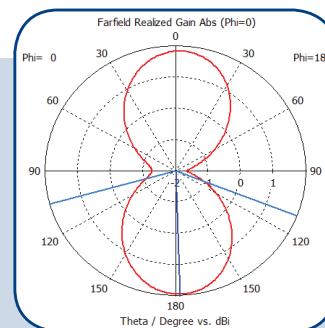
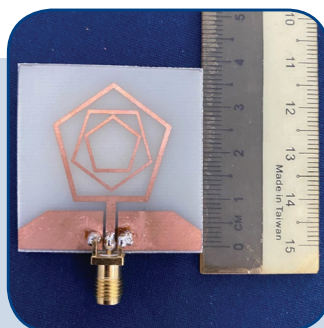
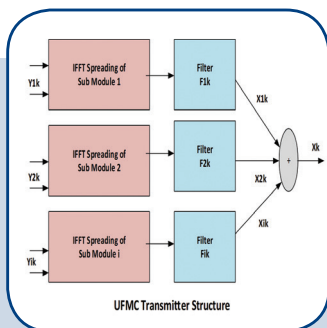
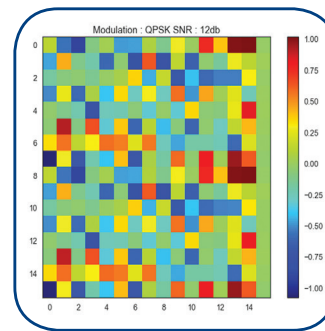
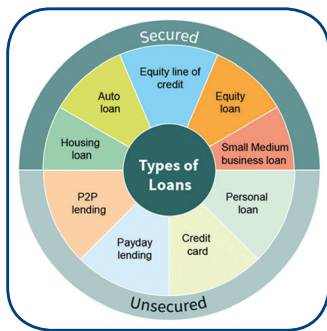


International Journal of Electrical and Computer Engineering Systems



INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia

Osijek, Croatia | Volume 14, Number 2, 2023 | Pages 119 - 239

The International Journal of Electrical and Computer Engineering Systems is published with the financial support
of the Ministry of Science and Education of the Republic of Croatia

CONTACT

**International Journal of Electrical
and Computer Engineering Systems
(IJECS)**

Faculty of Electrical Engineering, Computer
Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b, 31000 Osijek, Croatia
Phone: +38531224600, Fax: +38531224605
e-mail: ijeces@ferit.hr

Subscription Information

The annual subscription rate is 50€ for individuals,
25€ for students and 150€ for libraries.
Giro account: 2390001 - 1100016777,
Croatian Postal Bank

EDITOR-IN-CHIEF

Tomislav Matić
J.J. Strossmayer University of Osijek,
Croatia

MANAGING EDITOR

Goran Martinović
J.J. Strossmayer University of Osijek,
Croatia

EXECUTIVE EDITOR

Mario Vranješ
J.J. Strossmayer University of Osijek, Croatia

ASSOCIATE EDITORS

Krešimir Fekete
J.J. Strossmayer University of Osijek, Croatia

Damir Filko
J.J. Strossmayer University of Osijek, Croatia

Davor Vinko
J.J. Strossmayer University of Osijek, Croatia

Proofreader

Ivanka Ferčec
J.J. Strossmayer University of Osijek, Croatia

Editing and technical assistance

Davor Vrandečić
J.J. Strossmayer University of Osijek, Croatia

Stephen Ward
J.J. Strossmayer University of Osijek, Croatia

Dražen Bajer
J.J. Strossmayer University of Osijek, Croatia

EDITORIAL BOARD

Marinko Barukčić
J.J. Strossmayer University of Osijek, Croatia

Leo Budin
University of Zagreb, Croatia

Matjaz Colnarič
University of Maribor, Slovenia

Aura Conci
Fluminense Federal University, Brazil

Bojan Čukić
West Virginia University, USA

Radu Dobrin
Malardalen University, Sweden

Irena Galić
J.J. Strossmayer University of Osijek, Croatia

Radoslav Galić
J.J. Strossmayer University of Osijek, Croatia

Ratko Grbić
J.J. Strossmayer University of Osijek, Croatia

Marijan Herceg
J.J. Strossmayer University of Osijek, Croatia

Darko Huljenić
Ericsson Nikola Tesla, Croatia

Željko Hocenski
J.J. Strossmayer University of Osijek, Croatia

Gordan Ježić
University of Zagreb, Croatia

Dražan Kozak
J.J. Strossmayer University of Osijek, Croatia

Sven Lončarić
University of Zagreb, Croatia

Tomislav Kilić
University of Split, Croatia

Ivan Maršić
Rutgers, The State University of New Jersey, USA

Kruno Miličević
J.J. Strossmayer University of Osijek, Croatia

Tomislav Mrčela
J.J. Strossmayer University of Osijek, Croatia

Srete Nikolovski
J.J. Strossmayer University of Osijek, Croatia

Davor Pavuna

Ecole Polytechnique Fédérale de
Lausanne, Switzerland

Nedjeljko Perić
University of Zagreb, Croatia

Marjan Popov
Delft University, The Netherlands

Sasikumar Punnekkat
Mälardalen University, Sweden

Chiara Ravasio
University of Bergamo, Italy

Snježana Rimac-Drlje
J.J. Strossmayer University of Osijek, Croatia

Gregor Rozinaj
Slovak University of Technology, Slovakia

Imre Rudas
Budapest Tech, Hungary

Ivan Samardžić
J.J. Strossmayer University of Osijek, Croatia

Dražen Slišković
J.J. Strossmayer University of Osijek, Croatia

Marinko Stojkov
J.J. Strossmayer University of Osijek, Croatia

Cristina Secleanu
Mälardalen University, Sweden

Siniša Srblić
University of Zagreb, Croatia

Zdenko Šimić
University of Zagreb, Croatia

Damir Šljivac
J.J. Strossmayer University of Osijek, Croatia

Domen Verber
University of Maribor, Slovenia

Dean Vučinić
Vrije Universiteit Brussel, Belgium
J.J. Strossmayer University of Osijek, Croatia

Joachim Weickert
Saarland University, Germany

Drago Žagar
J.J. Strossmayer University of Osijek, Croatia

Journal is referred in:

- Scopus
- Web of Science Core Collection
(Emerging Sources Citation Index - ESCI)
- Google Scholar
- CiteFactor
- Genamics
- Hrčak
- Ulrichweb
- Reaxys
- Embase
- Engineering Village

Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003
Published: quarterly
Circulation: 300

IJECS online
<https://ijeces.ferit.hr>

Copyright

Authors of the International Journal of Electrical
and Computer Engineering Systems must transfer
copyright to the publisher in written form.

TABLE OF CONTENTS

Performance Optimization of Universal Filtered Multicarrier Technique for Next Generation Communication Systems.....	119
<i>Original Scientific Paper</i> Shatrughna Prasad Yadav	
CPW Fractal Antenna with Third Iteration of Pentagonal Sierpinski Gasket Island for 3.5 GHz WiMAX and 5.2 GHz WLAN Applications	129
<i>Original Scientific Paper</i> Amier Hafizun Ab Rashid Badrul Hisham Ahmad Mohamad Zoinol Abidin Abd Aziz Nornikman Hassan	
Multi-Head Attention-Based Spectrum Sensing for Cognitive Radio	135
<i>Original Scientific Paper</i> B.V. Ravisankar Devarakonda Venkateswararao Nandanavam	
Comparative Study and Performance Analysis of MANET Routing Protocol	145
<i>Original Scientific Paper</i> Chetana Hemant Nemade Uma Pujeri	
Human Face Emotions Recognition from Thermal Images Using DenseNet	155
<i>Original Scientific Paper</i> S. Babu Rajendra Prasad B. Sai Chandana	
A Performance Enhancement of Deepfake Video Detection through the use of a Hybrid CNN Deep Learning Model	169
<i>Original Scientific Paper</i> Sumaiya Thaseen Ikram Priya V Shourya Chambial Dhruv Sood Arulkumar V	
Feature Selection Model using Naive Bayes ML Algorithm for WSN Intrusion Detection System	179
<i>Original Scientific Paper</i> Deepa Jeevaraj B. Karthik T. Vijayan M. Sriram	
Ensemble Deep Learning Network Model for Dropout Prediction in MOOCs	187
<i>Original Scientific Paper</i> Gaurav Kumar Amar Singh Ashok Sharma	
Iterative Feature Selection-Based DDoS attack Prevention Approach in Cloud	197
<i>Original Scientific Paper</i> Sarah Naiem Ayman E. Khedr Amira M. Idrees Mohamed Marie	
NoSQL Databases: Modern Data Systems for Big Data Analytics - Features, Categorization and Comparison	207
<i>Original Scientific Paper</i> Atul O. Thakare Omprakash W. Tembhurne Abhijeet R. Thakare Soora Narasimha Reddy	
Design and Implementation of a Simulator for Precise WCET Estimation of Multithreaded Programs	217
<i>Original Scientific Paper</i> P. Padma Priya Dharishini P. V. R. Murthy	
Review of Loan Fraud Detection Process in the Banking Sector Using Data Mining Techniques	229
<i>Review Paper</i> Fahd Sabry Esmail Fahad Kamal Alsheref Amal Elsayed Aboutabl	
About this Journal	
IJECES Copyright Transfer Form	

Performance Optimization of Universal Filtered Multicarrier Technique for Next Generation Communication Systems

Original Scientific Paper

Shatrughna Prasad Yadav

Electronics and Communication Engineering Department,
Guru Nanak Institute of Technology
Hyderabad, India
spyadav68@gmail.com

Abstract – Next generation communication systems require better performance to support high - bandwidth, peak data rate, spectral efficiency, mobility, connection density, positioning accuracy, etc. Investigation on efficient modulation technique for next generation has become very important so as to meet its expectations. In this paper performance optimization of universal filtered multicarrier (UFMC) technique for next generation communication systems have been investigated. Dolph-Chebyshev (DC) and Kaiser-Bessel-derived (KBD) filters have been used to optimize power spectral density, channel equalization, bit error rate, and peak to average power ratio (PAPR). It has been observed that KBD filter response is comparatively better than DC filter. Effect of filter length also influences the system performance, filter with bigger length improves performance at the cost of computational complexity. Performance of UFMC has been compared with that of orthogonal frequency division multiplexing (OFDM) technique. The present work of investigations on UFMC that is based on subband filtering is our original research work that has been carried out for its suitability for next generation communication systems. It has simple design structure, lower computational complexities and better performance in terms of BER compared to OFDM and f-OFDM systems. It has comparatively low PAPR than GFDM and FBMC techniques.

Keywords: BER, Dolph-Chebyshev Window, Kaiser-Bessel-derived window, OFDM, OOB, UFMC

1. INTRODUCTION

5G communication systems require better performance in terms of heterogeneity for services and should support high - bandwidth, peak data rate, spectral efficiency, mobility, connection density, positioning accuracy, and low latency, etc. Investigation on efficient modulation technique for 5G and beyond has become very important so as to meet its expectations. Orthogonal frequency division multiplexing (OFDM) has been used as multicarrier communication system in 4G and performs better below 6 GHz signal transmission [1]. It is not suitable for 5G and beyond due to poor out of band (OOB) leakage, poor spectral efficiency, high peak to average power ratio (PAPR), synchronization of data, etc.

To overcome these limitations, several modulation techniques have been investigated in the recent past. These new techniques have been studied under novel orthogonal and non-orthogonal category. Non orthogonal wave shaping has been further investigated under power domain, code domain and multiple domain techniques. Whereas, novel orthogonal technique has been studied under pulse shaping, subband filtering

and few other techniques. Modulation based on novel orthogonal techniques uses either filtering or windowing in frequency or time domain [2].

FBMC and GFDM are pulse shape-based techniques, FBMC uses offset quadrature amplitude modulation (OQAM) and prototype filters: synthesis filter in transmitter and analysis filter in receiver. Among the different types of filters used, PHYDYAS filter has better frequency response [3]. FBMC is better than OFDM in terms of PAPR, channel achievable capacity, SNR, and OOB leakage [4]. GFDM uses circular convolution to apply filtering on a time-frequency block. GFDM has low complexity and better performance and it is suitable for burst signal transmission [5-6].

Universal Filtered Multicarrier (UFMC) and Filtered OFDM (f-OFDM) modulation techniques are based on subband filtering. UFMC is better than other techniques in terms of spectral efficiency, OOB leakage, robustness to time and frequency offset. Owing to its improved performance, UFMC can be used for high data rate transmission. Whereas, f-OFDM filters signal in time domain to reduce mutual interference and attenuation of side lobes. UFMC and f-OFDM have similar

power spectral density but f-OFDM has better timing offset due to use of receiving filters.

The present investigations on UFMC that is based on subband filtering is our original research work that has been carried out for its suitability in 5G and beyond cellular communication applications. Literature review reveals that in the recent past, there has been lot of investigation carried out by researchers that suggests UFMC has better performance than OFDM. Such as, in order to mitigate the effect of interference due to carrier frequency offset (CFO) in uplink systems, adaptive filter has been proposed in [7]. It has been demonstrated that the system performance is getting directly affected by the interference caused by CFO. The parameters of the filter can be adaptively designed to improve data transmission rate and bit error rate (BER). The proposed filter can also be used for different subband bandwidths. A least square (LS) technique-based complexity reduced receiver for UFMC has been proposed in [8] which is computationally efficient. Its symbol error rate (SER) and mean square error (MSE) performance are almost equal to the complex receivers. Its simulation results for number of subcarriers, $N = 128$, subbands, $B = 8$, and successive carriers, $Q = 16$ with a 6-ray Rayleigh fading channel, $Lh = 6$ indicates that symbol error rate decreases with increase in signal to noise ratio (SNR). In order to study the frequency response of overall subcarriers, an efficient channel estimation technique has been proposed in [9]. Simulation result with number of subcarriers, $N = 128$, subbands, $B = 8$, and successive carriers, $Q = 16$ and 40 dB of sidelobe attenuation indicates that MSE decreases and SER increases with an increase in the SNR value. They have demonstrated that their system has better performance with reduced computational complexities. A simplified UFMC structure has been proposed by [10], in which they have eliminated redundant IFFT computations by linking a direct relation between number of subcarriers and number of IFFT elements in a frequency block. They have demonstrated that for a single frequency block with 12 subcarriers, 42 % and 65% computations can be reduced for N (elements) = 64 and 1024 IFFT respectively. In [11], computational complexities of the UFMC system has been reduced by using a poly-phase filter with finite impulse response (FIR) structure. They have demonstrated that the system performance can be improved by adjusting their proposed filter structure. A multi user UFMC has been studied in [12] that is based on optimal filter and zero padding length. They have demonstrated that under a given set of criteria, the system capacity can be maximized with optimal filter length and zero padding / filter tail cutting length. A sparse code multiple access UFMC uplink system in the frequency domain has been investigated in [13]. Using maximum likelihood method, they have analyzed the odd and even component of multiuser detection of frequency domain received signal and demonstrated that the average symbol error probability is sub band independent. A low complexity reliability-based detection

of UFMC system has been proposed in [14] that demonstrated that a two-stage detection first, initial subcarrier wise estimation and then an update of the unreliable signal has better performance with less complexity. A baseband UFMC transmitter based on reconfigurable architecture has been proposed in [15] that has an option to choose number of subcarriers in a subband and type of pulse shaping filters as per the required figure of merit without having significant change in the hardware resources. Side lobe suppression in UFMC system using Kaiser-Bessel window has been investigated in [16] and its performance have been compared with Dolph-Chebyshev window. It has been demonstrated that Kaiser-Bessel has better side-lobe suppression capability even in noisy channel than Dolph-Chebyshev window with similar peak to average power characteristics. Comparative study of OFDM and UFMC systems based on uniform and probabilistic shaping have been reported in [17]. It has been observed that for UFMC system with uniform shaping, there is 3-dB improvement in the receiver sensitivity at the bit error rate of 3.8×10^{-3} , whereas it is reduced to 1.5 dB for the case of probabilistic shaping. It can support high order modulation with enhanced transmission rate [18].

In the remaining part of the paper, universal filtered multicarrier is presented in section-2, section-3 describes the proposed filter-Dolph-Chebyshev (DC) and Kaiser-Bessel-derived (KBD) window and effect of filter length on the system performance. Performance analysis has been presented in section-4, that describes system performance in terms of PSD, Effect of channel equalization, BER, and PAPR. Section-5 deals with result analysis and conclusion of the work is presented in section-6.

2. UNIVERSAL FILTERED MULTICARRIER

Universal Filtered Multicarrier (UFMC) is a subband filtering based modulation technique that has many advantages over other techniques such as, low OOB, low ICI, and better spectrum efficiency as cyclic prefix is not used. The given bandwidth is divided into multiple sub-bands. These sub-bands are made of a group of sub carriers [19]. These sub carriers are filtered individually with help of a finite impulse response (FIR) filters. The filters used are having low side-lobes that gives low OOB, low PAPR, and low inter block interference (IBI).

Fig. 1 shows the transceiver structure of UFMC. First the given 512 subcarriers is divided into 16 subbands and 32 carriers. Each subband input with 32 subcarriers is fed in the transmitter first to the N -point IFFT. Here the signal is de-spreaded and passed to the filter. Zero padding are done in the UFMC to make FFT of $2N$ point size. No cyclic prefix is added and because of independent subband filtering this system is considered to be more flexible [20]. It has total N number of subcarriers, B is the number of subbands and each subband consists of Q number of successive subcarriers in a particular subband, where, $N = Q \times B$.

The received signal X_k is represented by equation (1), where Y_{ik} is the baseband data symbols that is being sent on the i -th subband ($1 \leq i \leq B$), Z_{ik} is the N point IFFT, F_{ik} is the Topleft matrix that is impulse response of the FIR filter of length L [21].

The output signal of the filters is added together and transmitted through the channel after transforming it into radio frequency (bandpass) form, where signals from other users are also added along with additive white Gaussian noise (AWGN) n in the channel. In the receiver, the bandpass signal is re-transformed into baseband signal and processed in the time domain that includes zero padding and windowing [22]. Then it is converted into frequency domain with the help of $2N$ point FFT followed by symbol estimation and sub-carrier equalization [23].

$$X_k = \sum_{i=1}^B F_{ik} \cdot Z_{ik} \cdot Y_{ik} \quad (1)$$

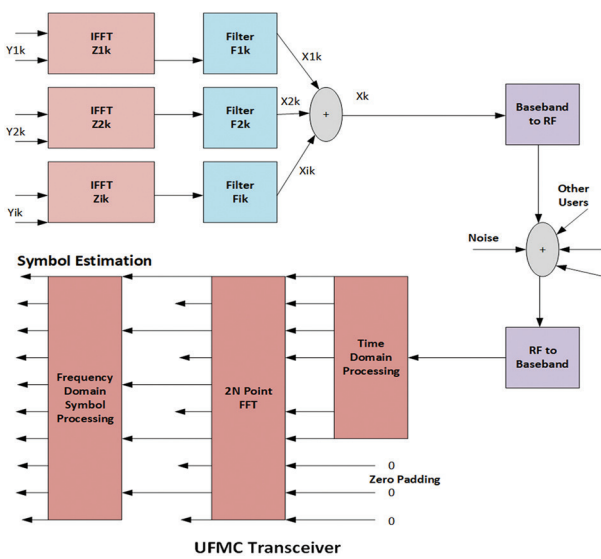


Fig. 1. UFMC Transceiver

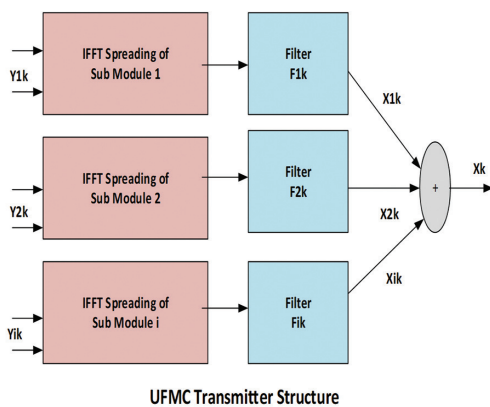


Fig. 2. UFMC Transmitter

Fig. 2 depicts the structure of a UFMC transmitter where the time domain baseband vector X_i is generated by the i th sub-module, it has B number of subbands with N number of samples per symbol. QAM technique has been used as the baseband modulation [24]. First the baseband QAM symbol vectors are spreaded and

converted into time domain using IFFT for a particular subband, then it is filtered first using a FIR Kaiser-Bessel window and then by a Dolph-Chevichev filter [25].

The output of the transmitter is represented by equation 1. The individual matrix vector can be represented by equations 2, 3 and 4.

$$\bar{F} = [F_1, F_2, \dots, F_B] \quad (2)$$

$$\bar{Z} = \text{diag}[F_1, F_2, \dots, F_B] \quad (3)$$

$$\bar{Y} = [Y_1^T, Y_2^T, \dots, Y_B^T]^T \quad (4)$$

After processing the data symbols into a single column, output is represented by equation 5.

$$X = \bar{F} \bar{Z} \bar{Y} \quad (5)$$

$$W_{ZF} = (\bar{F}\bar{Z})^+ = T^+ \quad (6)$$

UFMC receiver structure is shown in Fig. 3 where the received signal is given by $X_k + n$, n is the AGWN noise added in the channel. The receiver can be designed with any efficient filter, here zero forcing (ZF) and minimum mean square error (MMSE) filter has been considered that is represented by equation 6 and 7.

$$W_{ZF} = (\bar{F}\bar{Z})^+ = T^+ \quad (7)$$

$$W_{MMSE} = (T^H \cdot T + \sigma^2 I)^{-1} \cdot T^H \quad (8)$$

Where in, T^H is Hermitian transpose, T^+ is Moore-Penrose inverse, I is identity matrix and σ^2 is the variance of the noise.

After padding with zeros, FFT is of $2N$ length, where N is the number of elements [26].

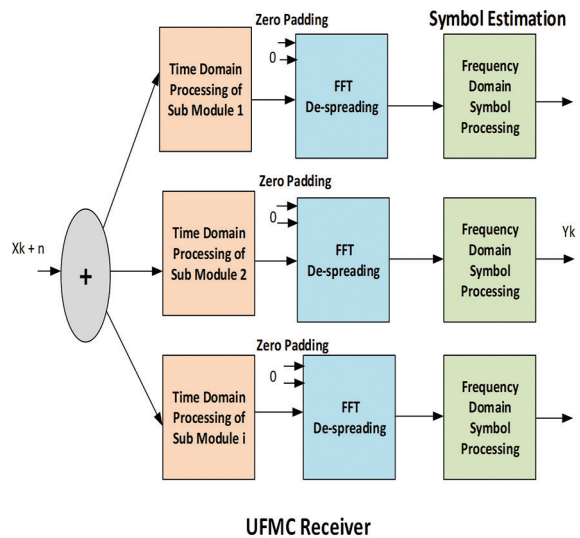


Fig. 3. UFMC Receiver

3. PROPOSED FILTER

Dolph-Chebyshev (DC) and Kaiser-Bessel-derived (KBD) window based low pass FIR filters have been used for investigation of the proposed UFMC transceiver system [27]. The choice of the filter is based on

the fact that it has accurate edges for both passband and stopband, low OOB, lower power leakage, lower sideband lobe, and better power spectral density performance, etc. [28].

It has been observed that KBD has comparatively lower spectral leakage than the DC window.

3.1 KAISER-BESSEL-DERIVED WINDOW

The coefficient of KBD window is represented by equation (8).

$$s_k(n) = \begin{cases} \frac{I_0(\beta \sqrt{1 - (\frac{n}{N/2})^2})}{I_0(\beta)}; & \text{for} \\ -\frac{N-1}{2} \leq n \leq \frac{N-1}{2} \\ 0; & \text{elsewhere} \end{cases} \quad (8)$$

Here, N is the length of the filter, β is the tuning parameter and I_0 is the modified Bessel function of first kind and zero order.

The Fourier transform of KBD window is given by equations 9 and 10.

$$S(f) = \frac{N}{I_0(\beta)} \frac{\sinh[\sqrt{\beta^2 - (\frac{Nf}{2})^2}]}{\sqrt{\beta^2 - (\frac{Nf}{2})^2}} \quad (9)$$

$$S(f) = \frac{N}{I_0(\beta)} \frac{\sin[\sqrt{(\frac{Nf}{2})^2 - \beta^2}]}{\sqrt{(\frac{Nf}{2})^2 - \beta^2}} \quad (10)$$

Where, the modified Bessel function of first kind and zero order, I_0 is given by equation (11).

$$I_0(y) = \sum_{k=0}^{\infty} \left[\frac{(\frac{y}{2})^{k-1}}{k!} \right]^2 \quad (11)$$

Fig. 4 shows the KBD window with sample length of 32 and side lobe of 40 dB in time and frequency domain.

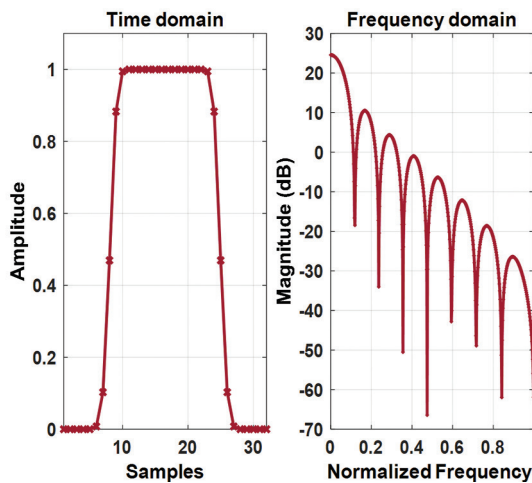


Fig. 4. Kaiser-Bessel-derived (KBD) window with $N = 32$ and side lobe = 40 dB.

