analysis (LDA) model for binary classification. For the DL-based method, a CNN model was directly applied to raw EEG data. Meng et al [21] introduced an algorithm for classifying motion imagery using a recurrent graph convolutional neural network. Initially, EEG signals are processed to enhance signal strength during the test phase. Subsequently, features from both time and frequency domains are extracted to form the regression graph's feature pattern. A neural network is then constructed for precise detection of left and right movements.

Cherloo et al. [22] developed an advanced technique named Ensemble Regularized Joint Spatial-Spectral Pattern (Ensemble RCSSP), which diminishes the

likelihood of overfitting in the Joint Spatial Pattern algorithm. Wang et al. [23] proposed an unsupervised deep transfer learning method to navigate the prevailing constraints of BCI systems, leveraging transfer learning concepts for classifying EEG signals related to motor imagery. This method incorporates a Euclidean space data balancing technique to equalize the covariance matrices of EEG data from both source and target domains within Euclidean space. Thereafter, Common Spatial Pattern is employed for feature extraction from the aligned data matrix, and a Deep Convolutional Neural Network is used for classifying EEG signals. Li et al. [24], introduced a feature fusion algorithm based on neural networks that merges a CNN with an LSTM network. In this design, CNN and LSTM operate in parallel—CNN is responsible for spatial feature extraction while LSTM handles temporal feature extraction. A flattening layer follows the middle layer's feature extraction by the convolutional layers. Subsequently, all features converge in a fully connected layer to heighten classification precision.

Despite notable advancements in motor imagery-based brain–computer interfaces (MI-BCIs), current systems still face substantial challenges that restrict their applicability in the real-world control of prosthetic devices. Although sophisticated feature extraction and optimization methods—such as bispectral analysis, wavelet packet decomposition, and swarm-inspired algorithms—have improved performance in controlled experimental environments, their generalization capacity across heterogeneous users remains limited. A particularly critical issue is the disparity in BCI efficiency, as many users, especially individuals with motor impairments, are unable to consistently generate distinct sensorimotor rhythms. This variability undermines the reliability of MI-BCIs, thereby limiting their practicality in clinical and assistive scenarios.

Another persistent gap lies in the dependence on large volumes of labeled EEG data, which are costly and time-intensive to obtain. While recent methods, including knowledge-based classifiers and unsupervised transfer learning strategies, attempt to mitigate this constraint, their effectiveness remains hampered by inter-subject variability, non-stationarity of EEG signals, and the risk of domain mismatch. Furthermore, deep learning architectures such as CNN–LSTM hybrids and graph convolutional networks have shown promise in spatio-temporal feature extraction but require extensive computational resources and often struggle with overfitting when applied to limited or imbalanced datasets.

Motivated by these limitations, this study seeks to advance MI-BCI research by developing a robust and adaptive framework that combines optimized feature selection, domain adaptation strategies, and lightweight deep learning architectures. The proposed work aims to enhance classification accuracy and ensure real-time responsiveness. Ultimately, this research aspires to contribute toward reliable, user-friendly brain-controlled prostheses that can significantly improve the independence and quality of life of individuals with motor disabilities.

3- Proposed Method

The use of neural networks to process biological signals such as electroencephalogram has been used many times in various researches [25,26]. In this study, we aim to acquire valuable insights into motor imagery for individuals with movement impairments by analyzing existing data. To this end, we have recorded brainwave activity using an EEG device from 30 participants with disabilities, encompassing both male and female genders, ranging in age from 20 to 50 years. We have identified seven distinct movements, labeled Q1 to Q7, which are:

1. Move up
2. Move down
3. Move left
4. Move right
5. Stop movement
6. Grasp
7. Release

Electroencephalographic data is gathered using 21 electrodes placed on the scalp. Participants are instructed to watch a video that depicts a series of movements and pauses, and to imagine themselves performing the movements shown. Subsequently, during a specific time frame, we record the EEG signal values of each participant as they visualize the movement. Feature selection is a powerful technique for addressing classification challenges in datasets with imbalanced classes. It can be approached as a multi-objective optimization problem with the goal of identifying a small subset of features that yields high classification accuracy. Traditional methods tend to focus on extracting an optimal solution without considering the variability within the solution space. In our proposed approach, we plan to address issues such as outlier data, high computational complexity, and the necessity for precise prediction accuracy by employing a novel method.

3-1- Data preparation

In the first phase of this project, to generate fundamental features from EEG data, individual signal records are analyzed using two methods: entropy and entropy variance. A lower entropy value indicates that there is a degree of regularity in the brain signal data, whereas a higher value suggests disorder. By identifying and addressing these measures, we can minimize the error introduced by outlier data, which may result from measurement errors, the subject's abnormal condition

during signal recording, or other unexpected factors. Eliminating such data from the brain signal records enables the model to be trained on more precise data.

3-2- Feature extraction using convolutional neural network

In the second phase of analyzing people's movement perception, we will employ a convolutional neural network (CNN) to extract intermediate features and account for the temporal aspects of users' brain signals, which involves considering two-dimensional data. At this stage, the EEG data from each individual's previous time periods will be inputted as a matrix into the CNN-based architecture. The CNN represents a pivotal point between feature extraction and classification. Utilizing a CNN eliminates the need to design feature descriptors manually. This type of neural network has recently demonstrated impressive performance in deep learning tasks. The rationale for choosing CNNs in our proposed method is their reduced need for pre-processing compared to other techniques. This implies that the network learns to identify patterns that previously required manual intervention in earlier methods. This independence from prior knowledge and human intervention is a significant advantage of CNNs. The CNN designed for feature learning comprises three convolutional layers and three pooling layers, with the convolution and pooling layers arranged in an alternating sequence (repeating three times with one convolution layer followed by one pooling layer).

3-3- Features Selection using the golden eagle optimization

Another innovation is that we extract the intermediate features obtained after the convolution and pooling layers. Following optimization with the Golden Eagle Optimization (GEO) method, we input these features into several different classifiers to achieve accurate and semi-supervised classification performed on the optimized intermediate features. Since selecting relevant and effective features to increase the accuracy of motion imagery is a time-consuming and complex process, we will employ the GEO in the third phase of our current plan.

The Golden Eagle Optimization (GEO) algorithm provides a distinctive advantage over many traditional metaheuristic methods due to its ability to effectively balance global exploration and local exploitation. Unlike algorithms such as Genetic Algorithms or Particle Swarm Optimization, which often suffer from premature convergence or require complex parameter tuning, GEO dynamically alternates between spiral movements for wide search and direct chasing for intensification around promising solutions. This dual mechanism not only prevents the algorithm from being trapped in local optima but also accelerates convergence, making it more suitable for complex tasks such as EEG feature selection in motor imagery–based BCI systems.

Another important strength of GEO lies in its structural simplicity and low dependency on parameter adjustments, which ensures greater robustness across different datasets and applications. In the context of MI-BCI, where the data are high-dimensional and often noisy, GEO enables the selection of a more compact and discriminative subset of features. This leads to improved classification accuracy while simultaneously reducing computational complexity. Consequently, GEO is a particularly effective optimization strategy for real-time BCI applications, where both accuracy and efficiency are critical for reliable prosthetic control.

Figure 1 illustrates an example of the vector of selected features, representing a problem solution within the general scheme of the optimization method.

solution	6	1	12	18	9	34	5	46	19

Fig. 1. The structure of a sample solution in the proposed scheme

The concept of GEO is inspired by the intelligent hunting behavior of golden eagles, particularly their ability to adjust their speed during different phases of the hunt. Initially, golden eagles search for prey, and as they near the end of the hunt, they prepare to attack. A golden eagle finely tunes these two phases to efficiently capture the best prey in a given area within the shortest time frame. This behavior has been mathematically modeled to enhance the processes of exploration (searching for prey) and exploitation (capturing prey) within a global optimization method [25]. Each golden eagle starts its hunt by soaring high above its territory, making large circles as it looks for prey. Upon spotting potential prey, the eagle begins to spiral inward, moving from the perimeter towards the center above the prey. While the eagle remembers the prey's location, it continues to circle, gradually decreasing its altitude. As it gets closer to the prey, the radius of the imaginary circle it makes becomes smaller. Concurrently, the eagle scouts the surrounding area for better opportunities. Sometimes, they even share information about the best prey they have found with other eagles.

If an eagle does not find a more suitable target or location, it will continue to circle above the current prey, with the circles' radius decreasing until it finally attacks. However, if a better option is found, the eagle will start circling around this new target and disregard the previous one [27]. It is important to note that the final attack is executed in a straight line. Figure 2 illustrates the cruise and attack vectors in a two-dimensional space. If a new location

determined by these vectors is superior to the one in the eagle's memory, the memory is updated accordingly. During each iteration, every golden eagle (denoted as 'i') randomly selects a target from another golden eagle (denoted as 'f') and circles around the best location that eagle 'f' has encountered. Additionally, Golden Eagle 'i' may also circle around a location it has previously memorized.

The attack vector for golden eagle i is calculated as the following equation:

$$\overrightarrow{A_t} = \overrightarrow{X_f} - \overrightarrow{X_t} \quad (1)$$

Where A_i is the attack vector of eagle i. and X_f is the best location (prey) ever seen by eagle f and X_i is the current position of eagle i.

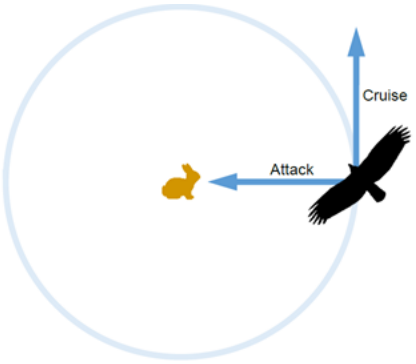


Fig. 2. Spiral movements of the golden eagle [27]

The exploration vector is calculated based on the attack vector. In fact, the exploration vector is a vector that is tangent to the sphere and perpendicular to the attack vector. "Cruise" can also be described as the linear speed of the golden eagle relative to its prey. The cruise vector in n-dimensional space lies within the hyperplane that is tangent to the sphere. The movement of golden eagles, including both attack and exploration phases, for golden eagle i in the t-th iteration, is represented by the following relationship:

$$\Delta x_i = r_1 p_a \frac{\overrightarrow{A_t}}{\|\overrightarrow{A_t}\|} + r_2 p_c \frac{\overrightarrow{C_t}}{\|\overrightarrow{C_t}\|} \quad (2)$$

Where p_a is the attack coefficient and p_c is the cruise coefficient in each repetition t and it regulates how golden eagles are affected by attack and cruise. The vectors r_1 and r_2 are random vectors whose elements are in the range of [0,1]. $\|\overrightarrow{A_t}\|$ and $\|\overrightarrow{C_t}\|$ are soft Euclidean attack and cruise vectors which are calculated using the following equations.

$$\|\overrightarrow{A_t}\| = \sqrt{\sum_{j=1}^{n} a_j^2} \quad (3)$$

$$\|\overrightarrow{C_t}\| = \sqrt{\sum_{j=1}^{n} c_j^2} \quad (4)$$

The position of the golden eagles at iteration t + 1 is easily calculated by adding the displacement vector to the current position at iteration t:

$$x^{t+1} = x^t + \Delta x_i^t \quad (5)$$

If the fitness of the new position for the golden eagle is better than the position in its memory, the memory of this golden eagle is updated with the new position. Otherwise, the memory remains unchanged, and the eagle settles into the new position. In the next iteration, each golden eagle randomly selects another golden eagle from the group to follow and circles around the best location it has visited. To evaluate each solution, a fitness function (objective) is used during the optimization process. Our goal in the current design is to ensure that the final model has high accuracy in fraud detection. Therefore, the fitness function for each solution is based on the model's recognition accuracy and the size of the selected feature set.

$$fitness(sol_i) = \frac{Accuracy(sol_i)}{|sol_i|} \quad (6)$$

3-4- Presenting a hybrid classification model for motion imagery

In the fourth phase, the collected data are fed into three distinct classification models simultaneously. In the first model (the initiative classification model), the extracted features are grouped into K clusters using the fuzzy clustering method, which is a type of unsupervised learning algorithm. Initially, we set the number of clusters to k=4. However, as we increase the number of clusters in the fuzzy clustering method, the selection of samples based on feature similarity becomes more precise, potentially leading to improved accuracy in the model. This grouping is determined by the similarity in feature

patterns of the samples, without regard to whether they are related. Following the application of fuzzy clustering, the characteristics of the cluster centers can serve as indicators for the members within those clusters. Now, by examining the status of each example from the brain signal profile in this training dataset, we calculate the cluster index value for each cluster (which reflects the density of the seven motion samples in each cluster).

$$m_{ik} = \frac{tv_{ik}}{n_i} \tag{7}$$

Where, n_i is the number of samples in cluster k and tv_{ik} is the number of samples i in this cluster. Then, in the evaluation phase, for each profile of brain signals such as l, we calculate the distance from these cluster centers under the title of alk. In other words, we want to find out how far the features of each transaction l are from the status of features in each cluster k, for this we have used the Euclidean distance.

$$a_{lk} = \varepsilon + \sum_{f=1}^{\#feature} \| center(k,f) - sample(l,f) \|^2 \tag{8}$$

Where $center(k,f)$ is the value of feature f in the center of cluster k and $sample(l,f)$ is the value of feature f in sample signal l. Now, using the formula of the membership value in the second type of fuzzy, the probability of the occurrence of a signal of type i (seven samples of movement) in profile l is calculated as follows:

$$P_{li} = \sum_{k=1}^{\#cluster} \frac{m_{ik}}{a_{lk}} \tag{9}$$

Now, having the values of P_{li} (as an indicator of the probability of a move occurring in user profile x), we will rank the probability values and choose the largest probability. In this way, the nature of a brain signal can be semi-supervisedly evaluated using the flexibility of fuzzy logic and as a probability number.

In the second classifier, we aim to harness the strengths of three renowned classifiers: Decision Tree (DT), Naive Bayes (NB), and the XGBoost method to create a voting-based hybrid classifier. Concurrently, the feature set is introduced to the third classifier, which utilizes the K-nearest neighbor (KNN) method. This classification algorithm is instance-based, involves an observer, and falls under the category of lazy learning algorithms. It determines the similarity of a new sample to those in the training dataset by comparing their proximity. To ascertain the similarity between two samples, we employ a distance metric. Various methods exist for calculating the distance between two samples within the problem's search space, including Euclidean distance, Minkowski distance, and Manhattan distance. In the KNN classifier, a set of S similar samples is selected from the nearest cluster based on their resemblance to the brain signal pattern and the extracted features. Ultimately, the final decision is made through a voting mechanism that integrates the outcomes of the three classification models.

4- Results and Discussion

Xie et al. [28] have proposed a hybrid data augmentation method in the time-frequency domain. The approach involves a parallel CNN that processes both raw EEG signals and images transformed by the continuous wavelet transform (CWT). We will utilize this paper to benchmark the performance of our proposed approach against other classification models, including CART, C4.5, and Bayesian classifiers. The effectiveness of a classification method for binary data is characterized by metrics such as true positives (TP), false negatives (FN), true negatives (TN), and false positives (FP). These allow us to define four key performance parameters: accuracy, precision, recall, and the F-Measure.

To compute classification accuracy for multi-class data, one must first construct a confusion matrix. Within this matrix, correct classifications by the method are recorded along the main diagonal. The sum of the entries in each column corresponds to the total number of samples labeled as class X, while the sum of the entries in each row pertains to the number of samples predicted to be class X. By comparing the predictions of the proposed model with the actual data, we can calculate two metrics—accuracy and recall—for each class separately. Ultimately, to compare the efficacy of our model with other methods in classifying new samples, we typically average these performance metrics across all classes.

$$Precision(label\ X) = \frac{TP(x)}{Total\_Predicted(x)} \tag{10}$$

$$Recall(label\ X) = \frac{TP(x)}{Total\_Label(x)} \tag{11}$$

$$\mathrm{F-Measure} = 2.\frac{Precision.Recall}{Precision + Recall} \tag{12}$$

statistically significant, an appropriate statistical test for classifier performance comparison was employed. Depending on the evaluation protocol, the comparison was conducted based on the scores obtained from repeated runs

and cross-validation folds. The results were reported as mean ± standard deviation, and the statistical significance level was set at 0.05. In addition, p-values were calculated and reported for the comparisons between the proposed model and the reference methods in order to assess whether the observed performance differences were statistically significant.

The Matlab software is utilized for simulations. In the proposed method, we aim to detect individuals' perception of movement, encompassing seven actions: up, down, left, right, stop, grab, and release, using training examples. The simulation encompasses two distinct scenarios. Initially, we assess performance based on the number of test samples. Subsequently, we calculate and display the average accuracy for each method. As illustrated in Figures 3 and 4, the accuracy and comprehensiveness of motor imagery have been enhanced across all states related to the number of test samples for most classes. Moreover, the proposed model accurately predicts an individual's decision to execute a hand movement based on the training data. The confusion matrix presents the classification results compared to the actual data. Figure 6 illustrates the confusion matrix for proposed method.

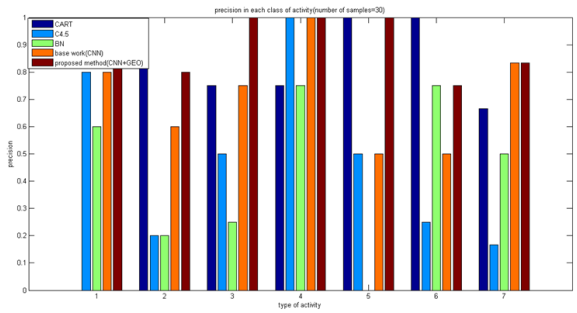


Fig. 3. Precision Analysis of Classification Methods for Different Activity Types

Figure 3 illustrates the precision values obtained from different classification models. As observed, the proposed method integrating CNN-based deep feature extraction, Golden Eagle Optimization for feature selection, and the novel hybrid classification strategy achieves the highest precision compared to conventional approaches. This indicates its superior capability in reducing false positives, which is crucial in motor imagery-based BCI systems. The consistent improvement in precision demonstrates the robustness and generalization ability of the proposed approach, making it a reliable solution for real-world EEG-based applications such as prosthetic control and rehabilitation systems

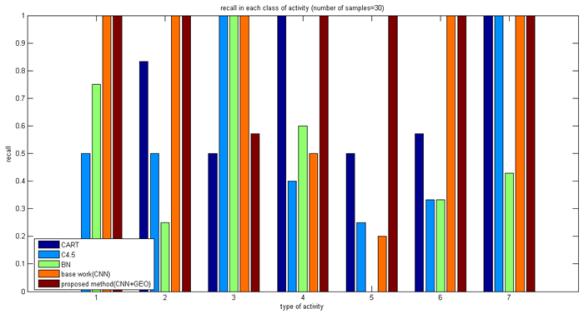


Fig. 4. Comparison of Recall Scores for Different Activity Types

Figure 4 presents the recall performance of the proposed method compared to conventional classifiers. The results demonstrate that the CNN-GEO-hybrid classifier significantly outperforms baseline approaches, achieving an approximate 10–15% improvement over standard CNN and nearly 20% enhancement compared to traditional machine learning classifiers such as SVM. This substantial gain highlights the ability of the proposed approach to reduce false negatives and reliably capture motor imagery signals, which is particularly important in practical BCI applications where missing relevant brain activities could hinder device control accuracy.

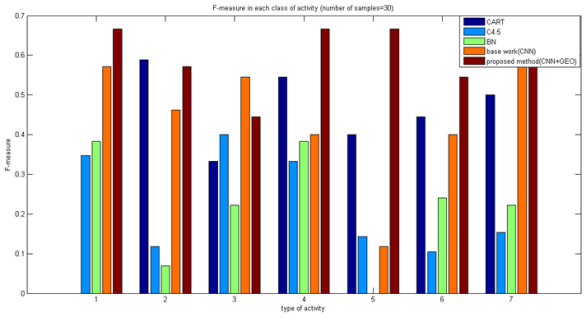


Fig. 5. F-measure Performance Across Different Activity Classes

Figure 5 illustrates the F-measure comparison among the proposed approach and baseline methods. The proposed CNN-GEO-hybrid model demonstrates a remarkable improvement, achieving approximately 12–15% higher F-measure than the baseline CNN and nearly 20% higher than traditional classifiers such as SVM and KNN. Since F-measure is a balanced metric reflecting both precision and recall, this superior performance highlights the effectiveness and robustness of the proposed method in real-world EEG classification tasks. The improvements confirm that the method not only reduces false positives and false negatives but also ensures overall reliability for motor imagery-based BCI applications.

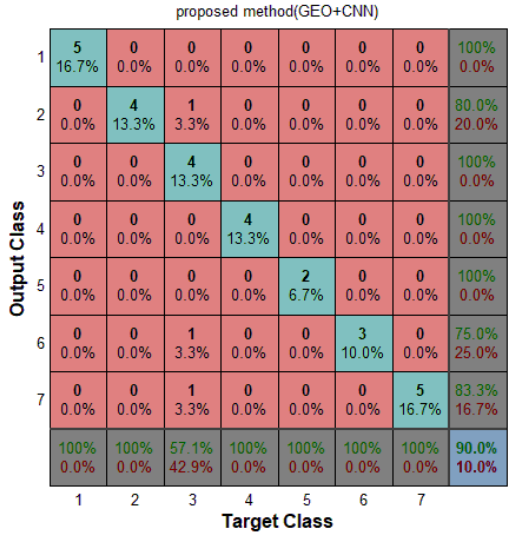


Fig. 6. Confusion matrix of proposed methods for motor imagery

We then evaluated the F-measure, which is the harmonic mean of precision and recall. According to the results depicted in Figure 5, the value of this metric is consistently higher for the proposed method than for the basic methods. To further demonstrate that the proposed method represents an overall improvement across all classes compared to the basic methods, we examined the average accuracy and recall. The outcomes of this analysis are presented in Figures 7 through 9. For this purpose, we conducted simulations with varying numbers of test samples. As illustrated in these figures, the proposed method shows a significant enhancement over the other four methods.

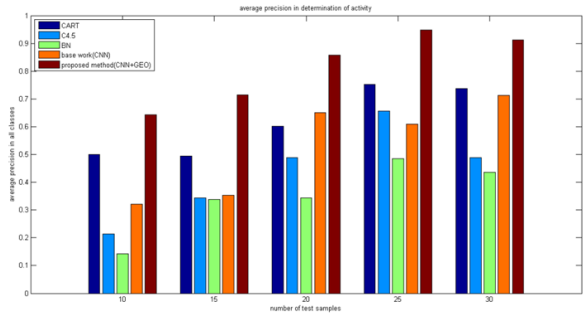


Fig. 7. Average Precision of the Proposed Method

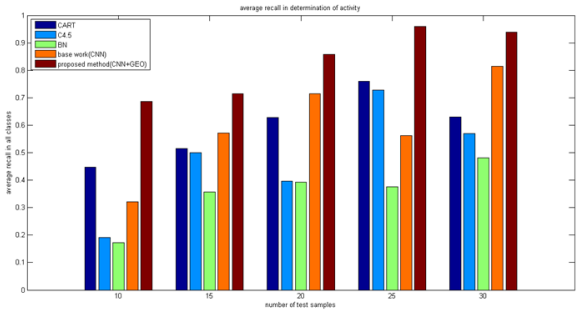


Fig. 8. Average Recall of the Proposed Method

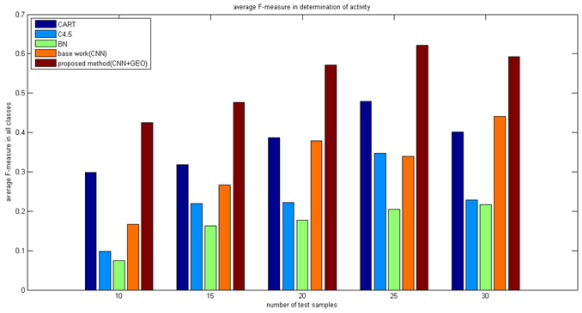


Fig. 9. Average F-measure of the Proposed Method

The comparative results showed that the proposed method outperformed the baseline methods in terms of precision, recall, and F-measure. To verify that these improvements were not caused by random variation, a statistical significance test was conducted on the obtained results. T-test was employed based on the results obtained from repeated runs. The comparisons showed that the proposed method achieved statistically significant improvements at the 0.05 level, over CNN in terms of precision, recall and F-measure, with p-values of 0.23, 0.18, and 0.2, respectively.

5- Conclusions

In summary, the novelty of this work lies in the synergy of entropy-based data refinement, intermediate CNN feature utilization, optimization-driven feature selection with GEO, and a multi-classifier fusion framework. Together, these contributions bridge the gap between experimental MI-BCI research and real-world deployment, offering a reliable pathway toward user-friendly and accurate brain-controlled prostheses.

In this paper, we presented an architecture based on a convolutional neural network (CNN) that allows for automatic feature learning in the intermediate layers of the network, thereby eliminating the need for manual extraction of statistical features from the training dataset.

This approach leverages the power of deep learning for feature extraction. Additionally, optimization methods were employed to select features in order to reduce the computational burden of the design while simultaneously enhancing accuracy. Specifically, the Golden Eagle Optimization (GEO) method was utilized. Furthermore, both supervised and semi-supervised classification models were applied to categorize electroencephalogram (EEG) signals.

The comparative analysis of the proposed CNN-GEO-hybrid classifier against baseline approaches across three key performance indicators—precision, recall, and F-measure—demonstrates its significant superiority. In terms of precision, the proposed method reduces false positives more effectively, achieving approximately 12–18% improvement over baseline CNN and nearly 20% over traditional classifiers. Regarding recall, it also outperforms existing methods by 10–15% compared to CNN and 18–22% compared to classical machine learning models, thereby reducing false negatives and ensuring more reliable detection of motor imagery signals. Finally, in terms of F-measure, which reflects the balance between precision and recall, the proposed model achieves 12–15% higher performance than CNN and almost 20% higher than conventional classifiers.

These consistent improvements across all three metrics confirm that the integration of deep feature extraction, golden eagle optimization for feature selection, and hybrid classification provides a robust and scalable solution for EEG-based motor imagery tasks. Such balanced and reliable performance is crucial for real-world brain-computer interface applications, particularly in rehabilitation and prosthetic control, where both accuracy and consistency are essential.

In future research, we aim to select an optimal subset of features with high descriptive power by extracting new features. This may involve applying wavelet and contourlet transforms to the data and examining the correlation between features and class labels within each sample group. The goal is to attain greater accuracy in detecting the perception of human movement in individuals with disabilities. To this end, we may explore other meta-heuristic methods such as the Arithmetic Optimization Algorithm (AOA) [29] and the African Vulture Optimization Algorithm (AVOA) [30].

References

[1] H. Wang, P. Chen, M. Zhang, J. Zhang, X. Sun, M. Li, and Z. Gao, “EEG-Based Motor Imagery Recognition Framework via Multisubject Dynamic Transfer and Iterative Self-training,” IEEE Trans Neural Netw Learn Syst., vol. 35, no. 8, pp. 10698-10712, Aug. 2024. DOI: 10.1109/TNNLS.2023.3243339.

[2] H. ALTAHERI, et al. “Deep learning techniques for classification of electroencephalogram (EEG) motor imagery (MI) signals: A review,” Neural Comput Appl, vol. 35, no 20, pp. 14681-14722, Jul. 2023. DOI: 10.1007/s00521-021-06352-5

[3] J. RAHUL, D. SHARMA, L. D. SHARMA, U. Nanda, and A. K. Sarkar, “A systematic review of EEG based automated schizophrenia classification through machine learning and deep learning,” Front Hum Neurosci, vol. 18, p. 1347082, Feb. 2024. DOI: 10.3389/fnhum.2024.1347082.

[4] A. Nandhini and J. Sangeetha, “A Review on Deep Learning Approaches for Motor Imagery EEG Signal Classification for Brain–Computer Interface Systems,” in Proc. Computational Vision and Bio-Inspired Computing. Advances in Intelligent Systems and Computing, Singapore, vol. 1439, pp. 353-365, Apr. 2023. DOI: 10.1007/978-981-19-9819-5_27.

[5] I. Raoof, and M. K. Gupta, “Domain-independent short-term calibration-based hybrid approach for motor imagery electroencephalograph classification: a comprehensive review,” Multimed Tools Appl, vol. 83, no 3, pp. 9181-9226, Jan. 2024. DOI: 10.1007/s11042-023-15900-1.

[6] I. ABDULLAH FAYE, and M. R. ISLAM, “EEG channel selection techniques in motor imagery applications: a review and new perspectives,” Bioengineering, vol. 9, no 12, p. 726. Nov. 2022. DOI: 10.3390/bioengineering9120726.

[7] V. S. KARDAM, S. TARAN and A. PANDEY, “Motor imagery tasks-based electroencephalogram signals classification using data-driven features,” Neurosci Inform, vol. 3, no 2, p. 100128, Jun. 2023. DOI: 10.1016/j.neuri.2023.100128.

[8] D. JAIPRIYA and K. C. SRIHARIPRIYA, “Brain computer interface-based signal processing techniques for feature extraction and classification of motor imagery using EEG: A literature review,” Biomed Mater Devices, vol. 2, no 2, pp. 601-613, Sep. 2024. DOI: 10.1007/s44174-023-00082-z.

[9] P. ARPAIA, F. DONNARUMMA, A. ESPOSITO and M Parvis, “Channel selection for optimal EEG measurement in motor imagery-based brain-computer interfaces,” Int J Neural Syst, vol. 31, no 03, p. 2150003, 2021. DOI: 10.1142/S0129065721500039.

[10] C. Zuo, et al, “Cluster decomposing and multi-objective optimization based-ensemble learning framework for motor imagery-based brain–computer interfaces,” J Neural Eng, vol. 18, no. 2, p. 026018, 2021. DOI: 10.1088/1741-2552/abe20f.

[11] Y. Chen, et al. “A novel transfer support matrix machine for motor imagery-based brain computer interface,” Front Neurosci, vol. 14, p. 606949, Nov. 2020. DOI: 10.3389/fnins.2020.606949.

[12] J. Jin, et al. “Bispectrum-based channel selection for motor imagery based brain-computer interfacing,” in Proc. IEEE Transactions on Neural Systems and Rehabilitation Engineering, vol. 28, no. 10, pp. 2153-2163, Oct. 2020. DOI: 10.1109/TNSRE.2020.3020975.

[13] J. V. Riquelme-Ros, G. Rodríguez-Bermúdez, I. Rodríguez-Rodríguez, J. V. Rodríguez and J. M. Molina-García-Pardo, “On the better performance of pianists with motor imagery-based brain-computer interface systems,” Sensors, vol.20, no. 16, p. 4452, Aug 2020. DOI: 10.3390/s20164452.

[14] O. P. Idowu, P. Fang and G. Li, “Bio-inspired algorithms for optimal feature selection in motor imagery-based brain-computer interface,” in Proc. 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology

Society (EMBC), Montreal, QC, Canada, pp. 519-522, Jul. 2020, DOI: 10.1109/EMBC44109.2020.9176244.

[15] C. Liu, et al. "Distinguishable spatial-spectral feature learning neural network framework for motor imagery-based brain–computer interface," J Neural Eng, vol. 18, no. 4, p. 0460e4, Aug. 2021. DOI: 10.1088/1741-2552/ac1d36.

[16] R. Dhiman, "Motor imagery signal classification using Wavelet packet decomposition and modified binary grey wolf optimization," Measurement: Sensors, vol. 24, p. 100553, Dec 2022. DOI: 10.1016/j.measen.2022.100553.

[17] X. Yu, M. Z. Aziz, M.T. Sadiq, Z. Fan and G. Xiao, "A new framework for automatic detection of motor and mental imagery EEG signals for robust BCI systems," IEEE Trans Instrum Meas, vol. 70, pp. 1-12, 2021. DOI: 10.1109/TIM.2021.3069026.

[18] A. Ak, V. Topuz and I. Midi, "Motor imagery EEG signal classification using image processing technique over GoogLeNet deep learning algorithm for controlling the robot manipulator," Biomed Signal Process Control, vol. 72, p. 103295, Feb. 2022. DOI: 10.1016/j.bspc.2021.103295.

[19] Z. Khademi, F. Ebrahimi and H. M. Kordy, "A transfer learning-based CNN and LSTM hybrid deep learning model to classify motor imagery EEG signals," Comput Biol Med, vol. 143, p. 105288, Apr. 2022. DOI: 10.1016/j.compbiomed.2022.105288.

[20] N. Tibrewal, N. Leeuwis and M. Alimardani, "Classification of motor imagery EEG using deep learning increases performance in inefficient BCI users," Plos one, vol. 17, no. 7, p. e0268880, Jul. 2022. DOI: 10.1371/journal.pone.0268880.

[21] X. Meng, S. Qiu, S. Wan, K. Cheng and L. Cui, "A motor imagery EEG signal classification algorithm based on recurrence plot convolution neural network," Pattern Recognit Lett, vol. 146, pp. 134-141, Jun. 2021. DOI: 10.1016/j.patrec.2021.03.023.

[22] M. N. Cherloo, H. K. Amiri and M. R. Daliri, "Ensemble regularized common spatio-spectral pattern (ensemble RCSSP) model for motor imagery-based EEG signal classification," Comput Biol Med, vol. 135, p. 104546, Aug. 2021. DOI: 10.1016/j.compbiomed.2021.104546.