Similarly, Fig. 5 depicts KBD window with sample length of 32 and side lobe of 20, 30 & 40 dB both in time and frequency domain.

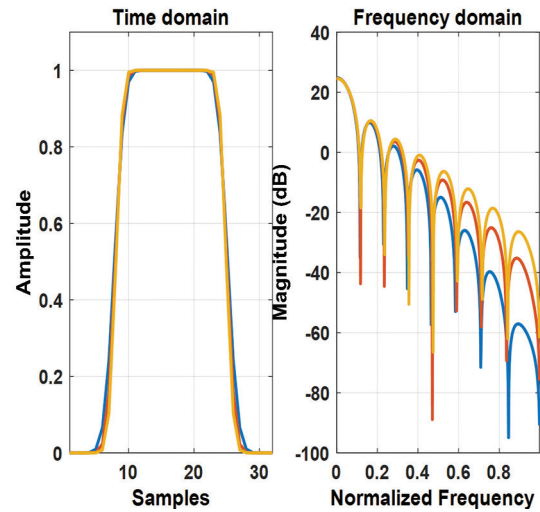


Fig. 5. Kaiser-Bessel-derived window with $N=32$, side lobe = 20, 30 & 40 dB

3.2 DOLPH-CHEBYSHEV WINDOW

The Dolph-Chebyshev window transform is represented by equation (12). Fig. 6 reflects the DC window with sample length of 32 and side lobe of 40 dB in time and frequency domain.

$$S(k) = -1^k \frac{\cos\{N \cos^{-1}(\beta \cos[\frac{\pi k}{N}])\}}{\cosh[\frac{1}{N} \cosh^{-1}(\beta)]} \quad (12)$$

β is defined in equation (13) and α is the representation of the side lobe attenuation.

$$\beta = \cosh[\frac{1}{N} \cosh^{-1}(10^\alpha)] \quad (13)$$

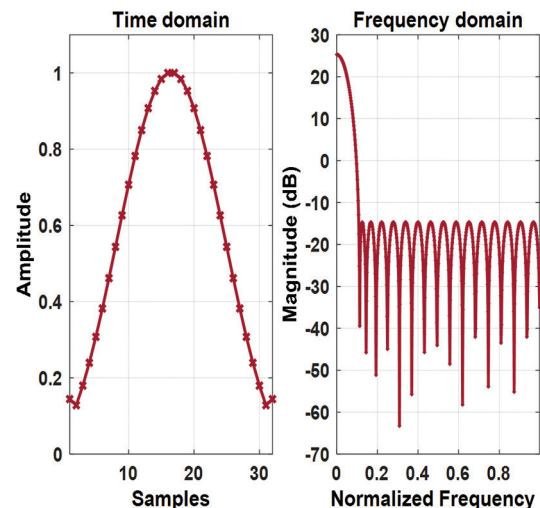


Fig. 6. Dolph-Chebyshev window with $N = 32$ and side lobe = 40 dB.

Whereas, Fig. 7 shows DC window with sample length of 32 and side lobe of 20, 30 & 40 dB both in time and frequency domain [29].

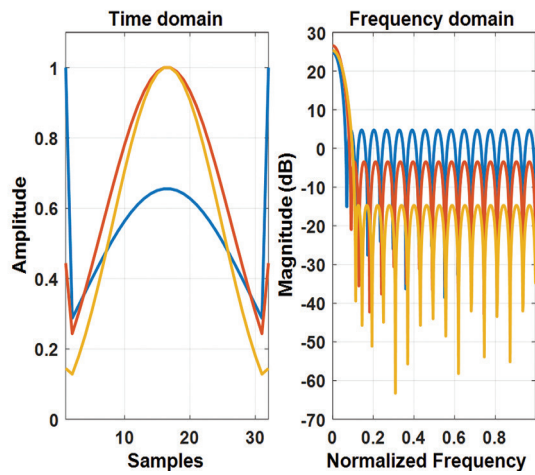


Fig. 7. Dolph-Chebyshev window with $N=32$ side lobe = 20, 30 & 40 dB

3.2 EFFECT OF FILTER LENGTH

The UPMC system performance depends upon the length of the filter. In the case of longer filter length its performance is better in terms of low OOB leakage, more robust to synchronization, and better frequency localization or frequency selectivity. On the other hand, longer filter leads to larger overhead, reduced transmission efficiency, narrow bandwidth, performance loss due to less effective power allocation for a subcarrier in a subband.

Cyclic prefix/ zero padding (CP/ZP) is required to be added in order to nullify the effect of multipath fading. But it causes overhead on the system, reducing spectrum and transmission efficiency with marginal improvement in system performance. To get the optimum efficiency of the system, trade off has to be made to justify the length of CP/ZP with transmission efficiency.

Similarly, filter tail cutting (TC) is required in order to reduce the overhead of the system. To make the system robust to imperfections like, inter carrier interference (ICI), inter symbol interference (ISI), carrier frequency offset (CFO) and timing offset, etc. it is required to choose the optimum length of the filter, CP/ZP, and TC.

4. PERFORMANCE ANALYSIS

Performance analysis using mathematical modelling and Matlab simulations have been carried out for the proposed UPMC system with Dolph-Chebyshev (DC) and Kaiser-Bessel-derived (KBD) window. It has been observed that KBD has comparatively lower spectral leakage than the DC window. Moreover, DC filter does not give optimal result for the UPMC system under considerations due to the fact that its high out of band emissions. On the other hand, performance of KBD filter is better in terms OOB and other desired parameters.

4.1 POWER SPECTRAL DENSITY

Power spectral analysis has been obtained using Matlab simulations using the parameters as depicted

in table 1 for UPMC systems using Dolph-Chebyshev window and Kaiser-Bessel-derived window. From Fig. 8, 9 and 10, it is observed that Kaiser-Bessel-derived window has low power leakage and gives better performance than Dolph-Chebyshev window.

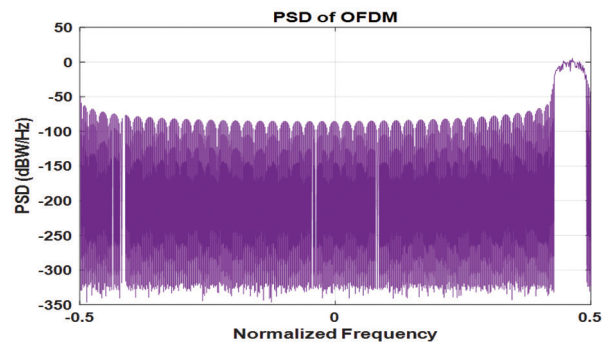


Fig. 8. PSD of OFDM

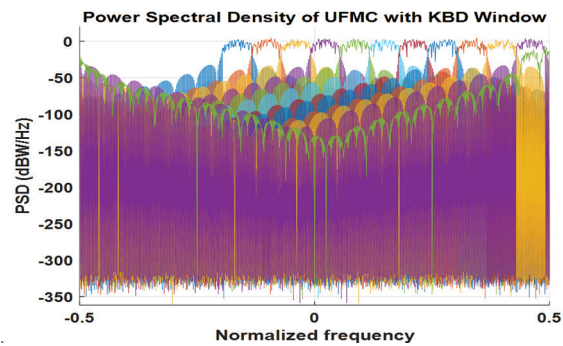


Fig. 9. PSD of UPMC with Kaiser-Bessel-derived window

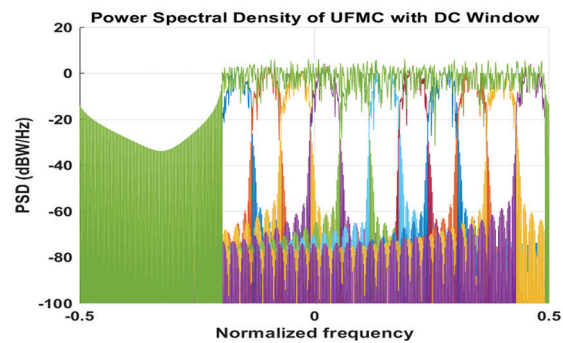


Fig. 10. PSD of UPMC with Dolph-Chebyshev window

Table 1. Simulation Parameters

Description	Value
FFT Size	512
Subband Size (No of Carriers)	32
Number of Subbands	16
subband Offset	156
Filter Length	42
Side Lobe Attenuation	40 dB
Modulation Type	16 QAM
Bits per Sub Carrier	4
SNR	15 dB

4.2. CHANNEL EQUALIZATION

Effect of channel equalization have been studied for the receiver imperfections and insufficient cyclic prefix, zero padding, and tell cutting length (CP, ZP and TC). For its analysis zero forcing (ZF) and minimum mean square error equalizers (MMSE) have been used. Performance of the equalizer for its n th subcarrier is expressed by equation 14.

$$S_n = \frac{\beta(n,n,0)^H}{|\beta(n,n,0)|^2 + m \sigma_{eff}^2 / p_{sym}^2} \quad (14)$$

Where, the parameter m is defined in equation (15) for zero forcing (ZF) and minimum mean square error equalizers (MMSE).

$$m = \begin{cases} 0 & \text{for ZF Receiver} \\ 1 & \text{for MSME Receiver} \end{cases} \quad (15)$$

The effective noise power, σ_{eff}^2 is represented by equation (16), where, P_{ISI} is the noise power due to inter symbol interference (ISI), P_{ICI} is the noise power due to inter carrier interference (ICI), L is the length of the filter and N is period of the received signal.

$$\sigma_{eff}^2 = P_{ISI} + P_{ICI} + \frac{L_2 k}{N} \sigma^2 \quad (16)$$

ZF receiver response is represented by equation (17). But ZF receivers amplify noise also along with received signal.

$$g = B^+ \cdot f_e \quad (17)$$

$$B^+ = (B^H B)^{-1} \cdot B^H \quad (18)$$

Whereas, MMSE receiver does not amplify noise as it uses transformation matrix and minimizes the mean square error distance between the transformed vector and the transmitted signal vector. Its BER performance is better than ZF receivers. The response of the MMSE receiver is represented by equation (19) and (20).

$$g = B^t \cdot f_e \quad (19)$$

and,

$$B^t = \left(\frac{\delta_n^2}{\delta_d^2} I + B^H B \right)^{-1} \cdot B^H \quad (20)$$

Fig. 11 depicts pre-equalization performance of 16QAM UFMC signals.

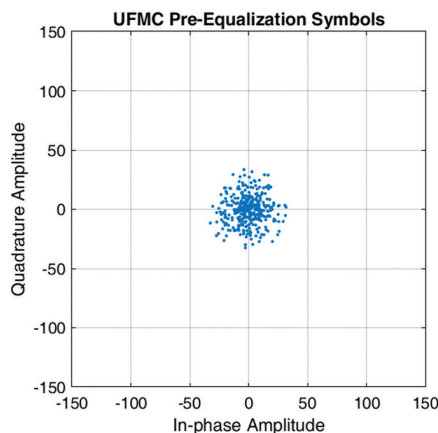


Fig. 11. UFMC Pre- Equalization Symbols

Whereas, Fig. 12 describes post-equalization performance of 16QAM UFMC signals. It is evident from figure 12 that performance of post-equalization is much better than pre-equalization operation.

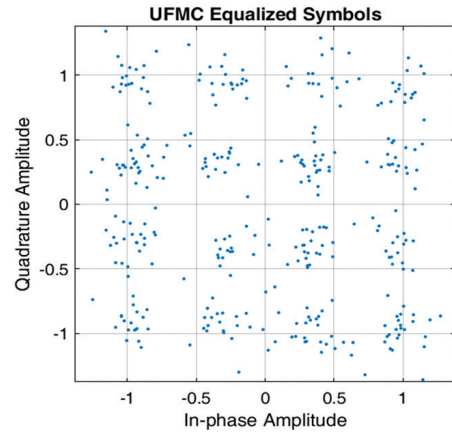


Fig. 12. UFMC Equalized Symbols

4.3. BIT ERROR RATE

BER of UFMC system for 16 QAM baseband format is expressed by equation 21.

$$\text{BER}(x) = 2 \left(1 - \frac{1}{\sqrt{16}}\right) Q \left(\sqrt{\frac{3 \text{SNR}(x)}{16-1}} \right) \quad (21)$$

But analysis using equation 21 is tedious and complex due to the use of Q-function. So, in order to get the result in simplified way, approximation of Q-function has been used as represented in equation 22.

$$Q(n) \approx \frac{1}{12} e^{-\frac{n^2}{2}} + \frac{1}{6} e^{-\frac{2n^2}{3}} \quad (22)$$

Fig. 13 depicts the BER performance of OFDM and UFMC technique with KBD and DC filters. It can be observed that BER performance of OFDM is better than UFMC technique.

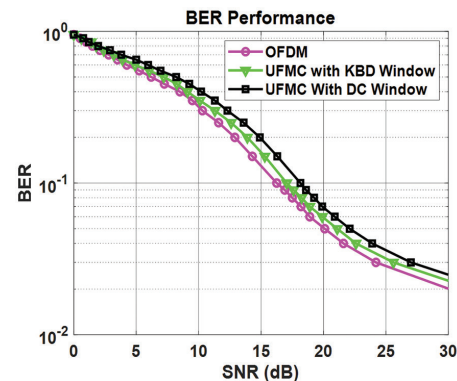


Fig. 13. BER Performance of UFMC with Filters

Performance of the UFMC system in terms of BER has been analyzed using mathematical modelling and Matlab simulations and it has been compared with that of OFDM, f-OFDM, GFDM and FBMC, systems as depicted in Fig. 14. It has been observed that for a given signal to

noise ratio, FBMC has highest BER, followed by GFDM, UPMC, f-OFDM, and OFDM has lowest BER.

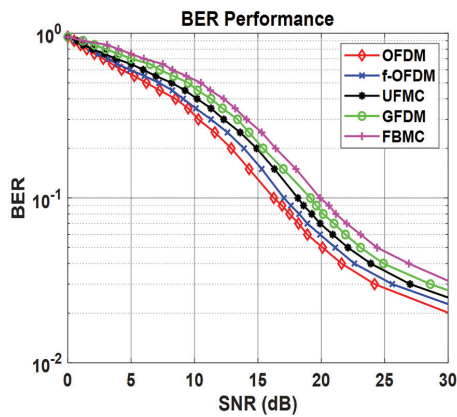


Fig. 14. BER Performance of Different Techniques

4.4. PEAK TO AVERAGE POWER RATIO

PAPR is defined as the ratio of peak power to average power of the given signal. It is a measure of fluctuations in the output of a multicarrier system. For a complex valued multicarrier signal $x(t)$, its PAPR is expressed by equation 23.

$$PAPR \{x(t)\} = \frac{\max\{x(t)\}^2}{Avg \{x(t)\}^2} \quad (23)$$

To find out the probability that PAPR of a system exceeds a given threshold value, complementary cumulative distribution function (CCDF) is used. CCDF has been expressed by equation 24, where x is the threshold value, P is the probability that the maximum value is greater than the threshold value x , and C is the CCDF.

$$\begin{aligned} Cx_{max}(x) &= P(x_{max} > x) \\ &= 1 - P(x_{max} \leq x) \\ &= 1 - Cx_{max}(x) \end{aligned} \quad (24)$$

Fig. 15 shows the CCDF performance of the UPMC system. It can be observed from the figure 15 that at 6 dB of PAPR value, CCDF value is 0.01 and 0.007 for Dolph-Chebyshev and Kaiser-Bessel-derived window respectively.

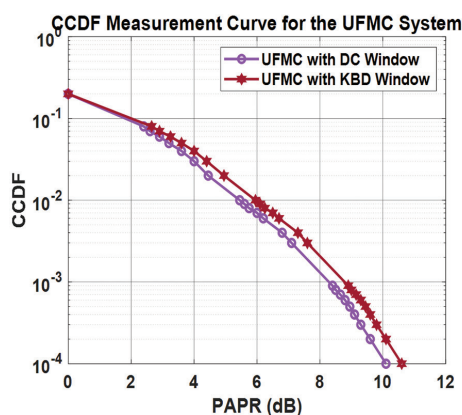


Fig. 15. CCDF Measurement of the UPMC System

Fig. 16 depicts the PAPR performance of OFDM and UPMC technique with KBD and DC filters. It can be observed that PAPR performance of UPMC with DC filter is better than UPMC with KBD filter, and OFDM technique. OFDM has the highest PAPR for the given number of sub carrier.

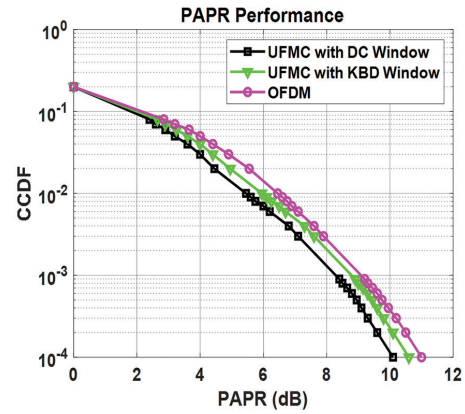


Fig. 16. PAPR Performance of UPMC with Filters

PAPR of UPMC has been computed and it has been compared with that of OFDM, f-OFDM, GFDM and FBMC, systems as depicted in Fig. 17. It Reveals that OFDM has highest PAPR followed by f-OFDM, UPMC, GFDM and FBMC has lowest PAPR. It can be concluded that BER and PAPR are inversely proportional to each other. UPMC system has better performance in terms of PAPR compared to the OFDM system.

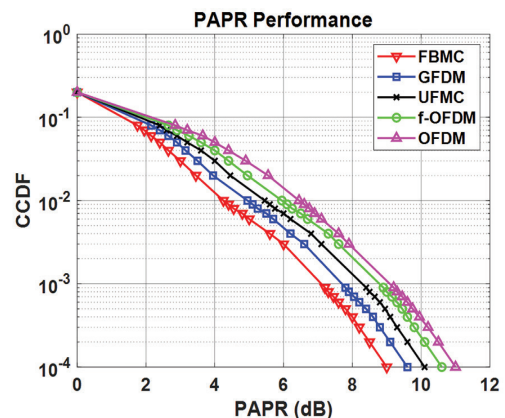


Fig. 17. PAPR Performance of Different Techniques

5. RESULT ANALYSIS

The study reveals that Kaiser-Bessel-derived window has low power leakage and gives better performance than Dolph-Chebyshev window. KBD filter response is comparatively better than DC filter in the case of power spectral density and out of band leakage, and side lobe area. The study further suggests that effect of post-equalization has greater impact on the received signal quality.

Table 2 depicts the performance of different modulation techniques in terms of its BER and PAPR values. It can be observed from the table that UPMC has com-

paratively average performance without any computational complexities with simple transceiver structure.

Performance of the UFMC system in terms of its BER and PAPR has been compared with that of OFDM, f-OFDM, GFDM and FBMC, systems. It has been observed that for a given signal to noise ratio, FBMC has highest BER, followed by GFDM, UFMC, f-OFDM, and OFDM. It further reveals that for a given number of subband carrier, OFDM has highest PAPR followed by f-OFDM, UFMC, GFDM and FBMC. The result obtained indicates UFMC performance is much superior in the case of PAPR, spectrum efficiency, etc. than OFDM technique.

Table 2. System Performance

Sl. No.	Modulation Technique	SNR (dB)	BER (dB)	PAPR (dB)	CCDF
1	OFDM	20	0.05	8	0.003
2	F-OFDM	20	0.06	8	0.001
3	UFMC	20	0.07	8	0.0008
4	GFDM	20	0.08	8	0.0006
5	FBMC	20	0.09	8	0.0004

Performance of the UFMC system in terms of its BER and PAPR has been compared with that of OFDM, f-OFDM, GFDM and FBMC, systems. It has been observed that for a given signal to noise ratio, FBMC has highest BER, followed by GFDM, UFMC, f-OFDM, and OFDM. It further reveals that for a given number of subband carrier, OFDM has highest PAPR followed by f-OFDM, UFMC, GFDM and FBMC. The result obtained indicates UFMC performance is much superior in the case of PAPR, spectrum efficiency, etc. than OFDM technique.

6. CONCLUSION

In the present work, investigations on UFMC that is based on subband filtering have been carried out for its suitability for next generation communication systems. It has been observed that UFMC is better than other techniques in terms of spectral efficiency, OOB leakage, robustness to time and frequency offset. Owing to its improved performance, UFMC can be used as multi carrier communication system with high data rate transmission capability. Its PAPR performance is far better than the OFDM system.

7. REFERENCES

- [1] B. F. Boroujeny. "OFDM versus filter bank multicarrier", *IEEE Signal Processing Magazine*, Vol. 28, No. 3, 2011, pp. 92-112.
- [2] Y. S. Prasad, B. S. Chandra, "Hardware Implementation of PAPR Reduction with Clipping and Filtering Technique for Mobile Applications", *International Journal of Engineering and Technology*, Vol. 8, No. 5, 2016, pp. 2018-2033.
- [3] J. Dang, Z. Zhang, L. Wu, Y. Wu, "A New Framework of Filter Bank Multi-Carrier: Getting Rid of Subband Orthogonality", *IEEE Transactions on Communications*, Vol. 65, No. 9, 2017, pp. 3922-3932.
- [4] Y. S. Prasad, "Filter Bank Multicarrier Modulation Techniques for 5G and Beyond Wireless Communication Systems", *European Journal of Electrical Engineering and Computer Science*, Vol. 6, No. 2, 2022, pp. 18-24.
- [5] P. Wei, Y. Xiao, L. Dan, L. Ge, W. Xiang, "N-Continuous Signaling for GFDM", *IEEE Transactions on Communications*, Vol. 68, No. 2, 2020, pp. 947-958.
- [6] Y. S. Prasad, "Orthogonal Versus Novel Orthogonal Pulse Shaped Waveforms for Future Generation Wireless Communication Systems", *Proceedings of the IEEE R10 HTC 2022 Conference*, Hyderabad, India, 16-18 September 2022.
- [7] X. Chen, L. Wu, Z. Zhang, J. Dang, J. Wang, "Adaptive Modulation and Filter Configuration in Universal Filtered Multi-Carrier Systems", *IEEE Transactions on Wireless Communications*, Vol. 17, No. 3, 2018, pp. 1869-1881.
- [8] T.-T. Lin, T.-C. Chen, "Complexity-Reduced Receiver for Universal Filtered Multicarrier Systems", *IEEE Wireless Communications Letters*, Vol. 8, No. 6, 2019, pp.1667-1670.
- [9] T.-T. Lin, T.-C. Chen, "Efficient Channel Estimation for Universal Filtered Multicarrier Systems", *IEEE Systems Journal*, Vol. 15, No. 3, 2021, pp. 3793-3796.
- [10] A. R. Jafri et al. "Hardware Complexity Reduction in Universal Filtered Multicarrier Transmitter Implementation", *IEEE Access*, Vol. 5, 2017, pp. 13401-13408.
- [11] Z. Guo, Q. Liu, W. Zhang, S. Wang, "Low Complexity Implementation of Universal Filtered Multi-Carrier Transmitter", *IEEE Access*, Vol. 8, 2020, pp. 24799-24807.
- [12] L. Zhang et al. "Optimal Filter Length and Zero Padding Length Design for Universal Filtered Multi-Carrier (UFMC) System", *IEEE Access*, Vol. 7, 2019, pp. 21687-21701.
- [13] C. Chen, J. Zheng, F. Si, "Sparse Code Multiple Access Over Universal Filtered Multicarrier", *IEEE Transactions on Vehicular Technology*, Vol. 70, No. 10, 2021, pp. 10335-10346.

- [14] F. Si, J. Zheng, C. Chen, "Reliability-Based Signal Detection for Universal Filtered Multicarrier", *IEEE Wireless Communications Letters*, Vol. 10, No. 4, 2021, pp. 785-789.
- [15] V. Kumar, M. Mukherjee, J. Lloret, "Reconfigurable Architecture of UFMC Transmitter for 5G and its FPGA Prototype", *IEEE Systems Journal*, Vol. 14, No. 1, 2020, pp. 28-38.
- [16] R. S. Yarrabothu, U. R. Nelakuditi, "Optimization of Out-of-Band Emission Using Kaiser-Bessel Filter for UFMC in 5G Cellular Communication", *China Communications*, Vol. 16, No. 8, 2019, pp. 15-23.
- [17] C. Zhang et al. "Experimental Comparison of Orthogonal Frequency Division Multiplexing and Universal Filter Multi-Carrier Transmission", *Journal of Lightwave Technology*, Vol. 39, No. 22, 2021, pp. 7052-7060.
- [18] Y. S. Prasad, "Pulse Based GFDM Modulation Technique for Future Generation Communication Systems", *European Journal of Electrical Engineering and Computer Science*, Vol. 6, No. 6, 2022, pp. 1-8.
- [19] L. Zhang, A. Ijaz, P. Xiao, R. Tafazolli, "multi-service system: An enabler of flexible 5G air interface", *IEEE Communications Magazine*, Vol. 55, No. 10, 2017, pp. 152-159.
- [20] "5GNOW, D3.2: 5G waveform candidate selection", Technical Report, 2014.
- [21] G. Wunder et al. "5GNOW: Non-orthogonal, asynchronous waveforms for future mobile applications", *IEEE Communications Magazine*, Vol. 52, No. 2, 2014, pp. 97-105.
- [22] X. Chen, S. Zhang, A. Zhang, "On MIMO-UFMC in the presence of phase noise and antenna mutual coupling", *Radio Science*, Vol. 52, No. 11, 2017, pp. 1386-1394.
- [23] H. Enver, S. Fatlum, "Spectrum Comparison between GFDM, OFDM and GFDM Behavior in a Noise and Fading Channel", *International Journal of Electrical and Computer Engineering Systems*, Vol. 6, No. 2, 2015, pp. 39-43.
- [24] J. Andrews, S. Buzzi, W. Choi, S. Hanly, A. Lozano, A. Soong, J. Zhang, "What will 5G be?", *IEEE Journal on Selected Areas in Communications*, Vol. 32, No. 6, 2014, pp. 1065-1082.
- [25] Y. Tao, L. Liu, S. Liu, Z. Zhang, "A survey: Several technologies of non-orthogonal transmission for 5G", *IEEE China Communications*, Vol. 12, No. 10, 2015, pp. 1-15.
- [26] M. Fuhrwerk, S. Moghaddamnia, J. Peissig, "Scattered pilot-based channel estimation for channel adaptive FBMC-OQAM systems", *IEEE Transactions on Wireless Communications*, Vol. 16, No. 3, 2017, pp. 1687-1702.
- [27] X. Wang, T. Wild, F. Schaich, A. F. Dos Santos, "Universal filtered multi-carrier with leakage-based filter optimization", *Proceedings of the 20th European Wireless Conference, Barcelona, Spain, 14-16 May 2014*, pp. 1-5.
- [28] X. Wang, T. Wild, F. Schaich, "Filter optimization for carrierfrequency- and timing-offset in universal filtered multi-carrier systems", *Proceedings of the IEEE 81st Vehicular Technology Conference, Glasgow, UK, 11-14 May 2015*, pp. 1-6.
- [29] P. H. Moose, "A technique for orthogonal frequency division multiplexing frequency offset correction", *IEEE Transactions on Communications*, Vol. 42, No. 10, 1994, pp. 2908-2914.

CPW Fractal Antenna with Third Iteration of Pentagonal Sierpinski Gasket Island for 3.5 GHz WiMAX and 5.2 GHz WLAN Applications

Original Scientific Paper

Amier Hafizun Ab Rashid

Centre for Telecommunication Research and Innovation (CeTRI), Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer (FKeKK) Universiti Teknikal Malaysia Melaka (UTeM), Malaysia amier.utem@gmail.com

Badrul Hisham Ahmad

Centre for Telecommunication Research and Innovation CeTRI), Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer (FKeKK) Universiti Teknikal Malaysia Melaka (UTeM), Malaysia badrulhisham@utem.edu.my

Mohamad Zoinol Abidin Abd Aziz

Centre for Telecommunication Research and Innovation CeTRI), Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer (FKeKK) Universiti Teknikal Malaysia Melaka (UTeM), Malaysia mohamadzoinol@utem.edu.my

Nornikman Hassan

Centre for Telecommunication Research and Innovation CeTRI), Fakulti Kejuruteraan Elektronik dan Kejuruteraan Komputer (FKeKK) Universiti Teknikal Malaysia Melaka (UTeM), Malaysia nornikman@yahoo.com

Abstract – Nowadays, the compact and multiband antennas are typically required for personal communication devices in the continuously developing wireless communication industry. The fractal antenna with the third iteration of the pentagonal Sierpinski gasket island for WiMAX and WLAN applications is presented in this work. It starts with the fundamental design of the square patch (Antenna A1) and pentagonal patch (Antenna A2). This simulation work is done using CST Microwave Studio simulation studio by applying the concept of the zero, first, second and third fractal iteration. Then, it goes on to use the fractal geometry concept of the Sierpinski gasket island structure with three designs step. The designs consist of the first iteration (Antenna B1), second iteration (Antenna B2) and third iteration (Antenna B3) of fractal geometry. The simulation work of Antenna B3 is compared with the fabrication work of the same design. After that, the measurement of the Antenna B3 is done in laboratory with -29.55 dB at 3.41 GHz and -20.40 dB at 5.28 GHz for its operating frequencies with bandwidth of 3.52 GHz and 5.48 GHz, respectively. At targeting 3.5 GHz WiMAX, 5.2 GHz WLAN application and 7.24 GHz of Antenna B3, the antenna shows the -17.78 dB, -29.63 dB and -22.73 dB, respectively, and this value is feasible for WiMAX and WLAN operation.

Keywords: Fractal Geometries, Patch Antenna, Sierpinski Gasket, Co-Planar Waveguide, Return Loss

1. INTRODUCTION

In the rapidly expanding wireless communication world, compact and multiband antennas are typically needed for personal communication devices. To fulfill the enormous demand for contemporary wireless applications, multiband, miniaturized microstrip patch antennas are needed [1]. Due to its appealing qualities, including its straightforward design, low profile, high efficiency, affordable manufacturing, and acceptable radiation properties, planar monopole antennas with various configurations are regarded as the most common in this field [2].

The fractal geometry is a useful technique for building multi-band and low-profile antennas to reduce the interference caused by the presence of other adjacent communication systems [3].

The fundamental resonant mode in the microwave and millimetre frequency ranges is demonstrated by fractal geometry structures (FGS), whose geometrical structures influence the resonant frequency [4]. These requirements can be met by using fractals in the antenna design. For example, a Hayder-Koch fractal geometry structure is applied [5]. Besides that, a frequency-selective surface also used the same concept to improve its performance by adding this fractal geometry design. For example, it uses using Koch fractal hexagonal loop [6]. Therefore, this is utilized to calculate the multi-band frequency of the antenna effect and is made up of several variations of a single fundamental form.

It has been demonstrated that fractal geometry is helpful in several fields. Fractal geometry is useful for designing tiny, multiband antenna arrays, and high-di-





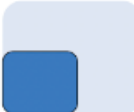
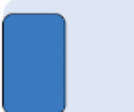
rective components in the field of antenna engineering. The fractal dimension, a complexity measure developed by Mandelbrot and based on his 1967 work on fractional dimensions, serves as the primary description of fractal sets [7]. Based on basic mathematical shapes, fractal structures can be divided into two categories. Firstly, it is a deterministic fractal structure with examples of the Sierpinski gasket [8-9], fern fractal [10], von-Koch snowflakes [11], and Minkowski [12]. Then, the creation of random fractal structures is random by natural phenomena such as dendrites and lightning bolts.

The self-similar structure of the fractal antenna was created by iterative design. Repeating the iteration endlessly results in a shape with a finite length but an infinite area within a limited border. Several studies have described novel fractal designs that provide a variety of geometry for lowering compact magnitudes, enlarging antennas, and maintaining the same radiating properties over the bandwidth [13].

Sets known as fractals display a repeating pattern at various scales. Table 1 shows an example of the similarity and not similarity of the first iteration from the zero iteration stage. Contraction decreases shapes, and it's possible that the shrinking is more significant in one direction than another, according to the inequality in the definition of a contraction [14].

The fractal antenna's subsequent phases are generated by the seed antenna. The initial frequency of the Sierpinski antenna is determined by the seed antenna's size. The development of the fractal is determined by the desired number of bands [15].

Table 1. Example of the similarity and not similarity of the first iteration from the zero iteration stage

Zero iteration	First iteration	
	Apply a similarity	Not apply a similarity
		
		

Fractal geometry structures have some benefits, including small size, improved input impedance, wide-band/multiband support for numerous applications, consistent performance over a broad frequency range, and—not least—the ability to add inductance and capacitance without the need for additional components. Furthermore, low-side lobes arrays, under-sampled arrays, and high-directivity elements can be made using mass fractals and boundary fractals [16].

Waclaw Sierpinski discovered the Sierpinski fractal geometry in 1915, and Sierpinski gaskets have been thoroughly studied for monopole and dipole antenna layouts. Over the last two decades, researchers have investigated a variety of fractal geometries. However, since its introduction in 1998 by Puente-Baliarda, the Sierpinski gasket fractal antenna geometry has been studied more than any other geometry [17].

There is several research on the Sierpinski gasket fractal antenna have been done. Kaur [18] introduces a complementary Sierpinski gasket fractal antenna array for wireless MIMO portable devices with 8.2 % bandwidth at the center frequency of 4.94 GHz (4.74 GHz - 5.15 GHz). In another work, Ramli [19] designed a Sierpinski Gasket Fractal Antenna (SGFA) with slits for multiband applications at 2.4 GHz and 5.0 GHz [19]. In other work, Chaouche had been introduced a modified Sierpinski gasket fractal antenna for tri-band applications with a bandwidth range between 1.6 GHz to 2.05 GHz, 4.88 GHz to 6.13 GHz, and 9.86 GHz to 10.34 GHz [20]. Other works also are shown to use the Sierpinski gasket fractal in their antenna design [21-23].

The fractal antenna is designed in this work with the third iteration of the pentagonal Sierpinski gasket island for 3.5 GHz WiMAX and 5.2 GHz WLAN applications. CST Microwave Studio is the simulation software used for this experiment to define the wanted design. Then, the fabricating work of the same design is contrasted with the simulation work of the suggested antenna. The proposed antenna is then measured in a laboratory for return loss, resonance frequency, and gain.

2. ANTENNA DESIGN

The antenna is designed to step by step, starting with the basic design antenna patch. Antenna *A*, which has a design of a basic patch of rectangular shape (Antenna *A1*) and pentagonal shape (Antenna *A2*), had been done. Then it moves to the next design of Antenna *B*, consisting of three different sub-shaped iterations – Antenna *B1*, Antenna *B2* and Antenna *B3*.

2.1. SIERPINSKI GASKET FRACTAL ITERATION CONCEPT

In this work, the Sierpinski gasket fractal is designed in the iteration stage. It starts with the zero iteration concept designed in a pentagonal shape. Then, it goes to the first iteration fractal with a Sierpinski gasket island shape. Fig. 1 represents the Sierpinski gasket fractal iteration concept from zero to the third iteration stage.

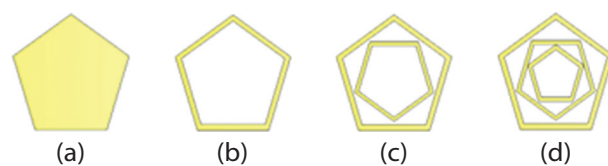


Fig. 1. Sierpinski gasket fractal iteration concept, (a) zero iteration, (b) first iteration, (c) second iteration, (d) third iteration

Next, a similar shape with a reduced size of the Sierpinski gasket island shape is added to become a second iteration fractal. Lastly, the step is repeated with a third iteration fractal step consisting of three Sierpinski gasket islands of different sizes.

2.2. ANTENNA A

The first part of the design is the Antenna *A1*, which is made of copper that is 0.035 mm thick and is mounted on an FR4 substrate that is 1.6 mm thick and has a dielectric constant of 4.3. It has a substrate, a co-planar waveguide (CPW) structure, a patch with a pentagonal shape, and a feeding line that connects to the antenna source (waveguide). This rectangular patch is the basic structure design, then it follows as the pentagonal patch structure design.

It shows that Antenna *A1* measured 22.0 mm in width and 22.0 mm in length, respectively. While the other design of Antenna *A2* has a dimension of 25.2 mm width and 25.1 mm length. Besides that, the width of a pentagonal island line is 1.26 mm. The fundamental square and pentagonal antenna structure of Antenna *A1* and Antenna *A2* is shown in Fig. 2.

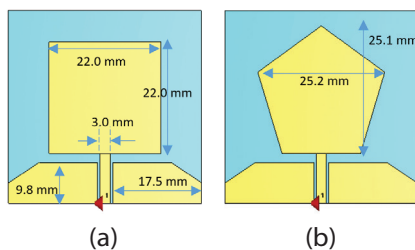


Fig. 2. Basic antenna design with CPW of Antenna *A*, (a) Antenna *A1* – square (b) Antenna *A2* – pentagonal

2.3. ANTENNA B

Then, it goes to the second part of Antenna *B*, which applies the fractal concept. It follows with Antenna *B1*, Antenna *B2* and Antenna *B3* with the first iteration, second iteration and third iteration step of the Sierpinski Gasket Island fractal. Fig. 3 shows the proposed design with CPW of Antenna *B*. Each iteration had an addition of a similar shape but with reduced dimension sizes of pentagonal island ring-shaped from one, two and three depending on the iteration number steps.

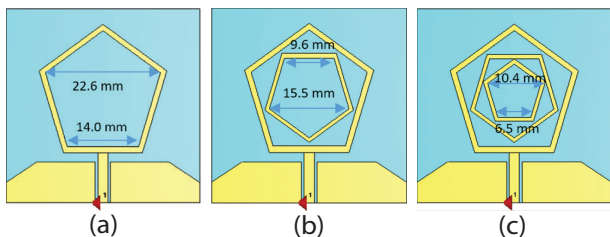


Fig. 3. Proposed design with CPW of Antenna *B*, (a) Antenna *B1* – first iteration, (b) Antenna *B2* - second iteration, (c) Antenna *B3* – the third iteration

3. RESULTS

The desired effect performance result of the microstrip patch antenna design with fractal geometric structure is supported by several notable findings. The key conclusions are bandwidth in GHz, return loss in GHz (dB versus frequency), antenna gain in GHz (dB versus frequency), and antenna radiation pattern. For resonance frequency, this antenna must transmit and receive at least 90 % of the signal. Besides that, the antenna's suitable result for antenna gain must be greater than 1 dB.

Antenna *A1* performance result is shown in Fig. 4. It shows the return loss at the first resonant frequency point of 3.35 GHz with – 12.45 dB, and it was found that this antenna's gain and bandwidth range performance were 2.00 dB and 2.96 GHz – 3.73 GHz, respectively. It goes to the second stage of Antenna *A2* at two different resonant frequencies of 3.52 GHz and 5.99 GHz with – 20.69 dB and – 24.77 dB of return loss. The antenna's gain shows are 5.04 dB. Besides that, the bandwidth of 1.61 GHz and 2.69 GHz for the first and second resonant frequencies of Antenna *A2*. Table 2 represents the performance results of Antenna *A1* and *A2*.

Table 2. Performance results of Antenna *A*

Ant	Resonant frequency, f_r (GHz)	Return loss (dB)	Bandwidth (GHz), $f_{High} - f_{Low}$ (GHz)	Gain (dB)
<i>A1</i>	3.35	- 12.45	0.77, 2.96 – 3.73	2.00
<i>A2</i>	3.52	- 20.69	1.61, 2.66 – 4.27	2.18
	5.99	- 24.77	2.69, 4.97 – 7.66	5.04

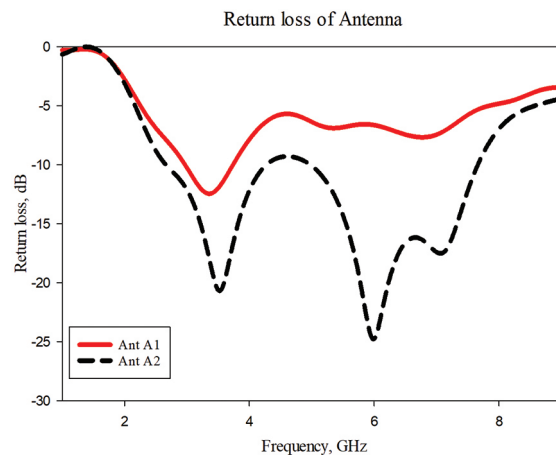


Fig. 4. Return loss of basic antenna design with CPW of Antenna *A*, (a) Antenna *A1* - rectangular, (b) Antenna *A2* – pentagonal

Fig. 5 and also Table 3 represent the performance results of Antenna *B*. It shows performance between several antennas of *B1*, *B2*, and *B3* with the first, second and iteration structure of the Sierpinski gasket fractal. Antenna *B1* effects of operating at two resonant frequencies of 2.34 GHz and 5.45 GHz with a return loss of – 12.28 dB and – 20.59 dB. It shows the bandwidth of 0.38 GHz and

1.92 GHz for the first and second resonant frequencies. Next, it goes to the Antenna B2, which operates at two different resonant frequencies of 2.26 GHz and 5.23 GHz with a return loss of - 11.15 dB and - 30.79 dB, respectively. It displays the 0.16 GHz and 0.79 GHz bandwidths for the first and second resonant frequencies, respectively.

The last stage shows the Antenna B3 with three locations of the resonant frequencies at 3.46 GHz, 5.19 GHz, and 7.24 GHz with return loss shown as - 23.43 dB, - 30.27 dB and - 22.72 dB, respectively.

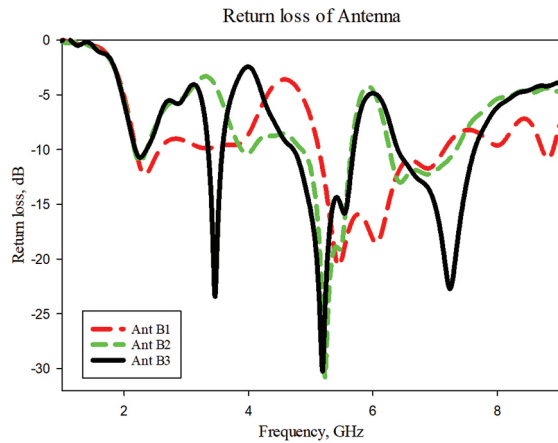


Fig. 5. Return loss of proposed design with CPW of Antenna B

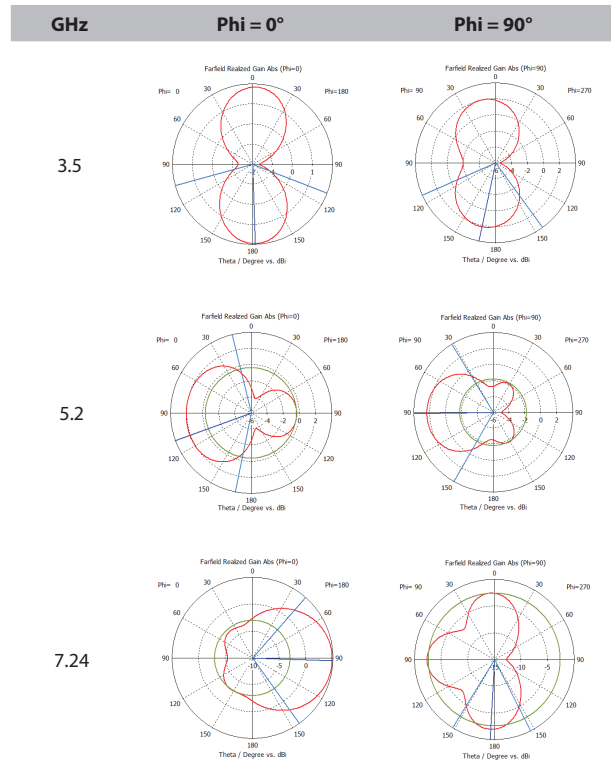
Table 3. Several performance results of Antenna B

Ant	Resonant frequency, f_r (GHz)	Return loss (dB)	Bandwidth (GHz), $f_{High} - f_{Low}$ (GHz)	Gain (dB)
B1	2.34	- 12.28	0.38, 2.19 – 2.57	1.72
	5.45	- 20.59	1.92, 5.23 – 7.15	2.76
B2	2.26	- 11.15	0.16, 2.16 – 2.32	1.50
	5.23	- 30.79	0.79, 4.86 – 5.65	3.31
B3	3.46	- 23.43	0.23, 3.35 – 3.58	2.04
	5.19	- 30.27	0.06, 4.72 – 5.68	4.05
	7.24	- 22.72	1.24, 6.40 – 7.64	5.06

Table 4 shows that $\phi = 0^\circ$ and $\phi = 90^\circ$ display the radiation pattern of the microstrip patch antenna from Antenna B3, where the radiation pattern symbolizes the distribution of electromagnetic power in free space. At $\phi = 0^\circ$ and $\phi = 90^\circ$ of 3.5 GHz, it displays an eight-shaped pattern for the first resonant frequency, with some lobes facing forward in the 00 direction and others facing toward the 180° direction.

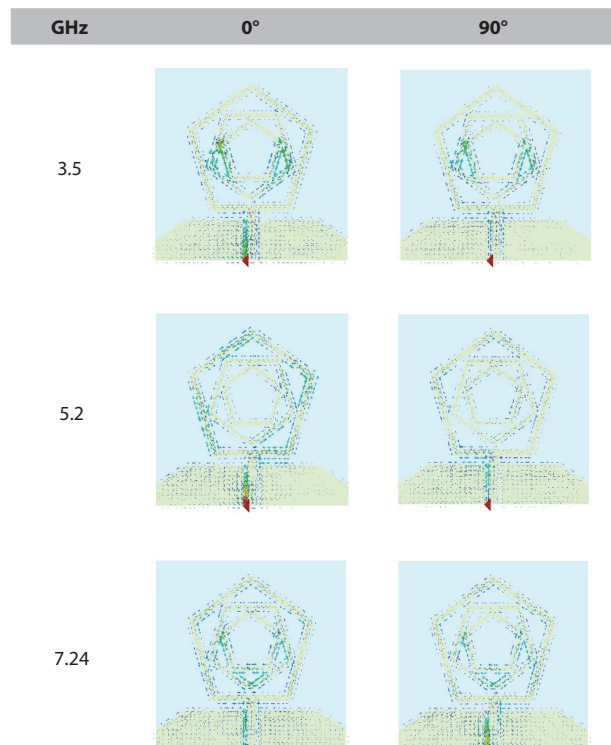
For 5.2 GHz, it shows the right direction major lobes and left path minor lobes are visible. In contrast to $\phi = 90^\circ$, it shows nearly the same as 0° . The right direction main lobes and left path minor lobes are evident for 7.24 GHz of $\phi = 0^\circ$. On the other hand, the $\phi = 90^\circ$ shows three locations of lobes at 0° , 90° and 180° .

Table 4. Performance results of the radiation pattern of Antenna B3



Then, it goes to Table 5 of surface current for the Antenna B3 at three different resonant frequencies at $\phi = 0^\circ$ and $\phi = 90^\circ$. The observation shows that the 3.5 GHz is more concentrated at the second ring, 5.2 GHz at the first ring and the 7.24 GHz effect at the third ring.

Table 5. Surface Current of the Antenna B3



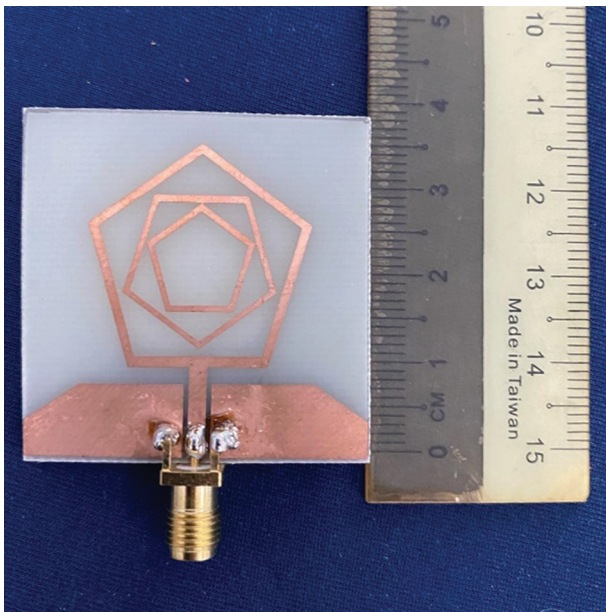


Fig. 6. Fabricated Antenna *B3*

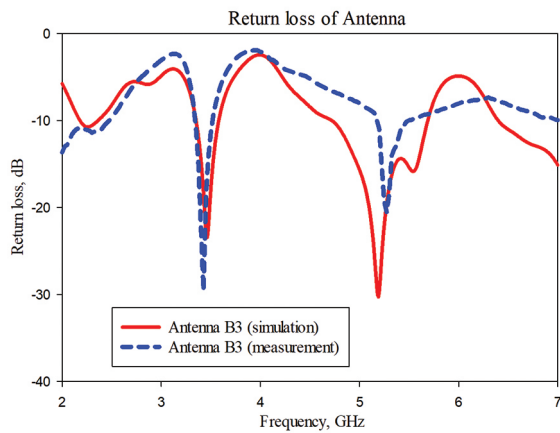


Fig. 7. Return loss of proposed design with CPW of Antenna *B3* (simulation and measurement)

Table 6. Several performance results of Antenna *B3* (simulation and measurement)

Ant	Resonant frequency, f_r (GHz)	Return loss (dB)	Bandwidth (GHz), $f_{High} - f_{Low}$ (GHz)	Gain (dB)
<i>B3 sim</i>	3.46	-23.43	0.23, 3.35 - 3.58	2.04
	5.19	-30.27	0.06, 4.72 - 5.68	4.05
<i>B3 meas</i>	3.41	-29.55	0.16, 3.36 - 3.52	1.98
	5.28	-20.40	0.31, 5.17 - 5.48	3.89

4. CONCLUSION

This work presents various pentagonal microstrip patch antenna designs with several advancement techniques of fractal iteration. Antenna *A1* and *A2* are the first and second basic square and pentagonal patch antenna shapes. Then the Sierpinski gasket island fractal structure's first, second and third iterations are ap-

plied into three parts of Antenna *B1*, Antenna *B2* and Antenna *B3*. As a result of the antenna's designs, the performance of the return loss, gain, and the first, second and third Sierpinski gasket iterations impacted the radiation pattern. For measurement performance, it indicates that -29.55 dB at 3.41 GHz and -20.40 dB at 5.28 GHz for its operating frequencies with bandwidth of 3.52 GHz and 5.48 GHz, respectively. The Antenna *B3* shows -17.78 dB and -29.63 dB in the targeted applications of 3.5 GHz WiMAX and 5.2 GHz WLAN, respectively. The Antenna *B3* exhibits -17.78 dB, -29.63 dB, and -22.73 dB for 3.5 GHz WiMAX, 5.2 GHz WLAN application, and 7.24 GHz, respectively. These values are practical for WiMAX and WLAN operation that can be apply to the future research with other types of novelty fractal antenna design. This antenna also can be applied to the complete communication system.