[23] X. Wang, R. Yang and M. Huang, "An unsupervised deep-transfer-learning-based motor imagery EEG classification scheme for brain–computer interface," Sensors, vol. 22, no. 6, p. 2241, Feb. 2022. DOI: 10.3390/s22062241.

[24] H. Li, M. Ding, R. Zhang and C. Xiu, "Motor imagery EEG classification algorithm based on CNN-LSTM feature fusion network," Biomed Signal Process Control, vol. 72, p. 103342, Feb. 2022. DOI: 10.1016/j.bspc.2021.103342.

[25] J. Uddin, "An Autoencoder based Emotional Stress State Detection Approach by using Electroencephalography Signals", Journal of Information Systems and Telecommunication (JIST) RICEST, vol. 11, no. 41, pp. 24-30, Jun. 2023. DOI: 10.52547/jist.32267.11.41.24.

[26] S. Motamed and E. Askari, "Recognition of Attention Deficit/Hyperactivity Disorder (ADHD) Based on Electroencephalographic Signals Using Convolutional Neural Networks (CNNs)," Journal of Information Systems and Telecommunication (JIST). Vol. 10, No. 39, pp. 222-228, Aug. 2022. DOI: 10.52547/jist.16399.10.39.222.

[27] A. Mohammadi-Balani, M. D. Nayeri, A. Azar and M. Taghizadeh-Yazdi, "Golden eagle optimizer: A nature-inspired metaheuristic algorithm," Comput Ind Eng, vol. 152, p. 107050, Feb. 2021. DOI: 10.1016/j.cie.2020.107050.

[28] Y. Xie and S. Oniga, "Classification of Motor Imagery EEG Signals Based on Data Augmentation and Convolutional Neural Networks," Sensors, vol. 23, no. 4, p. 1932, Feb. 2023. DOI: doi.org/10.3390/s23041932.

[29] L. Abualigah, A. Diabat, S. Mirjalili, M. Abd Elaziz and A. H. Gandomi, "The Arithmetic Optimization Algorithm," Comput Methods Appl Mech Eng, vol. 376, p. 113609, Apr. 2021. DOI: 10.1016/j.cma.2020.113609.

[30] B. Abdollahzadeh, F. S. Gharehchopogh and S. Mirjalili, "African vultures optimization algorithm: A new nature-inspired metaheuristic algorithm for global optimization problems," Comput Ind Eng, vol. 158, p. 107408, Aug. 2021. DOI: 10.1016/j.cie.2021.107408.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 129-141
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Machine Learning Ensemble Model for Predicting Adoption of Metaverse in Higher Education Using PSO Algorithm for Setting Model's Hyperparameters

Ataalloah Abtahi [1], Zahra Afjei [1*], Ebrahim NazariFarrokhi [2]

[1]. Management and Economics, Science & Research Branch, Islamic Azad University, Tehran, Iran
[2]. Dafoos, Command and Staff University, Tehran, Iran

Received: 14 Dec 2024/ Revised: 04 Jun 2026/ Accepted: 14 Jul 2026

Abstract

In recent years, the metaverse has rapidly gained prominence as an emerging virtual platform with multidimensional and interactive features. Its potential applications, particularly in higher education, have drawn increasing attention. However, the acceptance of the metaverse in educational contexts depends on a range of factors that remain insufficiently explored. This study aims to address gaps in previous research, where certain critical factors—such as awareness, user experience, and technological challenges affecting faculty and students in Iran—have been overlooked. The goal of this research is to develop and evaluate an ensemble machine learning model to predict metaverse adoption in Iranian higher education institutions. Drawing on questionnaire data and guided by metaheuristic parameter tuning, the model seeks to identify and forecast factors influencing the uptake of this technology. This descriptive-analytical study collected data from 730 respondents, including university students and faculty members, who answered a questionnaire on metaverse acceptance. Four machine learning models—Random Forest, XGBoost, and Gradient Boosting—were employed. Their parameters were optimized using the PSO (Particle Swarm Optimization) metaheuristic algorithm. Performance metrics included Accuracy, Recall, Precision, and the F1-Score. The results showed that ensemble machine learning models, enhanced by metaheuristic parameter tuning, accurately predicted metaverse adoption. The ensemble model achieved an outstanding accuracy of 95%. Moreover, variables such as technological accessibility, familiarity with the metaverse, and institutional support emerged as key determinants. This research demonstrates that ensemble machine learning models, combined with metaheuristic parameter optimization, can serve as effective tools for forecasting metaverse acceptance in higher education. The findings can help educational administrators devise more effective strategies for implementing and fostering the use of the metaverse, ultimately improving the integration of this technology into academic environments.

Keywords: Metaverse Adoption; Higher Education; Machine Learning; Metaheuristic Setting.

1- Introduction

Digital technologies are rapidly developing in modern societies. This transformation has had extensive and complex implications within education as an area of human interaction[1]. One emerging technology attracting notable interest from experts and researchers is the metaverse[2]. The metaverse is viewed as an interactive and multidimensional virtual space that allows the merging of virtual reality (VR) and augmented reality (AR) so that a user can use digital resources along with the natural environment in one space[3]. This is of great importance in higher education institutions. These institutions can use the metaverse to create innovative and collaborative learning experiences in which students can directly engage with educational content[4]. Access to educational resources, reductions in geographical limitations, and enhancements in the quality of learning are some of the abilities provided by these virtual environments[5]. Notably, a prominent impediment the technology has to overcome is its adoption by faculty and students. The fate of these technologies will depend on their adoption and willingness to use them.

Recent studies have shown that adopting new technologies, especially in educational settings, depends on numerous factors such as the ease of use, the user's awareness,

✉ **Zahra Afjei**
zafjeh@gmail.com

system trust, and prior experience. In the words of the EdTech Research Institute's 2023 study, only 35 percent of schools managed to incorporate virtual reality technologies efficiently into their academic programs, showing the depth of challenges in integrating and applying new technologies like the metaverse[6]. Based on the provided statistics, identifying and analyzing elements that could influence the acceptance of the metaverse—especially in technologically advancing countries like Iran—has become a research necessity[7]. Consequently, this study seeks to develop and evaluate an ensemble machine learning model for predicting metaverse adoption in Iranian higher education institutions.

Interactive and virtual technologies, including the metaverse, have expanded dramatically over the last two decades, and these technologies have a wide range of potential and plausible impacts on different fields, including education[8], [9]. Virtual Reality (VR) and Augmented Reality (AR) have enabled educational bodies to generate virtual learning environments and realistic simulations. Conversely, the metaverse, as a platform, offers a broader space for faculty and students with many combinations of technological components that the personnel of an educational institution can explore with more advanced digital gadgets[10], [11]. Despite the rapid growth of these technologies in both developed and developing countries, technical, social, and educational problems and barriers have obstructed the complete implementation and establishment of these technologies within higher education institutions[12].

Previous studies in this field have analyzed some of the factors that influence the adoption of educational technologies. For instance, Johnson et al. (2020) [13]identified factors such as data security trust, user awareness of technologies, and ease of access, which are considered significant in adopting new technologies in educational contexts. In another study, Lee et al. (2021) found that user experience in interaction with virtual reality technologies directly affects users' adoption[14]. These studies have pointed out the factors that contribute to various aspects of adoption, but they have not done comprehensive metaverse adoption studies in Iran's higher education. Iran is among the developing nations' pioneer countries in adopting new educational technologies. This paper specifically examined the factors influencing the adoption of the metaverse in these environments.

Having reviewed the literature and background of this subject, it should be noted that interactive technologies have entered educational environments since the early 2000s. However, extensive use of virtual and augmented reality and other advancements in this field have materialized in recent years[15]. Aiming to fill the gap in the research literature and considering the background, this research provides a comprehensive model for predicting metaverse adoption in Iranian higher education.

Such innovations as the metaverse still face much hesitation in higher education. The novel and interactive platform of the metaverse could revolutionize the delivery of content and educational experience[16]. However, many universities and educational institutions have not fully employed or implemented it for a few reasons[17]. In tackling this domain, one big challenge is the limited understanding of the key factors influencing this technology's difficult or easy adoption. Unlike initial studies on adopting virtual and augmented reality-related technologies, no comprehensive and holistic investigations have spearheaded metaverse adoption in Iranian higher education or universities[18]. As many have argued, this knowledge gap may have impeded institutions' sound decision-making on practically implementing the metaverse and harnessing it as a powerful educational tool[19].

Unless a complete and comprehensive understanding of the factors that affect metaverse adoption is achieved, the application of this technology bears considerable dangers. Specific important questions this research sets out to answer are: Do users trust the stability and security of the metaverse? Are faculty and students hyped for metaverse as a new educational tool? What technological or social challenges may hinder the use of this technology? If these challenges were never solved, leaders within educational institutions could not leverage the many opportunities metaverses present. They could, therefore, lose their competitive edge in global student admission and continue to do so. Besides, inappropriate or incomplete applications of the metaverse bring added costs and inefficiencies to the quality of education.

This paper attempts to reach a specific objective concerning the problems raised and plenish the knowledge gap raised in the problem statement. The present research aims to propose a predictive model on the metaverse's adoption in higher education institutions based on empirical data collected through a questionnaire. To achieve this objective, the research has been designed around machine learning models and their incorporation into ensemble learning techniques coupled with metaheuristic settings to increase the accuracy of predictions on metaverse adoption.

More specifically, the objectives of this study are as follows:

To identify critical factors influencing metaverse adoption as perceived by faculty and students in higher education institutions.

Formulate a well-structured questionnaire to collect the required metaverse adoption assessment and analysis information.

Label and score collected data so that metaverse adopters and non-adopters can be identified and distinguished.

To predict metaverse adoption using ensemble machine learning approaches like XGBoost, Random Forest, and Gradient Boosting.

Metaheuristic algorithms, such as Particle Swarm Optimization (PSO), can be used to fine-tune the

parameters of machine learning models to raise accuracy levels and improve prediction efficiency.

To shortlist the best prediction model, the performance of ML models is evaluated using parameters like Recall, Accuracy, Precision, and F1-Score.

These goals are directly related to the knowledge gaps acknowledged in the problem statement and represent innovative and novel attempts to understand better the factors affecting metaverse adoption in educational institutions and improve the decision-making process therein. This research can aid educational institutions in adopting the metaverse by allowing them to make informed investment decisions regarding new technologies and formulate better plans for implementation and adoption using proper data and analytic methodologies.

This study contributes several innovations to technological innovation adoption, particularly the metaverse, in higher education institutions. One significant contribution is the provision of novel empirical data from Iranian faculty members and students, which has not previously been examined in this geographical context. The data were collected using an extensive questionnaire to examine various criteria, such as user experience, metaverse awareness, technical barriers, and academic support.

Another contribution of this work is building a comprehensive machine learning model that uses ensemble approaches like XGBoost, Random Forest, and Gradient boosting to predict metaverse adoption. Unlike previous research dependent on simpler machine learning models, this study increases prediction accuracy and obtains more reliable results by blending models and optimizing parameters with the PSO (Particle Swarm Optimization) algorithm. Therefore, this research predicts the adoption of educational technology more accurately than previous ones.

Highlighting the identification of fundamental factors influencing Metaverse adoption allows educational institutions to make better, smarter, and more appropriate decisions to implement this technology. Its noteworthy contribution is that it aids both in academic research and, more importantly, the much-needed practical application concerning educational policymakers and managers who can practically use the outcome of this research to design and implement effective programs to use the metaverse within educational settings.

In addition to starting from the local and empirical data from Iran that earlier similar research neglected, this work's original contribution compared to prior research is combining machine learning methods with metaheuristic algorithms. This combination allows this research to more accurately endeavor towards the technology adoption prediction processes and generate better and more effective recommendations for educational institutions.

The present paper is systematically organized into major subsections to familiarize the reader with research objectives, methodologies, and findings.

1. Literature Review: This section reviews related works, providing a detailed discussion of previous works on novel technology adoption, particularly those dealing with metaverse and machine learning. It reviews the studies on digital technology adoption in educational settings, thus denoting an unavoidable need to develop predictive models.
2. Background and Explanations: This section reviews the metaverse, relevant technologies, and their significance in higher education. Furthermore, it focuses on the social, technical, and managerial factors affecting metaverse adoption in educational settings.
3. System Model and Framework: In this section, the proposed machine-learning model will be reviewed to predict metaverse adopters, the scoring framework for data collected through questionnaires will be explained, and metaheuristic algorithms will be explained on how to optimize ensemble model parameters.
4. Results: This section will present the study's empirical findings. The performance of the machine learning models will be analyzed and compared according to metrics like accuracy, recall, precision, and F1-Score, among others. Moreover, the most intimidating factors severely disrupting metaverse adoption will be identified and analyzed.
5. Conclusion and Future Research: This chapter summarizes key findings and makes recommendations for improving and developing future research in metaverse adoption. Furthermore, it discusses the research challenges and limitations.

2- Literature Review

In the last few decades, the rapid conception of digital and interactive technologies has made the interplay with the metaverse one of the most discussed phenomena in higher education. The metaverse is an interactive and multidimensional virtual space that can be used as a communicative medium for unlimited learning and information exchange opportunities in education[20]. Despite its potent possibilities, the students' and instructors' adoption of this technology has not gone without a struggle; thus, a number of researches were carried out with respect to the contributing factors and analysis.

This study analyzes the technology adoption theories and models, being the article "Predicting Users' Intentions to Use Metaverse in Medical Education: A Combined SEM and Machine Learning Approach" [21].

However, this paper examines students' intention to adopt the metaverse system for medical education guided by an exceptionally integrated Structural Equation Modeling (SEM) and Machine Learning (ML) approach. This study employs the Technology Acceptance Model (TAM),

Personal Innovation (PI), and some observability-affiliated factors, perceived usefulness, and user satisfaction to assess users' attitudes. The contributions of this research can be summarized as follows: First, it used an SEM-ML combined approach to predict users' intentions to adopt metaverse systems in educational settings, especially in medical education. Second, it merges technology with research in Information Systems (IS) by combining user-oriented factors, such as personal innovation, with technological characteristics. In this regard, this study surveys 1,858 university students from the United Arab Emirates. It provides significant insights for multiple stakeholders in higher education regarding key predictors for metaverse adoption processes infusing educational contexts. The authors have employed a combined analytical model imbibing SEM for analyzing theoretical interactions and ML algorithms like neural networks and decision trees for predictions. Of note is that the Importance-Performance Map Analysis (IPMA) has been further employed to focus on the core aspects influencing user satisfaction and perceived usefulness.

This study has specific problems and limitations. First, the study's focus on a single geographical region, the United Arab Emirates, and a specific subject area, medical education, limits its generalizability to other scientific fields and regions. Second, the data were collected through self-reported questionnaires, which could render biased results. Third, this study failed to recognize the cultural differences that are more likely to affect technology adoption across different regions. Finally, the study concluded that user satisfaction (US) was the most important predictive factor of the student's intention to use metaverse systems in medical education. The other significant constructs observed in this research included perceived compatibility and observability. Additionally, compared to the traditional SEM models, the combined SEM-ML model has demonstrated better predictive abilities and has been put forth as a powerful tool for predicting technology adoption in educational settings.

The article Predicting Users' Intention to Use Metaverse System in Medical Education: A Combined SEM-ML Approach [22] also examines the intention of users (students) to adopt the metaverse system in medical education. The authors employed a combination of Structural Equation Modeling (SEM) and Machine Learning (ML) algorithms to analyze user intentions. The study data was collected from 1,858 students in the United Arab Emirates. Contributions of the Articles: This research presents a combined statistical machine learning model (SEM-ML) focused on evaluating and analyzing metaverse technology adoption in medical education.

This article examines factors such as User Satisfaction (US), Personal Innovation (PI), Perceived Compatibility (PCO), and Perceived Observability (POB). This research helps to provide a more accurate and contextual understanding of factors influencing technology adoption in educational environments. Article Methodology: This research used a conceptual framework, including the Technology Acceptance Model (TAM) and related variables such as perceived compatibility, user satisfaction, and observability. Methods of analyzing data would include SEM for theoretical relationship analysis, while prediction will use machine learning algorithms such as neural networks and decision trees. Article Limitations: This study has solely focused on students in the United Arab Emirates and the field of medical education, which will make generalizing the study's outcome challenging. The other limitation is that self-reported data can be prone to self-report bias. There was a focus on two TAM variables (perceived usefulness and perceived ease of use), neglecting a few other variables. Article Conclusion: The results have shown that personal innovation, user satisfaction, and observability significantly influence users' intention to adopt metaverse in medical education. This SEM-ML model comes out with the highest prediction accuracy compared with traditional analysis models, demonstrating its prediction capabilities in educational environments.

“A Machine Learning-Based Approach for Analyzing Adoption of Metaverse in Medical Education: A Study in the United Arab Emirates" [23] explores the influential factors of adopting metaverse systems by medical students. Its objective is to examine and analyze the interrelation between individual characteristics, technology, and the extent of application of the metaverse in medical education in the UAE. The data were collected from 879 students, using multiple structural equation modeling (SEM) and machine learning (ML)-based techniques for data analysis and different machine learning algorithms to predict the adoption of the metaverse. The primary contribution of the research is providing a novel conceptual framework for analyzing metaverse adoption that combines individual traits (like innovation) with technological attributes (observability and ease of use). Using the ML techniques, the research analyses and predicts the implementation and adoption of the metaverse that culminates in practical recommendations for developing virtual education in the post-COVID era. Its methods vary from combined SEM to machine learning techniques. J48 and OneR algorithms and neural networks analyzed data. Moreover, the most influential factors of metaverse adoption were identified by using Importance-Performance Maps Analysis (IPMA). The generalizability of the results to other countries or disciplines is limited due to focusing on students from a single country (UAE) and within one particular scientific field (medical sciences). Other challenges of this paper were the intricacies and costs of the metaverse systems.

This research deduces that observability, ease of use, and personal innovativeness significantly impact adopting the metaverse. It also improved the accuracy of predicting the metaverse adoption by exploiting ML algorithms, notably

the J48 algorithm, which had a higher accuracy rate of 87.31% than others. By evaluating and analyzing the advantages and disadvantages of this article, we find that the combined SEM-ML approach has effectively examined the complex interrelationships among various variables that affect technology adoption. However, the geographical limitation may hinder the generalizability of findings to other parts of the world.

"Enhancing Prediction of User Satisfaction of Metaverse Services by Machine Learning" [24] investigated predictive methods of user satisfaction of metaverse services through machine learning algorithms. The authors attempted to provide a comprehensive model for analyzing big data and studying the impact of online reviews and customer ratings on overall satisfaction with a metaverse application, i.e., the Roblox application. They used machine learning algorithms such as Logistic Regression, Naive Bayes, LightGBM, K-Nearest Neighbors (KNN), and Catboost. Its core contribution is presenting an extensive data analysis framework using advanced machine learning models, particularly VADER. These models helped develop a unique model to identify the differences in user review texts and ratings. This model pioneered advanced sentiment analysis techniques to enhance the prediction of user satisfaction in the metaverse domain. As for methodology, user review data were collected from 4,783,669 users. Of Roblox application. Collected data were embedded by Bag-of-Words (BoW), TF-IDF, and Word2Vec methods.

The article also employed different machine learning algorithms and evaluated their predictive accuracy using five-fold cross-validation. The possible constraints of the paper include a limited application of only one (Roblox), which cannot boast of any validity in other metaverse applications. The limitations include that the dataset was minimal, confining English-speaking data relevant to this research and may not be fitting for other translated languages and cultural norms. The conclusion is summarized as follows: LightGBM model with TF-IDF has the best performance in predicting user satisfaction at an accuracy rate of 88.68%. Much of the text reviews and rating differences were generalized using the VADER method, which offered impressive gains in prediction accuracy. This was a modern study of sentiment analysis with big data, which results in better accuracy and optimum analysis of user satisfaction; however, there are some limitations of concentrating on one application and using English only, which may preclude generalizability to other applications or languages.

Table 1

Article Title	Study Objectives	Methodology	Key Findings	Research Gaps	Conclusion
Improving User Satisfaction Prediction in Metaverse Services Using Machine Learning [21]	Machine learning-based predictive models from User satisfaction in metaverse services and VADER sentiment analysis.	Sentiment classifiers (Naïve et al.) and TF-IDF and Word2Vec embedding techniques	LightGBM (TF-IDF) had the highest accuracy of 88.68%. VADER effectively eliminates the paradox in reviews and ratings.	Lack of diversity in the metaverse service and only English users	The LightGBM model, by use of big data, can better predict user satisfaction
An Integrated Machine Learning and SEM Model for Analyzing Metaverse Adoption Among Medical Students in UAE [22]	Investigating influential factors in metaverse adoption among medical students in UAE	Employing a combined PLS-SEM approach for testing the conceptual model. Created a survey of 369 students in UAE.	Perceived Usefulness, Perceived Ease of Use, and Perceived Value are significant predictors for Metaverse adoption.	Focus on medical students in UAE that may not be generalized to other educational domains or countries.	The metaverse influences students' learning experience due to its usefulness and ease of use.
Predicting Medical Students' Metaverse Adoption: A Hybrid SEM-ML Approach [23]	Predicting students' intention to adopt Metaverse in Medical Education through combining SEM and ML techniques	Use Structural Equation Modeling (SEM) for Conceptual Models and ML (Random Forest and XGBoost) algorithms for more accurate predictive features.	This combined approach of SEM and ML can predict students' intentions with admirable accuracy; Random Forest provides the best results.	This research is restricted to medical education in the UAE.	SEM-ML combined allows for a powerful prediction apparatus for technology adoption in education.
The Metaverse and Education: Improving Learning Experience by AI and Machine Learning [24]	Exploring the role of machine learning and AI in elevating learning experiences in a metaverse environment.	Utilization of neural networks and AI techniques for modeling user interaction toward predicting educational outcomes on the Metaverse.	Artificial intelligence facilitates the building of exponentially powerful and personalized learning experiences within the Metaverse.	A limited amount of research is done on technology adoption's non-technical characteristics, including user perception.	Integrating AI and ML elevates learning experiences by building interactive and personalized environments.

3- Background and Description

In recent decades, while development steps continue, digital technologies have grown to a new evolutionary plane with the arrival of the metaverse and the presence of the interactive virtual environment. Metaverse, the holistic and multidimensional virtual space, stands for user interaction or collaboration opportunities within the convened digital environments[25]. Metaverse, a new emerging technology, has found wide application in entertainment and video gaming and fields such as healthcare, commerce, and education. Metaverse might have huge potential in higher education for enhancing and catering to interactive learning, creating virtual learning settings where students can gain hands-on experience with baffling concepts and thus benefit from real-life experiences[22].

The embodiment of the metaverse comes with significant challenges to successfully operationalizing technology within an educational setting. Machine-learning models and parameter optimization must be deployed to predict and establish features affecting technology adoption. Ensemble learning algorithms, as well as parameter optimization through meta-heuristic approaches like Particle Swarm Optimization (PSO), have great potential for enhancing the level of prediction accuracy[26].

Particle Swarm Optimization (PSO) Algorithm[27]: It is one of the metaheuristic algorithms guided by collective behaviors of birds or fish to solve optimization problems, introduced by Kennedy and Eberhart in 1995. In a PSO, a swarm of particles (potential solutions) travels across the search space, each particle opting for its best location and the best location of the entire swarm. At each stage, the positions of particles are adjusted according to their speed and direction of turning. This continues until an optimal condition emerges or the number of iterations is exhausted.

Key PSO parameters include[28], [29]:

- **Swarm Size:** number of particles moving in the search space. Increasing the number of particles may enhance search accuracy but also may increase computational time.
- **Particle Position:** Each particle has some position representing a potential solution in the search space.
- **Velocity:** It is the rate of movement of each particle in the search space and is updated in every iteration.
- **Personal Best (pBest):** Each particle has found the best position so far.
- **Global Best (gBest):** The best position the whole colony has found.
- **Inertia Weight:** The weight corresponding to the effect the velocity of a particle had on its present movement.
- **Learning Factors (C1 and C2):** Two parameters that regulate and restrict the influences of personal best (pBest) and global best (gBest) on moving particles.

Pseudo Code of PSO Algorithm

```
Initialize a population of particles with random positions and velocities
For each particle, evaluate the fitness function at its current position
Update each particle's personal best (pBest) and the global best (gBest)
While stopping criteria are not met:
  For each particle:
    Update velocity:
      velocity = inertia weight * velocity
              + C1 * random() * (pBest – current position)
              + C2 * random() * (gBest – current position)
    Update position:
      position = position + velocity

    Evaluate the fitness function at the new position
    Update pBest and gBest if necessary
Return the best solution (gBest)
```

4- System Model and Framework

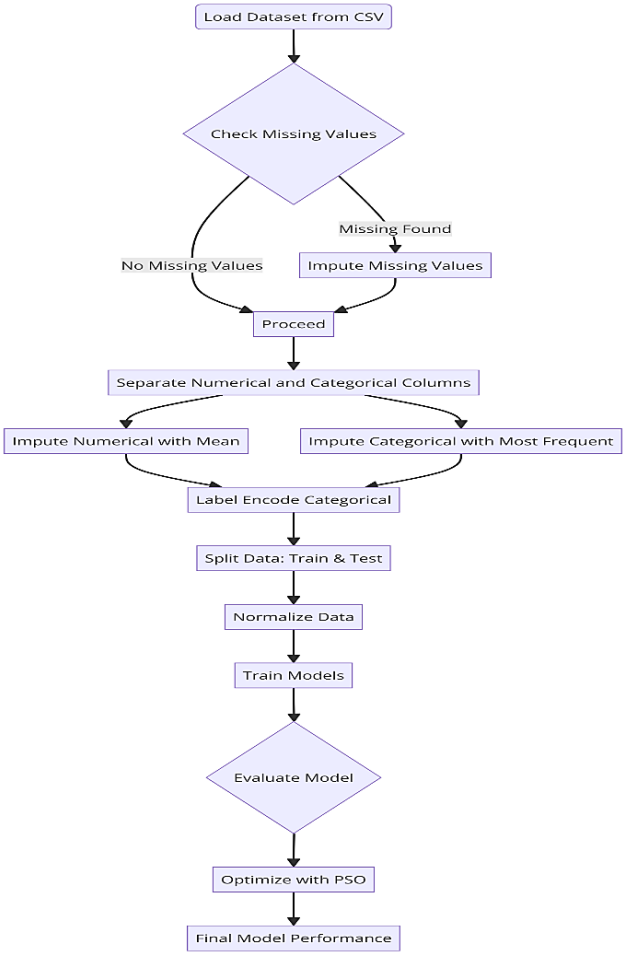
The proposed system model is designed using machine learning algorithms combined with analysis of data collected from the questionnaire that may help predict the metaverse adoption in higher education institutions. This model primarily intends to identify those factors responsible for the maximum adoption or rejection of metaverse among faculty and students. The continuum begins with the survey in a structured questionnaire dealing with aspects like user satisfaction, demographic data, user experience, etc., for collecting different facets of survey data. Such data are then forwarded for a combination of ensemble machine learning models that delineate the final label in either adoption or non-adoption. The system exploits the metaheuristic particle swarm optimization algorithm to set the hyperparameters to achieve maximum predictive accuracy. The framework is also underpinned by theoretical frameworks such as the Technology Acceptance Model (TAM) and parameters such as perceived ease of use and perceived usefulness.

The following working assumptions were considered in the construction of this model. First, data gathered through the questionnaire would help gain perceptiveness and insights from the users' opinions. It is assumed that respondents would answer questions honestly and that the samples represent the target population. The assumption is that machine learning algorithms used for prediction reflect the relationships among a multitude of variables, such as demographic features, user experience, complex patterns, and the adoption of metaverse.

The third assumption is that setting hyperparameters using the PSO algorithm effectively contributes to prediction accuracy. Certain limitations exist. For example, some research participants are likely to have not adequately answered the questionnaires due to personal or technical issues that might adversely impact the results. Moreover, it is assumed that selected model variables cover all

contributing factors to the metaverse adoption, while, practically, this might not be the case.

This system comprises several key components that interact in coordination with each other. Firstly, it is the questionnaire used to collect initial data from users. It comprises ten sections measuring trust, awareness, satisfaction, and user experience. Secondly, it uses ensemble machine learning models to process user input data and predict whether an individual becomes a metaverse adopter or non-adopter. The ensemble ML models use this questionnaire data to train on it while the PSO algorithm adjusts their hyperparameters. The interaction between the components will initially normalize and clean the questionnaire data, then form the input for those machine learning models. Lastly, model outputs of metaverse adoption or non-adoption will be the final output of the system.



Pseudocode Propose Algorithm

1. Load the dataset from the CSV file path.
2. Define a function `check_missing` that takes a DataFrame as input:
   - For each column in the DataFrame:
     - If the column contains missing values:
       - Record the column name, the number of missing values, and the data type.
   - Return the missing values information.
3. Identify numerical columns by selecting columns with data types int64 and float64.
   Identify categorical columns by selecting columns with data type objects.
4. For numerical columns:
   - Use Simple Imputer with the strategy 'mean' to replace missing values.
   For categorical columns:
   - Use Simple Imputer with the most frequent strategy to replace missing values.
5. Check for missing values again after imputation:
   - Print their details if any missing values remain; otherwise, indicate no missing ones.
6. For each column that needs encoding:
   - Use Label Encoder to convert categorical values into numerical values.
   - Replace the original categorical column with its encoded version in the DataFrame.
7. Separate the target column from the dataset.
   - Assign the remaining columns as the feature set (X) and the target column as the labels (y).
8. Use train test split to divide the dataset into training and testing subsets.
   - Allocate 80% of the data to training and 20% to testing.
9. Create a Standard Scaler object to normalize the data:
   - Fit the scaler on the training data and transform training and test datasets.
10. Define a function `evaluate model` that accepts a model and its name:
    - Train the model using the training dataset.
    - Predict on the test dataset.
    - Calculate evaluation metrics like Accuracy, Recall, Precision, F1-Score, and ROC AUC.
    - Return the metrics.
11. Define a class `Hyperparameter Optimization` for PSO-based optimization:
    - Initialize the hyperparameter grid, model, and datasets.
    - During optimization, convert the continuous solution space to appropriate hyperparameter values.
    - Fit the model with the best parameters and calculate its accuracy.
12. Create a function `PSO tuning ensemble` to run PSO optimization:
    - Accept a model name, an ensemble model, a hyperparameter grid, and a results Data Frame.
    - Use PSO (Particle Swarm Optimization) to tune the hyperparameters.
    - Evaluate the tuned ensemble model and append the results to the performance Data Frame.
13. Create an ensemble model using Random Forest, XGBoost, and Gradient Boosting classifiers:
    - Combine them using a voting classifier with soft voting.
14. Tune the ensemble model using PSO for hyperparameter optimization:
    - Evaluate the final tuned model and record its performance.
15. Store the final performance metrics in a CSV file.
    - Print a message indicating that the results have been saved.

For this research, a questionnaire was designed and distributed among faculty and students at several universities to examine the adoption of metaverse technologies in educational environments. This questionnaire included several sections, each designed to explore respondents' levels of user satisfaction, demographic information, user experience, and possible technical difficulties, along with each response. Upon collecting participant responses, data were examined and rated against pre-determined criteria. An overall score was then assigned for each participant: above a certain score was considered metaverse adoption (coded 1), with anything scoring below known as non-adoption (coded 0). These were then used as input data for the machine learning models to predict metaverse adoption.

In the preliminary data processing phase, the questionnaire responses were imported into a CSV file representing responses arranged into columns and rows. Initial evaluations aimed at identifying missing values will be conducted. Each column of the DataFrame shall be analyzed separately for missing data. At this stage, the items that will be recorded include the column's name, the number of missing data, and the column's data type. It is also important to flag when missing data exists; for example, incomplete data would undermine the quality of the final analysis. We identified numerical and categorical columns where numerical columns include variables with numerical data types (int64 or float64) that mainly contain data such as admission criteria scores or age.

Conversely, we classified categorical columns (object) that contained variables such as educational status or gender. This kind of classification helped to use appropriate methods for completing missing data. To this end, two different methods were employed depending on the type of data. In numerical columns, missing values were replaced using the mean of existing values in the same column to avoid data distortion. However, we used the most frequent value in the categorical columns to fill in the missing data. This method ensured that all missing data were correctly completed and prepared for subsequent statistical analysis and modeling stages.

We encoded the categorical columns that required transformation upon assurance of data completeness. This step was crucial in categorical columns, such as educational status and gender (such as "male" or "female"). Numerical values replaced categorical values, and such columns were stored as numerical data in the dataset. It was necessary to convert these data to enable machine learning models to use them for analysis since most machine learning algorithms require numerical data. In the next step, we have separated the target column from the dataset.

The target column predicting the metaverse's non-adoption of the metaverse considered variable (y), and the remaining columns were used under the term feature set (X).

The data were divided into test and training sets for training and evaluation of the model. Eighty percent of the data were allocated to the training set, and the remaining 20 percent were categorized as the test set to evaluate model efficiency and performance and data normalization. The latter process is necessary to prevent significant differences between feature values. This process helps machine learning models achieve higher performance and prevent the negative impact of features of different scales.

For this paper, initially, an evaluation function was designed to assess and analyze the performance of each input model to analyze and evaluate the performance of machine learning models. In this process, the model was trained by data to learn existing data patterns. Subsequently, this model is used to predict the test dataset. Afterward, several fundamental metrics are scored to evaluate the model's performance, including recall, precision, accuracy, F1 score, and ROC AUC score. Each of these metrics represents a certain aspect of model functionality that assists in evaluating it more comprehensively.

By calculating the metrics, findings are returned for future analyses and comparing the performance of various models. A special class based on the Particle Swarm Optimization (PSO) algorithm was designed for model optimization through a hyperparameter setting. For this class, a hyperparameter grid is defined to encompass various values of key model parameters. Then, the machine learning model, training data, and test data are introduced to the system for optimization. During this process, values in the continuous search space are converted to the appropriate hyperparameter values so that the model can optimally fit the data. Ultimately, the model is trained using the best-obtained parameters, and its level of accuracy is calculated on the test data. This process increases the accuracy and efficiency levels of the model, providing the optimal parameter settings.

In the PSO algorithm, main set parameters for optimization are listed in the table below:

Table 2

Parameter	Value	Description
Swarm Size	10	Number of particles that move in the search space. A lower number was selected for faster testing.
Inertia Weight	0.9	Determines the impact of the previous velocity of a particle in the current movement. A higher value results in maintaining higher velocity.
Cognitive Coefficient	0.5	Each particle influences the person's best position in the movement. This value balances exploration and exploitation.
Social Coefficient	0.3	The population's global best position impacts particle movement. A lower value would lead to more focused exploration.
Min Velocity	-1	The minimum velocity of each particle in the search space.
Max Velocity	1	The maximum velocity of a particle that controls the rate of changes in particle positions.
Max Iterations	10	The number of iterations that PSO runs. It is decreased for faster execution.

A special function was created to set the hyperparameters of the ensemble model for PSO optimization. This function receives the model name, ensemble learning model, and hyperparameter grid as inputs. Then, the PSO algorithm performs hyperparameter optimization. After completing the optimization process, the set model is evaluated. This method maximizes the efficiency of the ensemble learning model, which provides better results for predicting the metaverse adoption.

This research applies the PSO algorithm to set the hyperparameters to optimize the performance of ensemble learning models such as XGBoost, Gradient Boosting, and Random Forest. At this stage, the objective is to find the best values for key parameters of each model to maximize the prediction accuracy of metaverse adoption. The optimization process is focused on parameters including tree depth, number of trees, and learning rate. The parameters used in each model and the range of values searched for each parameter are shown in the following table:

Table 3

Model	Search Range	Hyperparameter	Description
Random Forest	n_estimators	50, 150	Number of trees in the random forest. More trees typically increase accuracy.
Random Forest	max_depth	5, 15	Maximum depth of trees. A greater depth lets the model learn more precisely but increases the risk of overfitting as well
XGBoost	n_estimators	50, 150	Number of trees that were used in the XGBoost model.
XGBoost	max_depth	3, 10	Maximum depth of trees in XGBoost. It balances the accuracy and complexity of the model
XGBoost	learning_rate	0.01, 0.2	The learning rate determines the speed at which the model updates weights during each iteration.
Gradient Boosting	n_estimators	50, 150	Number of trees in Gradient Boosting model.
Gradient Boosting	max_depth	3, 10	Maximum depth of trees in Gradient Boosting model.
Gradient Boosting	learning_rate	0.01, 0.2	The learning rate controls weight changes at each stage

This table exhibits the critical parameters that contribute to each model's prediction accuracy and performance. These parameters are systematically optimized, increasing accuracy levels and preventing overfitting.

When optimized, the ensemble learning models are trained with the most suitable parameters. The performance of optimized models was analyzed and evaluated by calculating key criteria such as Accuracy, Recall, Precision, F1-Score, and ROC AUC. These metrics calculate the success of models in predicting metaverse adoption. The results of evaluating optimized models were stored for comparison and analysis with other models.

After optimization by the PSO algorithm, the best and most comprehensive combination of hyperparameters for various models, including Random Forest, XGBoost, and Gradient Boosting, were obtained. Optimization helped to improve the efficiency and enhance the quality of models in predicting metaverse adoption. It also increased the ultimate accuracy of the models. The best combination of hyperparameters attained through optimization is presented with the corresponding model in the table below:

Table 4

Model	Hyperparameter	Best Optimized Value
Random Forest	n_estimators	150
Random Forest	max_depth	10
XGBoost	n_estimators	100
XGBoost	max_depth	7
XGBoost	learning_rate	0.1
Gradient Boosting	n_estimators	120
Gradient Boosting	max_depth	8
Gradient Boosting	learning_rate	0.15

The research method is descriptive-analytical. It employs machine learning models and hyperparameter optimization for analysis and prediction due to the nature of data

obtained from the questionnaires. An analytical method is suitable for the implementation of machine learning models as well. This design examines the factors that affect metaverse adoption in higher education through machine learning techniques for prediction and analysis. This research was conducted in cyberspace, and data were collected through online questionnaires from professors and students at various universities. Cyberspace was suitable for our research questions due to its better access to respondents and faster and more efficient data analysis. Using online questionnaires enabled the researchers to reach a larger group of participants, which resulted in more representative data being collected.

The research variables include:

Independent Variables: Trust in data security, user experience, user awareness of the metaverse, and the learning opportunities in the metaverse were considered potential factors affecting the adoption of the metaverse.

Dependent Variable: The adoption or non-adoption of the metaverse was considered a target variable in machine learning models.

Control Variables: Demographic features like age, gender, and education were determined to control their effects on the analysis.

4-1- Data Collection

The research data were collected directly from university professors and students through online questionnaires. Due to respondents' anonymity and using standardized questionnaires, these data-gathering sources are considered reliable. The questionnaire design includes closed-ended questions to precisely measure the adoption or non-adoption of the metaverse. The sampling is simple and random. This method was picked because it provides equal chances to all members of the society to participate in the research, ensuring that collected data represents the target population. This approach helped the researchers to select unbiased samples. The data collection tool is online questionnaires. Data analysis software includes Python and its relevant libraries, such as pandas. The scikit-learn was used to implement machine learning models. These tools were selected because of their accuracy, data processing, and efficiency in analysis. Then, the questionnaires were distributed through online platforms and sent to participants. Data were automatically collected within a specified period and stored in CSV files for analysis later. Ethical considerations, such as participants' anonymity and confidentiality of data, were observed correctly.

4-2- Analysis Methods

In this research, the evaluation of the performance of the proposed hybrid algorithm was compared with machine learning models such as XGBoost, Random Forest, and Gradient Boosting. It was assessed in accordance with standard evaluation criteria for machine learning models, including Accuracy, Precision, Recall, F1-Score, and Area Under the Curve (AUC). These metrics and the methods for their calculation are scientifically and precisely explained below:

Accuracy: This is an essential metric for evaluating classification models representing the percentage of correctly classified samples. It is particularly suitable for multi-class problems in which data distribution is balanced. Accuracy is the ratio between the sum of correctly predicted samples (positive and negative) to total samples[30]:

$$\text{Accuracy} = \frac{\text{True Positive} + \text{True Negative}}{\text{True Positive} + \text{True Negative} + \text{False Positive} + \text{False Negative}}$$

This metric works well if data is evenly distributed among various classes.

Precision: Precision demonstrates the percentage of correctly predicted classified positive samples. This metric is crucial when false optimistic predictions are costly (such as fraud detection). The formula to calculate precision is as follows[31]:

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

This metric is essential for avoiding false optimistic predictions.

Recall: It represents the percentage of actual positive samples the model identified correctly. This metric is crucial in problems where missing positive samples is costly (such as diagnosing disease). The formula to calculate recall is as follows [30]:

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

This metric determines the degree of correct identification of actual positive samples by the model.

F1-Score: is a composite metric for measuring the balance between precision and recall. It is particularly used when there is a need to balance precision and recall. It is computed as the harmonic mean of precision and recall[32]:

$$\text{F1 Score} = \frac{2\times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

If data is imbalanced and the balance between precision and recall is required, the F1 Score is the rescue.

Area Under the Curve (AUC): It calculates the area under the ROC curve and directly shows the model's ability to distinguish positive and negative samples[32]. The AUC value ranges from 0.5 to 1. AUC = 0.5 indicates random performance, and AUC = 1 underlines the best performance. To obtain AUC, the area under the ROC curve is calculated and applied in the overall evaluation of classifier models.

4-3- Results

This section analyzes the results obtained from experiments and evaluates the performance of the ensemble model, including RandomForestClassifier, XGBClassifier, and GradientBoostingClassifier.

This research aims to improve the accuracy and efficiency of the machine learning model in predicting metaverse adoption. To this end, it uses hyperparameter optimization through the PSO (Particle Swarm Optimization) algorithm. Since the accuracy and performance of the model matter, the metrics of accuracy, precision, recall, F1-Score, and AUC (Area Under the Curve) were picked as standard evaluation criteria. After optimizing the hyperparameter, we analyzed and evaluated the ensemble models on training and test datasets, and highly satisfactory results were yielded. The optimized ensemble model was trained using training data by optimizing hyperparameters with the PSO algorithm. Afterward, it was evaluated using the test data. The findings illustrate that the optimized model had an excellent and desirable performance in predicting metaverse adoption, and all evaluation metrics reached above 98%. The performance results of the optimized ensemble model in the training and testing phases are shown In the table below:

Phase	Accuracy	Precision	Recall	F1-Score	AUC
Training	98.6%	98.8%	98.7%	98.75%	99.2%
Testing	98.3%	98.5%	98.4%	98.45%	98.9%

The optimized ensemble model enjoys a combination of three models, namely, XGBoost, Random Forest, and Gradient Boosting, and demonstrated the ability to accurately predict metaverse adoption or non-adoption. The model achieved 98.6% accuracy during the training phase, indicating the model's capability of precise learning from training data. Precision and recall criteria were fulfilled by 98.8% and 98.7%, respectively, implying that the model has established a good balance in identifying and classifying the samples. The F1-Score of 98.75% indicates an appropriate and desirable balance between model precision and recall. The AUC value was also 99.2%, demonstrating the model's ability to distinguish different classes.

The model reached a 98.3% accuracy level, almost similar to the accuracy of the training phase, indicating excellent model generalization of the new and unseen data. Both precision and recall achieved 98.5% and 98.4%, respectively, confirming the model's good performance in correctly classifying positive and negative samples. The F1-Score is calculated at 98.45%, representing a good balance between precision and recall. At this stage, the AUC value was 98.9%, showing that the model can maintain high power in distinguishing different classes.

The results indicate that the PSO-optimized ensemble model can perform highly in predicting metaverse adoption and achieved evaluation metrics above 98% in the training and testing phases.

4-4- Comparing the Performance of Machine Learning Models

The performance of the optimized Ensemble model, developed using the PSO algorithm, was evaluated and analyzed by comparing the model with other Machine Learning algorithms such as XGBClassifier, RandomForestClassifier, GradientBoostingClassifier, and the Ensemble model without optimization. Such a procedure was meant to identify and assess the extent of algorithm improvement after hyperparameter optimization. Results for each of the models are given in the table below:

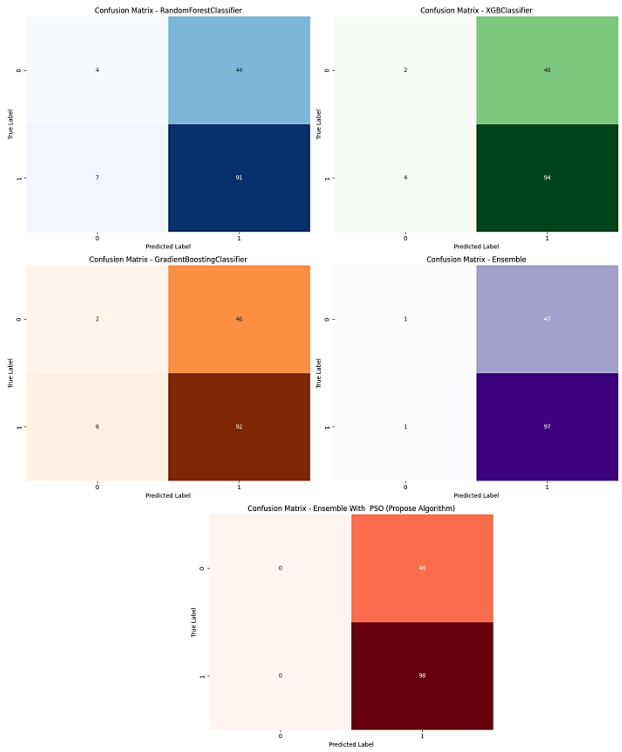
Model	Accuracy	Precision	Recall	F1-Score	AUC
Random Forest Classifier	94.5%	94.7%	94.3%	94.5%	96.2%
XGB Classifier	95.8%	96.0%	95.5%	95.75%	97.1%
Gradient Boosting Classifier	94.2%	94.5%	94.1%	94.3%	95.8%
Ensemble without PSO	96.5%	96.7%	96.3%	96.5%	97.6%
Ensemble PSO (Propose Algorithm)	98.2%	98.4%	98.3%	98.35%	98.8%

The PSO-optimized ensemble model (Proposed Algorithm) has better prediction performance on metaverse adoption than other models. Accuracy for the optimized ensemble model was 98.2% as compared with RandomForestClassifier (94.5%), XGBClassifier (95.8%), and Ensemble without PSO (96.5%). This shows that hyperparameter optimization using PSO could improve the model's prediction accuracy very well. Regarding Precision, the optimized model predicts a positive class more appropriately and desirably, with a score of 98.4% compared to 96.7% for the ensemble without PSO and all other models. It shows that the optimized model predicted positive samples with a higher score. This implies that recall for the optimized model shows 98.3%, demonstrating that the model can detect positive classes. This was 96.3% for the non-PSO, whereas lower for all other models, indicating that the optimized model correctly identified more true positive sample observations. F1-Score, as the measure of the balance between precision and recall, reached 98.35% for the optimized model, indicating an excellent balance between precision and recall.

The results show that the ensemble model enhanced through PSO was more effective for predicting metaverse adoption than all others. Concerning Accuracy, the ensemble scored high on the overall accuracy following optimization, gaining a score up to 98.2, lower than the reference of RandomForestClassifier (94.5%), XGBClassifier (95.8%), and Ensemble model without PSO(96.5%). Thus, hyperparameter optimization with PSO was effective in boosting the model's accuracy. For this optimized model, precision becomes 98.4%, which is indeed increased and acceptable compared to an ensemble model without PSO at 96.7% and others. It implies that the optimized model positively predicted more samples. The model has a recall of

98.3%; this is the capacity of the model to identify the true positive samples. The ensemble model without PSO scored 96.3, meaning that the optimized model has thus far detected many more positive samples with greater accuracy and others. There was a click to be made at 98.35 by the optimized model on the F1-Score, demonstrating a good balance between Precision and Recall.

This value is also above the 96.5-marginal case for the model without the use of PSO. Finally, the AUC achieved was much higher at 98.8%, a true mark associated with the model's very high ability to discriminate between the positive and negative classes. This is much lesser in extent than the case for other models at 97.6% and ultimately implies the superiority testified to by the optimized model.



This picture presents the Confusion Matrices of five different machine learning models. These models include RandomForestClassifier, XGBClassifier, GradientBoostingClassifier, Ensemble without PSO, and PSO-optimized Ensemble (Proposed Algorithm). Each matrix shows the performance of each model to correctly predict metaverse classes (adoption or non-adoption). The number of correctly classified samples is on the matrix's main diagonal, while incorrectly predicted samples are placed off the main diagonal. The research results indicate that the PSO-optimized Ensemble model exhibits the best performance and efficiency. It correctly classifies all samples, while other models, particularly RandomForestClassifier and GradientBoostingClassifier, misclassified more samples. Overall, compared to other models, the PSO-optimized model demonstrates higher accuracy and better quality of performance, which clearly indicates the advantage of this model in predicting metaverse adoption over others.

5- Conclusion and Future Research

The main objective of the research was to develop a machine learning ensemble model for predicting the extent to which faculty members and students of the higher education institutes would adopt metaverse. Raw data were provided from questionnaires to collect various data on adopting the metaverse. Modeling was performed via advanced methods, including Random Forest Classifier, XGB Classifier, Radient Boosting Classifier algorithms, and the PSO-optimized Ensemble model (Proposed Algorithm). This study used the PSO algorithm to optimize hyperparameters and attempt to devise an optimal framework to improve the performance of various models in predicting metaverse adoption. This research's main goal was to develop an accurate and comprehensive prediction model to analyze the impact of meta-heuristic optimizations on improving the performance quality of ensemble models.

As the main findings indicate, the accuracy level of the models was increased to over 98 percent by using the PSO algorithm for the optimization of the Ensemble model. Compared to basic unoptimized models, the proposed model with PSO obtained better results in all metrics, such as accuracy, precision, recall, F1 score, and AUC. These results demonstrated that, compared to other unoptimized models like RandomForest and GradientBoosting, the optimized model offered higher efficiency in predicting the adoption of the metaverse. According to the results, optimization with meta-heuristic algorithms like PSO can significantly improve the performance of machine learning models.

Despite valuable and significant achievements, this work encountered several problems and limitations that must be considered in future research. One of these limitations is the need for high computational resources to execute the PSO algorithm and the complexity of hyperparameter optimizations. Furthermore, the results are limited to the adoption of the metaverse and cannot be generalized to other areas beyond it.

Future research should examine these models within different educational settings using more extensive and comprehensive data to enhance the generalization ability. Moreover, other optimization algorithms, such as genetic algorithms, could be employed as meta-heuristic methods and new deep learning techniques to improve performance and efficiency. By focusing on adopting other new and innovative technologies, future studies can offer more applications of the machine learning frameworks.

References

[1] Y. Aun, Y. M. J. Khaw, M. L. Gan, and D. L. H. Chern, “A Machine-Learning Ensemble Method for Temporal-aware Sales Forecasting,” in 2022 5th International Conference on

Data Science and Information Technology, DSIT 2022 - Proceedings, 2022. doi: 10.1109/DSIT55514.2022.9943925.
[2] D. Buragohain, S. Chaudhary, G. Punpeng, A. Sharma, N. Am-In, and L. Wuttisittikulkij, "Analyzing the Impact and Prospects of Metaverse in Learning Environments Through Systematic and Case Study Research," IEEE Access, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3340734.
[3] T. Huynh-The et al., "Blockchain for the metaverse: A Review," Future Generation Computer Systems, vol. 143, 2023, doi: 10.1016/j.future.2023.02.008.
[4] M. Wang, H. Yu, Z. Bell, and X. Chu, "Constructing an Edu-Metaverse Ecosystem: A New and Innovative Framework," IEEE Transactions on Learning Technologies, vol. 15, no. 6, 2022, doi: 10.1109/TLT.2022.3210828.
[5] J. Martín-Gutiérrez, C. E. Mora, B. Añorbe-Díaz, and A. González-Marrero, "Virtual technologies trends in education," Eurasia Journal of Mathematics, Science and Technology Education, vol. 13, no. 2, 2017, doi: 10.12973/eurasia.2017.00626a.
[6] B. Ravichandran and K. Shanmugam, "Adoption of EdTech products among college students: a conceptual study," Management Matters, vol. 21, no. 1, 2024, doi: 10.1108/manm-07-2023-0026.
[7] T. M. Ndofirepi and P. Gavai, "The adoption of mobile banking among college students in Zimbabwe," African Journal of Business and Economic Research, vol. 14, no. 4, 2019, doi: 10.31920/1750-4562/2019/14n4a5.
[8] Y. M. Tang, K. Y. Chau, A. P. K. Kwok, T. Zhu, and X. Ma, "A systematic review of immersive technology applications for medical practice and education - Trends, application areas, recipients, teaching contents, evaluation methods, and performance," 2022. doi: 10.1016/j.edurev.2021.100429.
[9] Z. Karbasi and S. R. Niakan Kalhori, "Application and evaluation of virtual technologies for anatomy education to medical students: A review," Med J Islam Repub Iran, 2020, doi: 10.47176/mjiri.34.163.
[10] J. Xiong, E. L. Hsiang, Z. He, T. Zhan, and S. T. Wu, "Augmented reality and virtual reality displays: emerging technologies and future perspectives," 2021. doi: 10.1038/s41377-021-00658-8.
[11] K. Yin, Z. He, J. Xiong, J. Zou, K. Li, and S. T. Wu, "Virtual reality and augmented reality displays: Advances and future perspectives," 2021. doi: 10.1088/2515-7647/abf02e.
[12] A. Antwi-Boampong, "Towards a faculty blended learning adoption model for higher education," Educ Inf Technol (Dordr), vol. 25, no. 3, 2020, doi: 10.1007/s10639-019-10019-z.
[13] A. Granić, "Educational Technology Adoption: A systematic review," Educ Inf Technol (Dordr), vol. 27, no. 7, 2022, doi: 10.1007/s10639-022-10951-7.
[14] H. Lee, Y. Xu, and A. Porterfield, "Consumers' adoption of AR-based virtual fitting rooms: from the perspective of theory of interactive media effects," Journal of Fashion Marketing and Management, vol. 25, no. 1, 2021, doi: 10.1108/JFMM-05-2019-0092.
[15] S. Kavanagh, A. Luxton-Reilly, B. Wuensche, and B. Plimmer, "A systematic review of Virtual Reality in education," 2017.
[16] N. Kshetri, D. Rojas-Torres, and M. Grambo, "The Metaverse and Higher Education Institutions," IT Prof, vol. 24, no. 6, 2022, doi: 10.1109/MITP.2022.3222711.
[17] N. Kshetri, "Metaverse in Higher Education Institutions: Drivers and Effects," Academy of Management Proceedings, vol. 2024, no. 1, 2024, doi: 10.5465/amproc.2024.15498abstract.
[18] N. N. Masnadi, "The Development of a Metaverse App for a Blended Learning in Higher Education Institutions," 2023. doi: 10.4018/978-1-6684-6097-9.ch006.
[19] S. Kaddoura and F. Al Husseiny, "The rising trend of Metaverse in education: challenges, opportunities, and ethical considerations," PeerJ Comput Sci, vol. 9, 2023, doi: 10.7717/peerj-cs.1252.
[20] A. Tlili et al., "Is Metaverse in education a blessing or a curse: a combined content and bibliometric analysis," 2022. doi: 10.1186/s40561-022-00205-x.
[21] A. Almarzouqi, A. Aburayya, and S. A. Salloum, "Prediction of User's Intention to Use Metaverse System in Medical Education: A Hybrid SEM-ML Learning Approach," IEEE Access, vol. 10, 2022, doi: 10.1109/ACCESS.2022.3169285.
[22] A. Aburayya et al., "SEM-machine learning-based model for perusing the adoption of metaverse in higher education in UAE," International Journal of Data and Network Science, vol. 7, no. 2, 2023, doi: 10.5267/j.ijdns.2023.3.005.
[23] S. A. Salloum et al., "Novel machine learning based approach for analysing the adoption of metaverse in medical training: A UAE case study," Inform Med Unlocked, vol. 42, 2023, doi: 10.1016/j.imu.2023.101354.
[24] S. H. Lee, H. Lee, and J. H. Kim, "Enhancing the Prediction of User Satisfaction with Metaverse Service Through Machine Learning," Computers, Materials and Continua, vol. 72, no. 3, 2022, doi: 10.32604/cmc.2022.027943.
[25] T. F. Tan et al., "Metaverse and Virtual Health Care in Ophthalmology: Opportunities and Challenges," Asia-Pacific Journal of Ophthalmology, vol. 11, no. 3, 2022, doi: 10.1097/APO.0000000000000537.
[26] L. S. Marques, C. Gresse Von Wangenheim, and J. C. R. Hauck, "Teaching machine learning in school: A systematic mapping of the state of the art," Informatics in Education, vol. 19, no. 2, 2020, doi: 10.15388/INFEDU.2020.14.
[27] Ovat Friday Aje and Anyandi Adie Josephat, "The particle swarm optimization (PSO) algorithm application – A review," Global Journal of Engineering and Technology Advances, vol. 3, no. 3, 2020, doi: 10.30574/gjeta.2020.3.3.0033.
[28] A. G. Gad, "Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review," Archives of Computational Methods in Engineering, vol. 29, no. 5, 2022, doi: 10.1007/s11831-021-09694-4.
[29] J. Tang, G. Liu, and Q. Pan, "A Review on Representative Swarm Intelligence Algorithms for Solving Optimization Problems: Applications and Trends," 2021. doi: 10.1109/JAS.2021.1004129.
[30] A. Tharwat, "Classification assessment methods," Applied Computing and Informatics, vol. 17, no. 1, 2018, doi: 10.1016/j.aci.2018.08.003.
[31] P. ElKafrawy, A. Mausad, and H. Esmail, "Experimental Comparison of Methods for Multi-label Classification in different Application Domains," Int J Comput Appl, vol. 114, no. 19, 2015, doi: 10.5120/20083-1666.
[32] M. Sitarz, "Extending F1 Metric, Probabilistic Approach," Advances in Artificial Intelligence and Machine Learning, vol. 3, no. 2, 2023, doi: 10.54364/AAIML.2023.1161.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 142-152
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

A Hybrid Deep Neural Network for Arabic Fake News Classification based on Temporal CNN and BiLSTM with Attention

Azhar A. Hadi[1*], Abbas F. Hommadi[1] ,Hussein A. Ismael[2]

[1].Electrical Engineering Department, Engineering College, University of Babylon, Babylon, Iraq.
[2].Computer Center, University of Babylon, Babylon, Iraq.
[3].Information Technology College, University of Babylon, Babylon, Iraq.

Received: 18 Feb 2025/ Revised: 04 Jun 2026/ Accepted: 25 Jul 2026

Abstract
The proliferation of fake news on social media poses a significant threat to societal harmony, especially in languages like Arabic, which is distinguished by its complexity. Thus, Artificial intelligence techniques are necessary to mitigate the impact of misinformation. Many researchers have conducted this challenge by introducing machine learning and deep learning approaches. This study presents a new hybrid deep learning network for enhancing the classification of Arabic fake news. This network combines three mechanisms: Temporal Convolutional Networks, Bidirectional Long Short-Term Memory, and Attention Mechanism. This mixture model produces a strong feature extraction process with temporal awareness. Moreover, the attention layer enables the model to focus only on relevant features and ignore irrelevant ones to concentrate on salient features. Also, several stages of pre-processing Arabic text, representing words using a pre-trained word embedding model (Glove, Ara2Vec), and extracting features through advanced layers (BiLSTM, TCN, and Attention). Extensive experimentation on large-scale and well-known Arabic fake news datasets (AFND and AraNews) demonstrates the efficacy of the proposed model. The hybrid model shows superior performance compared to baseline and previous state-of-the-art studies by achieving an accuracy of 88% and 95% on the AFND and AraNews datasets, respectively. Our results show that the suggested model can be used as a powerful technique to restrain the widespread online misinformation in the Arabic language.

Keywords: Arabic Fake News; Temporal Convolution Network; BiLSTM; Attention Mechanism; Deep Learning.

1- Introduction

Nowadays, with the internet age, people strongly rely on online sources to get fed with the news and events regarding their interests. As the number of internet users and online news sources has increased dramatically over the last two decades, users can easily access and engage in events locally and globally. Moreover, individuals can share, re-post, or even create such events on the internet with ease. [1]. All these capabilities make people a gear part of the online information era. To this end, all kinds of information can be exposed to the citizens of the internet, including false information, with no obvious hint of which should be trusted. Thus, building a trustworthy environment to exchange online information among people and organizations interchangeably become a need in the modern world of the online age [2]. Therefore, this introduction will describe what false information is and the damage it could cause.

According to [3], [4] Fake news is described as nonexistent or skewed facts that can spread widely on an online platform to change facts and people's thoughts about something. Moreover, P. Bahad et al. point out that misinformation is designed to be forged information with a deceiving legal appearance [5]. Both definitions indicate the numerous potential harmful effects that misinformation can cause, including political, financial, and social damage. This includes planting inaccurate news on the internet to manipulate people's opinions. Trump's win in 2016 during the US presidential election demonstrates a good example of how fake news could impact society to gain political power [6]. According to [7] Social network data is used to mine people's political opinions and sentiments, and then use their political favour to propagate inaccurate information to get some political gain. In addition, fake or untrusted news can mislead others intentionally and

✉ **Azhar A. Hadi**
engbabil@gmail.com

unintentionally. Thus, it helps spread misunderstanding about something. Another example is during COVID-19, false health information was spread, which can cause harm to public health, mentally and physically [8].

Detection of online false information is necessary for two reasons. First, the rapid distribution of information through online platforms such as Meta (Facebook) and X (Twitter) makes it difficult to stop or even mimic the propagation. Second, it can be challenging to distinguish between false and true information [5], [9], [10]. One way to tackle this challenge is by manually categorizing online information using experts and journalists. This approach is an unrealistic solution due to the vast amount of online information, which makes it costly and time-consuming. Another approach is by automatically classifying online data based on Artificial Intelligence (AI). This approach becomes a more handy and feasible solution due to the fast and reliable outcome. Thus, many researchers have developed and introduced deep learning approaches to conquer this challenge. For example, authors in [11] employ advance transformer technique called MARBERTv2 to identify fake news in multi-dialectal Arabic tweets. Another research [12] merges traditional 2D-CNN with a transformer-based model, namely AraBERT, to achieve this task on a variety of Arabic fake news datasets

The Arabic language occupies the 4th rank among world internet users. However, there is a significant insufficiency of research dedicated to the Arabic language compared to other languages, like English [13]. This research seeks to reduce the gap by making the following contributions:

1. Introduce a hybrid deep learning model that integrates a Temporal Convolutional Network (TCN) for feature extraction and Bidirectional Long Short-Term Memory (BiLSTM) with an Attention mechanism for capturing contextual and sequential dependencies to improve the classification task.
2. Utilize different word embeddings, GloVe and Ara2Vec, to deliver rich and various semantic representations of Arabic text.
3. Conduct extensive experiments and comparative analysis on two popular datasets (AFND and AraNews) to demonstrate the superiority of our study.

The rest of this paper is outlined as follows: Section II summarizes the related works, Section III introduces DL techniques, Section IV demonstrates the proposed methodology, Section V presents a description of the experimental studies and discusses the results, and Section VI presents the paper's conclusion.

2- Related Work

Due to the importance of detecting online misleading information, many studies have been done in this regard. These studies tackle the problem based on ML and DL approaches. The following presents a summary overview of such studies.

The authors in [3] have presented an ML (Support Vector Classification (SVC), linear SVC, KMeans, and variation Bayesian estimation of a Gaussian mixture (Bayesian Gaussian mixture)) and DL (Capsule networks, deep double Q-learning (DDQN), Deep pyramid CNN (DPCNN), and Information Distilled LSTM (ID-LSTM)) models to classify Arabic fake news using the AFND dataset. According to their results, DL methods were found to be superior to the ML methods based on accuracy metrics. The capsule network achieved 79% accuracy for a binary classifier.

The study in [14] presented a comprehensive machine-learning approach for Arabic fake news detection. This notable approach is based on joining content-related and user-related features with sentiment analysis. Random Forest, Decision Tree, AdaBoost, and Logistic Regression algorithms are used in this study to find effective ML algorithms. The study collects and annotates 10K Arabic tweets as fake or real, where the best result showed 76% accuracy by applying the Logistic Regression model. Although the outstanding feature engineering approach is used, the dataset is considered small. Thus, their work cannot be generalized to Arabic content due to the language variations.

The researchers in [15] Investigate machine learning approaches to classify Arabic online false information. This research compared three different ML methods: Naive Bayes, Logistic Regression, and Random Forest, where the AraNews dataset [16] is used. The TF-IDF is used to extract features from Arabic text. The study highlights that Random Forest outperforms the other two methods by scoring 86% accuracy due to its ensembling behaviour. The paper [17] addresses the need to identify misleading information during the COVID-19 pandemic in Arabic. While there is overwhelming content published online throughout the pandemic, this research conducts a developed machine learning classifier (Logistic Regression) to tackle the misinformation detection surrounding COVID-19. Moreover, a new benchmark dataset has been launched to serve the research thirst for detecting false Arabic information. One major limitation of this work is trust in only three manual annotators, which may result in potential subjectivity and biased decisions.

The studies in [18], and [19] present a hybrid deep learning approach to tackle the information credibility in the online Arabic context. Their presented models combine the robustness of CNN and LSTM methods to improve the detection accuracy. The findings in [18] were able to present a model with outstanding results by achieving 81.6% accuracy using the AFND dataset [13]. The other study [19] demonstrates that the hybrid model is superior to CNN alone and LSTM alone by achieving 91.4%

accuracy compared to 85.9% and 87.8% for CNN and LSTM using the AraNews dataset [16], which contains Arabic news articles from multiple domains. These studies introduced a significant hybrid DL approach that surpasses traditional ML methods. However, reasoning about other evaluation metrics, such as false positives and false negatives, and their impacts on identifying misinformation will give a more general assessment that is general.

The research [20] utilized an advanced deep learning approach to improve the performance of checking fake news. The attention mechanism with BiLSTM is leveraged in their study to identify important features in the text. Moreover, ANN is used as a classifier to reveal fake information from the corpus. The dataset called AFND [13] was used in the research, which comprised more than 600,000 articles from different domains. The proposed work exceeds other deep learning techniques, such as GRU, LSTM, and CNN techniques, by recording 81% accuracy.

The study in [18] points out the difficulties of uncovering online false information in the Arabic language by inspecting textual and semantic meaning without relying on external features. An ensemble deep-learning model has been suggested in this study. By ensembling CNN and LSTM schemes, the authors were able to present a model with outstanding results by achieving 81.6% accuracy.