5. REFERENCES

- [1] J. Abraham, K. Suriyan, "Analysis of Tripleband Single Layer Proximity Fed 2x2 Microstrip Patch Array Antenna", International Journal of Electrical and Computer Engineering Systems, Vol. 13, No. 7, 2022, pp. 99-106.
- [2] Z. Yu, J. Yu J, X. Ran, C. Zhu, "A novel Koch and Sierpinski combined fractal antenna for 2G/3G/4G/5G/WLAN/navigation applications", Microwave and Optical Technology Letters, Vol. 59, 2017, pp. 2147-2155.
- [3] Y. Braham Chaouche, F. Bouttout, I. Messaoudene, L. Pichon, M. Belazzoug, F. Chetouah, "A compact CPW-Fed hexagonal antenna with a new fractal shaped slot for UWB communications", Proceedings of the 13th International Wireless Communications and Mobile Computing Conference, 2017, pp. 836-839.
- [4] R. K. Abdulsattar, T. A. Elwi, Z. A. Abdul Hassain, "A New Microwave Sensor Based on the Moore Fractal Structure to Detect Water Content in Crude Oil", Sensors, Vol. 21, 2021, p. 7143.
- [5] H. S. Ahmed, Z. S. Ahmed, R. S. Zamel, T. A. Elwi, "Compact MIMO Antenna Array for 5G Applications based Novel Hayder-Koch Fractal Geometry", Proceedings of the International Telecommunications Conference, 2022, pp. 1-5.
- [6] N. B. M. Nafis, M. K. A. Rahim, O. Ayop, H. A. Majid, S. Tuntrakool, "Characterization of the Koch fractal embedded hexagonal loop frequency selective surface structure for X-band application", Indone-

- sian Journal of Electrical Engineering and Computer Science, Vol. 20, No. 2, 2020, pp. 878-886.
- [7] J. Anguera A. Andújar, J. Jayasinghe, V. V. S. S. S. Chakravarthy, P. S. R. Chowdary, J. L. Pijoan, T. Ali, C. Cattani, "Fractal Antennas: An Historical Perspective", *Fractal and Fractional*, Vol. 4, No. 1, 2020, pp. 1-26.
- [8] M. K. C. Durbhakula, N. V. K Rao, "Sierpinski Monopole Antenna Reconfigurable System using Hairpin Bandpass Filter Sections", *Proceedings of the IEEE Indian Conference on Antennas and Propagation*, 2018, pp. 1-5.
- [9] S. B. Kumar, P. K. Singhal, "RF energy harvesting using Sierpinski's gasket fractal antenna with EBG geometry", *Journal of Information and Optimization Sciences*, Vol. 41, No. 1, 2020, pp. 99-106.
- [10] T. Mondal, S. Suman, S. Singh, "Novel Design of Fern Fractal Based Triangular Patch Antenna", *Proceedings of the National Conference on Emerging Trends on Sustainable Technology and Engineering Applications*, 2020, pp. 1-3.
- [11] X. Cao, B. Luo, Y. Zhu, Z. Xia, Q. Cai, "Research on the Defected Ground Structure with Von Koch Snowflake Fractals", *IEEE Access*, Vol. 8, 2020, pp. 32404-32411.
- [12] R. Gurjar, D. K. Upadyay, B. Kanaujia, A. Kumar, "A compact U-shaped UWB-MIMO antenna with novel complementary modified Minkowski fractal for isolation enhancement", *Progress in Electromagnetics Research C*, Vol. 107, 2021, pp. 81-96.
- [13] S. Palanisamy, B. Thangaraju, O. I. Y. Khalaf, Alo-taibi, S. Alghamdi, F. Alassery, "A Novel Approach of Design and Analysis of a Hexagonal Fractal Antenna Array (HFAA) for Next-Generation Wireless Communication", *Energies*, Vol. 14, 2021, p. 6204.
- [14] A. A. Rivera, A. Lankford, M. McClinton, S. Torres, "An IFS for the Stretched Level-n Sierpinski Gasket", *The PUMP Journal of Undergraduate Research*, Vol. 5, 2022, pp. 105-121.
- [15] V. A. S. Ponnappalli, P. V. Y. Jayasree, "Fractal Array Antennas and Applications, Emerging Microwave Technologies in Industrial", *Agricultural, Medical and Food Processing*, *InTech Open*, 2018.
- [16] C. Puente-Baliarda, J. Romeu, R. Pous, A. Cardama, "On the behaviour of the Sierpinski multiband fractal antenna", *IEEE Transactions on Antennas and Propagation*, Vol. 46, No. 4, 1998, pp. 517-524.
- [17] A. Kaur, S. Gupta, "A complementary Sierpinski gasket fractal antenna array for wireless MIMO portable devices", *Microwave and Optical Technology Letters*, Vol. 61, No. 2, 2018, pp. 436-442.
- [18] M. H. Ramli, M. Z. A. A. Aziz, M. A. Othman, H. Nornikman, M. S. N. Azizi, S. N. A. Azlan, A. H. Dahan, H. A. Sulaiman, "Design of Sierpinski gasket fractal antenna with slits for multiband application", *Jurnal Teknologi*, Vol. 78, No. 5, 2016, pp. 123-128.
- [19] Y. B. Chaouche, M. Nedil, B. Hammache, M. Bellazzoug, "Design of Modified Sierpinski Gasket Fractal Antenna for Tri-band Applications", *Proceedings of the IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting*, 2019, pp. 889-890.
- [20] Y. B. Chaouche, I. Messaoudene, M. Nedil, F. Bouttout, "CPW-Fed Hexagonal Modified Sierpinski Carpet Fractal Antenna for UWB Applications", *Proceedings of the IEEE International Symposium on Antennas and Propagation & USNC/URSI National Radio Science Meeting*, 2018, pp. 1045-1046.
- [21] A. R. Sankhe, U. Pandit Khot, "Combined Sierpinski Carpet and Koch Fractal Patch Antenna with Fractal DGS for Wireless Applications", *Proceedings of the 2nd International Conference on Electronics, Materials Engineering & Nano-Technology*, 2018, pp. 1-7.
- [22] V. K. Bal, Y. Bhomia, A. Bhardwaj, "Carpet structure of combination of crown square and Sierpinski Gasket fractal antenna using transmission line feed", *Proceedings of the International Conference on Electrical and Computing Technologies and Applications*, 2017, pp. 1-5.
- [23] T. N. Cao, W. J. Krzysztofik, "Design of multiband Sierpinski fractal carpet antenna array for C-band", *Proceedings of the 22nd International Microwave and Radar Conference*, 2018, pp. 41-44.

Multi-Head Attention-Based Spectrum Sensing for Cognitive Radio

Original Scientific Paper

B.V. Ravisankar Devarakonda

ANUCET, ANU
Guntur, India
dbvravisankar@gmail.com

Venkateswararao Nandanavam

Department of ECE, Bapatla Engineering College
Bapatla, India
nvrao68@gmail.com

Abstract – Spectrum sensing is one of the key tasks of cognitive radio to monitor the activity of the primary user. The sensing accuracy of the secondary user is dependent on the signal-to-noise ratio of the primary user signal. A novel Multi-head Attention-based spectrum sensing for Cognitive Radio is proposed through this work to increase the detection probability of the primary user at a low signal-to-noise ratio condition. A radio machine learning dataset with a variety of digital modulation schemes and varying signal-to-noise ratios served as a training source for the proposed model. Further, the performance metrics were evaluated to assess the performance of the proposed model. The experimental results indicate that the proposed model is optimized in terms of the amount of training time required which also has an increase of 27.6% in the probability of detection of the primary user under a low signal-to-noise ratio when compared to other related works that use deep learning.

Keywords: cognitive radio, primary user, spectrum sensing, multi-head attention, additive attention, deep learning

1. INTRODUCTION

Cognitive Radio (CR) [1] is introduced to mitigate the problems of spectrum scarcity and also to provide strategies for efficient communication owing to a huge increase in wireless traffic. Earlier studies on wireless traffic identified the underutilization of the available spectrum by licensed primary users (PUs). The idle time of PU transmission facilitates the unlicensed secondary users (SUs) to dynamically and opportunistically access the spectrum of PU without causing any interference to its transmission [2]. Through spectrum sensing the activity of the PU is monitored continuously by a SU to detect the spectral occupancy of the PU.

The diverse studies published in the area show that the accurate detection of the PU by the SU is highly impacted by the signal-to-noise ratio (SNR), fading, multipath, and shadowing effects [3]. Traditional spectrum sensing algorithms like Energy Detection [4-5], Cyclostationary-based detection [6], and Eigen-value-based detection [7] have been published earlier in the literature. The effects of various types of fading, multipath, and shadowing on spectrum sensing have been inves-

tigated in [8-10]. The capacity of fading channels and the data transmission rate through these channels are extremely important for reliable communication between SU and PU [11]. Reliable, efficient, and secured data transmission over wireless channels requires physical layer security and a controllable wireless propagation environment [12-13]. The requirement of an optimal threshold value for different channel conditions is the main drawback of traditional methods of spectrum sensing.

With the rapid advancement in technology and availability of a huge amount of data, Machine Learning (ML) based spectrum sensing algorithms are being implemented in place of traditional spectrum sensing techniques. In the ML-based approach, spectrum sensing is a binary classification to detect the presence or absence of PU. The accuracy of detection of PU presence or absence is significantly impacted by the range of SNR on which the spectrum sensing is performed. Some of the observations on early implementations of the ML models for spectrum sensing are presented in [14-19].

One of the limitations of the ML-based approach is the extraction of the appropriate features as test statistics for accurate decision making. While it is observed that Deep Learning (DL) based approaches help to overcome this limitation as they automatically learn features from the network. DL-based techniques are further used to detect patterns in various applications of natural language processing (NLP), computer vision, and signal processing-related tasks for wireless communications. In recent times, DL-based architectures have gained popularity in the implementation of spectrum sensing algorithms. The various studies that cite DL architectures employ Convolution Neural Networks (CNNs) for spectrum sensing.

Sandeep Kumar et al. [20] proposed a performance analysis of Cooperative Spectrum Sensing (CSS) over α - η - μ and α - κ - μ fading channels using a clustering-based technique. The interesting findings in this model show the use of an energy feature vector.

Dimpal Janu et al. [21] proposed a novel graph convolution network-based adaptive CSS in a cognitive radio network that handles dynamic channel conditions with multiple antennas experiencing different types of fading.

Dimpal Janu et al. [22] presented the performance comparison of machine learning-based multi-antenna CSS algorithms under a multi-path fading scenario. The CNN-based CSS method adopted in this model has obtained intriguing results.

Chang Liu et al. [23] proposed a Deep CM-CNN for spectrum sensing in cognitive radio that takes the covariance matrix as the input to CNN. The CM-CNN-based test statistics generated in this model achieved a higher detection probability.

Youheng Tan et al. [24] implemented CSS based on CNN. Though the model has resulted in better performance, its architecture is complex.

Surendra Solanki et al. [25] presented deep learning for spectrum sensing in cognitive radio and developed DLsenseNet architecture. The authors have not specified the threshold for detection probability, despite the model's interesting results.

Jiabao Gao et al. [26] proposed DLDetectNet architecture. The model was not able to achieve the desired probability of false alarm for various modulation schemes employed in this work.

Kai Yang et al. [27] proposed a blind spectrum sensing method based on deep learning to handle low SNR scenarios using a one-dimensional CNN and long short-term memory (LSTM). At a low SNR value, this method has a low detection probability percentage.

Jiandong Xie et al. [28] proposed a DL-based spectrum sensing in cognitive radio using the CNN-LSTM approach. The range of SNR on which the probability of detection is performed was not properly specified in this work.

Since spectrum sensing is a signal processing-related application that handles time series data, Neural net-

work architectures that can handle and account for the chronological order of data are needed. DL-based architectures such as Recurrent Neural Networks (RNNs) and LSTM are generally used in these applications. However, the issue of vanishing and exploding gradients plagues RNNs. There is a need for DL based architecture that will be unaffected by these gradients.

Of late Transformer based DL architecture became popular and is used in many NLP applications. An architecture of a similar nature has been designed and deployed for this signal processing related spectrum sensing task. The proposed work is motivated by the self attention approach implemented in [29], which is based on an attention mechanism that primarily attends to those parts of the input that would significantly affect the model's prediction.

This paper proposes Multi-Head attention based Spectrum Sensing (MHASS) for CR. An attention function is applied to the input sequence multiple times in parallel in Multi-Head attention (MHA) based on the number of heads. The output of the multiple attention blocks is concatenated to get the overall attention function. The performance metrics considered in this work are the probability of detection (P_d) which is similar to the true positive rate, the probability of false alarm (Pf) which is similar to the false positive rate, precision (Pr), the area under the curve (AUC) and F1 score (F1). The proposed MHASS model is trained, validated, and tested on the signal sensed by SU over a wide range of SNR from -20 dB to 20 dB.

The main contributions of the proposed work are:

- Use of MHASS for the first time in the literature.
- Reducing the number of computations and the amount of time spent to train the model.
- Best performance metrics obtained at low SNR.

The forthcoming sections in this paper are organized as follows. Section 2 discusses the DL system model, section 3 describes the proposed MHASS model, section 4 presents the experimental setup, section 5 analyzes the results & discussions and conclusions are presented in section 6.

2. DL SYSTEM MODEL

The DL system model for spectrum sensing is depicted in Fig.1. It can be seen that the N observation vectors are utilized for spectrum sensing.

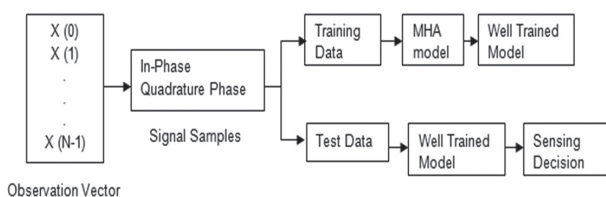


Fig. 1. DL System Model

Consider $X(n) = \{x_1(n), x_2(n), \dots, x_M(n)\}^T$, where M represents the signal's sample length l , $n=0,1,\dots, N-1$,

denotes the n^{th} received observation vector and $x_i(n)$ denotes the n^{th} discrete time sample.

The sensing decision in DL-based spectrum sensing has been formulated as a binary hypothesis as represented by (1).

$$X(n) = \begin{cases} R(n) + W(n) & : H_1 \\ W(n) & : H_0 \end{cases} \quad (1)$$

Here $R(n)$ denotes the PU signal samples vector, which also suffers from path loss and fading. $W(n)$ is the noise samples vector and $X(n)$ is the received SU observation vector. H_1 denotes the PU presence and H_0 denotes the PU absence.

The received observation vector consists of In-Phase (IP) and Quadrature Phase (QP) signal samples. The dataset of IP and QP samples is obtained from a Radio Machine Learning (RML) pickle file 'RML2016.10a_dict.pkl' [30] which is composed of eight digital modulations, multipath loss, Rician, and Rayleigh fading effects. A noise vector having a similar length of signal samples is generated using the Additive White Gaussian Noise (AWGN) scheme.

The real and imaginary parts of the SU observation vector are denoted as X_I and X_Q respectively. The received complex signal with both real and imaginary parts is given by (2).

$$\hat{X} = (X_I, X_Q) \quad (2)$$

The energy of this received complex signal is calculated as given by (3).

$$E = \sum_{i=1}^M (|\hat{X}_i|)^2 \quad (3)$$

To scale all the amplitude values of the signal samples to a similar magnitude, energy normalization is performed as given by (4).

$$\hat{X}_{norm} = \frac{\hat{X}}{E} \quad (4)$$

In the training phase, the proposed MHA model is trained on annotated data of the energy-normalized signal samples and noise samples such that the occurrence of signal samples can be considered as PU presence (Label=1) and the occurrence of only noise samples can be considered as PU absence (Label=0). The implementation details of the MHA model are discussed in section 3.

3. PROPOSED MHA MODEL

In the proposed model for spectrum sensing, MHA has been used on the input layer consisting of IP and QP samples along with the noise samples. It has been observed that MHA attends to only those parts of the input tensor which are used to compute the decision of PU presence or absence. The use of MHA facilitated the subsequent convolution layer to have pre-computed attention over the input convolution volume. As a result of this, the network converged in less number of epochs. The main operation involved in MHA was the

Scaled Dot Product Attention followed by concatenating the attention functions obtained from multiple attention blocks.

3.1. SCALED DOT PRODUCT ATTENTION

An attention function can be described as mapping a query (Q) and a set of key-value pairs K and V respectively to an output. Q , K , and V are matrices representing the input sequence. The output is computed as a weighted sum of the values, where the weight assigned to each value is computed by a compatibility function of the query with the corresponding key.

Query: The query Q is a feature vector of dimension $m \times n$ that describes the aspects being looked for in the input sequence.

Key: For each input element, a key K of dimension $m \times n$ is associated which is also a feature vector, that can identify the elements for which attention has to be paid based on the query.

Value: For each input element, value vector V of dimension $m \times n$, the output can be computed as a weighted sum of the values. Each value can be accessed using the key K .

The scaled dot product attention is shown in Fig.2.

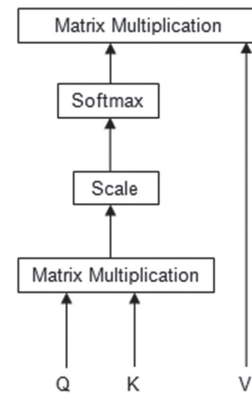


Fig. 2. Scaled dot product attention

Table 1. Parameters/ Hyperparameters of the MHA model

Parameters / Hyper-parameters	Description	Value
m	Number of tokens in the input sequence	2 (I, Q)
n	Dimensionality of hidden/ embedding layer	128
dk	Dimension of Q,K and V	128
Q	Query	2x128 matrix
K	Key	2x128 matrix
V	Value	2x128 matrix
dmodel	Model dimension	128
h	Number of attention heads	16
wq	Query Transformation Matrix	128x128
wk	Key Transformation Matrix	128x128
wv	Value Transformation Matrix	128x128

The attention function performed on three matrices Q , K , and V of order $m \times n$ is given by (5).

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right) \cdot V \quad (5)$$

In the proposed model, instead of the original input sequence, a parameterized form of attention has been used as given in (6).

$$\text{Attention}(w_q Q, w_k K, w_v V) = \text{softmax}\left(\frac{w_q Q (w_k K)^T}{\sqrt{d_k}}\right) \cdot w_v V \quad (6)$$

The steps involved in scaled dot product attention are:

- Initialize the three matrices Q , K , and V .

$$Q = \begin{bmatrix} I_p \\ Q_p \end{bmatrix}_{2 \times 128} \quad K = \begin{bmatrix} I_p \\ Q_p \end{bmatrix}_{2 \times 128} \quad V = \begin{bmatrix} I_p \\ Q_p \end{bmatrix}_{2 \times 128}$$

- Compute the weight matrix by performing the dot product $Q \cdot K^T$.

$$\begin{bmatrix} I_p \\ Q_p \end{bmatrix}_{2 \times 128} \cdot [I_p Q_p]_{128 \times 2} = \begin{bmatrix} w_1 & w_2 \\ w_3 & w_4 \end{bmatrix}_{2 \times 2}$$

- Apply the softmax on the scaled weight matrix and multiply it with V to calculate the attention function as (7).

$$\text{softmax}\left(\begin{bmatrix} \frac{w_1}{\sqrt{128}} & \frac{w_2}{\sqrt{128}} \\ \frac{w_3}{\sqrt{128}} & \frac{w_4}{\sqrt{128}} \end{bmatrix}_{2 \times 2}\right) \begin{bmatrix} I_p \\ Q_p \end{bmatrix}_{2 \times 128} = \text{Attention}(Q, K, V) \quad (7)$$

3.2. MHA MECHANISM

In the MHA mechanism, instead of a single attention function, linearly project Q , K , and V to a lower dimension and perform attention in parallel on the projected versions of Q , K , and V . Concatenate the output from h individual attention functions and perform final linear projection.

Fig. 3 illustrates the block diagram representation of the MHA mechanism with Attn-1, Attn-2, ..., and Attn-h representing the attention functions of h individual heads respectively.

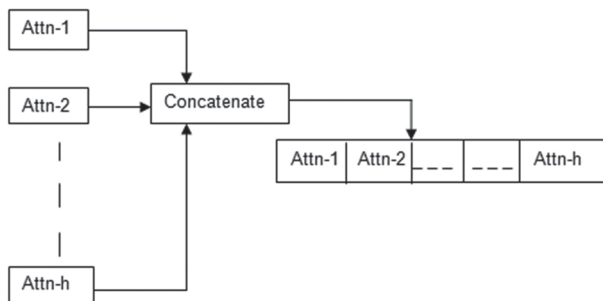


Fig. 3. MHA mechanism

The steps involved in the MHA mechanism are:

- Select the value of h as a factor of model dimension. The value of h in the proposed model is 16.
- Calculate the value of $dk = d_{model} / h$.

- Perform the linear projection of Q , K , and V each of dimension $2 \times dk$, h number of times.
- Calculate the individual attention functions Attn-1, Attn-2, ..., Attn-h.
- Compute the overall attention function by concatenating the individual attention functions.
- The concept of MHA for $h=2$ is illustrated as follows:
- Calculate $dk = d_{model} / h = 128 / 2 = 64$
- Let Attn-1 and Attn-2 will be the individual attention functions and they are computed as (8).

$$\text{Attn} - 1 = \text{Attn} - 2 = \text{softmax}\left(\frac{[Q]_{2 \times 64} \cdot [K^T]_{2 \times 64}}{\sqrt{64}}\right) \quad (8)$$

- Concatenate Attn-1 and Attn-2 to compute the overall attention function.

3.3. ARCHITECTURE OF MHA

The architecture of MHA used for the experimental evaluation is shown in Fig. 4.

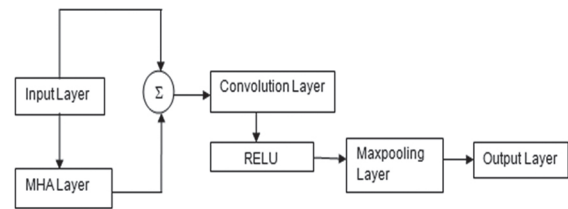


Fig. 4. Architecture of MHA

A 2×128 length sequence of the RML dataset serves as a training source for the input layer. Multi-Head attention is applied on the input layer followed by additive attention of the MHA layer with the input layer. The MHA attention input is then fed to the convolution layer, where the entire network will be trained end to end and validated on the dataset to build a well trained model as shown in the DL system model presented in section 2. The well trained model is tested on unseen samples of the dataset to predict the sensing decision of PU presence or absence at the output layer. The hyperparameters of the convolution layer are listed in Table 2.

Table 2. Hyperparameters of the convolution layer

Hyperparameters	Value
Number of filters in the convolution layer	64
Kernel size of convolution layer	5x5
Dropout ratio	0.25
Max Pooling kernel size	2x2
Batch Size	256
Optimizer	Adam

4. EXPERIMENTAL SETUP

The specifications of the dataset and the performance metrics used are discussed in this section. The proposed model makes use of the RML dataset, the parameters of which are tabulated in Table 3.

Table 3. RML Dataset Parameters

Parameters	Value
Modulation scheme	8PSK,BPSK, CPFSK, GFSK,PAM4,QAM16, QAM64, QPSK
Fading effects of the channel	Rician, Rayleigh
Sample Length	128
SNR Range	-20 dB to 20 dB in 2 dB increments
Training Samples	112000
Validation Samples	32000
Test Samples	16000

4.1. PERFORMANCE METRICS

The proposed model is trained and validated on the RML dataset, and the performance metrics like P_d , P_f , AUC, and $F1$ are observed to evaluate its performance. P_d denotes the probability of PU presence when PU occupies the spectrum, and P_f denotes the probability of PU presence when PU is not utilizing the spectrum. AUC is the overall area occupied by the receiver operating characteristics (ROC) and the quality of the model is indicated by $F1$ which is dependent on precision and recall.

4.2. TEST STATISTIC

The threshold (TH) value for various values of probability of false alarm (PFA) is calculated and it is compared with the test statistic T as in (9).

$$T = \frac{P(H_1)}{P(H_0)} \quad (9)$$

Where $P(H_1)$ and $P(H_0)$ represent the probability of PU presence and absence respectively. The proposed model predicts the likelihood of PU presence or absence using the two approaches listed below.