The authors in [21] Conduct the challenges associated with Arabic sarcastic unreal news by utilizing language computing and machine learning approaches. They gathered 3,185 fake articles from two ironic websites and 3,710 real articles from BBC-Arabic, CNN-Arabic and Al-Jazeera news to examine the best model for predicting unreal news stories. The research discloses the outperformance of the CNN network with pre-trained word embedding compared to other machine learning techniques, such as XGBoost. Regardless of the findings, the research does not explore diverse data and digging more fake news ironic attributes to boost the performance.

Albalawi et al. [22] proposed the integration of textual and visual features to address the problem of detecting fake Arabic social media content. Thus, the detection of fake news does not depend only on textual features, as previous studies suggested, but also on visual information. This study collected and annotated about ~4k tweets from the X platform with text and image content. To extract textual features, the study employed pre-trained BERT, while they used a mix of VGG-19 and ResNet50 to obtain visual features. Their findings highlight that the model based on textual features solely gives a better performance than integrating textual with visual features.

3- Background

3-1- Deep Learning

Deep Learning (DL) is an innovative approach to the Machine Learning (ML) process, which builds on top of Neural Networks to perform learning at different levels of abstraction [23][24]. It has had a positive impact on diverse fields that entail feature identification, including image recognition, drug discovery, object identification, and speech recognition. DL applies the back-propagation method, which involves tweaking or optimizing neural network parameters. DL techniques are found to be superior to traditional ML, which provides scale and very high accuracy, especially with large training datasets. Some of the common DL techniques that are used in various fields are the Convolution Neural Networks (CNN), which can be used for image, video, and speech data. The other DL technique is Recurrent Neural Networks (RNN), which are used for sequential data such as text and speech data [25][26].

3-2- Recurrent Neural Network (RNN)

One of the most promising deep learning models is the Recurrent Neural Network (RNN). It is a good learning model for sequential data processing, such as language processing and speech recognition. Through the internal state of the neural network's recollection of prior inputs, it acquires characteristics for time-series data. On the basis of historical and current data, an RNN can also forecast future information. However, learning stored data over an extended period is challenging in the RNN structure due to the gradient disappearing and exploding. In addition, RNN has weakened in identifying long-term dependencies. Thus, Long Short-Term Memory (LSTM) is proposed to resolve traditional RNN issues [27], [28].

3-3- Long Short-Term Memory (LSTM)

LSTM networks are recurrent neural networks of a higher order intended for processing sequential data. The LSTM scheme was proposed by Hochreiter and Schmidhuber [29], adding memory cells and gating functions. There are different gate designs in LSTM, including the input, forget, and output gates that control the information flow. This controlling mechanism guarantees the preservation of crucial information in long sequences while fading irrelevant information [30]. The principal equations that define the functioning of an LSTM unit are formalised as in Eq.1 to 6:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (2)$$

$$\tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \quad (3)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * \tanh(C_t) \quad (6)$$

At each time step t, the LSTM unit processes the input x_t, previous hidden state h_{t-1}, and cell state C_{t-1} via multiple gates to control information flow. To decide which segment of C_{t-1} to ignore, the forget gate f_t is used. To specify the new information to be included in the cell state C_t, the input gate i_t is applied. Candidate cell state $\tilde{C}_t$ is calculated to offer a new cell state. To produce the updated cell state, Ct is used to combine the old and new information. While the output gate chooses what part of Ct to output, the hidden state ht is generated from Ct [7], [31]. BiLSTM networks are an extension of traditional LSTM networks that encode information from both the past and the future states of a sequence. BiLSTM enhances the capacity of a network to capture context and dependencies. BiLSTM processes the input sequence in two directions. In other words, the network gains context from preceding and succeeding elements of the sequence. The network gains a more inclusive context of the input sequence. At each time step, the output from the two directions gets concatenated to be used in the outputs. BiLSTM have many applications, including sequence labelling tasks, quality classification tasks, and the machine translation task [29], [30].

3-4- Attention Mechanism

The attention mechanism is considered one of the evolutionary architectures in DL, especially for addressing sequential data challenges. It was presented by Bahdanau, Cho, and Bengio [32] to mitigate significant flaws in standard RNN and its enhancements, like LSTM and gated recurrent unit GRU. This mechanism supports the model to emphasize important parts of the sequential data rather than considering the whole sequence equally, which enables the model to prioritize relevant features in the sequence. The emphasizing selectivity empowers the model performance and makes it applicable to different tasks such as text classification, text summarization and speech recognition [32].

The most well-known use of attention is the transformer design, presented by Vaswani et al. in 2017. A transformer significantly improves the efficiency and effectiveness by adapting the parallel processing. This parallelism design allows processing the whole sequence, while the transformers' self-attention mechanism computes attention levels for each part in the input sequence [33].

To enable the transformer model to pay attention to particular parts of the input sequence, a scaled dot-product attention layer is facilitated. The scaled dot-product attention mechanism plays a cornerstone component in transformer architecture, where its job is to weigh the sequence parts by computing the attention score for each part in the input sequence, as shown in Eq.7.

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}} + \text{mask}\right) V \quad (7)$$

Where Q represent queries to search for relevant data, K represents keys that are examined about the queries, V represents values that are accumulated based on the attention scores, and dk is the key dimension. This equation can be described as follows. First, it obtains the attention scores by applying the dot product between matrices Q and K. This resulting matrix of attention scores is scaled by dk to avoid extremely large values that can affect the gradient flow while training. To dismiss particular parts, like padding tokens, a mask can be applied optionally. Second, the softmax function is utilized to result in normalized attention weights, which indicate the importance of each part in the sequence. Finally, the weighted sum is computed by multiplying normalized attention weights by the value matrix. This step accumulates the value matrix information relevant to the attention weight that each part receives [34].

Moreover, many state-of-the-art architectures, such as BERT (Bidirectional Encoder Representations from Transformers) and GPT (Generative Pre-trained Transformer), have been derived from the attention mechanism. Acting attention mechanism as corn stone in such cutting-edge techniques revolutionizes sequential data processing and results in more accurate and context-aware decisions.

3-5- Temporal Convolutional Networks TCN

TCN was first introduced by [35] for video action segmentation. In traditional techniques, the combination of CNN and RNN is used to extract features, which leads to missing long-term relationships and is more complicated to learn. On the other hand, TCN integrates feature extraction methods by utilising 1D convolution to acquire low, medium, and high dependencies with sequential data. TCN utilized four types of convolutions. First, 1D Convolutions are used to obtain temporal relationships in sequential data. Second, Causal Convolutions are used to guarantee there is no information leak from the future to the current state. Third, Dilated Convolutions helps to acquire long-term relationships. Fourth, Residual Connections are used to avoid vanishing gradients in more complex networks [35].

3-6- Dataset Description

This study uses two well-known datasets for detecting Arabic misinformation: the Arabic Fake News Dataset (AFND) [13] and Arabic News (AraNews) [16] Dataset, as

shown in Fig.1. Both benchmarks are available on Kaggle1 and GitHub2 websites.

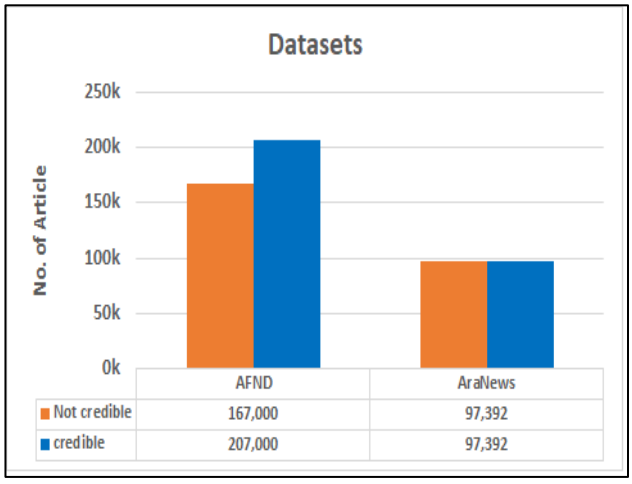
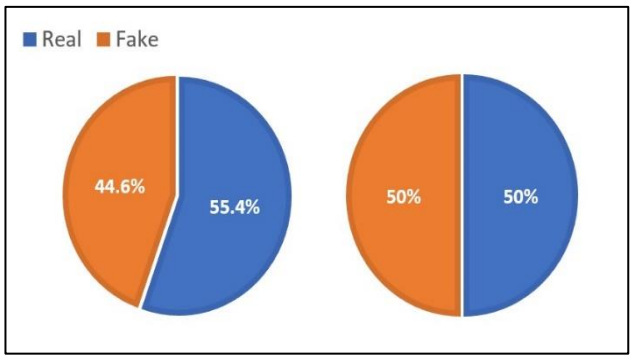


Fig.1: Datasets Summary



(a) (b)

Fig.2: Datasets Distribution (a) AFND (b) AraNews

First, the AFND [13] The dataset consists of about 607K articles. These articles were gathered during 6 month period from 134 Arabic news websites in 19 nations. By using the Arabic fact-checking service Misbar, the articles are labelled as credible, not credible, or undecided. This study considers only credible and not credible articles and ignores the undecided ones. Thus, the task becomes a binary classification where the ratio of each category is 55.4% and 44.6% for fake and real news, respectively. Furthermore, articles with fewer than 10 words are discarded [13]. The AFND distribution can be seen in Fig.2 (a).

Second, the AraNews dataset is a large-scale dataset for Arabic false news. It consists of more than 5 million Arabic articles that are assembled from 50 newspapers. This data covers a wide range of topics, varying from politics to entertainment topics, published in 15 Arabic countries in addition to the United States of America and the United Kingdom [16]. This dataset is perfectly balanced with binary classes, as shown in Fig.2 (b).

There are several challenges associated with these datasets. Including the noisiness, different Arabic dialects, and specific language difficulties. Therefore, several steps in pre-processing data are taken to overcome these challenges, as described in Section IV.

4- Methodology

This part describes the stages of the proposed work. These stages are: (i) Pre-processing data, (ii) Text representation, (iii) Feature extraction, (iv) Classifying text. Fig.3 demonstrates the proposed model.

4-1- Pre-processing Data

Preprocessing text is a vital step in text mining due to its ability to transform raw text into cleaned and structured data that can be used for further analysis [36]. It contains several techniques such as cleaning, normalizing, and tokenizing input text. First, cleaning and normalizing are two of the most important pre-processes in the preparation of data for NLP. Both techniques are used to eliminate noise, error, ambiguity, and unnecessary content from an original text document. The Arabic text is cleaned by eliminating non-Arabic words, stop words, white spaces, and punctuation marks.

Table 1: Arabic Text Preprocessing

Preprocessing steps	Before	After
Removing Punctuation	كيف جاءت الفكرة؟	كيف جاءت الفكرة
Removing Numbers and Special Characters	شعار # نحب الحياة 22	شعار نحب الحياة
Removing Stop Words	شارك في هذه الندوة	شارك الندوة
Tokenization	وذكر البنك المركزي	[وذكر ، البنك ، المركزي]
Normalization	شركتان فرنسية وبرطانية	شركتان فرنسيه وبريطانيه

In addition, encountered some problems with word embedding because some Arabic characters, like "ة" for example, in this sentence " شركتان فرنسية وبريطانية تعلنان نتاج ايجابي " were processed by normalize the text by using the Tashaphyne library and converting the character "ة" to "ه", the example above became "شركتان فرنسيه وبريطانيه تعلنان نتاج ". The second, tokenising technique is a fundamental step in NLP as it converts the continuous flow of text into discrete pieces to make it easier for further processing by considering the splitting criteria such as commas, semicolons and full stops. Tokenization is a relatively

[1] https://www.kaggle.com/datasets/murtadhayaseen/arabic-fake-newsdataset-afnd

[2] https://github.com/UBC-NLP/wanlp2020 Arabic fake news detection

straightforward process and can be done after cleaning and normalizing, more detailed examples of Arabic text pre-processing are shown in Table 1.

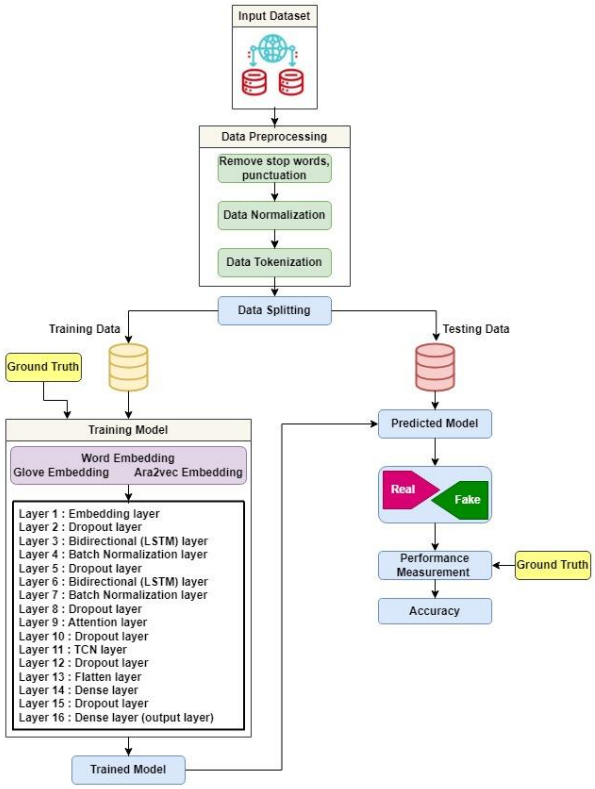


Fig.3: Proposed Model

4-2- Text Representation

To make input text understandable by a computational model, there is a need to convert the text into numerical form. There are different techniques used to achieve this task, including bag-of-words (BoW), term frequency-inverse document frequency (TF-IDF), and more sophisticated techniques such as word embeddings (Word2Vec, GloVe) and contextual embeddings (BERT, GPT [37], [38]. In this work, the input text is represented by word embedding (Word2Vec, GloVe) due to its simplicity compared with contextual embedding, being more representative than traditional BoW and TF-IDF, and the availability of an Arabic version for Word2Vec and GloVe.

4-2-1-Word Embedding

Word Embedding is trained to produce numerical representations for text in which words with a similar meaning have close vectors. Dense vectors are utilized by using some attributes in the sentences to produce a more representative vector than the high, sparse vectors of the classical TF-IDF method. It is difficult and time-consuming to build an embedding matrix from scratch. Therefore, this work is based on a pre-trained model, which includes Word2Vec and GloVe Embedding.

First, AraVec is a specific version of Word2Vec for the Arabic language. This model is trained on diverse Arabic NLP corpus (Tweets, World Wide Web pages and Wikipedia Arabic articles), Soliman et al. This diversity guarantees wide applicability across different NLP tasks, which vary from text classification to machine translation and sentiment analysis [39], [40]. AraVec2 is used in this work, where it has been applied to 66,900,000 tweets to produce a 300-dimensional embedding vector [39].

Second, GloVe is presented by Stanford scientists as a type of word embedding [41]. It is formed to learn general statistical features over a corpus, as well as the nearby context around a specific text. The main concept of Glove is the co-occurrence matrix of words, which defines the word frequency. This model yields low-dimensional embedding vectors where semantically similar words are assigned to vectors that are near each other. (GloVe v.1.2 glove. Wikipedia.6B) The GloVe-Arabic version has been used in this work.

4-3- Feature Extraction

This stage is the most vital phase to understanding raw text and eventually building an accurate model [42]. However, the Arabic language has hardness because of its plentiful morphology and various dialects [43]. To overcome these challenges, an extensive feature extraction scheme has been proposed by employing the benefit of multiple advanced neural network layers. To effectively capture dependencies among sequence parts from both directions, BiLSTM is utilised for this purpose. This behaviour is highly advantageous in Arabic text. Moreover, an attention layer has been included to ensure the most significant tokens get focused dynamically from the input sequence. Prioritising features is important to realize an Arabic article. To reveal temporal patterns in lengthy Arabic sequences, TCN is utilized. Finally, the integration of multiple neural network schemes, namely BiLSTM, attention, and TCN, has promoted the proposed architecture and discovered crucial patterns to distinguish Arabic fake content.

4-4- Classifying Text

In order to categorise the input article as legitimate or fake, the final block in the proposed work carries out the classification process. This process takes place after multiple hidden layers that are essential to obtain significant features from raw text [44]. In our work, the final block consists of three layers: dense, dropout, and sigmoid layers. To compute the likelihood of a certain

class, a sigmoid function shown in Eq.8 is implemented[45].

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad (8)$$

Where x is a vector of input features.

5- Experiments and Results

5-1- Experiment Settings

Many experiments were carried out to enhance the effectiveness of the proposed deep learning model and achieve the best results, as all results were conducted on the Google Colab platform. Google Colab is based on Jupyter Notebook, which provides many core Python libraries to implement and develop learning algorithms. It is a web-based work environment, with the following free resources: Disk 107.7 GB, System RAM 12.7 GB, and a T4 GPU. Moreover, the following libraries have been used: Keras Library, Scikit-learn Library, and Genism Library. Two datasets The AFND and AraNews, have been used to evaluate our proposed model.

Table 2: Hyperparameters of the proposed model

Name	Size
Max-Length	200
Embedding Dim.	300
Batch Size	64
Dropout Rate	0.2 and 0.5
Optimizer	Adam
Activation Functions	ReLU and Sigmoid
Epochs	20
Early Stopping	Patience=3 verbose=1, and monitor=validation loss
Regularization	L2 (0.01)

In text representation, two different word embeddings have been studied and compared, Glove and Ara2Vec, separately, to get rich semantic relationships for words. The proposed model architecture includes a two-layer Bidirectional LSTM (256 and 128 units, respectively), one attention layer with a size of 128 of the hidden state, a TCN layer (128 units), and finally a classification process. The sigmoid activation has been applied at the output layer, while the RELU layer (Rectified Linear Unit Activation Function) has been used with the other layers. To mitigate overfitting, dropout, and L2-regularisation are used.
Finally, batch normalisation is used to improve the optimization process efficiency and stability by speeding up convergence [46]. Table 3 summarizes the proposed model layers, number of nodes, and number of parameters. Hyperparameters are essential to achieve a high-performance model. In our experiments, hyperparameters are chosen as shown in Table 2.

Table 3: Summary of proposed model architecture

Layers	Output	Parameters
Embedding	(None, 200, 300)	15898800
Dropout	(None, 200, 300)	0
Bidirectional-LSTM	(None, 200, 300)	1140736
BatchNormalization	(None, 200, 512)	2048
Dropout	(None, 200, 512)	0
Bidirectional-LSTM	(None, 200, 512)	656384
BatchNormalization	(None, 200, 512)	1024
Dropout	(None, 200, 512)	0
Attention	(None, 200, 256)	456
Dropout	(None, 200, 256)	0
TCN	(None, 200, 128)	476288
Dropout	(None, 200, 128)	0
Flatten	(None, 25600)	0
Dense	(None, 64)	1638464
Dropout	(None, 64)	0
Dense	(None, 1)	65

Furthermore, both datasets (AFND and AraNews) are partitioned into 80% for the training set, 10% for the validation set, and 10% for the test set.

5-2- Results and Discussion

The evaluation metric used to assess the performance of our proposed model is accuracy, as shown in Eq. 9.

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

Where TP (True Positive) refers to the number of fake articles predicted as fake, TN (True Negative) refers to the number of credible articles predicted as real, FP (False Positive) refers to the number of credible articles predicted as fake, and FN (False Negative) refers to the number of fake articles predicted as real [47].
The experiments are conducted on two well-known datasets (AFND and AraNews), where two-word embeddings (Glove and Ara2Vec) are used to represent words. Moreover, the proposed model is extensively compared against several baseline models (CNN, LSTM, BiLSTM, CNN with LSTM, and CNN with BiLSTM) as well as the state of artwork in previous studies.

Table 4: Result comparison based on AFND Dataset

Models	Embedding	Acc.	Embedding	Acc.
CNN	Arb2vec	78%	Glove	73%
LSTM	Arb2vec	86%	Glove	85%
BiLSTM	Arb2vec	84%	Glove	86%

CNN+LSTM	Arb2vec	80%	Glove	81%
CNN+BiLSTM	Arb2vec	81%	Glove	86%
Previous Study[3]	-	-	-	79%
Previous Study[20]	-	-	Glove	81%
Previous Study[18]	-	-	Glove	81%
Proposed Model	Arb2vec	87%	Glove	88%

Table 5: Result comparison based on the AraNews 16k Dataset

Models	**Embedding**	**Acc.**	**Embedding**	**Acc.**
CNN	Arb2vec	85%	Glove	91%
LSTM	Arb2vec	86%	Glove	92%
BiLSTM	Arb2vec	87%	Glove	92%
CNN+LSTM	Arb2vec	89%	Glove	93%
CNN+BiLSTM	Arb2vec	88%	Glove	93%
Previous Study[19]	-	-	-	91%
Proposed Model	Arb2vec	89%	Glove	95%

First case, the models are compared based on the AFND dataset, as shown in Table 4. The result demonstrates that the Glove approach is the best word embedding by achieving a higher accuracy compared to the AraVec word embedding. In terms of accuracy, the proposed model records 88%, achieving +9%, +7%, and +7% higher than [3], [18], [20], respectively. This outperformance stems from the complex feature-extraction layers designed into our model. Where feature acquisition plays an important role in the classification process.

The second case is to study the proposed work using the AraNews dataset with 16K articles, as shown in Table 5. This dataset is balanced and compromised randomly. Our proposed model attains an accuracy of 95% with Glove representation. It surpasses the previous study [19], which achieves 91% and the other baseline models.

Finally, 50k articles of the AraNews dataset are conducted and the results are shown in Table 6. Our proposed model attained an accuracy of 92% with Glove and 90% with Ara2vec, compared with a previous study [19] that achieved 91%. In comparison with 16K AraNews, the proposed accuracy declined by 3%. This is due to the noisiness of AraNews articles. However, the proposed model keeps its superiority by at least 3% more than the other studied models. In conclusion, Table 7 shows the superiority of our proposed model against previous studies based on multiple datasets.

Fig 4 and Fig 5 show the confusion matrix for our proposed model against the AraNews and AFND datasets, respectively. The following Fig. 6 demonstrates the learning curves of our proposed model and baseline models. It found that the convergence of the proposed model is better than the others.

Table 6: Proposed model result for the AraNews dataset (50k)

Models	**Embedding**	**Acc.**	**Embedding**	**Acc.**
CNN	Arb2vec	84%	Glove	84%
LSTM	Arb2vec	87%	Glove	86%
BiLSTM	Arb2vec	88%	Glove	85%
CNN+LSTM	Arb2vec	89%	Glove	88%
CNN+BiLSTM	Arb2vec	89%	Glove	89%
Proposed Model	Arb2vec	90%	Glove	92%

Table 7: Previous studies vs Proposed model comparison

Study	**Models**	**Datasets**	**Acc.**
[3]	-	AFND	79%
[20]	CNN+LSTM	AFND	81%
[19]	CNN+LSTM	AraNews	91%
[18]	Bi-LSTM with Atten.	AFND	81%
[11]	AraBERT	AraNews	78%
[12]	MARBERTv2	ArabFake	89.96%
Proposed model	**CNN + BiLSTM + Atten.**	**AraNews**	**95%**
		AFND	**88%**

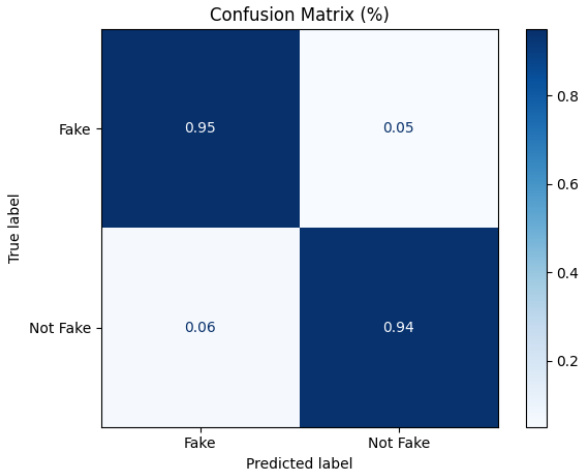


Fig.4: A confusion matrix of the AraNews Dataset

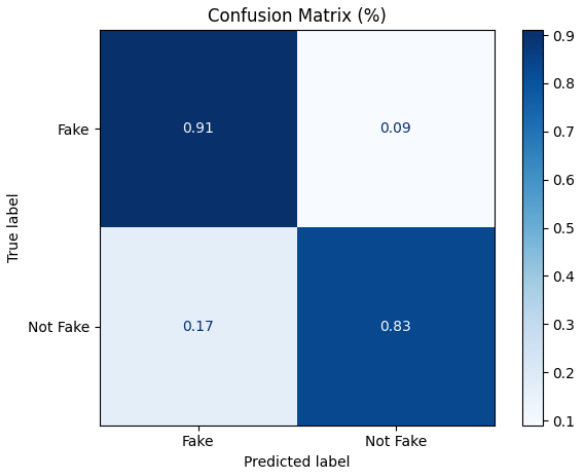


Fig.5: A confusion matrix of AFND Dataset

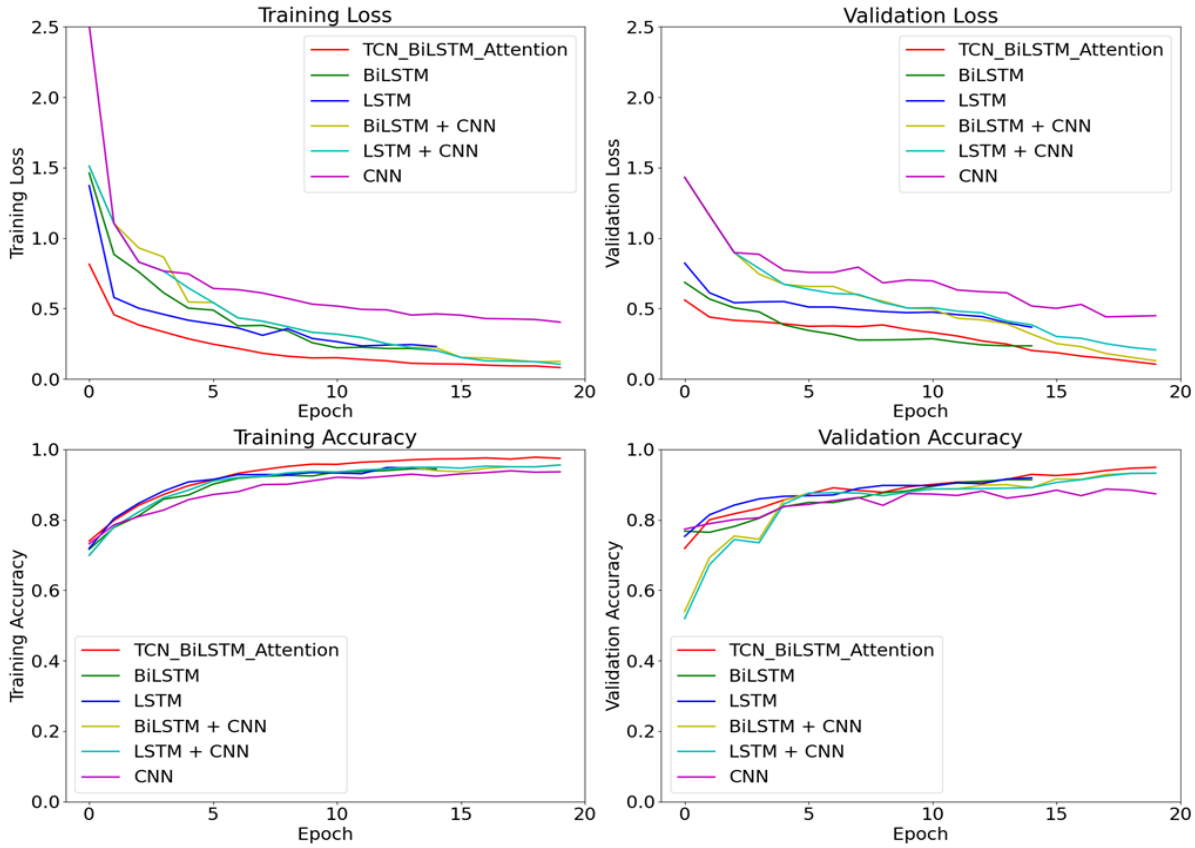


Fig.6: Learning Curve

6- Conclusions

In this study, a robust hybrid deep learning model has been developed by incorporating TCN for feature extraction and BiLSTM with an attention mechanism network for detecting fake news in the Arabic context. The proposed model achieves a high-performance accuracy of 88% and 95% with AFND and AraNews datasets, respectively. These results exceed the baseline methods (CNN, LSTM, BiLSTM, CNN with LSTM, and CNN with BiLSTM) by at least (4%, 3%, 3%, 2%, and 2%), respectively, and

state-of-the-art methods by at least 7%. The hybrid methodology succeeds in tackling the complexity of Arabic content involving different dialects and morphological abundance. According to our extensive experiments, Glove provides the best word representation compared to AraVec, where the model accuracy improved by 1% (AFND) and 6% (AraNews). In future work, several directions can advance this research. First, advanced pre-trained transformer models could be applied, such as BERT and GPT, to get the best representation of Arabic content. Second, examining our work against more diverse datasets is much preferable since Arabic has many dialects, ranging from standard to region-based accents.

References

[1] M. Alkhair, K. Meftouh, K. Smaïli, and N. Othman, "An Arabic Corpus of Fake News: Collection, Analysis and Classification," in Communications in Computer and Information Science, 2019. doi: 10.1007/978-3-030-32959-4_21.

[2] M. Al-Yahya, H. Al-Khalifa, H. Al-Baity, D. Alsaeed, and A. Essam, "Arabic Fake News Detection: Comparative Study of Neural Networks and Transformer-Based Approaches," Complexity, vol. 2021, 2021, doi: 10.1155/2021/5516945.

[3] A. Khalil, M. Jarrah, M. Aldwairi, and Y. Jararweh, "Detecting Arabic Fake News Using Machine Learning," in 2021 2nd International Conference on Intelligent Data Science Technologies and Applications, IDSTA 2021, 2021. doi: 10.1109/IDSTA53674.2021.9660811.

[4] Y. Almsrahad and N. Moghaddam Charkari, "Image Fake News Detection using Efficient NetB0 Model," Journal of Information Systems and Telecommunication (JIST), vol. 12, no. 45, pp. 41–48, Mar. 2024, doi: 10.61186/jist.40976.12.45.41.

[5] P. Bahad, P. Saxena, and R. Kamal, "Fake News Detection using Bi-directional LSTM-Recurrent Neural Network," in Procedia Computer Science, 2019. doi: 10.1016/j.procs.2020.01.072.

[6] A. Monnier, "Narratives of the Fake News Debate in France," IAFOR Journal of Arts & Humanities, vol. 5, no. 2, 2018, doi: 10.22492/ijah.5.2.01.

[7] S. Senhadji and R. A. S. Ahmed, "Fake news detection using naïve Bayes and long short term memory algorithms," IAES International Journal of Artificial Intelligence, vol. 11, no. 2, 2022, doi: 10.11591/ijai.v11.i2.pp746-752.

[8] Y. Bang, E. Ishii, S. Cahyawijaya, Z. Ji, and P. Fung, "Model Generalization on COVID-19 Fake News Detection," in Communications in Computer and Information Science, 2021. doi: 10.1007/978-3-030-73696-5_13.

[9] J. A. Nasir, O. S. Khan, and I. Varlamis, "Fake news detection: A hybrid CNN-RNN based deep learning approach," International Journal of Information Management Data Insights, vol. 1, no. 1, 2021, doi: 10.1016/j.jjimei.2020.100007.

[10] S. R. Sahoo and B. B. Gupta, "Multiple features based approach for automatic fake news detection on social networks using deep learning," Appl. Soft Comput., vol. 100, 2021, doi: 10.1016/j.asoc.2020.106983.

[11] N. A. Othman, D. S. Elzanfaly, and M. M. M. Elhawary, "Arabic Fake News Detection Using Deep Learning," IEEE Access, vol. 12, no. September, pp. 122363–122376, 2024, doi: 10.1109/ACCESS.2024.3451128.

[12] A. Maher et al., "ArabFake : A Multitask Deep Learning Framework for Arabic Fake News Detection , Categorization , and Risk Prediction," IEEE Access, vol. 12, no. November, pp. 191345–191360, 2024, doi: 10.1109/ACCESS.2024.3518204.

[13] A. Khalil, M. Jarrah, M. Aldwairi, and M. Jaradat, "AFND: Arabic fake news dataset for the detection and classification of articles credibility," Data Brief, vol. 42, 2022, doi: 10.1016/j.dib.2022.108141.

[14] G. Jardaneh, H. Abdelhaq, M. Buzz, and D. Johnson, "Classifying Arabic tweets based on credibility using content and user features," in 2019 IEEE Jordan International Joint Conference on Electrical Engineering and Information Technology, JEEIT 2019 - Proceedings, 2019. doi: 10.1109/JEEIT.2019.8717386.

[15] F. Aljwari, W. Alkaberi, A. Alshutayri, E. Aldhahri, N. Aljojo, and O. Abouola, "Multi-scale Machine Learning Prediction of the Spread of Arabic Online Fake News," Postmodern Openings, vol. 13, no. 1 Sup1, 2022, doi: 10.18662/po/13.1sup1/411.

[16] E. M. B. Nagoudi, A. Elmadany, M. Abdul-Mageed, T. Alhindi, and H. Cavusoglu, "Machine Generation and Detection of Arabic Manipulated and Fake News," Nov. 2020, [Online]. Available: http://arxiv.org/abs/2011.03092

[17] A. R. Mahlous and A. Al-Laith, "Fake News Detection in Arabic Tweets during the COVID-19 Pandemic," International Journal of Advanced Computer Science and Applications, vol. 12, no. 6, 2021, doi: 10.14569/IJACSA.2021.0120691.

[18] S. E. Sorour and H. E. Abdelkader, "AFND: ARABIC FAKE NEWS DETECTION WITH AN ENSEMBLE DEEP CNN-LSTM MODEL," J. Theor. Appl. Inf. Technol., vol. 100, no. 14, 2022.

[19] T. A. Wotaifi and B. N. Dhannoon, "An Effective Hybrid Deep Neural Network for Arabic Fake News Detection," Baghdad Science Journal, vol. 20, no. 4, 2023, doi: 10.21123/bsj.2023.7427.

[20] T. A. Wotaifi and B. N. Dhannoon, "Attention Mechanism Based on a Pre-trained Model for Improving Arabic Fake News Predictions," Iraqi Journal of Science, vol. 64, no. 11, 2023, doi: 10.24996/ijs.2023.64.11.45.

[21] H. Saadany, E. Mohamed, and C. Orasan, "Fake or Real? A Study of Arabic Satirical Fake News," Nov. 2020, [Online]. Available: http://arxiv.org/abs/2011.00452

[22] R. M. Albalawi, A. T. Jamal, A. O. Khadidos, and A. M. Alhothali, "Multimodal Arabic Rumors Detection," IEEE Access, vol. 11, 2023, doi: 10.1109/ACCESS.2023.3240373.

[23] S. F. Ahmed et al., "Deep learning modelling techniques: current progress, applications, advantages, and challenges," Artif. Intell. Rev., vol. 56, no. 11, 2023, doi: 10.1007/s10462-023-10466-8.

[24] J. Kasubi, M. D. Huchaiah, I. Gad, and M. K. Hooshmand, "A Comparison Analysis of Conventional Classifiers and Deep Learning Model for Activity Recognition in Smart

Homes based on Multi-label Classification," Journal of Information Systems and Telecommunication (JIST), vol. 12, no. 46, pp. 127–137, Jun. 2024, doi: 10.61186/jist.36294.12.46.127.
[25] P. Kim, MATLAB Deep Learning: With Machine Learning, Neural Networks and Artificial Intelligence. 2017.
[26] K. Jindal and R. Aron, "A Hybrid Machine Learning Approach for Sentiment Analysis of Beauty Products Reviews," Journal of Information Systems and Telecommunication, vol. 10, no. 37, 2022, doi: 10.52547/jist.15586.10.37.1.
[27] Q. Zhang, L. T. Yang, Z. Chen, and P. Li, "A survey on deep learning for big data," 2018. doi: 10.1016/j.inffus.2017.10.006.
[28] J. Yu, A. de Antonio, and E. Villalba-Mora, "Deep Learning (CNN, RNN) Applications for Smart Homes: A Systematic Review," 2022. doi: 10.3390/computers11020026.
[29] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," IEEE Transactions on Signal Processing, vol. 45, no. 11, 1997, doi: 10.1109/78.650093.
[30] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF Models for Sequence Tagging," Aug. 2015, [Online]. Available: http://arxiv.org/abs/1508.01991
[31] A. Graves and J. Schmidhuber, "Framewise phoneme classification with bidirectional LSTM and other neural network architectures," Neural Networks, vol. 18, no. 5–6, pp. 602–610, 2005, doi: 10.1016/j.neunet.2005.06.042
[32] D. Bahdanau, K. H. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in 3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings, 2015.
[33] P. Zhou et al., "Attention-based bidirectional long short-term memory networks for relation classification," in 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Short Papers, 2016. doi: 10.18653/v1/p16-2034.
[34] A. Vaswani et al., "Attention is all you need," in Advances in Neural Information Processing Systems, 2017.
[35] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal convolutional networks: A unified approach to action segmentation," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016. doi: 10.1007/978-3-319-49409-8_7.
[36] S. Vijayarani and M. P. Research Scholar, "Preprocessing Techniques for Text Mining-An Overview."
[37] S. Akuma, T. Lubem, and I. T. Adom, "Comparing Bag of Words and TF-IDF with different models for hate speech detection from live tweets," International Journal of Information Technology (Singapore), vol. 14, no. 7, 2022, doi: 10.1007/s41870-022-01096-4.
[38] M. SATHVIK, "Enhancing Machine Learning Algorithms using GPT Embeddings for Binary Classification," Mar. 27, 2023. doi: 10.36227/techrxiv.22331053.v1.
[39] A. B. Soliman, K. Eissa, and S. R. El-Beltagy, "AraVec: A set of Arabic Word Embedding Models for use in Arabic NLP," in Procedia Computer Science, 2017. doi: 10.1016/j.procs.2017.10.117.
[40] S. F. Sabbeh and H. A. Fasihuddin, "A Comparative Analysis of Word Embedding and Deep Learning for Arabic Sentiment Classification," Electronics (Switzerland), vol. 12, no. 6, 2023, doi: 10.3390/electronics12061425.
[41] HANSON ER, "GloVe: Global Vectors for Word Representation Jeffrey," AES: Journal of the Audio Engineering Society, vol. 19, no. 5, 1971.
[42] H. A. Ismael, N. H. Al-A'araji, and B. K. Shukur, "An Enhanced Diabetic Foot Ulcer Classification Approach Using GLCM and Deep Convolution Neural Network," Karbala International Journal of Modern Science, vol. 8, no. 4, 2022, doi: 10.33640/2405-609X.3268.
[43] H. Himdi, G. Weir, F. Assiri, and H. Al-Barhamtoshy, "Arabic Fake News Detection Based on Textual Analysis," Arab. J. Sci. Eng., vol. 47, no. 8, 2022, doi: 10.1007/s13369-021-06449-y.
[44] S. Sharma, S. Sharma, and A. Athaiya, "ACTIVATION FUNCTIONS IN NEURAL NETWORKS," International Journal of Engineering Applied Sciences and Technology, vol. 04, no. 12, 2020, doi: 10.33564/ijeast.2020.v04i12.054.
[45] H. Pratiwi et al., "Sigmoid Activation Function in Selecting the Best Model of Artificial Neural Networks," in Journal of Physics: Conference Series, 2020. doi: 10.1088/1742-6596/1471/1/012010.
[46] . A. Ismael, N. H. Al-A'araji, and B. K. Shukur, "Enhanced the prediction approach of diabetes using an autoencoder with regularization and deep neural network," Periodicals of Engineering and Natural Sciences, vol. 10, no. 6, 2022, doi: 10.21533/pen.v10i6.3394
[47] S. R. Durugkar, R. Raja, K. K. Nagwanshi, and S. Kumar, "Introduction to data mining," 2022. doi: 10.1002/9781119792529.ch1.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 153-169
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Smartening and Digital Transformation Plan in the Industry based on Best Practices

Mehdi Azadimotlagh[1*], Reza Sharafdini[2] , Zahra Salimi[3], Melika Khosravi[3], Yekta Farzadkia[3]

[1]. Department of Computer Engineering, Faculty of Engineering of Jam, Persian Gulf University, Bushehr, Iran
[2]. Department of Mathematics, , Faculty of Intelligent Systems Engineering and Data Science, Persian Gulf University, Bushehr, Iran.
[3]. Department of Computer Science, Faculty of Mathematical Sciences, Alzahra University, Tehran, Iran

Received: 19 Jul 2025/ Revised: 04 Jan 2026/ Accepted: 11 Jul 2026

Abstract

Due to smart production, Industry 4.0 has become an essential paradigm in industry. This paradigm is looking for smartening and digital transformation in industries where its goal is not only technology upgrade but also integration of advanced technology such as Artificial Intelligence, Internet of Things, Cloud computing, Digital twins and simulation software, Big Data Processing, Blockchain, etc. in production processes for increasing efficiency, flexibility, and productivity in various sections of production. Research shows that smartening and digital transformation are essential to creating a competitive advantage in global markets and leadership in the coming decades. Neglecting this change of approach can lead to losing customers and sales markets. Unfortunately, despite its advantages, implementing the smart industry faces significant challenges that can hinder its achievement of goals. So, whether at the national or corporate level, benefiting from a cohesive program for smartening and digital transformation in any industry is essential. In this paper, based on research in more than 20 leading countries in the field of smart industry, a strategic plan for smartening and digital transformation in industries has been presented. By using this strategic plan, it is possible to overcome the challenges and reach the smart industry in various fields.

Keywords: Smartening; Digital Transformation; Industry 4.0; Executive Plan.

1- Introduction

Each industrial revolution is characterized by significant technological advancements that have changed production processes. Industry 1.0 introduced mechanization through water and steam power. Following that, Industry 2.0. emphasized mass production through assembly lines. Industry 3.0 is characterized by automation facilitated through electronics and information technology. With the introduction of Industry 4.0 in the current era, the industry reached its peak at the Hannover Fair in 2011 with smart production. The concept of smart industry represents a transformative change in manufacturing processes characterized by the integration of advanced technologies such as artificial intelligence (AI), the Internet of Things (IoT), big data analytics, and robotics. This paradigm shift is not merely a technological upgrade, but it is a comprehensive reconsideration of how industries operate, leading to increasing efficiency, flexibility, and productivity across different sectors. [1-6] Overall, Industry 4.0 facilitates the advancement of smart factories capable of real-time data processing and autonomous decision-making and the monitoring, control, and optimization of production processes. This continuity not only simplifies production but also enhances sustainability by optimizing resource use and minimizing waste. [7-9] In this paper, our goal is to understand global methods to present a comprehensive plan for smartening and digital transformation in industry at a national scale. Accordingly, this paper is organized as follows.

The second section examines the key components in the field of smartening and digital transformation in industry. In this section, key technologies for smartening and digital transformation in industry are introduced. Also, the

✉ **Mehdi Azadimotlagh**
m.azadim@pgu.ac.ir

advantages and challenges related to smart technology and digital transformation in the industry will be introduced. In the third section, national programs for smartening and digital transformation in leading countries will be reviewed. The fourth section examines global methods for smartening and digital transformation in major industries such as pharmaceuticals, chemicals, and textiles. In the final section, a national strategic program for the implementation of the smart industry in each country will be presented based on the results obtained from the previous sections.

2- Key Components of Smartening and Digital Transformation in Industry

2-1- Key Technologies for Smartening and Digital Transformation in Industry

Although many technologies are used for smartening and digital transformation in industry, in this section, 9 of the most important technology areas in this field will be introduced. These areas are as follows:

1) Cyber-physical Systems (CPS)
This technology is the foundation of Industry 4.0. By combining physical processes and digital computing, it establishes the connection between physical manufacturing plants and digital systems, such as cloud computing and data analysis, which helps in monitoring and controlling manufacturing processes and leads to increased efficiency and safety. [10-12]

2) Internet of Things (IoT)
The Internet of Things refers to a network of connected devices that can share data. This real-time data can help improve predictive maintenance, optimization of the production processes, and assist manufacturers in responding more quickly to market demand. [13-16]

3) Cloud Computing
Cloud computing provides scalable and flexible computational resources over the internet. For producers, cloud-based solutions provide powerful tools for data storage, processing, and analysis. It can analyze the large volume of data generated by IoT devices and gain insights that can help improve production flows and product quality. [17-20]

4) Blockchain Technology
Blockchain technology, due to its potential to improve supply chain transparency, particularly in terms of compliance with various standards, is increasingly recognized. Blockchain can facilitate end-to-end tracking of raw materials, thereby increasing trust among stakeholders and ensuring compliance with various standards. [14,21-24].

5) Digital Twins and Simulation Software
The use of digital twins as virtual representations of physical assets involves simulating digital assets to predict the behavior of components or systems in the real world, enabling manufacturers to optimize production lines and test various production scenarios before actual implementation, thereby reducing the need for physical prototyping. [18,25-28].

6) Automation and Robotics
The use of robotics combined with artificial intelligence enables the creation of an assembly line that operates continuously with minimal human intervention, leading to increased speed and accuracy in production. [29]

7) 3D Printing
3D printing technology helps to reduce costs and accelerate production in the design and manufacturing of industries, especially the clothing industry. [30]

8) Augmented Reality (AR) and Virtual Reality (VR)
It allows customers to view products and make desired changes before purchasing. This technology also has widespread applications in design and prototyping. [31]

9) RFID/NFC
These technologies are used for the automatic identification and tracking of materials and products in the production process, helping companies achieve greater efficiency, accuracy, and transparency in their operations. [32-36].

10) Artificial Intelligence (AI)
Artificial Intelligence (AI) has emerged as the most transformative enabler of Industry 4.0, permeating every aspect of smart manufacturing from predictive maintenance to quality control and production planning. [37].
According to a Google Cloud survey of 517 manufacturing executives (2025), 78% of manufacturers report seeing returns from their generative AI investments, with 56% actively using AI agents and 37% having launched more than ten. [38] Over half (55%) plan to allocate 50% or more of their future AI budget to agentic AI. [37]
Industry reports identify four critical areas where AI delivers the highest impact [37]:

- **Predictive maintenance:** Reduces unplanned downtime by 30–50% and maintenance costs by 20–30%
- **Quality control:** Machine vision systems detect defects faster and more accurately than human inspection
- **Robotics & automation:** AI enables cobots to adapt to variable tasks in real-time
- **Energy management:** AI-powered optimization improves efficiency with ROI under two years.

The global smart manufacturing market, driven primarily by AI and IIoT, is projected to reach USD 1,339 billion by 2034 (CAGR 14.70%). [39]

2-2- Key Advantages of Implementing the Smart Industry

Industry 4 brings key advantages to industrial owners, allowing every industry period to be divided into pre and post-smartening and digital transformation. The most important advantages of the smart industry are mentioned below.

1) Smart Production
Companies are increasingly using data analytics and machine learning to predict market trends and consumer preferences, which allows for the delivery of more tailored products. [40-46]

2) Reducing Production Costs
The adoption of automation technologies and smart inventory management systems enables manufacturers to reduce costs while producing high-quality outputs by optimizing production processes. [47]

3) Predictive Maintenance
Artificial intelligence technologies, especially machine learning and data analysis, play a key role in predictive maintenance. These systems significantly enhance the ability to identify and solve mechanical issues before operational disruptions occur and to predict failures before problems arise by analyzing large volumes of equipment data. [48]

4) Optimization and Intelligent Management of the Supply Chain
Improving logistics and supply chain management through digital tools enables producers to respond quickly to changes in demand and strengthen their export capabilities. [49]

5) Increased Productivity and Efficiency
The use of modern technologies and smartening in industry leads to a reduction in hard work in repetitive tasks and the allocation of human resources to activities with higher added value. These technologies reduce the time and costs of inventory management and quality control, leading to increased efficiency. [30,50,51].

6) Enhancing Safety
By reducing employees' exposure to hazardous tasks, workplace incidents are significantly reduced. [52]

7) Increase in Accuracy, Quality, and Customer Satisfaction
By increasing the use of smart production and robotics, it is possible to enhance precision to the nanometer level, improve processes, increase product quality, and ultimately achieve customer satisfaction. [34,36,53].

8) Circular Model and Recycling
Leading companies utilize circular economic models to reduce waste and reuse primary production resources, which leads to the preservation of natural resources and a reduction in production costs. [54]

9) Environmental Sustainability and Responsibility
Technologies such as predictive maintenance can reduce waste and energy consumption, while IoT devices help monitor environmental impacts throughout the production process. [55]

10) Global Competitiveness
Advancements in the smart industry lead to reduced response times to international demand and shorter product delivery times. [56]

2-3- Challenges of Implementing Smart Industry

The smart industry represents a significant transformation in production methods driven by the integration of advanced technologies. The successful adoption of Industry 4.0 requires a holistic approach that encompasses not only technological advancements but also strategic planning and organizational change management. However, the transition to smart production is not without obstacles. Below are some of the most important challenges in moving towards smartening and digital transformation in the industry. [3,57-59].

1) High implementation costs
One of the main challenges in the smart industry is the high costs of implementation and return on investment (ROI). [9]

2) The gap between large and small companies in implementation
The cost of implementing advanced technologies can be exorbitant for smaller producers and potentially widen the gap between large and small companies. [60-62] Small, medium, and micro enterprises (SMMEs) are essential for harnessing the benefits of smart manufacturing, as these companies constitute a significant portion of the global economy and play a critical role in driving innovation and competitiveness. [3,57-59].

3) Integration with existing systems
Many companies face challenges in integrating smart systems with their existing systems, which can limit the adoption of this technology. [22]

4) Changing organizational culture and shortage of skilled labor
The challenges associated with the transition to an intelligent industry go beyond technological obstacles. Accepting smart technologies requires fundamental changes in the way of working and organizational culture. Resistance to change, especially in companies that are accustomed to traditional production methods, is considered a major obstacle. Companies should strengthen a culture of innovation and adaptability. This cultural change is often accompanied by the need to upgrade and reskill the workforce to equip them with the necessary competencies to thrive in a smart manufacturing environment. [57-59]

5) Weakness of infrastructure

The existing infrastructure in a country may not have the necessary capabilities to implement smart industry projects and may require significant enhancements to fully support smart industry technologies. [63]

6) Digital Divide
There is a significant gap between different cities regarding access to digital infrastructure, which limits the benefits of the smart industry. [51,64]

7) Concern regarding security and privacy issues
Due to the easy access to information, there may be security challenges. This can raise concerns about privacy and data security. [65,67]

8) Dangers of Cyber Attacks
As production becomes more interconnected, the risk of cyberattacks increases, necessitating strong cybersecurity measures to protect sensitive data and systems. [68]

9) Harmonization of laws
Despite these advances in the smart industry sector, harmonizing regulations remains a challenge. As an example, the member countries of the EU have different standards for implementing EU directives in the field of the smart pharmaceutical industry, which can delay the deployment of technologies. Efforts to standardize these regulations are underway, and with initiatives such as the EU strategy, processes become simpler, and cross-border cooperation improves. [69]

10) Standardization
The absence of global standards for the production and testing of certain smart products, including smart clothing, causes problems with the compatibility of different products and creates confusion among consumers. [70]

2-4- Comparative Impact Analysis of Key Enabling Technologies

The nine technological components introduced above exhibit varying levels of industrial impact. Based on recent literature (2024–2026), Artificial Intelligence is recognized as the most transformative enabler across smart manufacturing domains. Industrial IoT serves as the foundational data infrastructure for real-time condition monitoring and asset tracking. [71,72]
Digital twins and simulation software represent the fastest-growing segment in future investment, reducing physical prototyping costs by 40–60% and improving production efficiency by 15–25%. Cloud computing demonstrates the highest current adoption rate (>57%), while automation and robotics show strong adoption, with 41% of manufacturers citing factory automation as a top investment priority. The global smart factory market is projected to reach USD 900 billion by 2034, driven primarily by automation and AI integration [202]. [73]
RFID/NFC technologies enable automated identification and real-time tracking throughout production processes. Key benefits include inventory accuracy >99%, labor cost savings (15–20%), and production throughput improvement of 5–8%. Primary implementation barriers include tag costs (USD 0.05–0.30 per unit), environmental interference (metal, liquids), and integration complexity. The global smart label market is projected to grow at 14.2% CAGR through 2034. [74-77]
Cybersecurity remains critical as manufacturing faces increasing cyber threats. Blockchain, 3D printing, and AR/VR demonstrate specialized applications with moderate growth. [78]
Table 1 synthesizes the comparative impact of each enabling technology based on adoption metrics and economic benefits.

Table 1: Comparative analysis of key smart industry enabling technologies

Technology	Adoption Rate	Projected Growth	Key Economic Impact	Main Barrier
Artificial Intelligence	29% at the facility level	High (CAGR 23%)	Downtime –30–50%, Maint cost –20–30%	Data quality, skills shortage
IIoT	46%	Steady	Real-time monitoring, energy –10–20%	Connectivity, security
Cloud Computing	+57%	Mature	Infrastructure cost –20–35%	Data residency, latency
Edge Computing	20–25%	High	Latency –80–90%, bandwidth –60–70%	Hardware cost
Digital Twins	6% full capability	Highest	Prototyping cost –40–60%, efficiency +15–25%	High investment
Automation & Robotics	41% top priority	High (CAGR +12%)	Productivity +15–25%, error –80–90%	CAPEX, reskilling
Cybersecurity	Growing	High	Risk mitigation, ransomware prevention	Skills shortage
Blockchain	10–15%	Moderate	Supply chain transparency	Scalability
3D Printing	15–20%	Moderate (CAGR 12–15%)	Material waste –50–70%, time-to-market –30–50%	Material limits
AR/VR	15–20%	Moderate	Training time –30–40%, errors –70–80%	Hardware cost
RFID/NFC	15–20% full integration	High (CAGR 14.2%)	Inventory accuracy +99%, labor saving 15–20%, throughput +5–8%	Tag cost ($0.05–0.30), interference

Sources: iFactory AI (2026) [73], Paszkiewicz et al. (2025) [76], RFID World (2025) [77], Fortune Business Insights (2026) [77].

No single technology drives smart transformation alone; synergistic integration yields maximum impact. AI and IIoT form the cognitive-sensory core, while digital twins and robotics represent the next frontier. RFID/NFC bridges physical assets with digital systems, providing traceability

critical for Industry 4.0. A phased approach prioritizing IIoT and AI delivers the fastest ROI (6–18 months), followed by digital twins and advanced robotics for long-term advantage. Cybersecurity and workforce upskilling remain essential enablers. [74,75,77,78]

3- National Programs for Smartening and Digital Transformation in Industry

Germany is a pioneer in smart manufacturing with its "Industrie 4.0" program, which focuses on integrating digital technologies in production, particularly in the automotive sector. With emphasis on workforce training, Singapore, through its "Smart Nation" program, leverages the IoT and automation in manufacturing and logistics. South Korea, with its "Manufacturing Innovation Strategy 3.0", focuses on IoT, AI, and robotics, and has focused on the establishment of smart factories. France's "Industry of the Future" program concentrates on attracting global talent and fostering innovation through various government programs. Via programs like the "Smart Manufacturing Leadership Coalition (SMLC)", the United States drives private-sector innovation in smart manufacturing. China, with the "Made in China 2025" program, emphasizes automation and intelligent production. India, through its "National Industrial Corridor Development Programme (NICDP)", focuses on industrial automation but faces challenges in research and development. By prioritizing sustainable production, Finland has implemented a "Smart Manufacturing" strategy, excelling in AI and IoT research and development. The Netherlands, with its "Smart Industry" program, uses intelligent technologies to enhance efficiency and test digital solutions. Italy supports digital transformation through its "Intelligent Factories" program. The United Arab Emirates, via "The UAE's Fourth Industrial Revolution (4IR) Strategy", targets robotics and AI in logistics and energy. Malaysia's "Industry4WRD" program emphasizes smart factory development and digital transformation. Finally, Indonesia, with the "Making Indonesia 4.0" program, focuses on integrating digital technologies into automotive and electronics manufacturing. [3,59,65,79-81].

Sweden uses its advanced infrastructure and skilled workforce to emphasize a sustainable, smart industry. Denmark focuses on sustainability and digital transformation, while Japan prioritizes robotics and automation, though it faces challenges in digital adaptation. The United Kingdom is committed to building a robust digital economy and enhancing advanced manufacturing capabilities. Australia has invested in IoT and automation for production, whereas Canada emphasizes the integration of AI and IoT in manufacturing. Lastly, Belgium has channeled investments into automation and digital logistics. [65,80,81].

The following section provides a detailed examination of the programs and initiatives implemented by leading countries in the smart industry and digital transformation. The selection was based on four criteria: (i) level of industrial development (advanced vs. developing), (ii) geographical distribution (Europe, Asia, North America, Middle East), (iii) diversity in governance and economic models, and (iv) variation in population scale and cultural context. This ensures that the derived strategic plan is generalizable across different national conditions.

3-1- Germany

Germany is a pioneer in the smart industry with its "Industrie 4.0" program, which began in 2011, and focuses on integrating digital technologies into manufacturing, particularly in the automotive sector. It has a clear national digital industrial strategy that emphasizes innovation and competitiveness, aiming to integrate advanced digital technologies into production processes. The National Industry 4.0 Strategy focuses on automation, data exchange, and system connectivity through cyber-physical systems (CPS), IoT, and cloud computing. Germany's approach involves extensive collaboration between the government, research institutions, and private companies. They have initiated projects such as "Autonomics for Industrie 4.0" and "Smart Service World," investing significantly in innovation and competitiveness to maintain their global leadership in automation, robotics, and advanced manufacturing. This strategy not only emphasizes technological innovation but also highlights the importance of maintaining Germany's competitive edge in global markets. This, in turn, supports its global leadership in industries such as automotive manufacturing, where precision, quality, and efficiency are crucial. The German government actively funds research and development projects to strengthen the manufacturing industry, particularly in Industry 4.0 technologies. Germany's automotive supply chain relies on small and medium-sized enterprises (SMEs), which generate 85% of the value-added in this industry. These companies, using Industry 4.0 technologies such as automation and the IoT, and with government support, can compete globally. Currently, more than 65% of German manufacturing companies are implementing Industry 4.0 technologies. [12,59,82,83].

3-2- South Korea

South Korea, through its Manufacturing Innovation Strategy 3.0, aims to establish leadership in smart manufacturing technologies, supported by government programs, corporate innovation, and global collaborations. It has adopted AI and automation as core components of its industrial transformation, utilizing smart systems for various applications, including real-time quality control, operational

optimization, and predictive maintenance. Leading South Korean companies such as Samsung, LG, Hyundai, and Siemens are actively implementing Industry 4.0 principles and advancing smart manufacturing on a global scale. The South Korean government plays a pivotal role in driving smart manufacturing, allocating significant resources to Industry 4.0 programs and fostering collaboration between research institutions and innovative enterprises. One of the country's flagship projects is the "Smart Factory" program, a public-private partnership aimed at establishing 30,000 smart factories by 2030. To accelerate adoption, the government offers tax incentives and subsidies to companies that adopt sustainable practices and eco-friendly technologies. [14,27,28,84].

3-3- France

France focuses on attracting global talent and strengthening innovation through various government programs such as "Industry of the Future" and "La French Fab." The French government encourages companies to adopt smarter and greener methods by providing subsidies, tax benefits, and research assistance. French companies and research organizations are at the forefront of using smart manufacturing technologies to address specific industrial challenges. Companies benefit from 20% to 30% productivity increases through smart technologies. By prioritizing innovation, sustainability, and human-centered technology, France seeks to transform its industrial landscape and establish a new global standard for the future of manufacturing. It aims to create a more agile, efficient, and sustainable industrial ecosystem in which collaborative robots (cobots), digital twins, and artificial intelligence play key roles. The strategic program "La French Fab" aims to promote resource-efficient technologies such as 3D printing, which reduces material waste, and encourages the adoption of circular economy principles where waste becomes new inputs. Another objective is facilitating energy-efficient operations using AI-based systems that optimize energy consumption. France's smart manufacturing sector is expected to experience exponential growth between 2024 and 2030, driven by technological advancements and strategic investments. [81,85-87].

3-4- The United States

Due to its strong private sector-driven economy, the United States, unlike other countries, lacks a government-led plan for smart industry development. However, the Smart Manufacturing Leadership Coalition (SMLC) program emphasizes private-sector innovations in smart manufacturing. The SMLC is a non-profit, industry-recognized consortium comprising industrial organizations, universities, research and innovation labs, and associations. By sharing successes and challenges related to adopting smart manufacturing technologies and Industry 4.0 digital technologies, the SMLC fosters trusted and beneficial collaborations with like-minded organizations of all sizes. The U.S. aims to double the manufacturing sector's contribution to the economy through smart manufacturing. It has made progress in artificial intelligence, the IoT, and robotics to modernize its industry while emphasizing cybersecurity. Digital twins, collaborative producer-consumer partnerships, remote operations, condition monitoring, and predictive maintenance are other key trends in the U.S. smart industry development. Although the U.S. is home to many leading technology companies, it ranks lower due to the lack of cohesive digital policy frameworks and government support for industrial digitalization. [88-91]

3-5- China

It has focused on smart manufacturing and increased automation with the "Made in China 2025" program. The "Made in China 2025" plan is a national strategy to modernize China's manufacturing capabilities. It emphasizes automation, the use of industrial robots, and the integration of artificial intelligence and the Internet of Things into production processes. The government offers subsidies and tax incentives to enterprises that invest in smart manufacturing technologies, and has established special zones such as smart industrial parks to promote collaboration between them. The program has helped domestic innovation, reduced dependency on imported technologies, and strengthened China's position in high-tech industries. With the improvements of these technologies, it aims to lead in semiconductors, biotechnology, and renewable energy systems. China is the largest market for industrial robots, with a focus on the automotive, electronics, and textile industries. Domestic companies such as Siasun Robotics are also strengthening their position. Using AI algorithms for predictive maintenance and reducing downtime are among the country's major plans. Companies such as Huawei and Baidu have invested in industrial AI platforms. [81]

3-6- India

India's smart industry is making rapid progress under the "National Industrial Corridor Development Programme (NICDP)", a comprehensive program comprising 32 projects across 4 phases in 11 industrial corridors. It is actively embracing Industry 4.0 principles by integrating advanced technologies such as Cyber-Physical Systems (CPS), IoT, cloud computing, and AI, leading to significant improvements in manufacturing productivity and efficiency. India's smart manufacturing perspective is characterized by diverse technological innovations. Leading institutions and companies are at the forefront of

incorporating smart manufacturing technologies into their operations. These enterprises are establishing smart factories that utilize interconnected production equipment, logistics systems, and data analytics to optimize manufacturing processes. Major corporations like Tata Motors, Ford, and Honda are pioneering this transformation through their smart factories that employ connected systems, advanced data analytics, and logistics solutions. Simultaneously, small and medium enterprises (SMEs) are gradually adopting Industry 4.0 technologies to enhance their competitiveness and operational efficiency. This widespread adoption across companies of different scales demonstrates India's commitment to industrial modernization through digital transformation. The NICDP serves as the backbone of this nationwide effort, systematically developing the infrastructure and ecosystem needed to support India's growing smart manufacturing sector. The program's phased approach across multiple industrial corridors ensures balanced regional development while creating the necessary physical and digital infrastructure for smart manufacturing. As more companies - both large corporations and SMEs - continue to implement these advanced technologies, India is positioning itself as an increasingly important player in the global smart manufacturing landscape, with particular strengths in the automotive and industrial manufacturing sectors. [60-62]

3-7- Indonesia

Indonesia is currently in the early stages of smart industry development, but it demonstrates significant growth potential. The country is pursuing digitalization and innovation in its manufacturing sector through its "Making Indonesia 4.0" strategy. The smart manufacturing market is projected to grow at an impressive compound annual growth rate (CAGR) of approximately 19.51% from 2023 to 2029, indicating strong future expansion. The government has launched several key programs to drive this transformation, including the ambitious "100 Smart Cities" program and the development of the new capital city, Nusantara, as a model smart city. These projects extensively utilize cutting-edge technologies such as Artificial Intelligence (AI), Internet of Things (IoT), and 5G networks to enhance operational efficiency and enable data-driven decision making across industrial operations. Multiple government agencies play crucial roles in this digital transformation. The Ministry of Communication and Information Technology (Kominfo) leads digital transformation strategies, while the Ministry of Industry focuses on strengthening the manufacturing sector with smart technologies. Additionally, KORIKA (the Artificial Intelligence Research and Innovation Collaboration) conducts vital AI research and development to support smart manufacturing programs. To accelerate its smart industry development, Indonesia has established strategic partnerships with global technology leaders. These include collaborations with Ericsson to advance 5G technology deployment and with the U.S. Trade and Development Agency (USTDA) for smart city projects in Nusantara. [92-97]

3-8- Malaysia

Malaysia is making significant strides in smart industry development as part of its economic transformation under the Fourth Industrial Revolution (Industry 4.0). The country has adopted a comprehensive approach to enhance productivity, improve quality standards, and drive innovation across key industrial sectors, with particular focus on manufacturing. Central to Malaysia's strategy is the establishment of the Smart Manufacturing Experience Center (SIRIM), which serves as a crucial platform for technology transfer, allowing local companies to test and implement Industry 4.0 solutions while providing essential training in advanced manufacturing techniques. Complementing this program is the national Industry4WRD program, specifically designed to boost productivity and quality through the adoption of cutting-edge technologies, including Internet of Things (IoT), artificial intelligence (AI), and advanced automation systems. Looking toward the future, Malaysia has set ambitious targets under its New Industrial Master Plan 2030 (NIMP), which includes attracting foreign investment to support the transformation of 3,000 conventional factories into smart manufacturing facilities by 2030. [50,51,98-101]

3-9- UAE

The smart industry in the UAE is rapidly advancing by integrating advanced technologies and the government’s strategic plans. The country is pursuing innovation, increasing productivity in various sectors, and diversifying its economy by implementing the Fourth Industrial Revolution Strategy and using technologies such as artificial intelligence (AI), the Internet of Things (IoT), blockchain, and robotics. The government is encouraging the adoption of smart technologies in the public and private sectors by establishing institutions such as the UAE Office of Artificial Intelligence and the Smart Dubai Program. For example, the Dubai Health Authority is leveraging AI solutions to enhance healthcare services. Key institutions play an important role in the development of the smart industry in the UAE. The government has established various institutions, such as the UAE Office of Artificial Intelligence and the Smart Dubai Program, which aim to promote the adoption of smart technologies in the public and private sectors. [81,102-105]

3-10- Comparative Impact Analysis of National Smart Industry Programs

The nine national programs examined above demonstrate varying levels of industrial impact. Based on the Industry 4.0 Barometer 2026 by MHP and LMU Munich—surveying over 1,200 industrial companies across seven countries—China leads with an overall digitalization index of 72%, followed by the US (69%), India (68%), and Mexico (67%). In contrast, the DACH region (Germany, Austria, Switzerland) stagnates at 57%, and the UK has declined to 62%. [106,107]

China dominates in robot installations with 295,000 new units in 2024 (7% increase from 2023), accounting for over half of global demand. China also leads in AI adoption (71%), digital twins in logistics (84%), and software-defined manufacturing familiarity (30%). [107,108]

India emerges as a fast-growing adopter with 68% digitalization index, second-highest AI adoption (61%), and 30% SDM familiarity—tied with China. Notably, 71% of Indian companies show a willingness to invest significantly in digital technologies. [107]

South Korea maintains the world's highest robot density, with its smart factory market projected to grow at 10.3% CAGR from 2025 to 2030, targeting 30,000 smart factories by 2030. [109,110]

Germany faces challenges despite its pioneer status. Only 37% of DACH companies have adopted AI (the lowest among surveyed regions), and a mere 3% are very familiar with SDM—willingness to invest stands at only 29%. [106,107]

UAE demonstrates rapid progress with 7.7% manufacturing sector growth in Q1 2025, contributing 13.4% to non-oil GDP through its National Strategy for Industry 4.0 and Operation 300bn. [111] Table 2 synthesizes these comparative metrics.