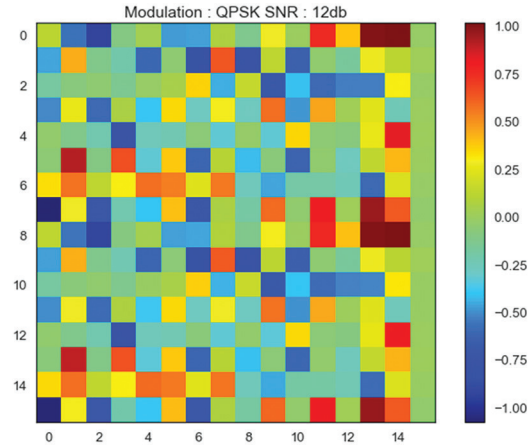
- A standard threshold value of 0.5 is used in all binary classification ML/DL algorithms.
- $T > TH$ denotes the PU presence and $T < TH$ denotes the PU absence.

5. RESULTS AND DISCUSSIONS

The proposed model's novelty and its effectiveness are discussed in this section. The results of the validation and test evaluation metrics are used to assess the performance of the proposed MHASS model. The additive attention signal pattern of the MHA layer is discussed in section 5.1. In section 5.2 the proposed model's performance is compared to that of a DL CNN model. The ROC of the proposed model is shown in section 5.3. Analysis of the results and improvement in the performance over the previous work is discussed in sections 5.4 and 5.5 respectively.

5.1. ADDITIVE ATTENTION SIGNAL PATTERN

The pattern of the input sequence that is added to the attention weights which are obtained from the MHA layer for the various modulated signals at various SNR values is considered as an additive attention signal pattern. The novel nature of the proposed model is evident from this pattern. An additive attention signal pattern for QPSK modulation at SNR= 12 dB is depicted in Fig. 5.

**Fig. 5.** Additive Attention Signal Pattern

In Fig. 5 the additive attention signal pattern is represented as a 16x16 image which is symmetric for both the IP and QP samples. Different colors in the image represent the signal intensities at each location of the image. The range of signal intensities represented by the vertical colour bar indicates the parts of the input sequence that should be considered for improved prediction.

The following inferences can be drawn from the additive attention signal pattern:

- A large part of the input has non-zero attention weights, which indicates that different parts of the input sequence are attended to increase the likelihood of PU presence.
- The peak levels of the signal at specific locations of the input have a very high magnitude, indicating the presence of PU.

5.2. PERFORMANCE COMPARISON

A DL CNN model without MHA is contrasted with the MHASS model's performance. Both models were compared for training, validation loss, and detection probability at low SNR.

The training and validation loss with respect to the number of epochs is shown in Fig. 6. MHA_TRAIN and MHA_VAL represent the training and validation loss of the MHASS model respectively. CNN_TRAIN and CNN_LOSS denote the training and validation loss of the CNN model respectively.

Fig. 6. depicts the fast convergence of the proposed MHASS model with optimized training time in comparison to CNN.

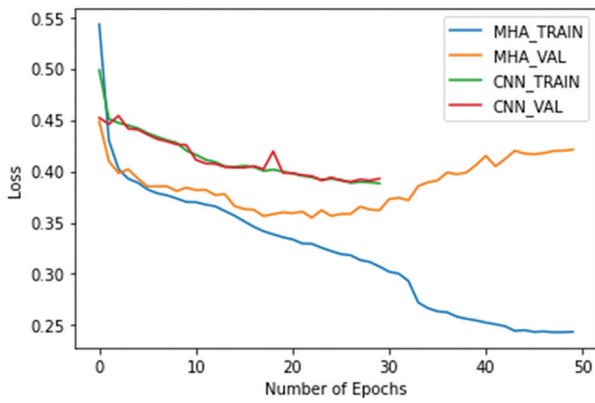


Fig .6. Loss VS Epochs for MHA, CNN

The above figure depicts the fast convergence of the proposed MHA model with optimized training time in comparison to CNN.

The plot of SNR and percentage of detection probability (P_d %) of the MHA and CNN models is shown in Fig.7.

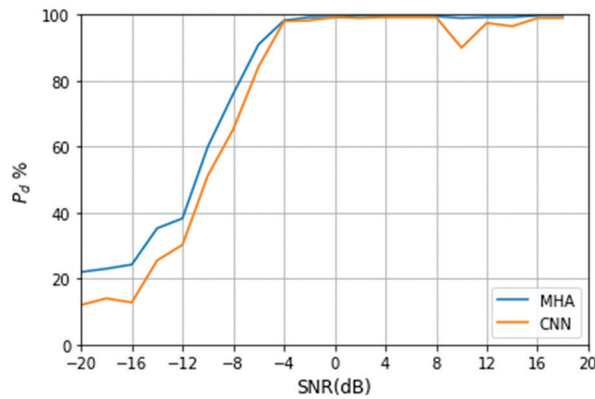


Fig. 7. SNR VS P_d

Fig.7 is an indication of the increased probability of detection of the proposed model when compared to CNN at low SNR.

5.3. ROC OF THE PROPOSED MHA MODEL

The ROC of the MHA model is given in Fig. 8.

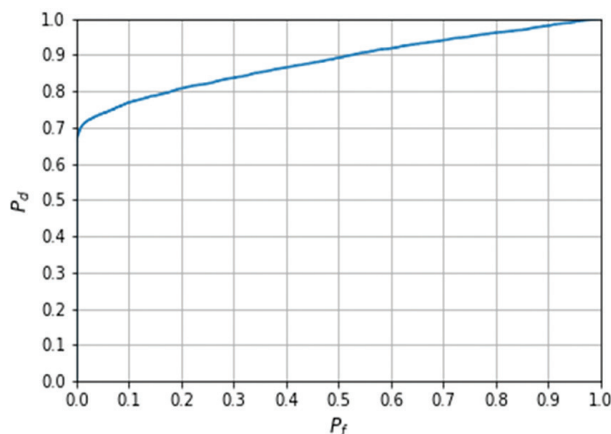


Fig. 8. ROC of MHA model

The ROC of the MHA model for SNR= - 20 dB and SNR = - 6 dB is shown in Fig. 9.

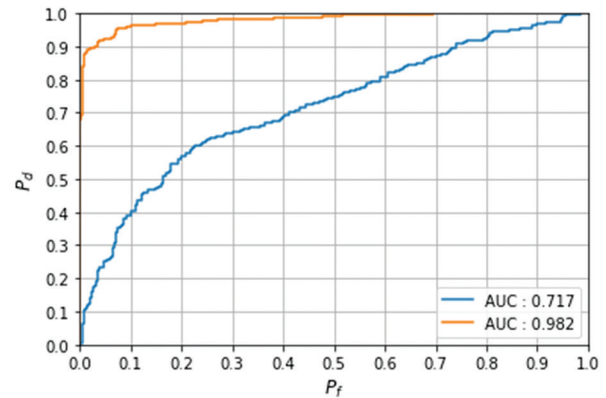


Fig. 9. ROC for SNR = -6 dB and SNR = -20 dB

Table 4 displays the proposed model's test evaluation metrics for different SNR values.

Table 4. Performance Metrics for Different SNR

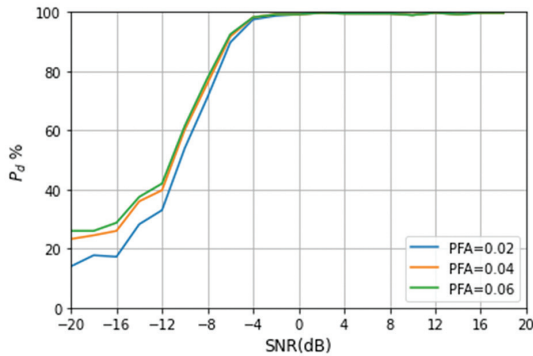
SNR(dB)	P_d	P_r	AUC	F1
-20	0.22	0.863	0.717	0.351
-18	0.23	0.868	0.744	0.364
-16	0.243	0.874	0.745	0.38
-14	0.353	0.91	0.795	0.51
-12	0.383	0.916	0.814	0.54
-10	0.6	0.945	0.89	0.734
-8	0.76	0.956	0.937	0.847
-6	0.91	0.963	0.982	0.936
-4	0.9823	0.966	0.995	0.974
-2	0.993	0.966	0.999	0.979
0	0.993	0.966	0.999	0.979
2	0.998	0.966	0.999	0.981
4	0.995	0.966	0.999	0.980
6	0.995	0.966	0.998	0.980
8	0.995	0.966	0.997	0.980
10	0.99	0.966	0.998	0.978
12	0.998	0.966	0.999	0.982
14	0.993	0.966	0.998	0.98
16	0.998	0.966	0.999	0.982
18	0.998	0.966	0.999	0.982

The impact of modulation schemes on percentages of detection probability (P_d %) and false alarm probability. (P_f %) at SNR = - 20 dB is displayed in Table. 5.

The P_d % of the proposed model to a varying range of SNR for different values of PFA based on the second test statistic mentioned in section 4.2 is shown in Fig. 10.

Table 5. Impact of modulation schemes

Modulation Scheme	Pd (%)	Pf (%)
8PSK	19	0
BPSK	23	0
CPFSK	18	0
GPSK	14	0
PAM4	24	0
QAM16	23	0
QAM64	28	0
QPSK	25	0
All modulations	22.5	4

**Fig.10.** SNR VS Pd for different values of PFA

5. 4. ANALYSIS OF THE RESULTS

The following observations can be made after analyzing the results :

- From Table 4 the values obtained for Pd, AUC, and F1 are 0.91, 0.982, and 0.936 respectively at SNR= -6 dB. These values indicate that the proposed model has a high detection probability at a low SNR.
- Fig. 9 denotes the best AUC values obtained for low SNR of -20 dB and -6 dB respectively.
- Table 5 indicates that the best values of Pd are obtained with zero percentage of Pf for various modulation schemes at a low SNR = -20 dB.
- Fig.10 shows that for various values of PFA best performance metrics are obtained for the proposed model at a low SNR.

5. 5. IMPROVEMENT IN PERFORMANCE

The proposed MHASS model's performance in comparison to prior work is shown in Tables 6 and 7.

Table 6. Comparison of the P_d and P_f at SNR=-20 dB

Model	Modulation Scheme	P_d (%)	P_f (%)
Gao et al. [26]	QPSK	<20	7.81
	QAM16	<20	6.54
	QAM64	<20	7.82
Proposed MHASS	QPSK	25	0
	QAM16	23	0
	QAM64	28	0

Table 7. Improvement in Pd % of the proposed model

SNR (dB)	Model	P_d (%)	% Improvement in P_d
-6	Kai Yang et al. [27]	80	13.8
	Proposed MHASS	91	
-12	Kai Yang et al. [27]	30	27.6
	Proposed MHASS	38.3	

From Table 6, it is observed that various modulation schemes used in the proposed model resulted in increased Pd % with 0% Pf when compared to the previously reported model.

Table 7 indicates that the proposed model has resulted in an improvement of 13.8 % and 27.6 % in the value of Pd for SNR values of -6 dB and -20 dB respectively when compared to the previously developed model.

6. CONCLUSION

Multi-Head attention based spectrum sensing for cognitive radio has been implemented in this work. The implemented model resulted in the best performance metrics such as Pd, Pf, AUC, and F1 over a wide range of SNR. The ROC and other plots obtained indicate that a higher value of detection probability is achieved at a low SNR for various modulations of the dataset used in this model. The use of multi-head attention has resulted in faster convergence of the proposed model with less number of computations. There is an improvement of 27.6 % in Pd (%) when compared to one of the previous works in deep learning. This work can be further extended by proposing a cooperative spectrum sensing scheme in which the secondary users are experiencing different levels of fading and other multipath effects.

7. REFERENCES

- [1] J. Mitola, G. Q. Maguire, "Cognitive radio: making software radios more personal", IEEE Personal Communications, Vol. 6, No. 4, 1999, pp. 13-18.
- [2] S. Haykin, "Cognitive Radio: Brain-empowered wireless communications", IEEE Journal on Selected Areas in Communications, Vol. 23, No. 2, 2005, pp. 201-220.
- [3] K. Kockaya, I. Develi, "Spectrum sensing in cognitive radio networks: threshold optimization and analysis", EURASIP Journal on Wireless Communications and Networking, Vol. 255, 2020.
- [4] J. Luo, G. Zhang, C. Yan, " An Energy Detection-Based Spectrum Sensing Method for Cognitive Radio", Wireless Communications and Mobile Computing, Vol. 2022, 2022, pp. 1-10.

- [5] S. Srinu, S. L. Sabat, "Effective cooperative wide-band sensing using energy detection under suspicious Cognitive Radio Network", *Computers & Electrical Engineering*, Vol. 39, No. 4, 2013, pp. 1153-1163.
- [6] G. R. George, S. C. Prema, "Cyclostationary Feature Detection Based Blind Approach for Spectrum Sensing and Classification", *Radioengineering*, Vol. 28, No. 1, 2019, pp. 298-303.
- [7] Y. Zeng, Y. C. Liang, "Eigenvalue-based Spectrum Sensing Algorithms for Cognitive Radio", *IEEE Transactions on Communications*, Vol. 57, No. 6, 2009, pp. 1784-1793.
- [8] S. Kumar, "Energy Detection in Hoyt/Gamma Fading Channel with Micro-Diversity Reception", *Wireless Personal Communications*, Vol. 101, No. 2, 2018, pp. 723-734.
- [9] S. Kumar, P. K. Verma, M. Kaur, P. Jain, S. K. Soni, "On the spectrum sensing of gamma shadowed Hoyt fading channel with MRC reception", *Journal of Electromagnetic Waves and Applications*, Vol. 32, No. 16, 2018, pp. 2157-2166.
- [10] S. Kumar, "Performance of ED Based Spectrum Sensing Over α - η - μ Fading Channel", *Wireless Personal Communications*, Vol. 100, No. 4, 2018, pp. 1845-1857.
- [11] P. Yadav, S. Kumar, R. Kumar, "A Review of Transmission Rate over Wireless Fading Channels: Classifications, Applications, and Challenges", *Wireless Personal Communications*, Vol. 122, No. 2, 2022, pp. 1709-1765.
- [12] P. Yadav, S. Kumar, R. Kumar, "A comprehensive survey of physical layer security over fading channels: Classifications, applications, and challenges", *Transactions on Emerging Telecommunications Technologies*, Vol. 32, No. 9, 2021, pp. 1-41.
- [13] S. Kumar, P. Yadav, M. Kaur, R. Kumar, "A survey on IRS NOMA integrated communication networks", *Telecommunication Systems*, Vol. 80, No. 2, 2022, pp. 277-302.
- [14] C. Clancy, J. Hecker, E. Stuntebeck, T. O'Shea, "Applications of Machine Learning to Cognitive Radio Networks", *IEEE Wireless Communications*, Vol. 14, No. 4, 2007, pp. 47-52.
- [15] D. Janu, K. Singh, S. Kumar, "Machine learning for cooperative spectrum sensing and sharing: A survey", *Transactions on Emerging Telecommunications Technologies*, Vol. 33, No. 1, 2022, pp. 1-28.
- [16] R. Sarikhani, F. Keynia, "Cooperative Spectrum Sensing Meets Machine Learning: Deep Reinforcement Learning Approach", *IEEE Communications Letters*, Vol. 24, No. 7, 2020, pp. 1459-1462.
- [17] N. Abbas, Y. Nasser, K. El Ahmad, "Recent advances on artificial intelligence and learning techniques in cognitive radio networks", *EURASIP Journal on Wireless Communications and Networking*, 2015, pp. 1-20.
- [18] C. H. A. Tavares, J. C. Marinello, M. L. Proenca Jr., T. Abrao, "Machine learning-based models for spectrum sensing in cooperative radio networks", *IET Communications*, Vol. 14, No. 18, 2020, pp. 3102-3109.
- [19] D. B. V. Ravisankar, N. Venkateswararao, "Ensemble Classifier with Heterogenous Fusion Center for Cooperative Spectrum Sensing in Cognitive Radio", *Journal of Interconnection Networks*, Vol. 22, No. Supp01, 2022, pp. 1-20.
- [20] S. Kumar, P. S. Chauhan, R. Bansal, M. Kaur, R. K. Yadav, "Performance Analysis of CSS Over α - η - μ and α - k - μ Fading Channel Using Clustering-Based Technique", *Wireless Personal Communications*, Vol. 126, No. 4, 2022, pp. 3595-3610.
- [21] D. Janu, S. Kumar, K. Singh, "A Graph Convolution Network Based Adaptive Cooperative Spectrum Sensing in Cognitive Radio Network", *IEEE Transactions on Vehicular Technology*, Vol. 72, 2022, pp. 1-11.
- [22] D. Janu, K. Singh, S. Kumar, "Performance Comparison of Machine Learning based Multi-Antenna Cooperative Spectrum Sensing algorithms under Multi-Path Fading Scenario", *Proceedings of the IEEE 4th International Conference on Cybernetics, Cognition and Machine Learning Applications*, Goa, India, 8-9 October 2022.
- [23] C. Liu, J. Wang, X. Liu, Y.-C. Liang, "Deep CM-CNN for Spectrum Sensing in Cognitive Radio", *IEEE Journal on Selected Areas in Communications*, Vol. 37, No. 10, 2019, pp. 2306-2321.

- [24] Y. Tan, X. Jing, "Cooperative Spectrum Sensing Based on Convolutional Neural Networks", *Applied Sciences*, Vol. 11, No. 10, 2021, pp.1-13.
- [25] S. Solanki, V. Dehalwar, J. Choudhary, "Deep Learning for Spectrum Sensing in Cognitive Radio", *Symmetry*, Vol. 13, No. 1, 2021.
- [26] J. Gao, X. Yi, C. Zhong, X. Chen, Z. Zhang, "Deep Learning for Spectrum Sensing", *IEEE Wireless Communications Letters*, Vol. 8, No. 6, 2019, pp. 1727-1730.
- [27] K. Yang, Z. Huang, X. Wang, X. Li, "A Blind Spectrum Sensing Method Based on Deep Learning", *Sensors*, Vol. 19, No. 10, 2019, pp. 1-17.
- [28] J. Xie, J. Fang, C. Liu, X. Li, "Deep Learning-Based Spectrum Sensing in Cognitive Radio: A CNN-LSTM Approach", *IEEE Communications Letters*, Vol. 24, No. 10, 2020, pp. 2196-2200.
- [29] A. Vaswani et al. "Attention Is All You Need", *Proceedings of 31st International Conference on Neural Information Processing Systems*, Long Beach, CA, USA, 4-9 December 2017, pp.6000-6010.
- [30] T. O'Shea, N. West, "Radio Machine learning Dataset Generation with GNU Radio", *Proceedings of the 6th GNU Radio Conference*, University of Colorado, Boulder, USA, 12-16 September 2016.

Comparative Study and Performance Analysis of MANET Routing Protocol

Original Scientific Paper

Chetana Hemant Nemade*

School of Computer Engineering and Technology,
Dr. Vishwanath Karad MIT World Peace University,
*School of Computer Engineering and Technology,
MIT AOE, Alandi (D).
chnemade@mitaoe.ac.in

Uma Pujeri

School of Computer Engineering and Technology,
Dr. Vishwanath Karad MIT World Peace University,
uma.pujeri@mitwpu.edu.in

Abstract – MANET (Mobile ad hoc networks) are famous in research due to their ad hoc nature and effectiveness during calamities across continents when no framework support is free. Wireless network interfaces have a limited transmission range; nodes may require multiple network hops to trade information across the organization. Each versatile node functions like a switch in such an organization, sending details to the other portable connected nodes. The nodes should not interrupt communication and associate themselves with the correct information transfer. Another significant issue was the development of expandable route discoveries capable of assessing rapid topography variations and numerous network detachments caused by high vehicle quality. This research article describes extensive technological changes, including the components and flaws of current progressive routing algorithms. Routing protocols designed for wired networks, such as the distance vector or connection state conventions, are inadequate for this application because they assume fixed geography and high overheads. This research article includes the MANET-supported routing protocols and their performance analysis across various performance parameters such as packet delivery ratio, average throughput, residual energy, and delay.

Keywords: AODV, AOMDV, DSR, DSDV, GSR, MANET, Network simulator

1. INTRODUCTION

A private organization is a data communication network that uses wireless connections to connect devices for information exchange [1] [2]. Remote organization innovation eliminates the costly strategy of introducing links for information association between gadgets in various areas.

Cell networks are an illustration of this sort of organization. The last option is ordinarily alluded to as "impromptu organizations." Stations in such an organization are fit for making and trading data in a multi-jump organization. Because of the highly dynamic nature of a mobile ad hoc network, network topology changes frequently and unexpectedly, adding difficulty and complexity to routing among mobile nodes. The directing region is the most dynamic examination region inside the Mobile ad hoc networks space because of the problems and intricacies, as well as the fundamental significance of the steering convention in laying out correspondences among portable hubs [3].

Mobile ad hoc networks can work independently or as part of a more extensive network. A highly dynamic independent topology has one or more different transceivers between nodes. MANET's biggest challenge is equipping each device with the knowledge to route traffic correctly. Starting with environmental sensors can be used for road safety.

A mobile peer-to-peer network can exchange data between disparate compact machines after providing a central machine. In that regard, it is not an entry point into the versatile peer-to-peer network. This act is the miracle of the spontaneous cellular network with which the mobile node communicates. Since there is no central or fixed infrastructure, the MANET (Mobile ad hoc network) attribute is an entirely different alternative node. It is completely different from all alternative networks.

In measuring Mobile ad hoc networks moving nodes, such as unfixed to occupy and enter into an effective net, each unique node becomes a part of the network any-

time and anywhere, just as each node leaves the network. MANET offers numerous excellent options in terms of topology flexibility, reliability, rapid configuration, intrinsic quality assistance, superpower geography, adaptation to internal failure, self-mending, and free-gathering framework. It has led to many visions based on functions [4][5].

We provide a comprehensive overview of the Mobile ad hoc network routing protocol and discuss which routing protocol is most efficient for emergency transformation. The second section compares survey documents related to Mobile ad hoc network routing protocols. Section 3 will introduce the different mobile ad hoc networks, followed by a discussion of the protocol's operating environment as simulated by NS2. Sections 4 and 5 show the performance analysis of routing protocols using parameters such as packet transfer rate, residual performance, and average throughput. Finally, this concludes the paper.

2. LITERATURE SURVEY

Many authors have previously worked on deep performance analysis of standard models of AODV (Ad Hoc On-Demand Distance Vector) and DSDV routing protocols for various chain frameworks [6] [7]. Later, they conducted several experiments on AODV and DSDV routing protocols by sterilizing core attributes of the protocol parameters and comparing their performances with standard models to realize performance progression [8].

There was an In-depth simulation study of more and more network nodes in a specific cellular network and the performance of DSDV (Destination Sequenced Distance Vector). The network uses a CBR (constant bit rate) flow mode of very different quality. It calculates the scientific score and delay of the generated data packets, attempting to assess the exhibition of the DSDV steering convention [9].

Considering AODV, DSR (Dynamic Source Routing protocol), and other routing protocols, the joint node density, packet length, and quality in ad-hoc cellular networks are studied to compare performance. Based on the analysis, they mainly looked at some protocol parameters for the simulation. Mobility has a significant impact on basic routing protocols [10].

In [11] [12] author used NS3 reenactment to think about and dissect the exhibition of AODV (Ad Hoc On-Demand Distance Vector), DSDV (Destination Sequenced Distance Vector), DSR (Dynamic Source Routing protocol), and more directing conventions. The recreation utilizes the customary models of these steering conventions for various organization hub sets [13].

Exhibition investigation of the all-inclusive DSDV (Destination Sequenced Distance Vector) convention for effective steering in arbitrary remote organizations, and the multicast parameters, mainly based on the

DSDV routing protocol, are introduced to increase energy savings in random networks [14].

Mobile ad hoc Network routing protocols studied for a real-world simulation scenario. In this scenario, they provide a Gauss-Markov movement model with a constant rate; that is, they send data packets at the same rate, change the three different outlines and measure the exhibition of various Mobile ad hoc Network routing protocols [15].

In [16], the authors showed a comparative study of passive and active routing protocols with different parameters (such as Packet Delivery Ratio and end-to-end delay); the author also mentions various attributes of routing protocols.

In [17], authors demonstrated research on proactive reagents combined with geographic routing protocols that are very suitable for sensor networks in which information aggregation effectively minimizes redundancy by eliminating the redundancy between data packets from multiple sources downstream transmission.

This article analyzes the updated MANET (Mobile ad hoc Network) standard routing protocol attribute model in mobile ad hoc networks. It provides a comparative analysis of geographic routing protocols, on-demand routing protocols, and table-based routing, as well as performance indicators, packet delivery ratio, residual energy, and throughput.

3. MANET ROUTING PROTOCOL

The routing protocol in mobile ad hoc networks can divide into a flat, hybrid, and hierarchical routing protocol. For example, active, reactive, and geographic routing protocols come under flat routing[18].ZRP under hybrid and zone-based, and cluster-based under hierarchical routing protocol.

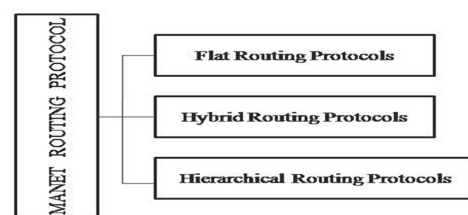


Fig.1. MANET Routing Protocol

3.1 FLAT ROUTING PROTOCOL

The geographic rules of hybrid management depend on the current geographic environment of the organization and correspond to the powerful ideas of mobile ad hoc networks. When we use contrast and geographic location-based control protocols (location-based control protocols), geographic-based control protocols have limited enforcement power. When interaction is required, these protocols use other data to determine geographic-based routing plans. Discretion of the hub

address on the hub geographical control can be further subdivided into active (table) protocols, operational (on-demand) control protocols, and hybrid control protocols [19][20].

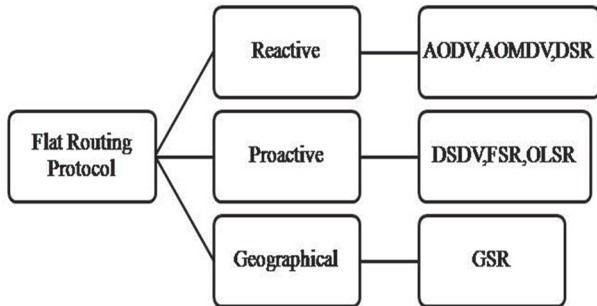


Fig. 2. Classification of Flat Routing Protocol

3.1.1 Proactive Routing Protocol

Another term for the present convention is table-based addressing. The current routine uses at least one address table to put existing address knowledge. Each center can generate data on changes in geography. This convention supports all legal procedures. It is time to move the package. Due to the update of the address table, all other centers' courses are always available when they are suitable. It uses a unique technique to update the address table. Examples of this convention include DSDV (Distance Sequencing Distance-Vector), OLSR (Optimized Link State Routing Protocol), and FSR (Fisheye State Routing).

3.1.1.1 Distance Sequenced Distance-Vector (DSDV)

The agreement builds on the "Bellman-Ford Act" with a few innovations, like the general aviation course on display, which offers a wise and trustworthy method to receive the most recent information. The first is referred to as a "full dump" and contains all the data required to update the table, whereas the second is a "fixed package" that only contains the most recent changes made to the data in the last complete dump. As a result, an incremental bulk transfer is quicker than dumping everything at once [21].

3.1.1.2 Fisheye State Routing (FSR)

FSR (Fisheye State routing) is a table-based strategy based on "calculating link state." It reduces the organization's total movement and also manages topography adjustment data. In FSR, a unique hub has modified data for maintaining tables. It is also very versatile for organizing large areas, but adaptability will reduce accuracy. The main problem with this strategy is the constant sending of interface updates, which can overwhelm organizations and air traffic [22].

3.1.1.3 Optimized Link State Routing Protocol (OLSR)

Proactive is the optimized link state routing protocol. The contact status protocol, which alerts the other net-

work centers of any geographic changes, is the foundation of this protocol. A multipoint repeater lessens the administrative load of employing a multipoint repeater by reducing duplicate message retransmission when messages are transmitted. Various junctions designate nearby hubs to receive information in an MPR (multipoint repeater). The number of retransmissions can be reduced by using any additional MPR-compatible hub to decrypt, measure, and send data packets but not retransmit them [23].

3.1.2 Reactive Routing Protocol

Another term for the present convention is the request address protocol. Adaptive conventions are the way to find courses. The focus of the agreement goes on reducing the organization's movement weight. This convention cannot keep the console aligned. According to geographic location, if the user needs to send information to the center, he can complete it as required. First, send a message to understand the purpose of this course. Till it processes, the found route will be the target center. The agreement also specifies the booking rate. It will reduce the company's traffic compared to the old routing protocol. The most common on-demand control protocols are DSR, AODV, and AOMDV [24] [25].

3.1.2.1 Dynamic Source Routing (DSR)

The dynamic source address uses the source address to send the message. Dynamic source routing allows a sender to specify the broad characteristics of a hub from which data send to the destination. This hub also connects the data of this process to the header of the forwarded data packet. Start at one center, then go on to the following center. Maintenance and exposure to the course are the two main components of this agreement. Looking for courses leading to the goal in the outreach process, of course, to provide support at every point of geographic change, he will see the setbacks that lead to the goal. Any time it is indicated that the original course is lost, the course will be resent. The main advantage of this method is that the center that finds the course will check it out after some time. When storing the rate and including the effective rate, the transmitter can search for it without searching, which is helpful for organizations with little versatility.

3.1.2.2 Ad-hoc On-Demand Distance Vector (AODV)

Any distance vector on-demand is a combo of dynamic source routing and distance sequencing distance vector, which can provide acyclic courses. The primary distinction between dynamic source routing and ad hoc on-demand distance vector is that a unique hub accepts complete network addressing instructions in dynamic source routing. The hub has only one target location in an ad hoc on-demand distance vector. It has been moving forward, so it will naturally react. Ad hoc on-demand distance vectors also added the direction of objective consistency to clarify this idea when geography changes during the broadcast.

3.1.2.3 An Optimized Ad-hoc On-demand Multipath Distance Vector (AOMDV)

An optimized ad hoc on-demand multipath distance vector stretches out the unmistakable ad hoc on-demand distance vector to find different connections in disjointed ways between the source and the objective in each course revelation. It extensively uses the directing input effectively accessible in the ad hoc on-demand distance vector convention. In addition, it utilizes AODV control bundles and a couple of additional fields in the parcel header, for example, publicized jump check, what is more, course list, which contains numerous ways. The primary issue, designated "course cutoff" in AOMDV, is that when there is at least one typical moderate center in a couple of disjointed ways, it cannot find both opposite ways. Therefore, it is vital to discover the current connection in disjointed ways [26].

3.1.3 Geographical Routing Protocols

Geological steering uses data from an area to plan and advance the looking course toward the objective [27] [28]. There is additionally a higher opportunity for enormous multi-jump remote organizations' geography to change as often as possible. Topographical steering needs the proliferation of single-jump geography data as the ideal neighbor to choose precisely on sending. The way it restricts its methodology diminishes the necessity of keeping up the directing tables, diminishes the control center, and dispenses with affecting requirement overloading [29]. The hub continues to send information parcels inside the stamped sending area. The source or moderate hubs can characterize this checked locale to avoid hubs that may hasten a diversion for sending the information bundle. The following property related to geological steering is position-based directing. A hub has to experience the location where its neighbor is present. The part related to this case is the exciting instrument whereby every center advances a group to an adjoining center. Since flooding for center disclosure and state expansion limit within a single leap, position-based control methods prepare to reduce overhead and energy. The organization thickness, the precise confinement of hubs, and the sending rule are the central considerations for the proficiency of the plan [30].

3.1.3.1 Geography Source Routing (GSR)

In GSR, the source hub registers the briefest path to the objective, utilizing Dijkstra's calculation dependent on distance measurements. It registers the separation from the source to moderated hubs through which information is to be sent [31]. The source hub questions the area and floods the parcel to the hubs, which squanders data transfer capacity.

Routing with Geographical Awareness It utilizes the GSR bundle-sending system to beat the issue of recuperation methodology in GPSR. It ascertains the most limited way by utilizing Dijkstra's calculation. A source

sets a GSR, which comprises a rundown of intermediate hubs installed in the header of all information parcels by a source. Each sending hub maps the situation of its neighbors into diagram hubs and picks the following hub having the most limited way from the objective [32].

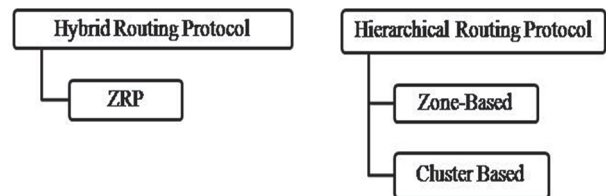


Fig. 3. Classification of hybrid and hierarchical protocol

Routing with Geographical Awareness It utilizes the GSR bundle-sending system to beat the issue of recuperation methodology in GPSR. It ascertains the most limited way by utilizing Dijkstra's calculation. A source sets a GSR, which comprises a rundown of intermediate hubs installed in the header of all information parcels by a source. Each sending hub maps the situation of its neighbors into diagram hubs and picks the following hub having the most limited way from the objective [32].

3.2 HYBRID ROUTING PROTOCOL

The advantages of reactive and proactive routing protocols are combined. Zone Routing Protocol is one of the maximum well-favored hybrid routing protocols (ZRP). Tracking occurs after segmenting the community into numerous zones, and the supply and vacation spot cellular nodes' arrangement happens. Proactive routing uses to transmit the facts packets among the supply and vacation spot cellular nodes if placed inside the identical zone. Additionally, reactive routing hires to transmit the facts packets among the supply and vacation spot cellular nodes if they locate in separate zones.

3.2.1 Zone directing convention (ZRP)

ZRP is a hybrid, including table-driven and adaptive contract management elements. With this method, the hop has a predefined address area that identifies the limits of each hub established upon the adequate opportunity of active organizations. Located in the downtown area, but for jumps outside these spaces, their paths have resolved, and these centers only use process leaders whose receptive consent is the ultimate source. Compared with the table convention, it reduces the corresponding channel. Compared with on-demand arrangements, it also limits the delay of package movement [33].

3.3 HIERARCHICAL ROUTING PROTOCOL

The idea of direct control is to divide self-organizing network hosts into different coverage areas or incompatible groups [34]. A hierarchical network uses when the network size within MANET increases significantly.

The direct control protocol classifies organizations into a cluster tree, where the tasks and elements of the hub are different at different levels of the direct system [35]. Heterogeneous forwarding protocols divide into two sub-categories: area-based and cluster-based. Multivalent Polish associations with management and leadership skills organize these agreements.

3.3.1 Zone-Based Hierarchical Routing Protocol

As correspondences pass across the covering extensions, each hub has a local scope and various directing techniques inside and outside the extension. With this adaptability, a more proficient general steering execution is possible. Moreover, by keeping up with steering data for all hubs in the organization, portable hubs in a similar zone realize how to arrive at one another with a more modest expense. In some zone-based directing conventions, explicit hubs are entryway hubs and complete between-zone correspondences. All along these lines, the organization will contain allotments or various zones. Examples: ZRP [36] is a MANET zone-based progressive directing protocol [37].

3.3.2 Cluster-Based Hierarchical Routing Protocol

The group control protocol is the most common method of sequence control. The division of organizations into interconnected sub-projects is called a "cluster," and the basis of interconnectedness is called a "cluster." The magnetic head acts as a non-permanent base station in its area or group. He also talked with other team leaders [38]. The group management agreement uses explicit group calculations for the political career of group leaders. The mobile center is standardized for new associations, and the association leader is responsible for effectively recruiting administrative staff and leadership skills. A universally voluntary agreement on organizational management, the device can support a multi-level group structure, such as hierarchical state routing (HSR) [39].

4. SIMULATION ENVIRONMENT

This paper's primary objective is to implement AODV in practice on a natural system and assess the protocol's effectiveness. The mobility model also plays a crucial part in performance comparison. Network Simulator 2.34 is employed to replicate the needed work. Discrete event simulator NS2 employs the C++ and OTcl programming languages. When sending data to the Internet domain, the MANET nodes use constant bit rate (CBR) traffic sources in the chosen simulation environment. The mobile nodes in the simulation environment move by our chosen random waypoint mobility model. Using the set dest software, we created the movement scenario files and the CBR gen tool used to create the traffic. Each simulation run lasts for a total of 300 seconds. Parameters in table 1 use To simulate a network.

Table 1. Simulation Parameter

Vehicle Density	5,10,20,30,40,50,60 nodes
Simulation Time	300 second
Mobility (Km/hr.)	40 km/hr.
MAC	802.11p
Propagation Model	Two-Ray Ground
Area	500 * 500
Mobility	Random Walk
Antenna	Omni Antenna
Traffic Model	CBR

This replay event uses NS-2, a discrete test system created at the University of California, Berkeley. The NS-2 organization simulator can plan new protocols, check various protocols, and estimate traffic [40]. The following created document saves on the board with *.tr and other script records. These scripts calculate the number of packages transported and the length of the path taken by each package. This information is also visible in the AWK script [41]. The number of hubs with the maximum length of each line is 50. Any waypoint model in a 500 m x 500 m rectangular field uses as the general model, and the station is also suitable for this rectangular field [42][43]. Each package tour departs from the incorrect location and travels to an unusual destination at a random speed.

5. PERFORMANCE RESULTS

Use the graph to display the results of the above recovery attempts. The metrics used to validate the results are output, packet transfer rate, residual energy, and average throughput output [44] [45].

5.1 PACKET DELIVERY RATIO

This ratio can derive from the entire sum of packets arriving at the target divided by the entire sum of packets directed by the origin, which is the packet transmission rate. Therefore, the delivery rate of packages is critical in evaluating the effectiveness of guidance arrangements in an organization.

The main constraints are the size of the location, the number of centers, the spread range, and the organizational structure [46]. The total amount of information determines the transportation part of the package. The data packet indicated in the objection decomposes into all information sources sent by the source [47]. In this manner, the bundle transmission rate is the proportion of the number of parcels received at the objective to the number of bundles sent by an origin. The presentation impact is better when the bundle transportation speed is high.

$$P = \left(\frac{P_r}{P_g} \right) \times 100 \quad (1)$$

Where are the number of received packets and the number of generated packets?

5.2 AVERAGE THROUGHPUT

Compared with the imaginary package delivery, this is a reasonable proportion of the actual package delivery. The expected message bandwidth tells the client the number of packets displayed on its target. The performance will show the amount of information removed from the source at any time. Organizational efficiency includes how much data can be moved from source to target in a given time. During the protest, the number of packages would display successfully. As the power limits estimate in bits per second, similarly, the data rate per second is in data units.

$$T = \left(\frac{R}{T^2 - T^1} \right) \times \left(\frac{8}{1000} \right) \quad (2)$$

Where R is the complete received packets at all destination nodes, is the simulation stop time, and the simulation starts time.

5.3. RESIDUAL ENERGY

Adding up the energy consumed when the concentrator is in each state gives the remaining energy of the sensor concentrator. The hub loses a certain amount of energy for each data packet sent and received. Therefore, the value of the initial energy is in the concentrator. After receiving or sending address packets, the current energy value in the concentrator is constant energy. The energy model refers to the energy level of the organization's hub. The energy mode described in the hub has a primary meaning: the energy that the center has at the beginning of the stationary phase. This energy is called the "initial energy." In playback, the variable "energy" refers to the energy level in the middle of a predetermined point in time. Transmitters will transmit the initial energy value as a payload. The hub loses a certain amount of energy for each data packet sent and received. As a result, the initial energy value decreases in the middle. After receiving or sending address data packets, the current energy cost of the center is excess energy. Information transmission is established between centers using CBR traffic and UDP experts. The center is estimated multiple times by obtaining the variable "energy" included in the energy search method.

5.4. DELAY

This metric calculates the average time between the packet origination time at all sources and the packet reaching time at all destination nodes. It is measured by:

$$D = \frac{\sum_{i=1}^N a_t^i + a_p^i + a_{pc}^i + a_q^i}{N} \quad (3)$$

Where N is the number of total transmission links, is the link's transmission delay, is the link's propagation delay, is the link's processing delay, and is the transmission delay of the link.

6. RESULTS AND DISCUSSION

The presentation measurements utilized for MANET directing convention investigation incorporate packet delivery ratio, average throughput, and residual energy consumed [48]. The packet delivery ratio is the proportion of the number of information bundles conveyed to the objective. A higher worth of packet delivery ratio shows that the convention is performing better [49] [50][51][52]. AODV and DSR convention gives the most noteworthy packet delivery ratio. However, because of successive course disappointments and its proactive nature, it has been found that DSDV gives a meager packet delivery ratio. Fig. 4 shows a graph of the packet delivery ratio over different nodes in a MANET with random mobility.

Fig. 4 shows that the exhibition of AODV is superior to the four protocols in the corresponding PDR. As the count of nodes rises, the data packet delivery rate of AODV decreases on all nodes.

DSDV remains unchanged as the count of nodes rises, but DSR and GSR rise and decrease as the count of nodes rises. The count of mobile nodes 5 for AODV, GSR, AOMDV, and DSDV are almost the same, but they have changed a lot afterward. The exhibition of DSDV is consistent with the count of nodes.

All nodes remain unchanged except for five for AODV, DSR, DSDV, AOMDV, and GSR. The remaining energy of DSR is the largest for all nodes, and all other remaining power is the same for different nodes. DSDV remains unchanged as the count of nodes rises.

Fig. 7 shows the delay for all routing protocols. From Fig. 7, the AODV delay is the lowest compared to other protocols.

7. CONCLUSION

This paper analyses the presentation of existing routing protocols like AODV, DSR, AOMDV, DSDV, and GSR. Packet transfer rate, residual energy, and average are execution measurements. The result is obtained after the dissection some proactive and on-request steering conventions. For low loads and low portability, dynamic DSDV conventions produce improved results.

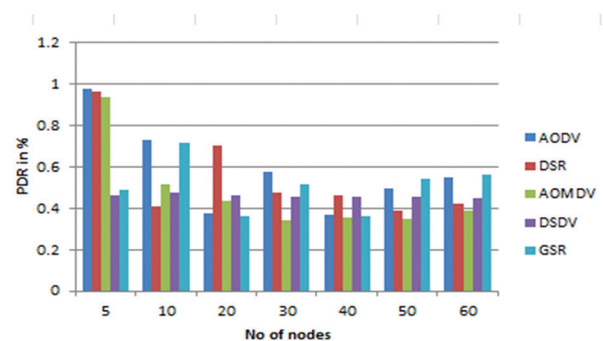


Fig. 4. Packet Delivery Ratio versus the number of nodes

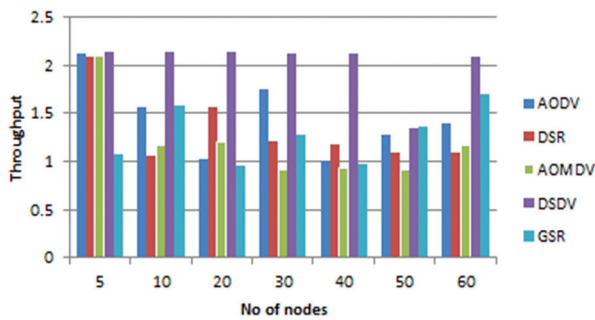


Fig. 5. Average Throughput versus the number of nodes

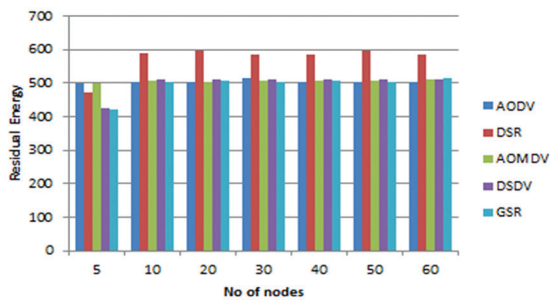


Fig. 6. Residual Energy versus the number of nodes

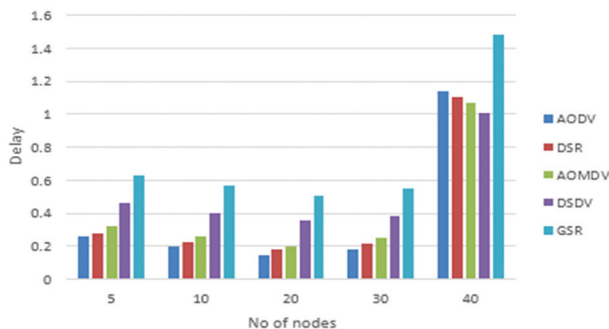


Fig. 7. Delay versus number of nodes

Most noteworthy packet delivery ratios are obtained by operating AODV and DSR conventions. AODV and DSR conventions are more suitable for high traffic because of their responsive nature, which creates less upward control. The DSDV convention gives a typical high throughput, paying little heed to organized traffic. For all hubs, the DSR convention has a below-the-norm throughput. In contrast with AODV and DSR, DSDV conventions consume less energy overall. Accordingly, the DSDV convention decreases energy utilization while expanding network lifetime.

8. REFERENCES

- [1] W. A. N. W. Abdullah, N. Yaakob, R. B. Ahmad, M. E. Elobaid, S. A. Yah, "Impact of clustering in AODV routing protocol for wireless body area network in the remote health monitoring system", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 13, No. 2, 2019, pp. 689-695.
- [2] Y. Zhang, Y. Shi, "Tactical wireless network visualization: Requirements and representations", *Proceedings of the IEEE Third International Conference on Data Science in Cyberspace*, 2018, pp.740-743
- [3] C. H. Nemade, U. Pujeri, "Execution Evaluation of AODV Protocol Using NS2 Simulator for Emergency Automobile", *Revista Geintec-gestao Inovacao e Tecnologias*, Vol. 11, No. 4, 2021, pp. 1792-1801.
- [4] S. R. Inamdar, R. M. Yadahalli, "Paradigm Shift in Routing Approach for High-Speed MANET Applications", *European Journal of Engineering and Technology Research*, Vol. 2, No. 1, 2017, pp. 59-64.
- [5] M. Vijaya Rao, "Efficient multicast routing protocol for MANETs", *International Journal of Computer Networks and Wireless Communication*, Vol. 6, 2016, pp. 2250-3501.
- [6] L. Naik, R. U. Khan, R. B. Mishra, "Analysis of Node Density and Pause Time Effects in MANET Routing Protocols using NS-3", *International Journal of Computer Network and Information Security*, Vol. 8, No. 12, 2016, pp. 9-17.
- [7] L. Naik et al. "Analysis of Node Velocity Effects in MANET Routing Protocols using Network Simulator (NS3)", *International Journal of Computer Applications*, Vol. 144, No. 4, 2016, pp. 1-5.
- [8] L. Naik, R. U. Khan, R. B. Mishra, "Analysis of Performance Improving Parameters of DSDV using NS-3", *International Research Journal of Engineering and Technology*, Vol. 3, No. 7, 2016, pp. 446-452.
- [9] E. Mahdipour, A. M. Rahmani, E. Aminian, "Performance evaluation of destination-sequenced distance-vector (DSDV) routing protocol", *Proceedings of the International Conference on Future Networks*, 2009, pp. 186-190
- [10] N. I. Sarkar, W. G. Lol, "A study of MANET routing protocols: Joint node density, packet length, and mobility", *Proceedings of the International Conference on Future Networks*, 2010, pp. 515-520.
- [11] R. K. Jha, P. Kharga, "A Comparative Performance Analysis of Routing Protocols in MANET using NS3 Simulator", *International Journal of Computer Network and Information Security*, Vol. 7, No. 4, 2015, pp. 62-68.