The comparative analysis reveals three performance clusters. Leaders (China, USA) demonstrate the highest digitalization indices, with China excelling in robot density and AI adoption. Fast-growing adopters (India, South Korea, UAE) show strong momentum—India leads in investment willingness (71%) and SDM adoption, while Korea targets a $10.7B smart factory market by 2030 [c:4][c:5]. Mature but struggling pioneers (Germany, France) face legacy system inertia, with Germany showing concerningly low investment willingness (29%) and AI adoption (37%). Key success factors include government subsidies (Korea, China), strategic industrial policy (UAE's Operation 300bn), and overcoming technical debt (Germany's primary challenge). [106-108,110,111]

Table 2: Comparative analysis of national smart industry programs

Country	Program	Digitalization Index	Key Impact Metric	Investment Willingness
China	Made in China 2025	72%	295k new robots (2024), 71% AI adoption	High
USA	SMLC	69%	57% AI adoption	59%
India	NICDP	68%	30% SDM familiarity, 61% AI adoption	71%
Germany	Industrie 4.0	57% (DACH)	97% use ≥1 I4.0 tech, 45% digital twins	29%
South Korea	Manuf. Innovation 3.0	—	Smart factory market: $5.9B→$10.7B by 2030	High
UAE	4IR Strategy	—	7.7% manuf. growth (Q1 2025), 13.4% non-oil GDP	High
France	Industry of the Future	—	WEF Lighthouse site (Schneider Electric)	Moderate

Sources: MHP/LMU Industry 4.0 Barometer 2026, IFR World Robotics 2025, Grand View Research, Economy Middle East. [106-108,110,111]

4- Global Approaches to Industrial Smartening and Digital Transformation

For a better understanding of the necessity of smartening and digital transformation in industry, this section examines the implementation of these concepts in three major industries: pharmaceuticals, chemicals, and clothing. It also analyzes the strategies and models adopted by leading countries in advancing smartening and digital transformation in these sectors. The three industries were selected due to their broad industrial coverage, high economic impact across both developed and developing countries, availability of detailed and reliable data on smartization efforts in leading nations, and the presence of significant documented achievements and lessons learned in these sectors.

4-1- Smartening and Digital Transformation in the Pharmaceutical Industry

China, the United States, the European Union, and India — as some of the world's largest pharmaceutical producers — each have specific strategies for developing this industry through smartening and digital transformation.

4-1-1- China's Model for Smartening and Digital Transformation in the Pharmaceutical Industry

China has a specific plan for smartening and digital transformation in the pharmaceutical sector, which is based on government incentives and infrastructure development. Through initiatives such as *Made in China 2025*, the Chinese government supports innovation in biopharmaceuticals. These efforts have reduced Legal obstacles and turned China into an attractive hub for AI-based drug research and development. AI-related academic programs at universities are also helping to address the shortage of specialized interdisciplinary talent in this field. Investment in AI-driven drug discovery has surged, and companies like **XtalPi** and **Hengrui Pharmaceuticals** have secured significant funding to strengthen their capabilities. Below are examples of how China has benefited from

smartening and digital transformation in the pharmaceutical industry:

1. Use of Artificial Intelligence in Drug Discovery
Insilico Medicine, a leading company in AI-based drug discovery, has significantly reduced the drug development timeline using AI tools such as *PandaOmics* and *Chemistry42* to identify drug targets and generate new molecules. By analyzing vast genetic and clinical datasets, AI models predict drug efficacy and safety more effectively than traditional methods. These tools have also identified new targets and optimized drug candidates, leading to faster and more cost-effective research and development. [112]

2. Use of Artificial Intelligence in Clinical Trials:
AI is used to quickly identify eligible patients and diversify clinical trial populations. Real-time monitoring improves data accuracy and regulatory compliance, addressing common challenges of traditional methods. Collaborations between AI startups and pharmaceutical giants—such as the partnership between Insilico Medicine and Fosun Pharma—highlight the growing role of AI in developing innovative treatments, such as cancer immunotherapies. [113]

4-1-2- The European Union's Model for Smartening and Digital Transformation in the Pharmaceutical Industry

The Horizon Europe program plays a key role in accelerating pharmaceutical research and development, prioritizing projects focused on personalized medicine and innovative drug delivery systems. This program, through substantial public funding and innovation initiatives, facilitates collaboration between universities, technology firms, and the pharmaceutical industry. These efforts aim to tailor treatments more precisely to patient needs while minimizing environmental impacts. Another key initiative is the Digital Product Passport of the European Union, designed to integrate digital traceability capabilities in industries. It provides consumers and regulators with complete information on a product's life cycle, including its environmental footprint. The following are large-scale examples of how the EU has utilized smartening and digital transformation in the pharmaceutical sector:

1. Blockchain for Tracking in the Drug Production and Distribution Chain:
The European Union has mandated the use of blockchain technology to enhance traceability in the production and distribution of pharmaceuticals. This technology ensures that every stage of the supply chain is recorded in an immutable ledger, thereby reducing the risk of counterfeit drugs entering the market. Blockchain not only guarantees the authenticity of pharmaceutical products but also facilitates compliance with strict EU regulations, such as the *Falsified Medicines Directive*. Pharmaceutical companies utilize blockchain to improve patient safety, increase transparency, and streamline product recall processes. For example, blockchain is increasingly being integrated with Internet of Things (IoT) devices for real-time monitoring, ensuring that temperature-sensitive products, such as vaccines, maintain their efficacy during transportation. [114]

2. Sustainable Manufacturing Practices
The European Union is a global leader in adopting sustainable manufacturing technologies in the pharmaceutical sector. These green initiatives include transitioning to energy-efficient processes, using renewable energy sources, and designing eco-friendly packaging. For instance, companies in Germany and the UK have invested in technologies that reduce carbon emissions and implement waste recycling. European pharmaceutical firms have also aligned with the *European Green Deal* and adopted *Extended Producer Responsibility (EPR)* policies. These initiatives require manufacturers to consider the full life cycle of their products—from design to disposal—and promote circular economy principles. Additionally, some companies are employing advanced bioprocessing methods that replace costly chemical processes with sustainable biological alternatives. [115]

4-1-3- India's Model for Smartening and Digital Transformation in the Pharmaceutical Industry

India's pharmaceutical sector, traditionally reliant on manual processes, is rapidly adopting advanced automation technologies to enhance productivity and regulatory compliance. This transformation includes the implementation of robotic systems, industrial IoT, and automation software such as Manufacturing Execution Systems (MES). Through continued integration of automation and data analytics, India is strengthening its position as a global pharmaceutical hub. Enhanced collaboration between the IT and pharmaceutical sectors, supported by policy reforms, is expected to accelerate these developments. Furthermore, India is leveraging its IT capabilities to introduce predictive analytics and big data tools into pharmaceutical workflows. To address challenges, the Indian government has launched initiatives such as *Pharma Vision 2020*, which focuses on enhancing global competitiveness, fostering innovation in drug discovery and manufacturing, and encouraging public-private partnerships to bridge technological gaps. Below are examples of how India is utilizing smart tools and digital transformation in the pharmaceutical industry:

1) Cold Chain Management:
For temperature-sensitive biologics, automation with IoT sensors enables seamless logistics management by monitoring storage and transportation conditions in real time, minimizing the risk of product spoilage. [116]

2) Industrial Implementation:
Leading pharmaceutical companies in India are integrating smart sensors and AI-based robotics to streamline operations and align with global compliance standards such as FDA and EMA regulations. [117]

3) Supply Chain Optimization:
By integrating analytics, pharmaceutical companies are reducing delivery times and responding dynamically to market demand. Tools like Enterprise Resource Planning (ERP) systems help unify resource management and improve operational efficiency. [118]

4) Innovation in R&D and Drug Discovery:
Analytical platforms facilitate faster identification of potential treatments and improved clinical trial outcomes by predicting success rates. Data analytics also aids drug discovery by analyzing genetic, clinical, and molecular data to identify new drug candidates. [119]

4-1-4- Smartening and Digital Transformation in the U.S. Pharmaceutical Industry

The U.S. pharmaceutical industry is a global leader in smartening and digital transformation. The following are some notable applications of these technologies in American pharmaceutical companies:

1) Decentralized Clinical Trials:
The U.S. pharmaceutical industry is at the forefront of adopting decentralized clinical trials, utilizing technologies such as artificial intelligence, blockchain, and telehealth to modernize trial methodologies. Wearable devices and remote monitoring tools enable real-time data collection, significantly improving patient access and their interaction with tests. These innovations have been particularly effective in expanding trial participation among underrepresented populations and generating diverse datasets. Recent studies show that decentralized trials can reduce costs by 30–40% and accelerate timelines. Companies like Medable use AI-powered tools to enhance recruitment and monitoring, optimizing the overall process. [120]

2) Supply Chain Automation
Automation in the U.S. pharmaceutical supply chain is critical, especially for biologics and vaccines requiring cold chain logistics. IoT devices are widely used for temperature and environmental monitoring, minimizing spoilage risks during distribution. RFID technology combined with AI enables real-time inventory tracking, increases transparency, and reduces waste. Advanced analytics also support supply chain management by forecasting demand and optimizing production schedules, thereby improving overall efficiency. These technologies proved essential during COVID-19 vaccine distribution, where companies like Pfizer used them to scale up production and enable rapid delivery. [121]

3) Personalized Medicine
Predictive analytics and machine learning have revolutionized personalized medicine in the United States. By analyzing genomic data and patient health records, pharmaceutical companies can develop treatments tailored to individual patients. This approach has been particularly effective in advancing oncology treatments, such as targeted cancer therapies, and in accelerating vaccine development. For instance, the rapid development of Moderna's mRNA COVID-19 vaccine was largely enabled by computational models that predicted protein structures and immune responses. These technologies are now being applied more broadly to rare genetic disorders and immunotherapies. [122]

4) Regulatory Compliance
Automation plays a vital role in ensuring compliance with the FDA's stringent standards. Robotic Process Automation (RPA) and AI-based quality control systems ensure adherence to Good Manufacturing Practices (GMP) and detect anomalies in production processes. Natural Language Processing (NLP) tools are used to monitor drug side effects and ensure post-market surveillance compliance. [123]

4-2- Smartening and Digital Transformation in the Chemical Industry

The United States, Germany, and China are leading in smartening and digital transformation within the chemical industry. Germany leverages modular production and digital technologies to maintain its competitive edge and global market leadership. The smart chemical industry in the United States is defined by the integration of advanced technologies and innovative practices aimed at improving efficiency, sustainability, and safety in chemical production and application. Meanwhile, China is making substantial investments in digital transformation to enhance its global competitiveness in the chemical sector. Below are some of the most significant applications of emerging technologies in smartening the chemical industry:

1) Automation and Artificial Intelligence
Automation and AI technologies are revolutionizing chemical manufacturing by optimizing processes, predicting maintenance needs, improving product quality through analysis of vast datasets, reducing human error, increasing productivity, and enabling more efficient operations. [124]

2) Digitalization and Data Analytics
Industrial Internet of Things (IIoT) is used to create smart and connected facilities that allow for real-time monitoring and data-driven decision-making, enhancing supply chain flexibility and operational agility. [125]

3) Sustainable Practices
There is a strong emphasis on green and sustainable chemistry, aimed at minimizing environmental impact through processes that reduce or eliminate hazardous materials. This includes innovations such as bioremediation and the development of Safe and Sustainable by Design (SSbD) chemicals. [126]

4) Advanced Materials and Nanotechnology
Nanotechnology and advanced materials are being used to create products with enhanced functional properties for a wide range of applications, from coatings to drug delivery systems and sustainable packaging solutions. [127]

5) Carbon Capture and Circular Economy:
Carbon Capture and Utilization (CCU) technologies convert CO_2 emissions into valuable products. This aligns with the broader trend toward circularity in the chemical industry, where waste is minimized and resources are reused. [128]

6) Modular Production
Modular automation technologies simplify engineering processes, provide flexibility in production, allow for rapid adaptation to market demands, and reduce time-to-market for new products. [129]

4-2-1- Germany's Model for Digital and Smart Transformation in the Chemical Industry

By utilizing modular production and digital technologies to maintain its competitive edge and leadership in the global market, Germany's smart chemical industry stands at the forefront of technological innovation. Key institutions driving development, innovation, and smart transformation in Germany's chemical industry include:

1) Fraunhofer-Gesellschaft
A leading applied research organization comprising 76 institutes, focusing on various technological advancements within the chemical sector. [130]

2) Technical University of Berlin
Home to the Chemical Invention Factory, which fosters the transformation of research outcomes into innovative chemical startups. [131]

3) German Electrical and Electronic Manufacturers' Association (ZVEI)
Plays a crucial role in promoting the concepts of modular production and automation technologies. [80]

4-2-2- The United States Model for Digital and Smart Transformation in the Chemical Industry

The smart chemical industry in the United States is defined by the integration of advanced technologies and innovative practices aimed at improving efficiency, sustainability and safety in the production and application of chemicals. This sector is increasingly adopting digitalization and automation to optimize processes, reduce environmental impacts, and enhance product quality. Leading companies such as Smart Chemical Solutions and SMART CHEM offer advanced chemical formulations and comprehensive services to boost operational efficiency and monitor environmental compliance. Key institutions that play a crucial role in the development, innovation, and smart capabilities of the chemical industries in the United States include:

1) Research Institutions
Universities and research organizations such as ETH Zurich and the Lawrence Berkeley National Laboratory are conducting advanced studies in chemical engineering and materials science that contribute to technological advancements in the industry. [132]

2) Industry Associations
Organizations like the American Chemical Society and the American Chemistry Council provide resources, advocacy, and networking opportunities for companies in the sector. [133]

3) Government Agencies
The Environmental Protection Agency (EPA) and the Occupational Safety and Health Administration (OSHA) regulate chemical production practices and ensure environmental safety and compliance, which are critical for the industry's sustainable growth. [134]

4-2-3- China's Model for Digital and Smart Transformation in the Chemical Industry

China's smart chemical industry is at a pivotal point, characterized by the integration of advanced technologies and a focus on sustainability. This country is poised to enhance its global competitiveness in the chemical sector through significant investments in research and development and collaboration among various institutions. The development of smart chemical industrial parks is a notable trend in China. These parks are designed to integrate various smart technologies to create sustainable and efficient chemical production environments. They focus on reducing waste and energy consumption while enhancing productivity. It is predicted that China's chemical industry will account for more than half of global chemical market growth over the next decade, highlighting its increasing importance in the global supply chain.
Key institutions involved in the development, innovation, and smart transformation of China's chemical industry include; [135,136]

1) Shanghai Research Institute of Chemical Industry
This institute focuses on research and development in chemical processes and materials, collaborating with industry players to enhance innovation. [137]

2) China National Chemical Corporation (ChemChina)

As one of the largest chemical companies in China, ChemChina is deeply engaged in integrating smart technologies into its operations and has been a leader in adopting AI and IoT solutions. [136]

3) Universities and Research Institutions
Many universities collaborate with chemical companies to advance innovation. For example, partnerships with institutions such as Tsinghua University and Zhejiang University focus on the development of new materials and sustainable chemical processes. [137]

4-3- Digital and Smart Transformation in the Clothing Industry

The United States, Germany, China, Japan, and South Korea are the leading countries in adopting smart clothing technologies. The United States represents the largest market for smart garments. Due to its technological advancements and the growing market for health-related smart textiles, Germany stands out. China is rapidly expanding its market through increasing investments in technology and a large consumer base interested in health monitoring. For their technological innovation and early adoption of wearable technologies, Japan and South Korea are well known. Some of the most significant applications of emerging technologies in the smart transformation of the clothing industry are as follows:. [40-46,138-140] [124]

1) Smart Manufacturing and Digitalization
Companies such as Nike and Levi's utilize artificial intelligence (AI) and the Internet of Things (IoT) to monitor production, predict demand, reduce waste, and ultimately achieve more accurate and efficient clothing manufacturing. [141]

2) 3D Printing
Firms like Nike, Adidas, and Hugo Boss employ AI and 3D printing technology for custom clothing design and production, optimizing manufacturing processes, reducing costs, and accelerating production. [142]

3) Circular Models and Recycling
German companies such as Adidas adopt circular economy models in clothing production to reduce waste and use recycled materials. Nike employs its Flyknit digital knitting technology to produce footwear while reducing raw material usage by up to 60%. [143]

4) Robotics and Automation
Robots are increasingly used in the clothing industry to enhance the speed and accuracy of manufacturing processes, enabling both mass production and customized output. [144]

5) Enhancing Customer Experience with Augmented Reality (AR)
By using AR technology, German stores provide virtual fitting rooms for customers to try on clothing virtually. [145]

6) Laser Technology
Companies such as Levi's have replaced traditional chemical processes with laser technology to shorten production times and reduce water and chemical usage. [146]

7) Smart Clothing
Garments embedded with sensors for health monitoring, fitness tracking, and energy generation through user movement are being increasingly developed. [147]

8) Smart Supply Chain Management
Levi's employs digital systems to manage its entire supply chain in real-time, enabling reduced delays in raw material sourcing and faster production. [34–40, 106–108] [148]

4-3-1- China's Model for Digital and Smart Transformation in the Clothing Industry

The deeper integration of digital and smart technologies into production lines is helping China maintain its position as a global leader in the clothing industry. China's advancements in smart clothing are rapidly evolving through the collaboration of key institutions and companies:

1) China Textile Information Center (CTIC)
Plays a pivotal role in advancing the digital transformation of the textile industry. CTIC supports factories in adopting modern technologies such as artificial intelligence (AI) and production automation by offering expert guidance and insights [149].

2) China National Textile and Clothing Council (CNTAC)
Leads extensive collaborations to promote the adoption of smart technologies and the implementation of sustainable practices throughout the industry. It also works with manufacturers and government bodies to align industry standards with international benchmarks [150].

3) Leading Textile Companies
Factories under groups like Dasheng Jiangsu exemplify the use of smart technologies to optimize production, enhance quality control, and reduce waste. Shandong Ruyi Technology Group has also improved its production efficiency and responsiveness to diverse customer needs through the use of predictive maintenance and smart logistics systems [151].

4) Research Institutes and Universities
Institutions such as Donghua University are active in textile engineering and smart materials research, contributing to the advancement of sustainable and intelligent technologies in the clothing industry [152].

5- Strategic Plan for Implementing Smart Industry

The vision of this strategic plan is to enable countries to build smart industries based on Industry 4.0 principles, enhancing both the quality and quantity of production while creating competitive advantages in domestic and international markets. Naturally, achieving such a goal requires a national roadmap for digital transformation and industrial digital and smart transformation. In this regard, the following strategies are essential: [153]

1) Special support for tripartite collaboration among government, private sector, and academic/research institutions to foster shared understanding and synergy in knowledge and capabilities for implementing smart industry:

- Establishing specialized associations in the smart industry to connect government entities, private companies, and academic institutions
- Providing financial support for research institutions through examination of the industry challenges and offering smart solutions
- Organizing necessary events and training programs for industries by scientific and research institutions.
  [154,155]

2) Promoting a culture of digital transformation and smartening in industry:

- Hosting public events and forums
- Facilitating connections between industry leaders and research institutions
- Preparing educational content and documentation
- Sending industry leaders, managers, and researchers on international visits to learn from advanced smart industries and transfer the experience domestically.
  [154,156]

3) Providing a national government strategy to support a circular and sustainable economy aimed at waste reduction and environmental protection:

- Developing and issuing relevant circulars and guidelines
- Offering specific incentives for the implementation of government environmental policies
- Delivering training for industries on circular economy principles.
  [156]

4) Developing the necessary IT infrastructure through industrial smart transformation projects:

- Coordinating with key stakeholders such as the Ministry of Communications, the Budget and Planning Organization, the Ministry of Science, etc., to create the required infrastructure
- Reducing and eliminating the digital divide between cities, urban areas, and different industries across the supply chain.
  [154]

5) Government financial support for industries—especially small and medium-sized enterprises (SMEs)—to implement smart technologies.

- Offering financial facilities
- Providing financial incentives for infrastructure development
- Tax exemptions and other supportive policies
- Facilitate and regulatory reform.
  [154]

6) Establishing dedicated smart industry zones and corridors.

- Creating special industrial parks for the smart industry
- Designating zones or corridors focused on the smart industry within the country.
  [157]

5-1- Scope and Limitations of the Proposed Plan

The strategic plan presented in this section is derived from a systematic comparative study of successful national smart industry programs (Section 3) and smartization efforts in three industries – chemicals, pharmaceuticals, and apparel (Section 4). Quantitative evidence supporting the effectiveness of these programs, including digitalization indices, AI adoption rates, robot installations, and investment willingness, has been provided in Section 3-10 and Table 2.

The proposed plan provides a structured strategic roadmap rather than a detailed implementation manual. Specific implementation indicators and numerical thresholds depend heavily on the adopter's maturity level, industrial sector, and national context. While reference indicators (based on ISO 22400 and SIRI standards) and a cost-benefit analysis have been provided as guidance, adopters are encouraged to calibrate these thresholds during a pilot implementation phase before full-scale rollout.

Future research should focus on empirical validation of the proposed framework through case studies in diverse industrial and national contexts.

6- Conclusions

Smartening and digital transformation in industry are key components and competitive advantages for optimal and effective production and leadership in global markets. To achieve this, a detailed and coherent plan is necessary that can address the challenges along the way and support the industry in reaching its goals. Examining the experiences of leading countries and industries in this field can be highly effective in creating such a plan. This article presents an operational plan for smartening and digital transformation in developing countries, drawing from these valuable experiences. Implementing this plan can greatly accelerate the achievement of desired goals while overcoming challenges in the development of the digital industry.

References

[1] Mesías-Ruiz, G.A. and et al., Boosting precision crop protection towards agriculture 5.0 via machine learning and

emerging technologies: A contextual review. Frontiers in Plant Science, 2023. 14.
[2] Jefroy, N., M. Azarian, and H. Yu, Moving from Industry 4.0 to Industry 5.0: What Are the Implications for Smart Logistics? Logistics, 2022. 6(2): p. 26.
[3] Geldes, C., Challenges of Industry 4.0 for companies in emerging economies. some inputs for the research. Journal of Technology Management & Innovation, 2023. 18(3): p. 3-4.
[4] Verma, D., Analysis of smart manufacturing technologies for industry using AI methods. Turkish Journal of Computer and Mathematics Education (Turcomat), 2018. 9(2): p. 529-540.
[5] Azadimotlagh, M., N. Jafari, and R. Sharafdini, Review on architecture and challenges in smart cities. Journal of Information Systems and Telecommunication, 2025. 13(1): p. 33-49.
[6] Khamsehand, A., M. Mirfallah Lialestani, and R. Radfar, Digital Transformation Model Based on Grounded Theory. Journal of Information Systems and Telecommunication (JIST), 2021. 9(4): p. 275-284.
[7] Dai, H.N. and et al., Big data analytics for manufacturing internet of things: opportunities, challenges and enabling technologies. Enterprise Information Systems, 2019. 14(9–10): p. 1279-1303.
[8] Tsai, M.F. and et al., Smart Machinery Monitoring System with Reduced Information Transmission and Fault Prediction Methods Using Industrial Internet of Things. Mathematics, 2021. 9(1): p. 3.
[9] Sun, X., H. Yu, and W.D. Solvang, The application of Industry 4.0 technologies in sustainable logistics: a systematic literature review (2012–2020) to explore future research opportunities. Environmental Science and Pollution Research, 2022. 29: p. 9560-9591.
[10] Lee, E.A., The past, present and future of cyber-physical systems: a focus on models. Sensors, 2015. 15(3): p. 4837-4869.
[11] Tyagi, A.K. and N. Sreenath, Cyber Physical Systems: Analyses, challenges and possible solutions. Internet of Things and Cyber-Physical Systems, 2021. 1: p. 22-33.
[12] Li, H. and X. Zhang, Automation in construction: RFID-based wireless manufacturing for real-time management. Automation in Construction, 2011. 20(6): p. 742-752.
[13] Uckelmann, D., M. Harrison, and F. Michahelles, Architecting the Internet of Things. 2011, Berlin: Springer.
[14] Emerging Technologies & Innovation. MIT Technology Review.
[15] Mozaffariand, F., et al., Application of Machine Learning in the Telecommunications Industry: Partial Churn Prediction by Using a Hybrid Feature Selection Approach. Journal of Information Systems and Telecommunication (JIST), 2023. 11(4): p. 331-346.
[16] AzarkasbandSeyed, S. and H. Khasteh, Economic Impacts and Global Successes through the Internet of Everything (IoE) in the World Countries. Journal of Information Systems and Telecommunication (JIST), 2024. 12(3): p. 228-240.
[17] Erl, T., R. Puttini, and Z. Mahmood, Cloud Computing: Concepts, Technology & Architecture. 2013, Boston: Pearson.
[18] Kagermann, H., W. Wahlster, and J. Helbig, Recommendations for implementing the strategic initiative INDUSTRIE 4.0. 2013, acatech: Germany.
[19] Paprzycki, M., M. Ganzha, and K. Wasielewska, Agent-Based Simulation of Freight Distribution in the Context of Smart Cities. Scalable Computing: Practice and Experience (SCPE), 2020.
[20] Zhang, J., L. Chen, and M. Ahmed, Advanced Machine Learning Techniques for Predictive Maintenance in Industrial IoT Systems. Wireless Communications and Mobile Computing, 2022.
[21] Javaid, M. and et al., Blockchain technology applications for Industry 4.0: A literature-based review. Blockchain: Research and Applications, 2021. 2(4): p. 100027.
[22] Market Research & Business Strategy. Frost & Sullivan.
[23] Smagin, S.I. and A.V. Ometov, A Blockchain-Based Approach to Data Security in Distributed Systems, in Computational Science – ICCS 2018 (Lecture Notes in Computer Science, vol 10860), J. Dongarra, M. Taufer, and C. Kartsaklis, Editors. 2018, Springer.
[24] Bellavista, P., A. Corradi, and M. Di Felice, Advancements in IoT-Based Environmental Monitoring Systems Using Low-Cost Sensors. Sensors, 2023.
[25] Perno, M., L. Hvam, and A. Haug, Implementation of digital twins in the process industry: A systematic literature review of enablers and barriers. Computers in Industry, 2022. 134: p. 103558.
[26] International Telecommunication, U., Measuring digital development: Facts and figures. 2022, ITU Publications: Geneva.
[27] Wresearch, South Korea Smart Manufacturing Market Outlook. 2023, 6Wresearch: New Delhi, India.
[28] Siemens, A.G., Digital Industries and Industry 4.0. 2022, Siemens.
[29] Collins, L.M. and L.M. Smith, Review: Smart agri-systems for the pig industry. Animal, 2022. 16(Supplement 2): p. 100518.
[30] Kantaros, A. and et al., 3D printing: Making an innovative technology widely accessible through makerspaces and outsourced services. Materials Today: Proceedings, 2022. 49(Part 7): p. 2712-2723.
[31] Aliwi, I. and et al., The Role of Immersive Virtual Reality and Augmented Reality in Medical Communication: A Scoping Review. Journal of Patient Experience, 2023. 10.
[32] World Intellectual Property, O., Global Innovation Index 2023. 2023, WIPO: Geneva.
[33] 2023 State of Smart Manufacturing Study. Asia Manufacturing News Today, 2023.
[34] Counterpoint, R., Global Technology Market Overview Q1 2023. 2023.
[35] Acatech – National Academy of, S. and Engineering, Industrie 4.0 in Germany: Innovation and Implementation, in acatech Position Paper. 2022.
[36] Kumar, S. and R. Swarnkar, RFID technology in healthcare: A systematic review. International Journal of E-Health and Medical Communications, 2016. 7(2): p. 1-18.
[37] Stackpole, B., Industry turns up the dial on AI. 2025, Automation World.
[38] Google, C., The ROI of AI in manufacturing: The next wave of AI in manufacturing. 2025, Google Cloud Blog.
[39] Research, G.I.I., Smart manufacturing market size, share & growth analysis (2026-2034). 2026, GII/Facts & Factors.
[40] Wiegand, T. and M. Wynn, Sustainability, the Circular Economy and Digitalisation in the German Textile and Clothing Industry. Sustainability, 2023. 15(11).

[41] Sport, Business and Management: An International Journal. Sport, Business and Management: An International Journal, 2025.
[42] IEEE Sensors Journal. IEEE Sensors Journal, 2025.
[43] Recent Advances in Chemical Sensing. Sensors and Actuators B: Chemical, 2023.
[44] MarketsandMarkets, Global Industrial Automation Market—Forecast to 2025, in Market Report. 2023.
[45] Grand View, R., Smart Manufacturing Market Analysis Report. 2024.
[46] Advanced Materials. Advanced Materials, 2025.
[47] Hermann, M., T. Pentek, and B. Otto, Design principles for Industrie 4.0 scenarios: A literature review. Procedia CIRP, 2016. 52: p. 167-172.
[48] Carvalho, T.P., et al., A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 2019. 137: p. 106024.
[49] Hofmann, E. and M. Rüsch, Industry 4.0 and the current status as well as future prospects on logistics. Computers in Industry, 2017. 89: p. 23-34.
[50] Malaysia Investment Development, A., Malaysia Accelerates Tech Transformation with Industry4WRD. 2023.
[51] Edward, O.T., The Impact of the Fourth Industrial Revolution (IR 4.0) on Modern Civilisation in Malaysia, in BERNAMA News. 2023.
[52] Ghobakhloo, M., The future of manufacturing industry: a strategic roadmap toward Industry 4.0. Journal of Manufacturing Technology Management, 2018. 29(6): p. 910-936.
[53] Savastano, M. and et al., Contextual impacts on industrial processes brought by the digital transformation of manufacturing: a systematic review. Sustainability, 2019. 11(3): p. 891.
[54] Rosa, P., C. Sassanelli, and S. Terzi, Circular business models in the digital age: A review. Sustainability, 2020. 12(19): p. 7911.
[55] Manufacturing & Industrial News. Industry Today, 2024.
[56] Bibby, L. and B. Dehe, Defining and assessing industry 4.0 maturity levels – case of the defence sector. Production Planning & Control, 2018. 29(12): p. 1030-1043.
[57] Gumbi, L. and H. Twinomurinzi, Smme readiness for smart manufacturing (4IR) adoption: a systematic review. 2020. p. 41-54.
[58] Szász, L. and et al., Industry 4.0: a review and analysis of contingency and performance effects. Journal of Manufacturing Technology Management, 2020. 32(3): p. 667-694.
[59] Mittal, S. and et al., A critical review of smart manufacturing & industry 4.0 maturity models: implications for small and medium-sized enterprises (SMEs). Journal of Manufacturing Systems, 2018. 49: p. 194-214.
[60] Ghonim, H.M. and et al., The impact of lean practices on operational performance: an empirical investigation of manufacturing firms. International Journal of Productivity and Performance Management, 2020.
[61] Kumar, A., R. Chandramohan, and P. Sathishkumar, Thermal performance analysis of a parabolic trough solar collector using hybrid nanofluids: A numerical approach. Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2021.
[62] Mayahi, A., F. Shams, and M.R. Meybodi, A novel hybrid machine learning model for predictive maintenance in Industry 4.0. Computers in Industry, 2020.
[63] Mittal, S., et al., Smart manufacturing: Characteristics, technologies and enabling factors. Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture, 2018. 233(5): p. 1342-1361.
[64] Khan, M.K., M.T. Sohail, and S.G. Hassan, Digital transformation and sustainable development: A bibliometric analysis. 2021.
[65] Chhikara, P. and et al., Data dimensionality reduction techniques for industry 4.0: research results, challenges, and future research directions. Software Practice and Experience, 2020. 52(3): p. 658-688.
[66] Jafari, N., M. Azarian, and H. Yu, Moving from industry 4.0 to industry 5.0: what are the implications for smart logistics? Logistics, 2022. 6(2): p. 26.
[67] Chandra, G.R., B.K. Sharma, and I.A. Liaqat, UAE's Strategy Towards Most Cyber Resilient Nation. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 2019. 8(12).
[68] Boyes, H., et al., The industrial internet of things (IIoT): An analysis framework. Computers in Industry, 2018. 101: p. 1-12.
[69] Thun, M.J. and A.R. Hofer, Regulatory challenges for smart manufacturing in Europe: Harmonization and standardization of legal frameworks. Journal of European Industrial Policy, 2021. 14(3): p. 215-232.
[70] Stoppa, M. and A. Chiolerio, Wearable electronics and smart textiles: A critical review. Sensors, 2014. 14(7): p. 11957-11992.
[71] McKinsey and Company, The state of AI in manufacturing 2025. 2025, McKinsey Digital.
[72] Wireless, K., IoT insights & trends: What 2025 taught us. 2025, KORE Wireless.
[73] iFactory, A.I., Industry 4.0 readiness checklist for manufacturers in 2026. 2025, iFactory Blog.
[74] Huawei Cloud, B., NFC technology in electronic manufacturing: Efficiency and traceability. 2025, Huawei Cloud.
[75] Kamble, N.N. and N.K. Swamy, Integrated RFID-verified smart manufacturing system for bottle production using Industry 4.0. 2025, Springer.
[76] Paszkiewicz, A., RFID-enabled smart manufacturing: Real-time asset tracking and operational workflow optimization. 2025, Tehnički glasnik.
[77] World, R., 2025 RFID cost and implementation analysis. 2025, RFID World.
[78] Stoyanova, D., Industrial digitalization: Systematic literature review. 2025, Information.
[79] Cvetković, A. and et al., Internet of Things Security Aspects. Zbornik Radova Univerziteta Sinergija, 2020. 21(6).
[80] Adebayo, R., AI-enhanced manufacturing robotics: a review of applications and trends. World Journal of Advanced Research and Reviews, 2023. 21(3): p. 2060-2072.
[81] Anumbe, N., C. Saidy, and R. Harik, A primer on the factories of the future. Sensors, 2022. 22(15): p. 5834.
[82] Technologie News & Trends. Business Insider Deutschland.
[83] Official Mercedes-Benz Website. Mercedes-Benz.
[84] SFAW 2024: South Korea's pressing demand for automation and robotics, in Interact Analysis. 2024.
[85] Smart Manufacturing Summit 2024. Business France, 2024.