- [12] Q. Razouqi, A. Boushehri, M. Gaballah, L. Alsaleh, "Extensive simulation performance analysis for DSDV, DSR and AODV MANET routing protocols", Proceedings of the 27th International Conference on Advanced Information Networking and Applications Workshops, 2013, pp. 335-342.
- [13] N. A. Haseeb Zafar, D. Harle, I. Andonovic, "Survey of Reactive and Hybrid Routing Protocols for Mobile Ad Hoc Networks", International Journal of Communication Networks and Information Security, Vol. 3, No. 3, 2011.
- [14] D. Loganathan, P. Ramamoorthy, "Performance Analysis of Enhanced DSDV Protocol for Efficient Routing in Wireless Ad Hoc Networks", Research Inventy: International Journal Of Engineering And Science, Vol. 2, No. 10, 2013, pp. 1-8.
- [15] C. Samara, E. Karapistoli, A. A. Economides, "Performance comparison of MANET routing protocols based on real-life scenarios", Proceedings of the IV International Congress on Ultra Modern Telecommunications and Control Systems, August 2018, pp. 870-877.
- [16] M. N. Alslaim, H. A. Alaqel, S. S. Zaghloul, "A comparative study of MANET routing protocols", Proceedings of the Third International Conference on e-Technologies and Networks for Development, 2014, pp. 178-182.
- [17] M. N. Abdulleh, S. Yussof, H. S. Jassim, "Comparative Study of Proactive, Reactive and Geographical MANET Routing Protocols", Communication Networks, Vol. 7, No. 2, 2015, pp. 125-137.
- [18] S. Lalar, A. Yadav, "Comparative Study of Routing Protocols in MANET", Oriental Journal of Computer Science and Technology, Vol. 10, No. 1, 2017, pp. 174-179.
- [19] B. A. Mahmood, D. Manivannan, "Position Based and Hybrid Routing Protocols for Mobile Ad Hoc Networks: A Survey", Wireless Personal Communications, Vol. 83, No. 2, 2015, pp. 1009-1033.
- [20] M. Dua, "Performance Evaluation Of Aodv, Dsr, Dsdv Mobile Ad-Hoc Protocols On Different Scenarios: An Analytical Review", International Journal of Advanced Computer Technology, Vol. 1, No. 1, 2012, pp. 26-45.
- [21] P. Papadimitratos, Z. J. Haas, "Securing mobile ad hoc networks", Mobile Computing Handbook, 2004, pp. 457-481.
- [22] M. Ruta et al., "Semantic-based resource discovery, composition, and substitution in IEEE 802.11 mobile ad hoc networks", Wireless Networks, Vol. 16, No. 5, 2010, pp. 1223-1251.
- [23] S. B. Elizabeth, M. Royer, G. I. O. T. Chai-Keong, "A review of current routing protocols ad hoc mobile wireless networks", IEEE Personal Communications, Vol. 6, No. 2, 1999, pp. 46-55.
- [24] A. Joshi, P. Srivastava, P. Singh, "Security Threats in Mobile Ad Hoc Network", SAMRIDDHI Physical Sciences, Engineering and Technology, Vol. 1, No. 2, 2015, pp. 1-22.
- [25] Y. YuHua, C. H. Min, J. Min, "An optimized Ad-hoc On-demand Multipath Distance Vector (AOMDV) routing protocol", Proceedings of the Asia-Pacific Conference on Communications, 2005, pp. 569-573.
- [26] M. K. Marina, S. R. Das, "Ad hoc on-demand multipath distance vector routing", Wireless Communications and Mobile Computing, Vol. 6, No. 7, 2006, pp. 969-988.
- [27] P. Kumar, A. Chaturvedi, M. Kulkarni, "Geographical location based hierarchical routing strategy for wireless sensor networks", Proceedings of the International Conference on Devices, Circuits and Systems, 2012, pp. 9-14.
- [28] P. Karkazis, H. C. Leligou, T. Orphanoudakis, T. Zahariadis, "Geographical routing in wireless sensor networks", Proceedings of the International Conference on Ad-Hoc Networks and Wireless, September 2014, pp. 19-24.
- [29] K. Sohrawy, D. Minoli, T. Znati, "wireless sensor networks technology, protocols, and applications", Wiley Telecom, 2007.
- [30] R. S. Battula, S. Dutt, "A Review of Location-Based Geographic Routing Protocols for Wireless Sensor Networks", International Journal of Engineering Research & Technology, Vol. 2, No. 6, 2013 pp. 1170-1174.
- [31] L. Liu, Z. Wang, W. K. Jehng, "A geographic source routing protocol for traffic sensing in an urban environment", Proceedings of the IEEE International

- Conference on Automation Science and Engineering, 2008, pp. 347-352.
- [32] S. Malhotra, P. Nasib, S. Gill, "Analysing Geographic based Routing Protocols in Manets", *International Journal of Computer Science and Mobile Computing*, Vol. 3, No. 5, 2014 pp. 1068-1073.
- [33] J. A. Sanchez, P. M. Ruiz, I. Stojmenovic, "GMR: Geographic multicast routing for wireless sensor networks", *Proceedings of the 3rd Annual IEEE Communications Society on Sensor and Ad Hoc Communications and Networks*, 2006, Vol. 1, pp. 20-29.
- [34] Sabah M. Ahmed Nabil Sabor, Shigenobu Sasaki, Mohammed Abo-Zahhad, "A Comprehensive Survey on Hierarchical-Based Routing Protocols for Mobile Wireless Sensor Networks: Review, Taxonomy, and Future Directions", *Wireless Communications and Mobile Computing*, 2017, pp. 1-23.
- [35] S. Barakovi, S. Kasapovi, J. Barakovi, "Comparison of MANET Routing Protocols in Different Traffic and Mobility Models", *Telfor Journal*, Vol. 2, No. 1, 2010, pp. 8-12.
- [36] S. Goyal, "Zone routing protocol in ad-hoc networks", *Journal of Research, Engineering and Applied Sciences*, Vol. 3, No. 3, 2013, pp. 92-98.
- [37] B. M. Susanto, A. Hariyanto, Surateno, "Performance Comparison of MANET Routing Protocol based on Random Waypoint Mobility Model", *ACM International Conference Proceedings Series*, 2017, pp. 183-187.
- [38] S. K. Bharti Kukreja, "Performance Comparison of Routing Protocols in MANET", *International Journal of Computer Science and Network Security*, Vol. 14, No. 8, 2014, pp. 108-114.
- [39] M. V. D. Khairnar, D. K. Kotecha, "Simulation-Based Performance Evaluation of Routing Protocols in Vehicular Ad-hoc Network", *International Journal of Scientific and Research Publication*, Vol. 3, No. 10, 2013, pp. 1-14.
- [40] S. Saranya, R. M. Chezian, "Comparison of Proactive, Reactive and Hybrid Routing Protocol in MANET", *International Journal of Advanced Research in Computer and Communication Engineering*, Vol. 5, No. 7, 2016, pp. 775-779.
- [41] S. Singla, S. Jain, "Comparison of routing protocols of MANET in a real-world scenario using NS3", *Proceedings of the International Conference on Control, Instrumentation, Communication and Computational Technologies*, 2014, pp. 543-549.
- [42] G. R. Vijayavani, G. Prema, "Performance comparison of MANET routing protocols with mobility model, derived based on realistic mobility pattern of mobile nodes", *Proceedings of the IEEE International Conference on Advanced Communication Control and Computing Technologies*, 2012, No. 978, pp. 32-36.
- [43] S. K. Srivastava, R. D. Raut, P. T. Karule, "Analyzing the performance of routing protocols based on evaluation of different parameters in MANETs", *Proceedings of the International Conference on Communication Networks*, 2015, pp. 258-261.
- [44] U. Draz, T. Ali, S. Yasin, A. Shaf, "Evaluation-based analysis of packet delivery ratio for AODV and DSR under UDP and TCP environment", *Proceedings of the International Conference on Computing, Mathematics and Engineering Technologies*, January 2018, pp. 1-7.
- [45] W. A. N. W. Abdullah, N. Yaakob, R. Badlishah Ahmad, M. E. Elobaid, S. A. Yah, "Corrupted packets discarding mechanism to alleviate congestion in wireless body area network", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 14, No. 2, 2019, pp. 581-587.
- [46] N. A. B. Zainal, M. H. Habaebi, I. J. Chowdhury, M. Rafiqul Islam, J. I. Daoud, "Gateway sink placement for sensor node grid distribution in LoRa smart city networks", *Indonesian Journal of Electrical Engineering and Computer Science*, Vol. 14, No. 2, 2019, pp. 834-842.
- [47] S. Shekhawat, S. Singh, S. K. Singh, "Designing of Unit EBG Cell Using Conductive Textile for Dual Band Operation", *Indian Journal of Science and Technology*, Vol. 15, No. 18, pp. 881-891.
- [48] Y. Zhang, Y. Shi, "Tactical wireless network visualization: Requirements and representations", *Proceedings of the IEEE Third International Conference on Data Science in Cyberspace*, 2018, pp. 740-743.
- [49] S. K. Srivastava, R. D. Raut, P. T. Karule, "Analyzing the performance of routing protocols based on

- evaluation of different parameters in MANETs", Proceedings of the International Conference on Communication Networks, 2015, pp. 258-261.
- [50] S. Vemuri, S. Mirkar, "A Performance Comparison of MANET Routing Protocols", Proceedings of the Innovations in Power and Advanced Computing Technologies, 2021, pp. 1-5.
- [51] U. Draz, T. Ali, S. Yasin, A. Shaf, "Evaluation-based analysis of packet delivery ratio for AODV and DSR under UDP and TCP environment", Proceedings of the International Conference on Computing, Mathematics and Engineering Technologies, January 2018, pp. 1-7.
- [52] S. Shekhawat, S. Singh, S. K. Singh, "A review on bending analysis of polymer-based flexible patch antenna for IoT and wireless applications", Materials Today: Proceedings, Vol. 66, No. 8, 2022, pp. 3511-3516.

Human Face Emotions Recognition from Thermal Images Using DenseNet

Original Scientific Paper

S. Babu Rajendra Prasad

VIT-AP University
School of CSE, Amaravathi, Vijayawada,
Andrapradesh, India.
baburajendraprasad655@gmail.com

B. Sai Chandana

VIT-AP University
School of CSE, Amaravathi, Vijayawada,
Andrapradesh, India.
saichandanas869@gmail.com

Abstract – In the current scenario face identification and recognition is an important technique in surveillance. The face is a necessary biometric in humans. Therefore face detection plays a major job in computer vision applications. Several face recognition and emotions classification approaches have been presented throughout the last few decades of research to improve the rate of face recognition for thermal pictures. However, in real-time, lighting conditions might change due to several factors, such as the different times of capture, weather, etc. Due to variations in lighting intensity, the performance of the facial expression recognition system is not good. This paper proposed a model for human thermal face detection and expression classification. Four main steps were involved in this research. Initially, the Difference of the Gaussian (DOG) filter is utilized to crop the input thermal images and then normalize the images using the median filter in pre-processing step. Then, Efficient Net is used for extracting features such as shape, location, and occurrences from thermal face images. After that, detect human faces utilized by the YOLOv4 technique to better emotions classification. Finally, classify the emotions on faces by using the DenseNet technique into seven emotions such as happy, sad, disgust, surprise, anger, fear, and neutral. The proposed method outperforms state-of-art techniques for face recognition on thermal pictures, and classifies the expressions, according to experimentations on the RGB-D-T database. The accuracy, precision, recall, and f1-score metrics will be utilized with the database to assess the efficacy of the proposed methodology. The proposed models achieve a high classification accuracy of 95.97% on the RGB-D-T database. Furthermore, the outcomes show good precision for various face recognition tasks.

Keywords: Deep learning, detection, face expressions classification, feature extraction, pre-processing, thermal face recognition, and thermal image.

1. INTRODUCTION

Recently, emotion detection has been used for a variety of purposes, including job interviews to identify whether a candidate is at ease, frightened, or confident, classrooms to assess whether students are paying attention, and supermarkets to consumer behavior with purchases. The primary concept behind emotion identification is based on features of the face, such as the shape of the mouth and eyes. To develop an emotion-detecting system, those expressions are crucial. The main problem is figuring out how to extract those expressions from high-resolution images where the face only takes up around 10% of the overall image space.

Face detection has numerous uses in the fields of information security, video surveillance, and identity authentication. The visible spectrum has received the majority of attention in face identification, this is dependent on outside factors like lighting. Measuring the light reflected by the face is necessary for visible spectrum imaging. As a result, changes in illumination can affect visual appearance significantly and impair the functioning of such devices [1]. Visible face identification is still difficult, mostly because of the many environmental changes that affect it, like poor lighting, uneven illumination, and viewing angles. In addition, hackers can recreate facial patterns to fool visible face detection systems. Since

thermal imaging records the heat radiation from the face and body temperature even in a completely dark area, it has been suggested as an answer in the latest years [2-4]. The likelihood of creating a fake face pattern is greatly diminished since the temperature of the skin on the face is directly tied to the underlying blood vessels that are specific to every unique person [5]. The performance of thermal face identification is, nonetheless, hampered by many problems, including pictures with a lot of noise, opacity to glass, low resolution, and sensitivity to temperature variation.

The last few years have seen the development of thermal IR imaging-based face recognition as a promising addition to traditional visible spectrum-based methods. Depending on their temperature and properties, several things emit a range of infrared energy [6-8]. The human body and face temperatures have a range that is very consistent and similar. As a result, the thermal signature is constant. Thermal emissivity from the facial surface is measured using IR cameras, and their images

are largely stable under changes in lighting. Subsurface characteristics are included in the anatomical data that infrared technology images. The objective of thermal face detection is to recognize a person who has been photographed in the thermal spectrum by comparing them to visible spectrum face photographs that are the most similar [9]. Thermograms, also known as thermal images are created using IR radiation by the thermal infrared (IR) camera as it picks up the heat that the surveillance target emits. These images are used in surveillance systems for recognition (e.g., the ability to categorize an object into one of the classes like a vehicle, human, or animal), the ability to define surveilled objects in detail (like a woman with a hat, a bear, a man with a coat, etc.), and object detection (e.g., the ability to tell an object apart from the backdrop). Since the camera can catch a face from a set distance away, it is a non-intrusive method of identification [10-12]. Thermal face biometrics can help with the difficult process of identifying or verifying individuals only based on their thermal characteristics.

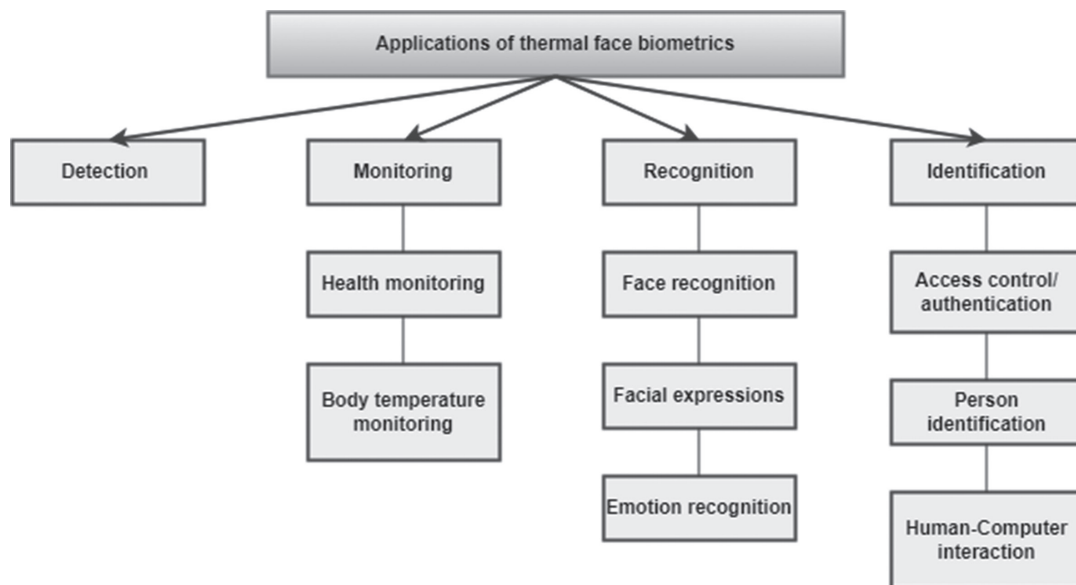


Fig. 1. Thermal biometrics applications

There are numerous applications for thermal face biometrics, including access control to protect computer systems and recognition and facilities like financial transactions, authentication for private banking, government buildings and automatic screening for terrorists at airports, use of surveillance footage, tracking body temperature in medical diagnostics, recognition of facial expression in high-end security applications, etc. Figure 1, De-identification techniques are being developed concurrently because privacy protection is a priority due to the prevalence of surveillance cameras.

A deep learning approach to thermal images is tried to solve all the existing challenges [13]. Deep learning-based techniques have been well-known in the research community as a result of their improved performance in recognizing a variety of difficulties,

such as object identification, face recognition, handwritten digit recognition, speech recognition, and human action recognition. The proposed method uses the DenseNet model, which is claimed to be superior to other approaches that use the CNN, SVM, ANN, and YOLO models. The paper's contributions are,

- In the pre-processing step, cropped the input thermal face images using by Difference of the Gaussian filter and then used the median filter to normalize the lighting variations of input thermal images.
- To extract the features such as positioning, shape, and the presence of a face from thermal face images, utilizing the Efficient Net technique.
- The YOLOv4 technique is proposed to detect the face using collected features to better classification of facial emotions.

- To classify facial expressions like happy, sad, disgusted, surprised, angry, fearful, and neutral from thermal face images, we utilized the DenseNet methodology.
- This paper proposes a method for detecting several facial emotions in thermal pictures utilizing the RGB-D-T database for the dataset of human facial expression recognition.
- The proposed model performs significantly better in the identification of the item and is capable of extracting information from photos.

The remainder of the paper is ordered as follows. Section 2 presents the related works in the paper. In section 3, the problem statement is mentioned. The proposed methodology is shown in section 4. In section 5, the result section is shown. And finally, in Section 6, the conclusion is presented.

2. LITERATURE REVIEW

A literature review is presented and discussed in this paper to offer a speculative background about thermal images and human thermal face emotions recognition methodologies.

To identify human targets in aerial view thermal pictures, Akshatha et al. [14] suggested Faster R-CNN and single-shot multi-box detector (SSD) algorithms with various backbone networks. To achieve this, two common aerial thermal datasets with variously sized human objects ResNet50, Inception-v2, and MobileNet-v1 are taken into consideration. By having a mean average precision (mAP at 0.5 IoU) of 100% for the test data from the OSU thermal dataset and 55.7% for the test data from the AAU PD T datasets, respectively, the evaluation results show that the Faster R-CNN model trained with the ResNet50 network architecture outperformed in terms of detection accuracy. The suggested framework outperforms some cutting-edge approaches compared here with a recognition accuracy of 87.46%.

Bhattacharyya et al. [15] suggested the use of the effective deep learning model IRFacExNet for the detection of facial expressions in thermal pictures. According to the direction of the research, they used a DCNN structure to build this model. To extract usable information from human faces that can be utilized for the recognition of different expressions, they have utilized two structural units, the transformation unit and the residual unit, each of which has unique strengths. The suggested framework outperforms some cutting-edge approaches compared here with a recognition accuracy of 81.16%.

Nayak et al. [16] suggested a three-stage HCI system for processing multivariate time-series thermal video sequences to identify human emotion and provide diversion recommendations. The first stage consists of following the face ROIs throughout the thermal video while simultaneously detecting faces, eyes, and noses

using a Faster R-CNN (region-based convolutional neural network) structure. The multivariate time series (MTS) data is created by calculating the mean intensity of ROIs. To categorize the emotional states induced by video stimulation, the smoothed MTS data are then sent to the Dynamic Time Warping (DTW) algorithm. In the third stage of HCI, the suggested framework offers pertinent recommendations from a physical and psychological distraction perspective. Both their created data set and the NVIE data set show 93.5% accuracy with the suggested Faster R-CNN architecture.

A method of thermal-visible face detection was suggested by Kamel et al. [17]. This approach, which is based on the YOLO v3 framework, offers enhanced solutions for face detection in both visual and thermal imaging, making it appropriate for a variety of applications including facial emotion recognition (FER) or liveness detection. The second is that they fully labeled a thermal face database and made it available to the scientific community in a GitHub repository. Thirdly, they have presented TVCycleGAN, a modified version of CycleGAN that enables the conversion of LWIR images into visible-like visuals. Finally, the networks synthesized-visible face images show great promise for thermal facial landmark identification. The suggested framework outperforms some cutting-edge approaches compared here with a recognition accuracy of 89.27%.

Siddiqui et al. [18] introduced a multimodel automated emotion recognition (AER) that is very accurate at discriminating between emotional expressions. The contribution comprises integrating speech with visible and infrared (IR) images to build an ensemble-based strategy for the AER. The architecture is implemented in two layers, with the first layer employing a single modality to identify emotions and the second layer fusing the modalities to categorize feelings. The classification and feature extraction processes have been carried out using convolutional neural networks (CNN). To merge the features and the decisions at various phases, a hybrid fusion strategy was used, consisting of late (decision-level) and early (feature-level) fusion. To arrive at the final determination, the output of the images classifier and the output of the CNN both of which were trained using speech samples from the RAVDESS database were combined. Similar f-score (0.87), precision (0.88), and recall (0.86) scores as well as an accuracy of 86.36% were attained.

3. PROBLEM STATEMENT

Visible cameras can be quite helpful in daytime situations, but they have significant problems with human face detection that may limit their utility. The difficulties are as follows:

- The street lights during the night may make it harder to see people.
- Visibility may be reduced by obstructions like trees, buildings, and cars.

- When there are many occlusions between humans or blurring as a result of cameras closing down gradually when they are placed on moving vehicles, it is very challenging to detect humans.
- It can be challenging to identify humans when the weather is foggy, rainy, snowy, or melting snow.

4. PROPOSED METHODOLOGY

Fig. 2 shows the proposed technique for the classification of the emotions of a person's face-based DenseNet technique. To recognize the human face in a thermal image, four steps are presented in this paper. They are pre-processing, feature extraction, detection,

and classification. In a pre-processing step, cropped the input thermal face images using by Difference of Gaussian (DOG) filter and used the median filter to normalize the lighting variations of input thermal images. After pre-processing step the generated output images are ready to feature extraction. An Efficient Net approach is proposed to extract the features from input thermal face images. It extracts the features of the face, such as positioning, shape, and the presence of a specific object. Then YOLOV4 technique is used to detect the human face from extracted features. And then to classify the emotions on faces in thermal images, we used the DenseNet technique. For this experiment, we collect the images from the RGB-D-T database.

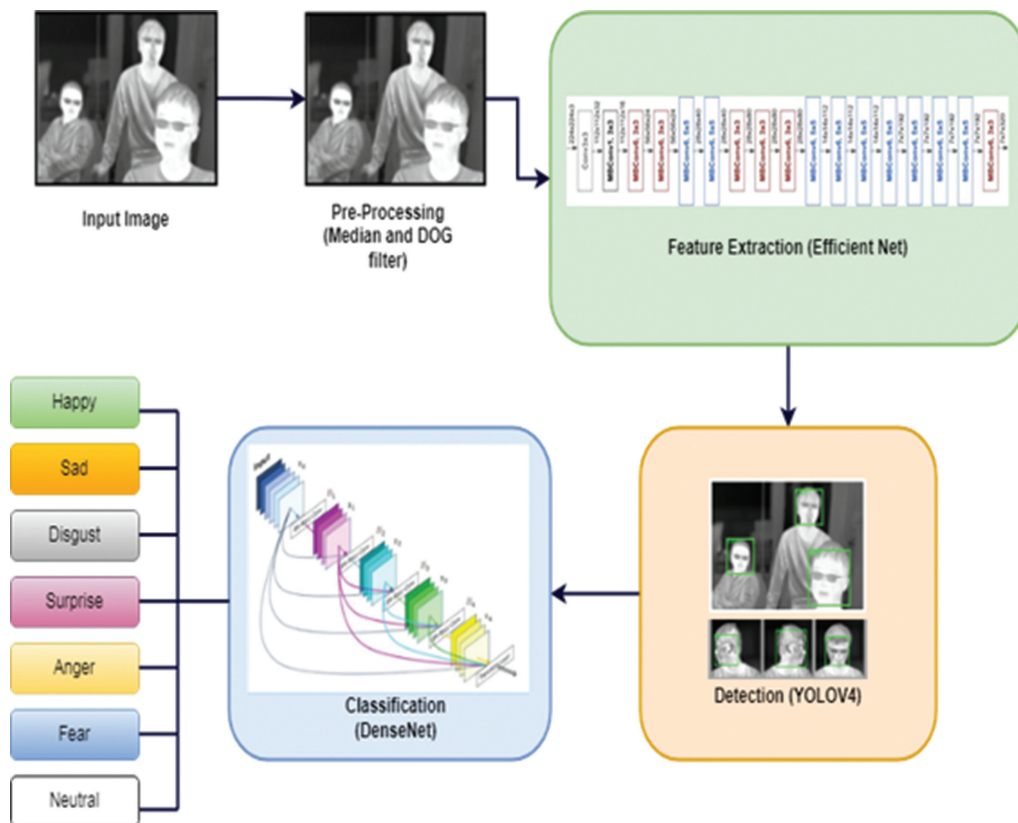


Fig. 2. The architecture of the proposed method

Pre-processing is a procedure that must be taken to extract useful information from face images. The Difference of Gaussian (DOG) filtering, a widely used method to reduce lighting variations for visible face detection, is then applied to the cropped thermal face images. An initial image and a DOG filter are convoluted to create a DOG filter image. A two-dimensional DOG filter is used for images. The one dimension is

$$G(u) = \frac{1}{\sqrt{2\pi}} \left(\frac{\exp(-(u+u)^2/2\sigma_1^2)}{\sigma_1} - \frac{\exp(-(u+u)^2/2\sigma_2^2)}{\sigma_2} \right) \quad (1)$$

And in two dimensions it is

$$G(r) = \frac{1}{2\pi} \left(\frac{\exp(-(r+P)^2/2\sigma_1^2)}{\sigma_1^2} - \frac{\exp(-(r+P)^2/2\sigma_2^2)}{\sigma_2^2} \right) \quad (2)$$

In addition to reducing local changes in the thermal imaging that result from the changing heat/tempera-

ture spread of the face, DOG filtering also minimizes lighting varieties in thermal facial imagery. Therefore, by lowering the local fluctuations in each technique while improving edge details, DOG filtering aids in closing the modality gap. The Gaussian filter is carefully chosen to include frequencies that are helpful for face recognition and reduce those frequencies that are negative for it. To lessen the variations and lighting in the face image the median filter can be used for normalization.

$$\hat{f}(x, y) = \text{median}_{(s,t) \in S_{xy}} \{g(s, t)\} \quad (3)$$

4.2. FEATURE EXTRACTION USING EFFICIENTNET

After pre-processing the images, we need to extract the features of the face for better prediction of the face

for classifying the emotions. The best shape information is discovered by feature extraction. Using these characteristics to categorize activities is simple with a systematic process. Features may appear differently to people and robots because they can be understood by machines. Almost all features serve to convey one aspect of an image, such as its shape, location, and occurrence of a certain face. Original images must first go through some preprocessing steps to ensure that they are suitable for feature extraction before they can begin.

After pre-processing the images, the resulting image is fed to the feature extraction method for extracting the facial feature using Efficient Net. Each of the modalities comprises a huge number of slices that form the segments using the Efficient Net technique.