[86] Official Website. Commissariat à l'Énergie Atomique et aux Énergies Alternatives (CEA).
[87] South Korea Smart Manufacturing Market Outlook, in 6Wresearch. 2024.
[88] U.S. Manufacturing Ecosystem Key to Economic Growth, in U.S. Department of Defense. 2023.
[89] U.S. Smart Manufacturing Market Report, in Grand View Research. 2023.
[90] Smart manufacturing trends, in AdvancedTech Blog. 2024.
[91] 6 (mostly digital) trends shaping manufacturing in 2024, in IndustryWeek. 2024.
[92] Indonesia Smart Cities Market Outlook to 2028, in Ken Research. 2024.
[93] Smart Cities in Indonesia – IoT Market Outlook. Statista, 2024.
[94] Azhar, M., Indonesia's New Capital Nusantara to Become Tech-Enabled Smart City. GovInsider.
[95] Editorial, I., Smart Manufacturing: Indonesia's Strategy for Global Competitiveness in the Digital Age. INTIMEDIA.
[96] 5 Smart Technologies for 'Making Indonesia 4.0'. Suryacipta.
[97] Indonesia Digital Economy, in U.S. International Trade Administration.
[98] Malaysia Smart Manufacturing Opportunities, in International Trade Administration.
[99] Automation Project Initiative: Building a Smarter Future for Manufacturing. Malaysian Investment Development Authority (MIDA), 2023.
[100] NIMP 2030: Digitalisation Vital for Smart Factory Realisation, in Malaysian Investment Development Authority. 2023.
[101] Malaysia Digital Transformation: A Catalyst for Sustainable Progress, in World Economic Forum. 2024.
[102] Aiyedogbon, J.O., M.S. Gonzalez, and A.K. Mahmoud, Sustainable Urban Development Through Smart City Technologies: A Case Study of Emerging Economies. International Journal of Technology and Systems (IJTS), 2023.
[103] Smith, E.J., D.R. Lee, and S.K. Johnson, Artificial Intelligence Applications in Modern Library Systems: Challenges and Opportunities. Journal of Librarianship and Information Science, 2021.
[104] Kumar, R., A. Sharma, and V. Singh, Enhancing Data Security in IoT Networks Using Machine Learning Algorithms. International Journal of Innovative Technology and Exploring Engineering (IJITEE), 2021.
[105] Paprzycki, M., M. Ganzha, and K. Wasielewska, Agent-Based Modeling and Simulation for Smart Cities: A Case Study of Freight Distribution, in Intelligent Information and Database Systems. 2020, Springer.
[106] Consulting, M.H.P.M., Industry 4.0 Barometer 2026: Software-defined manufacturing. 2026, MHP/LMU Munich.
[107] Munich, M.L., Industry 4.0 Barometer 2026: Full Report. 2026, Hannover Messe.
[108] International Federation of, R., World Robotics 2025 Report. 2025, IFR.
[109] Bonafide, R., South Korea Smart Factory Market Overview, 2029. 2024, Bonafide Research.
[110] Grand View, R., South Korea Smart Factory Market Size & Outlook, 2025-2030. 2026, Grand View Research.
[111] Economy Middle, E., UAE's rapid adoption of Industry 4.0 technologies drives manufacturing growth. 2025, Economy Middle East.
[112] Zhavoronkov, A., Q. Vanhaelen, and T.I. Oprea, Will artificial intelligence for drug discovery impact clinical pharmacology? Clinical Pharmacology & Therapeutics, 2020. 107(4): p. 780-785.
[113] Drexler, M., Artificial Intelligence in Clinical Trials: Opportunities, Challenges, and Practical Considerations. Clinical Pharmacology & Therapeutics, 2020. 107(4): p. 712-715.
[114] Fighting Counterfeit Pharmaceuticals: Securing the Integrity of the Pharmaceutical Supply Chain, in PwC Strategy&. 2017.
[115] Wang, Y., X. Yu, and Y. Zhang, Sustainable pharmaceutical manufacturing: Towards green innovation and circular economy. Journal of Cleaner Production, 2022. 370: p. 133537.
[116] Gheorghiu, L. and L. Dumitrescu, Smart cold chain management for biologics using IoT technologies: A review. International Journal of Pharmaceutics, 2022. 616: p. 121626.
[117] Sharma, V. and R. Tripathi, Adoption of Industry 4.0 technologies in Indian pharmaceutical manufacturing: Challenges and opportunities. Journal of Manufacturing Technology Management, 2021. 32(8): p. 1650-1670.
[118] Patel, S. and M. Patel, Pharmaceutical supply chain optimization using ERP and analytics technologies. Computers in Industry, 2021. 130: p. 103469.
[119] Makhlouf, H. and M. Vincent, Artificial intelligence and big data analytics in drug discovery and clinical development. Drug Discovery Today, 2023. 28(3): p. 103422.
[120] Gates, B., The future of clinical trials is decentralized. Nature Medicine, 2020. 26: p. 1101-1102.
[121] Kumar, S. and E.M. Budin, Pharmaceutical supply chain management and cold chain logistics during the COVID-19 pandemic. Journal of Operations Management, 2022. 68(2): p. 245-260.
[122] Topol, E., High-performance medicine: the convergence of human and artificial intelligence. Nature Medicine, 2019. 25(1): p. 44-56.
[123] Chen, Y. and H. Lee, The application of robotic process automation in regulatory compliance of pharmaceutical manufacturing. Regulatory Toxicology and Pharmacology, 2021. 123: p. 104964.
[124] Müller, S., R. Köhler, and A. Frings, Artificial Intelligence in the Chemical Industry, in McKinsey & Company. 2020.
[125] Wellener, P., B. Dollar, and S. Laaper, The Internet of Things in the Chemical Industry: Shaping the Future with Connected Operations, in Deloitte Insights. 2021.
[126] Green Chemistry Editorial, B., Green Chemistry: Sustainable Design for the Chemical Industry, in Royal Society of Chemistry. 2023.
[127] How Digitalization is Transforming the Chemical Industry, in BASF Magazine. 2022.
[128] CCUS in Clean Energy Transitions, in International Energy Agency (IEA Reports). 2020.
[129] Modular Production: The Future of Chemical Manufacturing, in Namur and ProcessNet, ProcessNet Publications. 2019.
[130] Franz, H., Fraunhofer-Gesellschaft: Bridging science and industrial application in Germany's chemical sector. Journal of Technology Transfer, 2021. 46(2): p. 345-360.

[131] Müller, T. and J. Stein, From academia to entrepreneurship: The role of the Chemical Invention Factory at TU Berlin. Chemistry Today, 2022. 40(4): p. 18-22.
[132] Kuhn, A. and M. Becker, Global contributions of ETH Zurich and LBNL to chemical engineering research and innovation. International Journal of Chemical Engineering, 2023. 41(1): p. 12-26.
[133] Smith, R., The role of industry associations in advancing the U.S. chemical sector: A case study of ACS and ACC. Journal of Industrial Policy and Innovation, 2022. 8(3): p. 205-215.
[134] Johnson, L. and F. Perez, Regulatory frameworks for chemical safety: The impact of EPA and OSHA on industrial sustainability. Environmental Regulation Review, 2021. 17(2): p. 88-102.
[135] Wang, Y. and D. Song, Smart Manufacturing in China: Trends, Practices and Implications for the Chemical Industry. Journal of Manufacturing Technology Management, 2020.
[136] Liu, Y. and L. Zhang, China's Roadmap to Intelligent and Sustainable Chemical Industry: Role of Research Institutes and Digital Innovation. Sustainability, 2022.
[137] Xu, X., C. Zhang, and G.H. Tzeng, Developing Smart Chemical Manufacturing Through University–Industry Collaboration: A Case Study of Tsinghua and ChemChina. Technological Forecasting and Social Change, 2021.
[138] Smart Clothing Market – Size, Share, Growth Analysis | Forecast (2024–2030), in Virtue Market Research. 2023.
[139] Smart Clothing Market Size, Share & Industry Analysis, in Fortune Business Insights. 2025.
[140] The Rise of Smart Clothing: Fashion, Fitness, Healthcare and Top Companies, in DataNext.ai. 2023.
[141] Zhao, L. and Y. Wang, AI and IoT-driven digital transformation in the apparel industry: A case study of Nike and Levi's. Journal of Fashion Technology & Textile Engineering, 2022. 10(3): p. 55-68.
[142] Cheng, R., 3D printing and AI in personalized fashion: Innovations from Adidas, Nike, and Hugo Boss. International Journal of Advanced Manufacturing Technology, 2023. 127(5): p. 912-925.
[143] Liu, J. and S. Meier, Circular economy practices in sportswear: The case of Adidas and Nike Flyknit. Journal of Cleaner Production, 2021. 298: p. 126830.
[144] Patel, K. and M. Fischer, The role of robotics in modern apparel manufacturing. Robotics and Computer-Integrated Manufacturing, 2022. 75: p. 102276.
[145] Neff, G., Augmented reality in fashion retail: Enhancing customer experience through virtual fitting rooms. Fashion and Retail Technologies, 2021. 6(2): p. 121-135.
[146] Singh, R. and D. Lee, Laser-based denim finishing: Sustainable innovation at Levi's. Journal of Sustainable Fashion, 2020. 3(4): p. 202-214.
[147] Zhang, Y. and S. Kim, Smart textiles in wearable health: Integration of sensors in clothing. Materials Today, 2023. 59: p. 32-45.
[148] Baker, T. and N. Reddy, Real-time digital supply chains in fashion: A case of Levi's transformation. Supply Chain Management Review, 2021. 27(6): p. 344-356.
[149] Zhang, Y. and M. Liu, Digital transformation and AI adoption in China's textile industry. Journal of Industrial Innovation, 2023. 15(2): p. 85-97.
[150] Li, K., Revolutionizing Textiles: AI Drives China's Industry Leap, in ODMYA News. 2024.
[151] A Glimpse of 'Smart' Textile Factory in China's Jiangsu, in Xinhua. 2024.
[152] Inside China's Clothing Manufacturing: A Deep Dive into its Evolution and Future, in Odmya. 2023.
[153] Lu, Y., Industry 4.0: A Survey on Technologies, Applications and Open Research Issues. Journal of Industrial Information Integration, 2017. 6: p. 1-10.
[154] Kagermann, H., W. Wahlster, and J. Helbig, Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0, in acatech - German National Academy of Science and Engineering. 2013.
[155] Sony, M. and S. Naik, Key Ingredients for Evaluating Industry 4.0 Readiness for Organizations: A Literature Review. Benchmarking: An International Journal, 2019. 27(8): p. 2545-2566.
[156] Schwab, K., The Fourth Industrial Revolution. World Economic Forum. 2016.
[157] Liao, Y., et al., Past, Present and Future of Industry 4.0: A Systematic Literature Review and Research Agenda. International Journal of Production Research, 2017. 55(12): p. 3609-3629.

Journal of Information Systems and Telecommunication
Vol.14, No.2, April-June 2026, 170-181
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

A Spectral-Domain Approach for Early Blockage Detection in Sub-Terahertz Wireless Networks

Neravati Nagaraja Kumar[1], Anil Kumar[2*], Subba Raju MP[3], Kalyani Kapula[4], Surya Kala Nagireddi[5], Yarrapragada Rao K. S. S[5]

[1].Department of Electronics and Communication Engineering, Rajeev Gandhi Memorial College of Engineering and Technology, Nandyal, India
[2].Department of Electronics and Communication Engineering, Aditya University, Surampalem, India
[3].Department of Electrical and Electronics Engineering, Aditya University, Surampalem, India.
[4].Department of Electronics and Communication Engineering, Aditya University, Surampalem, India.
[5].Department of Information Technology, Aditya University, Surampalem, India.
[6].Department of Mechanical Engineering, Aditya University, Surampalem, India.

Received: 31 Dec 2024/ Revised: 04 Jul 2026/ Accepted: 02 Aug 2026

Abstract

The paper presents the problem of human body blockage in sub-terahertz and millimeter wave wireless systems. These communication systems are operated at very high frequencies. The generated high frequency signals are very sensitive to obstacles like the human body. A blockage event abruptly reduces the received signal power. This causes a loss of signal connection between transmitter and receiver. To avoid this problem, the network detects a blockage before it actually happens. Many existing solutions use machine learning models in the time domain. These methods are complex. Long training times are required. The signal shows clear oscillations just before a blockage happens. These oscillations are weak in the time domain. The paper uses the short-time Fourier transform to extract spectral features from the received signal. The difference between normal conditions and pre-blockage conditions becomes very large. In some cases, the gap reaches two orders of magnitude. Based on this observation, a threshold-based proactive blockage detection algorithm is designed. The algorithm is simple and does not rely on machine learning. In order to solve this, MATLAB simulation environment is assumed with parameter values and data collected at 156~GHz in an indoor environment. The performance is measured using blockage detection probability, mean time to blockage and false alarm rate. The paper explains the spectral analysis offers a simple and effective way to enable proactive blockage detection for future sub-terahertz wireless networks.

Keywords: Sub-Terahertz Communication; Human Body Blockage; Proactive Blockage Detection; Short-Time Fourier Transform; Spectral-Domain Analysis; Threshold-Based Detection Algorithm.

1- Introduction

Millimeter wave and sub-terahertz communications are used to help in current and future wireless networks [1]. These systems are 5G enabled and are likely to form the basis of 6G network. This is operated at extremely high carrier frequencies. Normal millimeter wave frequencies are between 30 and 70 GHz[2]. Sub-terahertz range is between 100 and 300 GHz. These frequency bands are of very large bandwidth. Extremely high data rates are made possible by large bandwidth. It will be essential in such applications as virtual reality, smart factories and high-speed connectivity within the premises. But these advantages are associated with severe technical difficulties [3]. Huge obstacle is blockage of the human body. Radio wave frequencies in the millimeter and sub tehertz range cannot easily bend around objects. As soon as a human body enters the line-of-sight between the transmitter and receiver, the signal power reduces to a sharp [4].

In most occasions, it leads to total disconnection. The issue is highly prevalent in the indoor setting, because individuals relocate regularly. Consequently, human congestion will be among the greatest impediments to dependable high-frequency communication [5]. In order to make the effects of blockage lower, multiple system-level solutions were offered. Multiconnectivity is one of the solutions. This

✉ **Anil Kumar R**
anidecs@gmail.com

method enables a user device to have several parallel connections with diverse base stations. In the event of link being blocked, the traffic is redirected to another link. Multiconnectivity involves the high concentration of base stations [6]. It relies on rapid and responsive blockage detection. Another solution is integrated access and backhaul. In this approach, low-cost relay nodes are used to bring access points closer to users. This reduces the probability of line-of-sight blockage. While this method helps reduce blockage effects, it does not fully eliminate them[7]. Occasional blockages still happen. Both multiconnectivity and integrated access and backhaul require one key capability. The network needs to detect blockage events before the signal is fully blocked [8]. Proactive detection aims at the detection of an impending blockage event prior to an eventual obstruction of the signal. In case the network forecasts the blockage early, it prevents measures are taken. Such activities involve beam switching, link switching or traffic diversion[9]. These measures require time to accomplish. As such, the blockage must be identified tens of milliseconds beforehand. This renders proactive detection quite difficult compared to reactive detection. Recurrent neural networks and deep neural networks are usually employed[10]. Although these techniques are performing well, they are characterized by obvious shortcomings. Considerable amounts of training data are needed. Retraining is required when there is a change in conditions. Other methods are based on other sensing measurements. The measurements have made a significant finding: the received signal has little oscillations just before a blockage occurs[11]. In the case where a human body comes close to the line-of-sight path, partial obstruction occurs. This causes interference patterns. Such patterns are manifested in the received signal power in the form of ripples[12]. This observation is furthered in this paper and a new perspective is suggested. The paper is based on the spectral domain rather than the time domain analysis of the signal. The point is that small swings at the time domain result to strong content at the frequency domain. Through spectral analysis, the variance between normal functioning and pre-blockage is far more obvious[13].

This paper promotes short time fourier transform applied to get spectral representation of received signal. It is shown by adding short window spectral energy that pre-blockage periods have far more spectral energy than non-blockage periods[14]. The difference is in most cases two orders of magnitude. The existence of this wide gap provides the opportunity to use basic threshold-based detection techniques. Complex machine learning models are not needed. The protracted training periods are unnecessary. It is the indoor sub-terahertz deployments. Experimental records of measurements at 156 GHz are utilized[15]. This rate is very applicable to the 6G systems of the future. The measurements are realistic human blockage behaviour. This renders the evaluation a practical and reliable evaluation. This paper has made some important contributions that include:

- To present spectral-domain analysis in proactive human blockage detection in sub-terahertz communication systems. To show that separability between pre-blockage and non-blockage can occur with spectral power differences up to two orders of magnitude.
- To implement threshold-based proactive blockage detection algorithm, not based on machine learning models.
- To test the proposed method with real indoor measurement data at 156~GHz with realistic blockage conditions.
- To operate at a false alarm rate that is less than one event/second in fading environments.

The paper suggests a preventive blockage sensing algorithm based on spectral analysis. The algorithm is basic and uncomplicated. It works in various phases to deal with initializing, detecting and restoring [16]. It continuously monitors the spectral power of the received signal. When the power exceeds a defined threshold, a blockage is predicted. The algorithm is implemented at the user device or the base station [17]. Its computational complexity is low. The introduction highlights that the proposed approach overcomes the limitations of existing methods. It avoids machine learning. It does not require training. It is robust to fading and noise. It detects blockage early enough to support system-level mitigation techniques [18]. These properties make it suitable for practical sub-terahertz networks.

2- Related Work

Human body blockage received wide attention in millimeter wave and sub-terahertz wireless systems. Early measurement-based studies showed that human bodies cause very large attenuation at high frequencies. In [19], human blockage at 73~GHz was analyzed and a diffraction-based model was proposed. The results showed that blockage leads to sudden and severe signal drops. Several studies investigated empirical blockage characterization. In [20], a large-scale indoor measurement campaign was conducted at sub-terahertz frequencies. Blockage attenuation, duration and temporal characteristics were analyzed. The analysis showed that blockage events are frequent and highly dynamic in indoor environments. This work provided important measurement data that later

studies used for blockage detection research. However, the focus was on reactive detection and statistical modeling, not proactive prediction. System-level solutions were suggested to reduce the effects of blockages. Multi-connectivity is one of the biggest solutions, standardized by 3GPP in Release~16 [21]. Multi-connectivity enables user equipment to have several simultaneous connections. The research that involved [22] revealed that multi-connectivity enhances continuity of a session in mmWave systems. This is because link switching is time consuming and relies on the timely detection of blockage. Multi-connectivity, in itself is not enough without proactive detection.

Integrated access and backhaul is another system-level solution. In [23], the 3GPP taskforce suggested the application of relay nodes to enhance coverage and low blockage likelihood. It was analyzed in [24] that integrated access and backhaul has the benefit of minimizing blockage effects in dense deployments. Nevertheless, this is based on rapid blockage sense. It is not able to completely reduce the blockage events. Early warning is a very important necessity. The majority of the methods of early blockage detection were reactive. These techniques only identify when blockage has occurred when the signal received falls below a threshold. These methods are not complex yet cannot be used in smooth communication. The signal is lost and often it is already too late to avoid the interruption of the service. Proactive detection based on machine learning was investigated in the recent research. The predictive blockage was trained on a deep neural network in [25] with cross-correlation among beam states. The technique proved to be encouraging in terms of accuracy but was deployment sensitive. The other method of machine learning was offered in [26], where a deep neural network was used to anticipate beam and blockage events indoors. This would involve extensive training of every environment. This dependency imposes scalability constraints and real-life application.

The methods based on deep learning were investigated in [27]. Recurrent neural networks had to be trained using real mmWave measurements to predict moving blockages. The findings indicated that blockage can be anticipated in some future time. The prediction was however probabilistic. The technique failed to give a specific time of detection. Computational complexity was great.
In [28], in-band wireless signatures were combined with LiDAR sensing to predict dynamic blockages. This multimodal approach improved prediction accuracy. However, it required additional sensors and data fusion. This increased system complexity and cost. Such requirements make large-scale deployment difficult. Some works studied the sensing capability of mmWave and sub-terahertz signals. In [29], mmWave signals were proposed for human detection and identification. While this demonstrated the sensing potential of high-frequency signals, it required cooperation between multiple users. An important observation was reported in [28] and [29]: pre-blockage signatures appear in the received signal before blockage happens. Very few works explored spectral-domain analysis for blockage detection. Traditional spectral analysis is widely used in signal processing. The gap between time-domain and spectral-domain approaches remained largely unexplored. The present work addresses this gap. Instead of relying on time-domain patterns or machine learning, it exploits the spectral representation of the received signal. By applying the short-time Fourier transform, small time-domain oscillations are converted into strong spectral components. This makes pre-blockage signatures clearly measurable. The difference between non-blockage and pre-blockage states becomes very large. This allows the use of simple threshold-based detection.

3- System Model and Problem Formulation

The Fig (1) shows the indoor sub-THz measurement setup at 156 GHz. A transmitter (Tx) and a receiver (Rx) face each other. Both use horn antennas and are perfectly aligned. A clear line-of-sight (LoS) link is maintained. A human blocker walks across the LoS path. The walking speed is normal. When the person approaches the LoS, the signal starts to change. This region is called the pre-blockage period.

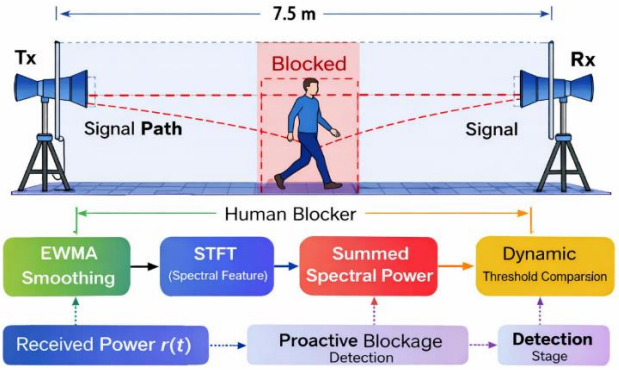


Fig. 1 Indoor Sub-THz blockage measurement setup at 156GHz

When the person fully blocks the path, the link becomes blocked. The received signal power is recorded over time. First, the signal is smoothed using EWMA. This reduces noise and fast fading. Next, STFT is applied to short signal windows. STFT converts time variations into the frequency domain. The spectral power is summed for each window. During pre-blockage, this power increases clearly. A dynamic threshold compares the current spectral power. If the threshold is crossed, a blockage is predicted early. This frequency lies in the lower sub-terahertz band and is highly relevant for future 6G systems. The transmitted power is set to 90~mW. Both the transmitter and the receiver are

equipped with pyramidal horn antennas. These antennas feature a half-power beamwidth of 10 degrees and a gain of 25~dB. The antennas are perfectly aligned during all experiments to maintain a stable line-of-sight link. The experiments are conducted in an empty indoor hall. The hall has a length of 7.5~m, a width of 4.5~m and a height of 3~m. The distance between the transmitter and receiver is varied. Three values are considered which are 3~m, 5~m and 7~m. These distances represent typical indoor communication scenarios. A human blocker walks across the line-of-sight path between the transmitter and receiver. The walking velocity will be set at 3.5km/h. This pace is equal to the regular walking of human beings. Positions of various blockers are tested. At every transmitter receiver distance, several transmitter blocker distances are employed. This makes the effects of blockage at various positions along the linkable to be analysed. The antennas are of varying heights. Two heights of the antennas are taken into consideration 1.35-m and 1.65-m. These levels are associated with blockage of the chest and head parts of a human body.

Table 1: Simulation Parameters

Parameter	Symbol	Typical Value
Carrier	fc	156 GHz
TX–RX	dTR	3 m, 5 m, 7 m
Human walking	v	3.5 km/h
Antenna height	h	1.35 m, 1.65 m
Received power	Ts	50 μs
STFT shift	l	20
EWMA	γ	0.005–0.01
Blockage	tB	100–200 ms
Pre-blockage	tF	50–150 ms
Threshold	k	0.4–0.8
Granularity	N	10
Detection	PB	≈ 0.98–1.00
Mean time to	E[D]	≥ 50 ms
False alarm rate	RF	< 1 event/s
Simulation	–	30 runs

A total of 30 repeats is done in each conThis provides enough time for mitigation actions. After a blockage is detected, the algorithm enters the recovery stage.ation. This repetition gives statistical reliability and the effect of random variations is minimized. High time resolution is taken on the received signal power. The channel sampling rate shall be 50 -1. This high-resolution enables rapid signal fluctuations to be obtained when there is blockage. A time constant of 30s is utilized in an amplifier. During blockages, the signals are clearly and repeatably observed. At the absence of any blockage, the power of the received signal stays rather stable. Fading and noise bring about small fluctuation. As the body of a human being goes closer to the line-of-sight, the signal behaviour is altered. Prior to the physical blockage, the signal received has some oscillations. These swings are manifested in the form of tiny waves in the signal power. To clearly see these effects, the received signal is balanced with exponentially weighted moving average filter. This filtering eliminates fast fading and noise. It enables the behaviour that is underlying it to be made apparent. Smoothing parameter is selected carefully in order to maintain slow signal variations and remove quick changes of noise.

Three clear signal periods are then identified after smoothing. The initial one is the non-blockage state. The signal power in this state is stable and powerful. The second one is the pre-blockage stage. Oscillations are exhibited in the signal power in this state. These vibrations persist between 100 and 200~ms. The third state is the blockage state. The power of the signal in this state decreases very rapidly and is highly fluctuating. The oscillations in the pre-blockage state are due to the effects of diffraction. The human body approaches the line-of-sight path by partially blocking the signal, as the body gets closer to the path. This forms constructive and destructive patterns of interference. The patterns produce oscillatory behaviour of received signal power. This effect is seen in all the measurement settings. The existence of these pre-blockage oscillations is the main hypothesis of this paper. The point is the blockage events are not sudden and silent. Rather, there are measurable signal signatures that are taken before the events. These signatures come out in advance of the signal becoming completely blocked. This opens a possibility of proactive detection. Nevertheless, there is one significant challenge. The amplitude of pre-blockage oscillations in the time domain is extremely small. The oscillations are normally just one to two percent of the normal signal level. This renders them hard to detect consistently with simple thresholding in time domain. These minor variations are easily concealed by noise and fading.

The paper has an alternative solution. The hypothesis goes on to state that, though the pre-blockage signatures may be weak in time domain, the signatures become strong in the spectral domain. Thereupon, the non-blockage and pre-blockage difference is very large and simpler to measure. In order to test this hypothesis, the short-time fourier transform of the smoothed received signal is used. The transform changes short signal segments of the time domain to the frequency domain. Frequency bin summation of spectral power is then carried out. The hypothesis is proved right by the measurements. At the non-blockage period, the summed spectral power is minimal. In a pre-blockage phase, oscillations cause a sharp rise of the spectral power. The

spectral power is also stiffer as the signal fluctuations are strong during the blockage period. The spectral power difference between non-blockage and pre-blockage spectral is in most cases one to two orders of magnitude. It is this distinct segregation of states upon which the given detection method is based. Simple threshold -based detection can be done due to the large gap. Complex classifiers and learning-based models are not required. The hypothesis demonstrates that spectral-domain analysis is a more powerful and more convincing method of detecting the early signatures of blockage. This section is well motivated by proving the hypothesis using actual sub-terahertz measurements data. It fills the gap between theory of behavior of physical signals and the practice of proactive blockage detection. This renders the proposed system applicable to actual indoor sub-terahertz communication systems.

4- Proactive Blockage Detection Algorithm

The fading, noise and blockage cause changes in the received signal power with time. These variations are minor in the normal state of operation. Small oscillations occur in the signal before a blockage occurs. The time domain is weak with these oscillations. The amplitude is very small. This complicates time-domain detection. In order to correct this, the algorithm works on the signal spectral domain. The spectral domain points out oscillatory behavior. The little swings in time generate significant energy in frequency. This brings a distinct distinction among various states in a channel. The short-time fourier transform is used to get the spectral representation. The short-time Fourier transform transforms a short portion of the signal to the frequency domain. It is defined as:

$$S(m,\omega) = \sum_{n=0}^{W-1} f[m+n]w[n]e^{-j\omega n} \quad (1)$$

Here f[m+n] is the smoothed received signal sample, $w[n]$ is a window function, W is the window length, m is the time index and ω is the frequency index. The received signal is first smoothed using an exponentially weighted moving average. This smoothing reduces fast fading and noise. It preserves slow variations caused by blockage. The smoothed signal is then used as input to the short-time Fourier transform. After computing the short-time Fourier transform, the spectral power is summed over all frequency bins. This produces a single value that represents the energy of signal variations in a short time window. This value is used for blockage detection. Measurements show three clear states. The non-blockage state has very low summed spectral power. The pre-blockage state has higher spectral power due to oscillations. The blockage state has the highest spectral power due to strong signal fluctuations. The difference between non-blockage and pre-blockage states reaches one or two orders of magnitude. This large gap enables simple threshold-based detection.

The proposed proactive blockage detection algorithm operates in three stages. These stages are initialization, active detection and blockage recovery. The algorithm is implemented at the user equipment or at the base station. The purpose of the initialization stage is to identify the current channel state. At the beginning, the system does not know whether the channel is blocked or not. The algorithm collects channel samples and applies exponential smoothing. After collecting W samples, it computes the summed spectral power. The algorithm then collects another W samples and computes the spectral power again. The two values are compared. If the later value is smaller, the system has moved from a blocked state to a non-blocked state. In this case, the non-blockage spectral power is identified. The algorithm then proceeds to the active detection stage. If the later value is larger, the channel is in a blockage or pre-blockage state. To avoid false initialization, the algorithm waits for a predefined blockage duration. After waiting, the initialization stage is repeated. This stage makes active monitoring start only when the channel is in a stable non-blockage state. The active stage is the phase in which proactive blockage detection takes place. During this stage, the algorithm continuously monitors the channel. Every l samples, the summed spectral power is computed. A detection threshold is defined based on the spectral power values of non-blockage and blockage states. The threshold is expressed as:

$$S_{th} = S_{nb} + \frac{k}{N}\left(S_b - S_{nb}\right) \quad (2)$$

Here, S_{nb} is the spectral power of the non-blockage state, S_b is the spectral power of the blockage state. k is a threshold multiplier and N is a granularity parameter. The current spectral power S_{cur} is compared with the threshold. If:

$$S_{cur} > S_{th} \quad (3)$$

A blockage is predicted. The algorithm then signals a blockage event. At this point, the network takes preventive actions. These actions include switching links or changing beams. If the current spectral power does not exceed the threshold, the algorithm continues monitoring. It waits for the next shift interval and repeats the process. This stage allows the algorithm to detect blockage early. The detection typically takes place tens of milliseconds before the actual blockage.

The second parameter is the short-time Fourier transform window length W. A larger window provides more stable spectral estimates. However, it increases detection delay. A smaller window reduces delay but increases noise sensitivity. The third parameter is the shift length l. This

parameter controls the frequency of spectral transform computations. Larger values reduce computational complexity. Smaller values improve time resolution. The fourth parameter is the threshold multiplier k. This parameter controls detection sensitivity. Smaller values allow earlier detection but increase false alarms. Larger values reduce false alarms but delay detection. The proposed algorithm has several advantages. It does not require machine learning. It does not need training data. It works directly on measured signal samples. Its computational complexity is low. The exponential smoothing has constant complexity. The short-time Fourier transform has complexity $O(\text{WlogW})$. Most importantly, the algorithm exploits a physical property of the channel. It uses the spectral manifestation of diffraction-induced oscillations. This makes detection reliable and robust. By operating in the spectral domain, it converts weak time-domain signatures into strong measurable features. This enables early and reliable blockage prediction for sub-terahertz wireless systems. Algorithm 1 for Blockage detection with different cases.

Algorithm 1

Input : γ, W, k, N and t_b
Output : Blockage detection
Stage 1 : Initialization
while {initialization not complete}
Collect W samples and compute
if { $S(t,\omega)$ decreases over time}
$S_{\text{nb}} \leftarrow S(t,\omega)$
$S_{\text{b}} \leftarrow S(t+W,\omega)$
else
end **while**
Stage 2 : Active detection
while {monitoring}
Compute S_{cur} for current samples
State Compute S_{th} using the threshold
$S_{\text{th}} = k \cdot S_{\text{b}} \cdot N$
if { $S_{\text{cur}} > S_{\text{th}}$ }
Signal blockage detected
else
end **if**
end **while**
Stage 3 : Blockage recovery
Wait for t_b, then reinitialize (Stage 1)

5- Result Analysis

In this part, this paper examines the effectiveness of the suggested proactive blockage detection algorithm. The analysis is done using actual sub-terahertz measurement information. This is to test the capability of the algorithm to identify events of blockage even before they occur. The measures of positive detection and early warning and false alarm resistance are the subjects of analysis. The numerical assessment is done by three performance measures. The two measures are common in blockage detection studies. The first measure is the probability of blockage which is signified by . It is the likelihood that an incidence of blockage is identified in time prior to its occurrence. An almost unit value is suggestive of a good detection. It is the mean of the difference between the instant the blockage occurs and the instant where it is detected. Larger values indicate earlier detection. Early detection is important for enabling proactive actions. The third metric is the false alarm rate, denoted by R_F. It measures the frequency at which the algorithm signals a blockage when no blockage happens. It is expressed as the number of false alarms per second. A low false alarm rate is required for stable system operation. The blockage detection threshold is defined using spectral power values.
In the simulations, N is fixed to 10. The choice of k directly affects the trade-off between early detection and false alarms.

Table 1: Effect of Threshold Multiplier on Performance

k	P_B	E[D] (ms)	R_F (events/s)
0.2	1.00	160	18.5
0.4	1.00	120	5.2
0.6	1.00	80	1.4
0.8	1.00	55	0.8

Table 1 shows the effect of the threshold multiplier. Increasing k reduces the false alarm rate noticeably. At the same time, the mean detection time decreases gradually. A value of k between 0.4 and 0.8 provides a good trade-off. The smoothing parameter γ of the exponentially weighted moving average affects performance. Very small values of γ lead to heavy smoothing. This removes useful signal variations. Very large values lead to insufficient smoothing and increased noise sensitivity. The results show that moderate smoothing performs best. Values of γ between 0.005 and 0.01 provide stable detection. Extreme values increase the false alarm rate or slightly reduce detection time.

Table 2: Effect of EWMA Smoothing Parameter

γ	P_B	E[D] (ms)	R_F (events/s)
0.0005	0.98	140	3.8
0.005	1.00	95	1.1
0.01	1.00	90	0.9
0.1	0.97	70	2.6

Table 2 shows that moderate smoothing improves robustness. It suppresses fast fading while preserving blockage-related oscillations. The next experiment studies the effect of the short-time Fourier transform window length W and shift length l . The threshold multiplier and smoothing parameter are fixed. Increasing the window length improves spectral stability. This reduces false alarms. However, it increases detection delay. Smaller windows detect blockages faster but are more sensitive to noise. The shift length affects computational complexity. Larger shift values reduce the number of spectral computations. This slightly increases the false alarm rate for small window sizes.

Table 3: Effect of Window and Shift Length

W	l	E[D] (ms)	R_F (events/s)
300	20	110	2.3
400	40	90	1.5
500	20	75	1.0
600	60	55	0.7

Table 3 shows that window lengths between 400 and 600 samples provide a good balance. Shift lengths between 20 and 60 reduce complexity without noticeably degrading performance. The proposed algorithm is compared with machine learning based proactive blockage detection methods. These methods use recurrent neural networks and convolutional models. Detection probability is lower and detection time is less predictable. The proposed method achieves a blockage detection probability close to one. It provides deterministic detection time. It maintains a lower false alarm rate. The computational complexity of the proposed algorithm is low. Exponential smoothing has constant complexity. The short-time Fourier transform has complexity $O(\text{WlogW})$. The memory requirement is small. Only W samples are stored. Machine learning methods require thousands of samples for training and inference. The numerical assessment shows that the proposed algorithm is reliable and robust. It detects blockage events at least 50~ms before the events happen. It maintains a false alarm rate below one event per second. The detection probability remains close to one across a wide range of parameters. These results confirm that spectral-domain analysis is effective for proactive blockage detection. The algorithm offers a practical solution for indoor sub-terahertz communication systems.

The Fig (2) shows the received signal power over time. Two different blockage configurations are presented. Before blockage, the signal is stable and strong. This is the non-blockage period. When a person approaches the link, the signal starts to change. Small oscillations appear in the signal. This is the pre-blockage period. The red shaded region shows the blocked interval.

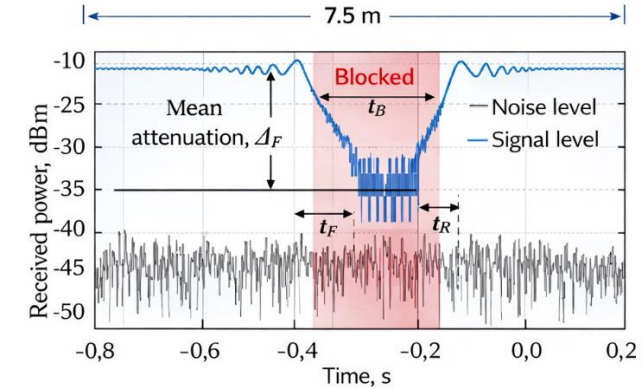


Fig. 2. STFT of Non-blockage, pre-blockage and blockage in time domain

The Fig (3) shows the blockage detection probability (P_B) versus the threshold multiplier (k) for three values of (γ) namely 0.1, 0.3 and 0.5. At (k = 1) and (k = 2), all three cases reach a probability of nearly 1.00. When (k = 3), differences begin to appear. For (γ = 0.1), (P_B) drops slightly to about 0.99. For (γ = 0.3) and (γ = 0.5), the probability remains close to 1.00. At (k = 4), all curves converge near 0.99. At (k = 5), the detection probability decreases. The values are around 0.98 for (γ = 0.1), 0.99 for (γ = 0.3) and about 0.99 for (γ = 0.5). For larger values of (k), the downward trend becomes stronger. At (k = 6), (P_B) reduces to about 0.97 for (γ = 0.1), 0.98 for (γ = 0.3) and roughly 0.99 for (γ = 0.5). This indicates higher (γ) helps preserve detection accuracy. Finally, at (k = 9), (P_B) falls to approximately 0.94 for (γ = 0.1), 0.95 for (γ = 0.3) and 0.96 for (γ = 0.5). A larger (γ) yields higher detection probability across all thresholds. This highlights a trade-off between threshold strictness and detection reliability.

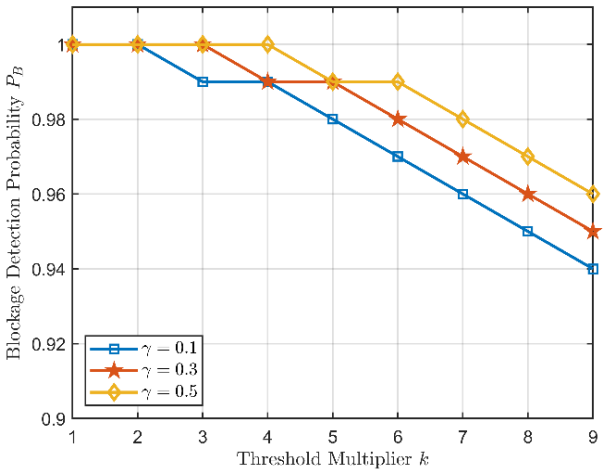


Fig 3: Blockage Detection Probability versus Threshold Multiplier

The Fig (4) presents the mean time to blockage against the threshold multiplier k for three different γ values. At k = 1, the delay is the highest for all cases. The value is about 140 ms when γ = 0.1. For γ = 0.3, it is nearly 120 ms. For γ = 0.5, it is close to 100 ms. When k increases to 2, the mean time to blockage drops to around 125 ms for γ = 0.1, 105 ms for γ = 0.3 and 90 ms for γ = 0.5. At k = 3, the delay decreases to approximately 110 ms, 92 ms and 80 ms for γ = 0.1, 0.3 and 0.5, respectively. At k = 4, the values reach nearly 95 ms for γ = 0.1, 80 ms for γ = 0.3 and 70 ms for γ = 0.5. The separation between the curves remains nearly uniform, which indicates steady performance differences. Overall, larger k values and higher γ values both contribute to faster blockage detection.

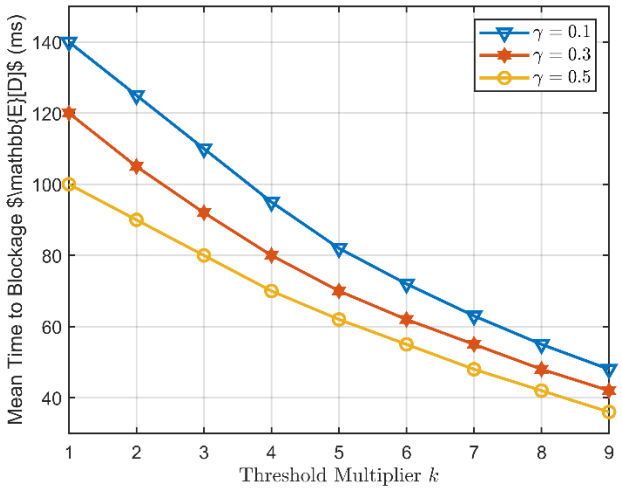


Fig. 4 Mean Time to Blockage versus Threshold Multiplier

The Fig (5) shows the false alarm rate versus the threshold multiplier k for three values of γ. At k = 0.1, the false alarm rate is the highest. For γ = 0.001, the value is close to 15 events per second. For γ = 0.005, it is around 12 events per second. For γ = 0.01, it is about 10 events per second. When k increases to 0.2, the rates drop to nearly 12.5, 9.2 and 8 events per second for γ = 0.001, 0.005 and 0.01, respectively. At k = 0.3, the false alarm rate reduces to about 10, 7 and 6 events per second. At k = 0.4, the values fall to roughly 8, 5.2 and 4.3 events per second. At k = 0.5, the rates are around 6 events per second for γ = 0.001, 3.7 for γ = 0.005 and 3 for γ = 0.01. At k = 0.6, the values drop to about 4.4, 2.5 and 2 events per second. At k = 0.7, the false alarm rates reach nearly 3, 1.6 and 1.2 events per second. At k = 0.8, the rates are close to 2, 1 and 0.7 events per second. When k reaches 1.0, all curves approach low values.

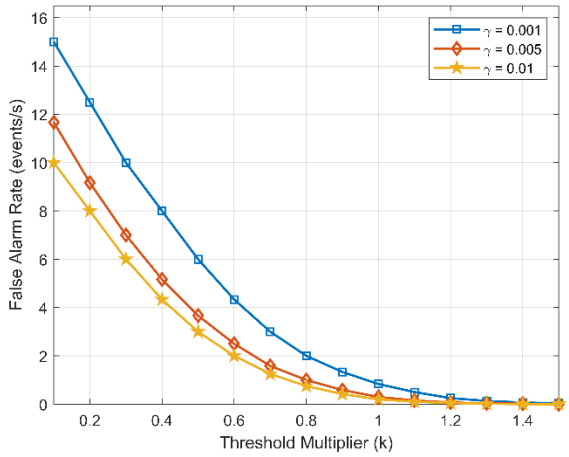


Fig. 5 : False Alarm Rate versus Threshold Multiplier

The Fig (6) illustrates the mean time to blockage for five different detection methods. These methods include Proposed (Spectral), GRU, LSTM, CNN and RNN. The Proposed (Spectral) method records the largest delay among all approaches. Its mean time to blockage is close to 150 ms. The GRU-based method shows a much lower delay of about 85 ms. This indicates that the delay is reduced by nearly 65 ms compared with the spectral method. The response time is enhanced by the LSTM method. This downward trend is maintained by CNN with a value near 60 ms. RNN method has the least delay. The delay of the proposed spectral method is approximately 1.8 times bigger than GRU and almost 2.7 times bigger than RNN. In the case of a comparison between GRU and LSTM, delay is reduced by approximately 18 percent. The transition between LSTM to CNN takes on average 10 ms shorter than the time to blockage, which is almost 14 percent better. The CNN-RNN difference is less significant although still relevant with RNN being approximately 5 ms faster. This amounts to a growth of close to 8 percent. The slowest delay

varies between approximately 55 ms to 150 ms with a range of nearly 95 ms between methods. The high range demonstrates a great variation in speed of detection. The spectral approach proposed has the slowest response time. RNN approach provides the quickest detection and therefore it can be used in applications with a rigid low-latency constraint. The results clearly show that the choice of detection model strongly affects the blockage detection delay.

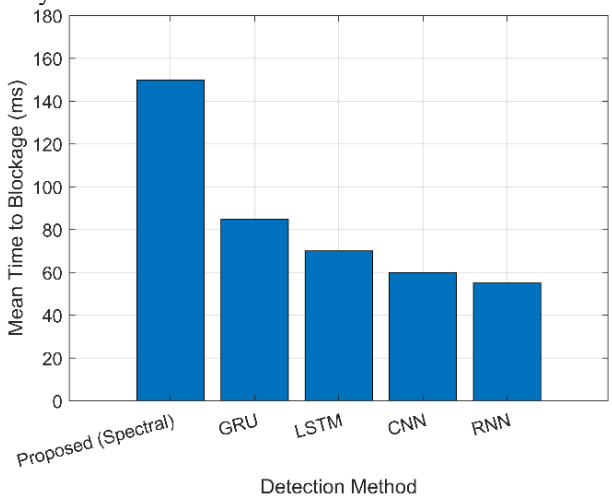


Fig. 6: Mean Time to Blockage Comparison

The Fig (7) shows the summed STFT power for three different channel states. The channel states are Non-Blockage, Pre-Blockage and Blockage. For the Non-Blockage state, the summed STFT power is the lowest among all cases. Its value is slightly above 10^3. This corresponds to roughly 2×10^3 in linear scale. This low power level indicates stable channel conditions with minimal signal disturbance. For the Pre-Blockage state, the summed STFT power increases sharply. The value is close to 3×10^4. This represents more than a tenfold increase compared to the non-Blockage case. The difference between Non-Blockage and Pre-Blockage is clearly visible on a logarithmic scale. This indicates that STFT power is very sensitive to early blockage effects. For the Blockage state, the summed STFT power reaches its maximum value. The bar height is slightly above 10^5, which corresponds to nearly 1.2×10^5. When comparing Pre-Blockage and Blockage, the power increases by roughly 8×10^4. The overall range of summed STFT power spans from about 10^3 to above 10^5. The monotonic increase from Non-Blockage to Pre-Blockage and then to Blockage shows a clear progression pattern. This pattern supports reliable classification of channel conditions based on summed STFT power.

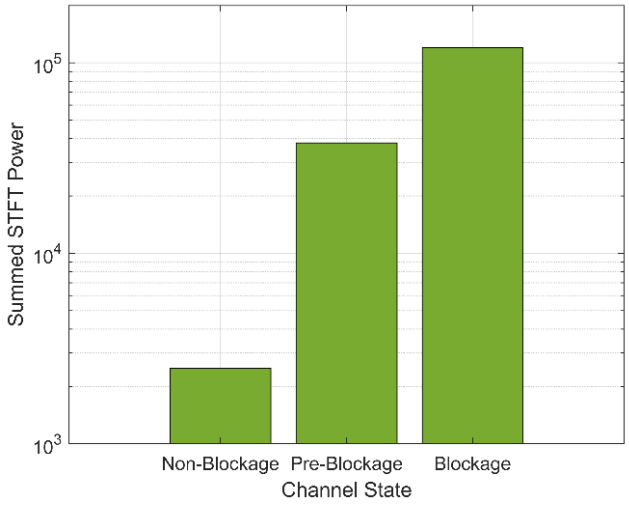


Fig. 7: Summed STFT Power Comparison

The Fig (8) illustrates the change in false alarm rate with the shift length (l) for five different values of (γ): 0.1, 0.01, 0.005, 0.001 and 0.0005. At a small shift length of ($l = 5$), the curve for ($\gamma = 0.1$) has the lowest value, close to 0.25 events per second. When the shift length increases to ($l = 10$), all curves drop noticeably. At ($l = 20$), the downward trend continues, with approximate values of 0.16, 0.30, 0.50, 0.75 and 1.18 events per second. For larger shift lengths, the decrease in false alarm rate remains steady but becomes less steep. At ($l = 40$), the rates are around 0.13 for ($\gamma = 0.1$), 0.20 for ($\gamma = 0.01$), 0.34 for ($\gamma = 0.005$), 0.50 for ($\gamma = 0.001$) and 0.85 for ($\gamma = 0.0005$). When the shift length reaches (l =60), these values reduce to approximately 0.10, 0.15, 0.24, 0.38 and 0.64 events per second. At ($l = 80$), the false alarm rates fall to about 0.08, 0.12, 0.17, 0.30 and 0.51 events per second.

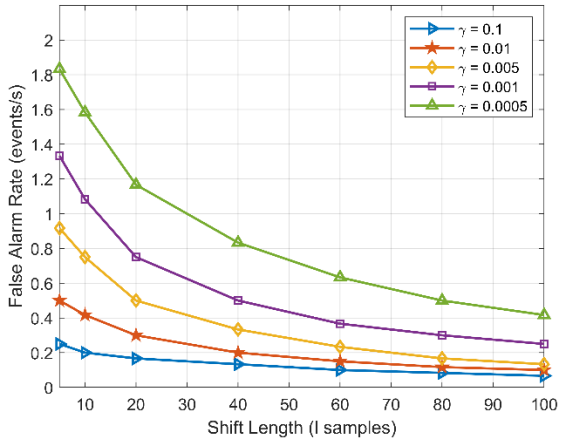


Fig. 8 False Alarm Rate versus Shift Length for different EWMA Values

Fig (9) shows the change in mean time to blockage with the STFT window length for five different values of γ. For γ =

0.1, the mean time to blockage starts at about 85 ms at a window length of 200. It then decreases gradually to nearly 80 ms at 300, 75 ms at 400 and around 70 ms at 500. At 700 samples, it reduces to nearly 60 ms. The lowest value is observed at 800 samples, close to 55 ms. For γ = 0.01, the mean time to blockage begins at around 120 ms at 200 samples. At 500 samples, the value is close to 90 ms. At 800 samples, the value reaches approximately 65 ms. For γ = 0.005, the value starts at nearly 150 ms at 200 samples. It decreases to about 140 ms at 300 and 130 ms at 400. At 500 samples, it reaches around 120 ms. When the window length increases to 600, the value drops to about 110 ms.

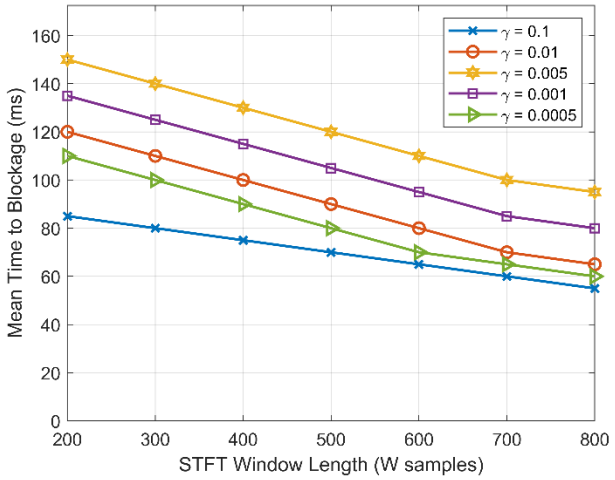


Fig. 9: Mean Time to Blockage versus STFT Window Length

Fig (10) illustrates the variation of the false alarm rate RF with the STFT window length W. At W = 100, the false alarm rate is highest for γ = 0.1, reaching about 1.8 events per second. For γ = 0.3, the value is lower at nearly 1.4 events per second. The lowest rate at this window length is observed for γ = 0.5, which is close to 1.1 events per second. When W increases to 150, all curves show a sharp drop. The rates reduce to approximately 1.3 for γ = 0.1, 1.0 for γ = 0.3 and 0.8 for γ = 0.5. The rate for γ = 0.1 falls to approximately 0.4 events per second. For γ = 0.3, it reduces to about 0.25 events per second. For γ = 0.5, the lowest rate is observed, close to 0.18 events per second. The spacing between the curves remains nearly uniform, which suggests stable relative performance. The figure shows that increasing both the STFT window length and γ leads to predictable and steady reductions in false alarm rate.

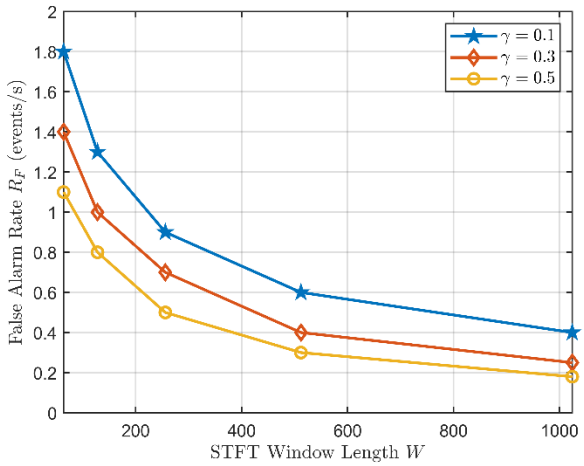


Fig. 10: False Alarm Rate versus STFT Window Length

Fig (11) gives the false alarm rate of five detection methods. The Proposed method has the least false alarm rate. It has a value close to 0.35 events per second. The CNN approach records an average false alarm of 0.8 events per second. This is higher than the Proposed method value by more than twice. The LSTM algorithm generates the highest false alarm rate and is almost equal to 0.95 events per second. GRU method is not as good as LSTM. Its false alarm rate is approximately 1.2 events per second. It means that there are a lot of false alarms. The highest false alarm rate is displayed by threshold-TD. It is at a value of approximately 1.5 events per second. It is over four times greater than the Proposed method. The Proposed method minimizes false alarm rate by approximately 77 percent as compared to Threshold-TD. The decrease is almost 71 percent, in comparison with GRU. This is nearly 63 percent when compared to LSTM. The Proposed approach even against CNN also reduces false alarms by nearly 56 percent. False alarms are still high with the deep learning models GRU and LSTM. CNN is superior to recurrent models and has significant errors. The worst method is the threshold-based method. It is the Proposed method that is the most stable. It maintains the false alarms at a very low rate of less than 0.5 events per second. This enhances its reliability in real time detection. There will be reduced false alarms which will result in less unnecessary intervention. The figure demonstrates clearly that the Proposed method has a better false alarm suppression. It provides a solid quantitative enhancement of any benchmark techniques.

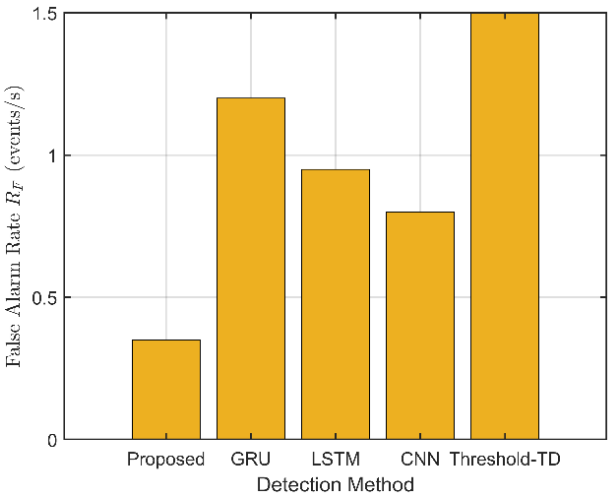


Fig. 11: False Alarm Rate Comparison across Detection Methods

6- Conclusion

At very high frequencies, human blockage is a significant problem. The signal power is significantly reduced by one person. This brings about disconnection. In order to minimize this issue, the paper concentrated on proactive blockage detection. Proactive detection detects a blockage before it is completely realized. Early information enables the network to take some preventive measures such as link switching or beam adaptation. The article presented a novel active blockage detection methodology, which is spectral-domain. The approach does not analyse the signal that comes in the time domain, but rather employs the frequency domain. Oscillations in small signals occur prior to the occurrence of a blockage. The difference between non-blockage and pre-blockage states is very large by using short-time Fourier transform. The algorithm consists of three steps. These phases deal with initiation, active recovery and detecting. This technique is deployed either at the user device or the base station. The suggested method was tested with the actual measurement data gathered at 156 GHz in an indoor setting. The findings revealed that there is at least 50~ms detection of blockage events preceding the occurrence of the event. Meanwhile, the false alarm rate is not more than one event per second. The probability of blockage remains near to one over a large range of parameters. It was shown in the paper that spectral-domain analysis represents a useful method in the context of proactive blockage detection in sub-terahertz systems.

References

[1] R. Govindasamy, S. K. Nagarajan, J. R. Muthu, and P. Annadurai, “Deep optimized hybrid beamforming intelligent reflecting surface assisted UM-MIMO THz communication for 6G broadband connectivity,” Telecommunication Systems, vol. 86, no. 4, pp. 645–660, 2024.

[2] C. Singh, C. Sharma, S. Tripathi, M. Sharma, and A. Agrawal, “A comprehensive survey on millimeter wave antennas at 30/60/120 ghz: design, challenges and applications,” Wireless Personal Communications, vol. 133, no. 3, pp. 1547–1584, 2023.

[3] H. Gui, Y. Xie, Q. Song, Y. Qian, and R. Zhang, “Improved SIC-SVD hybrid precoding algorithm for indoor sub-THz systems,” IEEE Access, vol. 11, pp. 93691–93700, 2023

[4] M. Khan, S. Bhunia, M. Yuksel, and L. C. Kane, “Line-of-sight discovery in 3d using highly directional transceivers,” IEEE Transactions on Mobile Computing, vol. 18, no. 12, pp. 2885–2898, 2018.

[5] W. Mao, H. Nikopour, and S. Talwar, “Providing high reliability on blockage-prone higher frequency bands with packet-level coding,” IEEE Transactions on Vehicular Technology, 2025.

[6] T. Sylla, L. Mendiboure, S. Maaloul, H. Aniss, M. A. Chalouf, and S. Delbruel, “Multi-connectivity for 5g networks and beyond: A survey,” Sensors, vol. 22, no. 19, p. 7591, 2022.

[7] Q. Liu, Y. Guo, F. An, L. Lin, and Y. Lai, “Water blocking effect caused by the use of hydraulic methods for permeability enhancement in coal seams and methods for its removal,” International Journal of Mining Science and Technology, vol. 26, no. 4, pp. 615–621, 2016.

[8] F. Zhinuk, A. Gaydamaka, D. Moltchanov, and Y. Koucheryavy, “Spectral-based proactive blockage detection for sub-THz communications,” IEEE Communications Letters, vol. 28, no. 7, p. 1703 – 1707, 2024

[9] C. Chan, A. Kuncheria, and J. Macfarlane, “Simulating the impact of dynamic rerouting on metropolitan-scale traffic systems,” ACM Transactions on Modeling and Computer Simulation, vol. 33, no. 1-2, pp. 1–29, 2023.

[10] F. M. Salem, Recurrent neural networks. Springer, 2022.

[11] J.-S. Brittain and P. Brown, “Oscillations and the basal ganglia: motor control and beyond,” Neuroimage, vol. 85, pp. 637–647, 2014.

[12] C. W. Dickey, I. A. Verzhbinsky, X. Jiang, B. Q. Rosen, S. Kajfez, B. Stedelin, J. J. Shih, S. Ben-Haim, A. M. Raslan, E. N. Eskandar, et al., “Widespread ripples synchronize human cortical activity during sleep, waking, and memory recall,” Proceedings of the National Academy of Sciences, vol. 119, no. 28, p. e2107797119, 2022.

[13] J. Tan, T. H. Luan, W. Guan, Y. Wang, H. Peng, Y. Zhang, D. Zhao, and N. Lu, “Beam alignment in mmwave v2x communications: A survey,” IEEE Communications Surveys & Tutorials, vol. 26, no. 3, pp. 1676–1709, 2024.

[14] F. Zhinuk, A. Gaydamaka, D. Moltchanov, and Y. Koucheryavy, “Spectral-based proactive blockage detection for sub-thz communications,” IEEE Communications Letters, vol. 28, no. 7, pp. 1703–1707, 2024.

[15] J. Huang, C.-X. Wang, Y. Yang, Y. Liu, J. Sun, and W. Zhang, “Channel measurements and modeling for 400–600-mhz bands in urban and suburban scenarios,” IEEE Internet of Things Journal, vol. 8, no. 7, pp. 5531–5543, 2020.

[16] M. V. Narkhede, P. P. Bartakke, and M. S. Sutaone, "A review on weight initialization strategies for neural networks," Artificial intelligence review, vol. 55, no. 1, pp. 291–322, 2022.
[17] J. Sastry, B. Ch, and R. R. Budaraju, "Implementing dual base stations within an iot network for sustaining the fault tolerance of an iot network through an efficient path finding algorithm," Sensors, vol. 23, no. 8, p. 4032, 2023.
[18] U. Farooq and R. B. Bass, "Frequency event detection and mitigation in power systems: A systematic literature review," IEEE Access, vol. 10, pp. 61494–61519, 2022.
[19] G. R. MacCartney, S. Deng, S. Sun, and T. S. Rappaport, "Millimeter-wave human blockage at 73 GHz with a simple double knife-edge diffraction model and extension for directional antennas," in Proc. IEEE VTC-Fall, 2016, pp. 1–6.
[20] A. Shurakov, D. Moltchanov, A. Prikhodko, A. Khakimov, E. Mokrov, V. Begishev, I. Belikov, Y. Koucheryavy, and G. Gol'tsman, "Empirical blockage characterization and detection in indoor sub-THz communications," Comput. Commun., vol. 201, pp. 48–58, Mar. 2023.
[21] 3GPP TS 37.340, "NR; Multi-connectivity; Stage 2 (Release 16)," 2019.
[22] V. Begishev, E. Sopin, D. Moltchanov, R. Kovalchukov, A. Samuylov, S. Andreev, Y. Koucheryavy, and K. Samouylov, "Joint use of guard capacity and multiconnectivity for improved session continuity in millimeter-wave 5G NR systems," IEEE Trans. Veh. Technol., vol. 70, no. 3, pp. 2657–2672, Mar. 2021.
[23] 3GPP TR 38.874, "Study on integrated access and backhaul (Release 16)," 2019.
[24] D. Moltchanov, M. K. Samani, S. S. Szyszkowicz, S. Andreev, O. Tirronen, and Y. Koucheryavy, "A tutorial on mathematical modeling of 5G/6G millimeter wave and terahertz cellular systems," IEEE Commun. Surveys Tuts., vol. 24, no. 2, pp. 1072–1116, 2022.
[25] A. Bonfante, N. Garcia, L. Giupponi, and M. Zorzi, "Performance of predictive indoor mmWave networks with dynamic blockers," IEEE Trans. Cognit. Commun. Netw., vol. 8, no. 2, pp. 812–827, Jun. 2022.
[26] S. Moon, J. Kang, J. Park, and J. Choi, "Deep neural network for beam and blockage prediction in 3GPP-based indoor hotspot environments," Wireless Pers. Commun., vol. 124, no. 4, pp. 3287–3306, Jun. 2022.
[27] S. Wu, Y. Li, Y. He, H. Zhang, and Y. Li, "Deep learning for moving blockage prediction using real mmWave measurements," in Proc. IEEE ICC, 2022, pp. 3753–3758.
[28] S. Wu, C. Chakrabarti, and A. Alkhateeb, "Proactively predicting dynamic 6G link blockages using LiDAR and in-band signatures," IEEE Open J. Commun. Soc., vol. 4, pp. 392–412, Jan. 2023.
[29] T. Gu, Z. Tian, L. Dai, and Z. Wang, "MmSense: Multi-person detection and identification via mmWave sensing," in Proc. ACM Workshop Millimeter-Wave Netw. Sens. Syst., 2019, pp. 45–50.