The architecture of EfficientNet originates from the compound scaling method. The family of EfficientNet is built on the baseline model EfficientNet-B0 (i.e., $\phi = 0$). The structure of EfficientNet-B0 is summarized in Table 1. An artificial neural network's performance can be improved by carefully balancing network width, resolution, and depth. Compound scaling necessitates balance and coordination between the three scaling dimensions because they are not independent. For this, a new baseline structure called EfficientNetB0 was created, which was then scaled up using compound scaling to produce the EfficientNetB0 to the EfficientNetB7 family of EfficientNets [19]. The feature extraction from the EfficientNet has been effectively combined in the proposed study. The acquired findings are comparable to those of cutting-edge networks.

Table 1. The EfficientNet [19] Framework

Level	Operator	Resolution	Channels	Layers
EfficientNetB0 architecture, the network baseline				
1	Conv1×1/Pool/FC	7×7	1,280	1
2	MBCConv6, k3×3	7×7	320	1
3	MBCConv6, k5×5	14×14	192	4
4	MBCConv6, k5×5	14×14	112	3
5	MBCConv6, k3×3	28×28	80	3
6	MBCConv6, k5×5	56×56	40	2
7	MBCConv6, k3×3	112×112	24	2
8	MBCConv1, k3×3	112×112	16	1
9	Conv 3×3	224×224	32	1
Additional layers				
10	FC/Softmax	1	NC	1
11	FC/BN/Swish	1	128	1
12	FC/BN/Swish/Dropout	1	512	1
13	B.N./Dropout	7×7	1280	1

EfficientNet presents a new method for scaling network proportions by consistently scaling all resolution variables and structure depth/width utilizing the compound coefficient, a highly efficient agent. When compared to other CNN methods, EfficientNet provides a highly effective

technique to improve accuracy while being more efficient. The EfficientNet family includes many distinct versions that are adapted to different input layers. In several dimensions, this DCNN has been built up [20]. Most CNN architectures are built up by including more layers as part of the ResNet family. Unlike previous methods, EfficientNet built up all width, depth, and resolution dimensions simultaneously. The compound scaling mechanism was introduced in the proposed EfficientNet structure to balance all of these scaling characteristics.

Convolutional neural networks belong to the EfficientNet family. In terms of layer depth, input resolution, layer width, and a combination of these criteria, EfficientNet models scale well. EfficientNet is a recent deep-learning model that aims to improve model efficiency while also improving accuracy. From B0 to B7, there are various variants. This network's fundamental building piece is MBCConv, which has compression and excitation optimization added to it. Between the starting and finish of a convolutional block, these blocks provide shortcuts. To improve the depth of the feature maps, the input activation maps are enlarged using 1x1 convolutions. The thin layers are connected by shortcut links in this paradigm, whereas the broader layers are positioned between the jump links. This structure aids in the reduction of both the model's size and the overall number of transactions needed. The EfficientNetB0 structure is shown in Fig. 3.

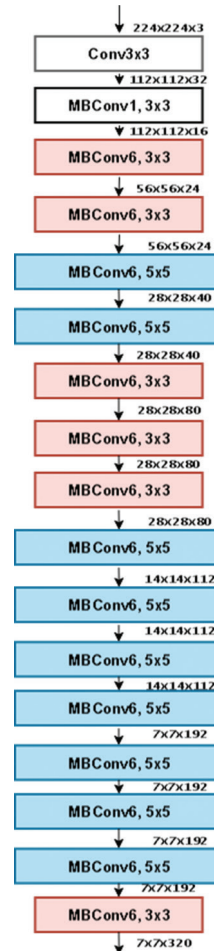


Fig. 3. EfficientNet B0 architecture [21]

Because of its higher prediction performance using a compound scaling technique across all parameters of the networks, such as depth, resolution, and breadth, EfficientNets has gotten a lot of attention [21]. To be clear, width defines the number of dimensions in any depth, the resolution is the size of the image, and the layer represents the number of levels in CNN. The theory behind compound scaling is that increasing any network dimension (width, image resolution, or depth) improves accuracy, but as the model grows larger, the accuracy gain diminishes. Compound scaling employs a compound coefficient to govern how many extra resources are useful for model scaling, and the parameters are scaled in the following fashion by the compound coefficient:

$$\begin{aligned} & \text{Resolution } R \gamma \Phi \\ & \text{Width } w \beta \Phi \\ & \text{While } \alpha \beta \gamma^2 \approx 2 \\ & \alpha \geq 1, \beta \geq 1, \gamma \geq 1. \end{aligned} \quad (4)$$

where the grid search determines the constants α , β , γ . The compound scaling yields the following coeffi-

cient after numerous experiments and considerable grid search:

$$\text{Depth } 1.20 / \text{Width } 1.10 / \text{Resolution } 1.15$$

In this feature extraction stage, it's extracting the features such as their shape, location, and occurrence to detect the face in thermal human face images.

4.3. YOLOV4 FOR DETECTION

YOLOV4 is proposed to detect the human face. After extracting the facial feature in the feature extraction stage, the detection methodology is ready to detect the faces in thermal images. This face detection step is for classifying the facial emotions clearly and providing better classification accuracy. As shown in Fig. 4, YOLOv4 is a single-stage detector that classifies and efficiently localizes the objects in an image in one pass. It was published in April 2020 and featured several data augmentation strategies, pre-and post-processing approaches, as well as minor model adjustments. The following is a concise description of YOLOv4.

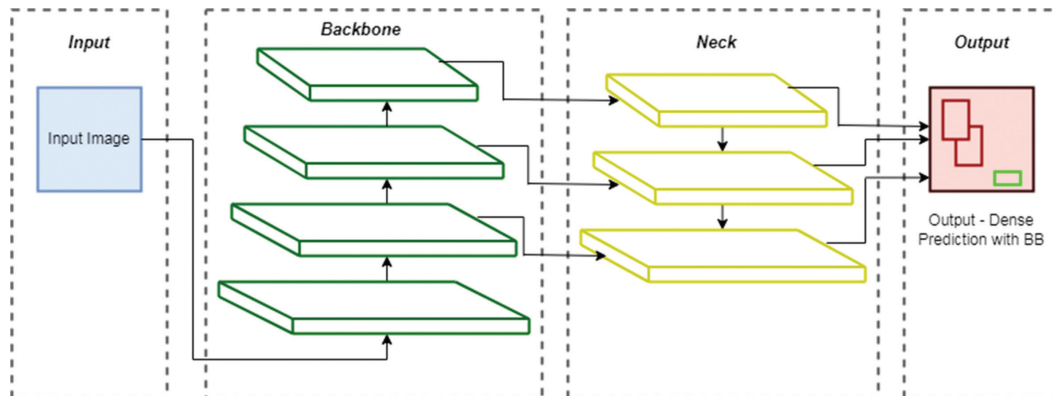


Fig. 4. YOLO: Single-stage architecture [22]

The goal of YOLO (You Only Look Once) v4 is to build a fast-functioning object detector that is also equipped for parallel processing for production systems. It had to be better in a variety of aspects, as compared to the present practices [22]. It is super-fast, high quality, and provides convincing results for object detection in terms of accuracy. Item detectors usually consist of several components: Input-This is where the picture is entered. Next, variants of Resnet-50, VGG16, ResNext50, or Darknet52, can be the backbone, which refers to the network that takes the image as input and retrieves the feature map. The neck and head are backbone sub-sets that maximize the discriminability and robustness of the function using FPN, PAN, RFB, etc., and the forecast-managing head [23-25]. For a single-stage detector such as YOLO and SSD, this may either be used for dense prediction or FRCNN and Mask RCNN, a two-stage detector also known as Sparse Prediction.

The option of architecture is the mechanism that can conjure up a suitable entity detector to be created. For the backbone, based on theoretical logic, there was an

alternative between CSPResNeXt50, CSPDarknet53, and EfficientNet B3, and several CSPDarknet53 neural network tests were found to be the most optimal model. The YOLOv4 concept, the CSPDarknet53 Spatial Pyramid Pooling block, also known as SPP, was used. Since the receptive region is significantly enhanced, it distinguishes the most important context characteristics and produces practically no decrease in the speed of network traffic [26]. As a form of parameter aggregation for various detector levels from different backbone levels, PANet, also known as the Path aggregation network, is also used, and this was used instead of the FPN, also known as the YOLOv3 Feature Pyramid Network. Eventually, they chose the head of YOLOv3, as YOLOv4's head. Different classifier training features Different training features of the detector Different backbones and pre-trained training weightings of the detector Different minibatch sizes of detector training Different training features of the detector since there are several features that they had to test, particularly in the bag of freebies and specials [27]. So the approach they used was to use a methodology called ablation

analysis to test every feature. An ablation analysis is when you manually remove parts of the input to see which parts of the input are relevant to the network performance.

Normally, it looks like a table like this with the observations on the right-hand side. Speaking of results, if we look at how YOLOv4 relates to others, you would be very impressed. But to ensure that we compare each other with oranges and apples. Depicts the steps involved in the object detection process in the YOLOv4. Separate GPU architectures are used for inference cycle checking to test broadly accepted GPU architectures as

competing models. The comparative GPU architectures used were the Maxwell, Pascal, and Volta architectures. You can see from these tables that YOLOv4 is comparable to the fastest and most reliable in terms of both speed and accuracy [30]. This analysis uses a state-of-the-art detector that on MS COCO datasets is faster in terms of Frames per Second (FPS) and more reliable than all available alternative detectors. On a traditional 8-16GB VRAM GPU that is readily accessible, YOLOv4 can be educated and used. The YOLOv4 is checked with a broad variety of features and the best ones are selected to improve both the classifier and the detector efficiency.



Fig. 5. Thermal images collected from RGB-D-T database



Fig. 6. The result of YOLOv4 detection in thermal images

Figure 5&6 use bounding boxes around recognized humans' faces showing the precision of thermal pictures. These detected faces are utilized to classify the face into their emotions category.

4.4. DENSENET FOR CLASSIFICATION

The classification of facial emotions in thermal images into emotions including surprise, anger, sadness, happiness, fear, disgust, and neutral. To classify human emotions from thermal face images, the DenseNet technique is used. The graphical representation of DenseNet creates a 5-layer dense block with a growth rate of $k = 4$ as seen in Figure 7. The 121 in DenseNet-121 stands for the total number of layers in the neural network. Combinations of different layers make up the conventional DenseNet-121 structure. It has five layers of pooling and convolution, three transition levels (6, 12, and 24), one classification layer (16), and two DenseBlocks (1x1 and 3x3 convs). The accuracy of the model for classifying the person's emotions from thermal photos is improved by DenseNet's feature reuse and parameter reduction [27]. After the composite function operation, the output of the first layer enters the second layer as an input. A non-linear activation layer, a pooling layer, a convolution layer, and a batch normalization layer make up the composite process.

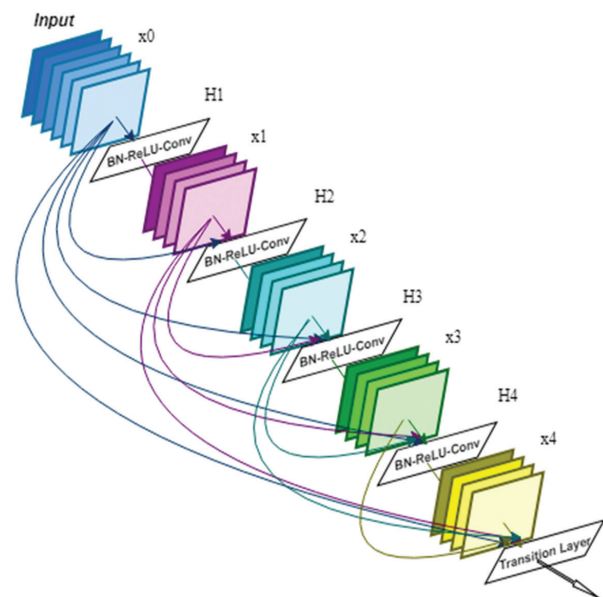


Fig. 7. Block diagram of DenseNet [27]

The Dense Block is an important part of the DenseNet for improving the information flow between layers. It is composed of BN, ReLU, and 3×3 Conv. The specific formula is shown as follows,

$$x_1 = H_1([x_0, x_1, \dots, x_{l-1}]) \quad (5)$$

Where $[x_0, x_1, \dots, x_{l-1}]$ refers to the concatenation of the feature maps produced in layers $0, 1, \dots, l-1$, $H_l(\cdot)$ is defined as a composite function of three consecutive operations on the input of l^{th} layer. The function (such as ReLU, sigmoid) to increase their nonlinearity. The convolution process can be expressed as,

$$z^l = W^l f_1(z^{(l-1)}) + b^l \quad (6)$$

Where z^l is the l^{th} -layer neuron status, $f^l(\cdot)$ the activation function, w^l and b^l are the weight matrix and bias from $(l-1)^{th}$ to the l^{th} , respectively. Contrary to popular assumption, DenseNets require few parameters than traditional CNNs since they do not need to know unnecessary feature maps. DenseNets layers are constrained and barely present any new feature maps. Re-using features produces highly compact versions and is the main idea behind DenseNet. Since no feature maps are duplicated, it requires fewer parameters than other CNNs. CNN encounters problems as they delve deeper. The reason for this is that the gradient in the opposite direction of the gradient from the inner layer to the outer layer gets very long that it can evaporate before hitting any farther side. By simply linking every layer to every layer, DenseNet makes this connectivity considerably simpler. By recycling features, DenseNets maximize the capacity of the network. As explained, DenseNet is a type of CNN. Typically, DenseNet architectures use dense blocks to connect all layers directly to one another, creating dense connections across layers. According to DenseNet, the more connections there are, the more accurate the system will be. In a DenseNet, each layer delivers its feature maps to the succeeding layers and receives new input from the lay-

ers above it. The idea of conjunction is utilized to describe how each layer gains collective wisdom from the levels above it. Multiple classifiers are combined into an ideal, DCNN and connected with a dense connection for effective picture categorization to maximize computation recycling between the classifiers. On the majority of them, DenseNet significantly surpasses the state-of-the-art while using the least amount of memory and processing possible to maximize its efficiency.

5. RESULTS AND DISCUSSION

This section begins by using our method to identify a face and classify their emotions on thermal pictures from the RGB-D-T database and comparing it to state-of-the-art techniques. Finally, we present the assessment findings according to experimental findings to evaluate our method in the subsections that follow.

5.1. EXPERIMENTAL DATASET

Our DenseNet model is trained using a portion of the RGB-D-T database. Thermal camera data was gathered as RGB-D-T. It has a 51-person capacity and a 384 x 288 resolution. Thermal face detection is influenced by three variables, including facial expression, head rotation, and illumination. We create the train set for our model using the thermal faces from a subset of the RGB-D-T database [28]. About 12K thermal facial photos are included. In addition, we randomly extract non-face regions in addition to the facial regions to train our model on roughly 12K photos also serving as our test set are 3K thermal photos from the database. We contrast how the various elements affect the face detection rate.



Fig. 8. Several illustrations of our thermal facial image database in various unrestricted settings.

However, the previous thermal face database is very easy to recognize when compared to the RGB face database. It is necessary to construct a database with multiple face thermal photos in unrestricted situations. RGB face database is the initial database of thermal pictures with various faces that we are aware of. The number of persons varies (from one to three), as does the head rotation (up, down, left, and right). The photographs have a resolution of 640 x 480 and were captured using 10K thermal cameras in 10 different environments. For

the sake of additional investigation, we also take the relevant RGB photos. Examples from five different contexts are shown in Figure 8. Figure 8's second column features three persons with various expressions and head rotations while the first column, which was taken at night, only has one human with a neutral expression. The head rotation of the person in the third column who is leaning against the window at nighttime changes. The fourth column's spacing changes and the last column has several expressions.

5.2. EVALUATION METRICS

Each detector generates a bounding box that shows the location of people in the input photos. We can identify which predicted bounding boxes are accurate by comparing them to ground truth boxes. The intersection over union (IOU), which calculates the ratio between a pair of boxes' union and intersection, is used to quantify overlap. A perfect disjoint alignment is 0, and a perfect alignment of the two boxes is 1. A bounding box is considered to be an accurate forecast if it overlaps the ground truth by at least 0.3. Typically, an overlap of 0.5 or greater is necessary for object detection. However, it was chosen to be a little leaner here because some of these things can be rather little. Only one prediction can be mapped to a ground truth box. A prediction is deemed accurate if it falls within a ground truth box; otherwise, it is deemed incorrect.

In terms of performance measures, looked at the proposed method's Accuracy (A), Precision (P), F1-score (F), and Recall (R). These metrics indicate:

5.2.1. Accuracy

The accuracy metric is calculated to determine whether the categorization of advertisements is accurate.

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (7)$$

5.2.2 Precision

The proportion of accurately predicted positive outcomes to all predicted positive observations is known as precision. The ability to carry out the following actions is precision:

$$Precision = \frac{TP}{TP+FP} \quad (8)$$

5.2.3. Recall

The terms for the recall are the True Positive Rate (TPR) and Sensitivity. The classifier's capacity to identify all positive samples is shown by the recall score. It is the total divided by TP, including FN. It can be described in the following terms:

$$Recall = \frac{TP}{TP+FN} \quad (9)$$

5.2.4. F-Measure

F-Measure determines the harmonic mean of recall and precision.

$$F1\ Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (10)$$

5.3 QUANTITATIVE EVALUATION

In this paper, we proposed a new deep learning classifier, the DenseNet technique. Our proposed method includes four steps, pre-processing, feature extraction, detection, and classification. In Pre-processing step, we crop the input image using the DOG filter to reduce lighting variations and then normalize the image utiliz-

ing the median filter. After pre-processing, the features like the shape and location of the face are extracted from input images using by EfficientNet. The YOLOv4 method is used to detect the human face in a single frame. Then classify the facial emotion from detected face images with the help of extracted features into seven categories using the DenseNet technique. Here, we provide some thermal images which are collected from the database to detect the faces in a single thermal image and also classify the face by their emotions.

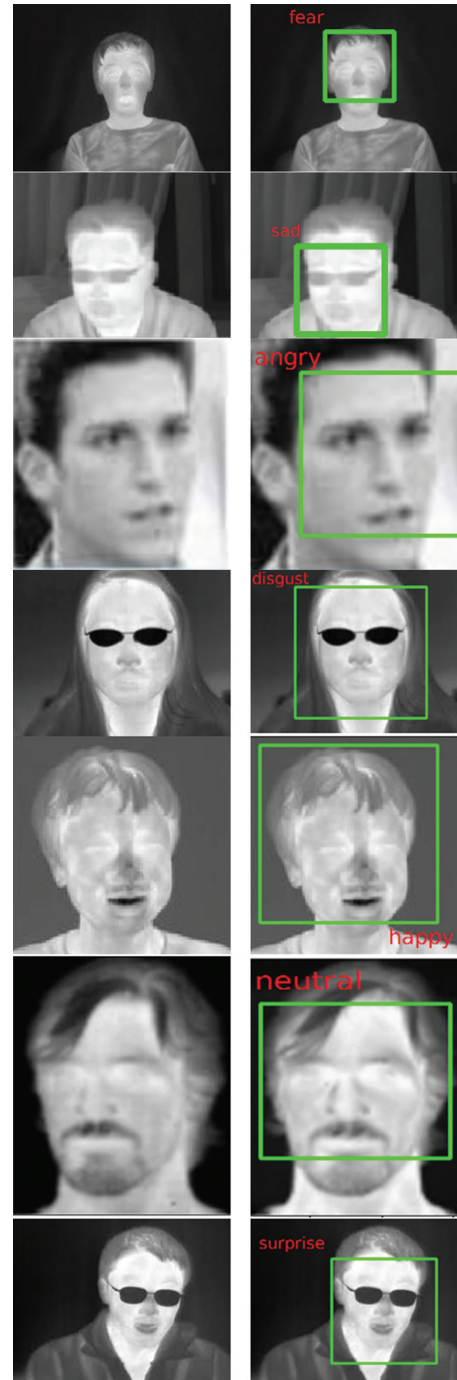


Fig. 9. The classification outcome of facial emotions using the DenseNet technique

The classified emotions on faces show in figure 9. This is classified into seven emotions.

5.4 PERFORMANCE METRICS

We compare various approaches with the RGB-D-T database studies. Table 2 shows that no matter the circumstances, our technique has the highest effectiveness and increases the recognition rate. We have no trouble identifying a single heat face opposing a dark foreground, without a doubt. Part of the reason is that the faces are closer to one another than in the photographs that featured three persons. When only

one person was present in the test photographs, our approach was still able to identify the thermal face regardless of the head's rotation, occlusion, or facial expression. The proposed approach operates nicely in our database, according to the results. Table 2 lists our findings based on accuracy, precision, recall, and computing time. Compared with other techniques to classify emotions, our proposed method gives higher classification accuracy with less computation time. It can classify emotions clearly and correctly.

Table 2. Using the proposed and compared approaches, calculate precision, F-Score, accuracy, and recall (%) while classifying the computing time.

	Dataset	Precision	F-Score	Accuracy	Recall	Computing Time (ms)
SSD [14]	AAU PPT	92.04	92.5	87.46	91.24	0.15
DCNN [15]	IR Database	75.60	68.47	81.16	63.28	0.23
Faster R-CNN [16]	NVIE	89.52	90.85	93.5	93.03	0.31
YOLO V3 [17]	NVIE and PUJ	84.02	84.49	89.27	85.98	0.28
Proposed (DenseNet)	RGB-D-T database	96.15	95.34	95.97	97.42	0.16

Table 2 lists the results for SSD [14], Faster R-CNN [16], DCNN [15], YOLOV3 [17], and the planned DenseNet on the RGB-D-T database in terms of Recall, Accuracy, F-Score, and Precision. Based on the results, we can see that the proposed methodology has higher classification accuracy values than other deep learning approaches in terms of recognition rate Recall, Precision, and F-Score. The achieved F-Score for the proposed technique is 95.34%, compared to 90.85% for faster R-CNN [16], 84.49% for YOLOv3 [17], 68.47% for DCNN [15], and 92.50% for SSD [14]. The proposed method's acquired Precision is quite similar to SSD [14]'s precision. Additionally, when compared to other existing methodologies, the proposed approach's recall is superior. DCNN [15] has the lowest accuracy rate, at 81.16 percent. The better results are achieved by conducting the training phase with 26 epochs and a learning rate of 0.01. For each epoch, the experiment analysis used 31 iterations.

Figures 10-14 show how the DenseNet improves the classification outcomes. It completes the task in 0.16 milliseconds with an F-score of 95.34 percent, recall of 96.64 percent, precision of 95.92 percent, and accuracy of 95.37 percent.

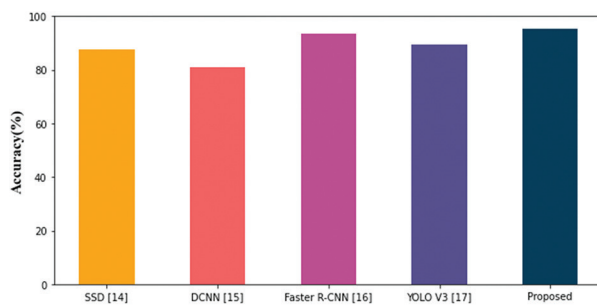


Fig. 10. Comparison of the suggested technique's classification Results with previous Methods in Terms of Accuracy

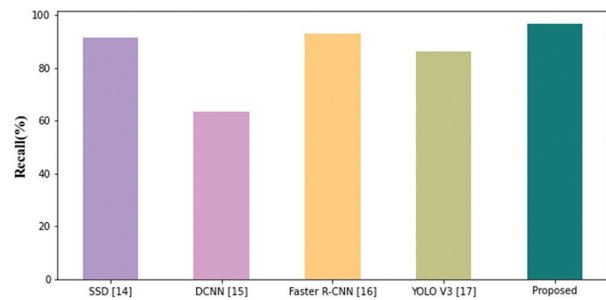


Fig. 11. Performance comparison of the classification with known methods in terms of Recall

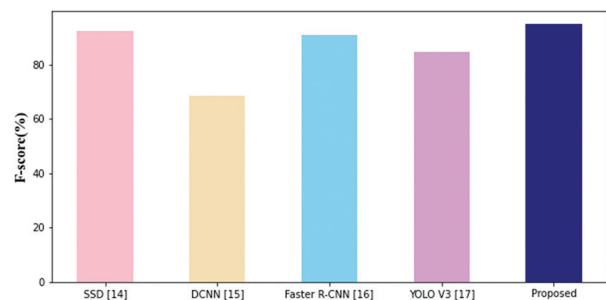


Fig. 12. Performance comparison of the classification with known methods in terms of F-score

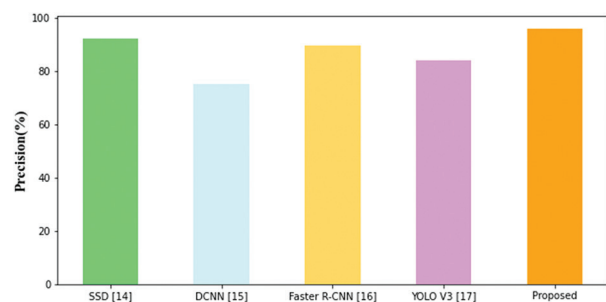


Fig. 13. Performance comparison of the classification with known methods in terms of Precision

Calculation time is another factor that is compared. Deep learning methods aim to lessen the complexity of computation. Fig. 14 displays how long the cutting-edge methods and the proposed model needed to compute using the RGB-D-T database.

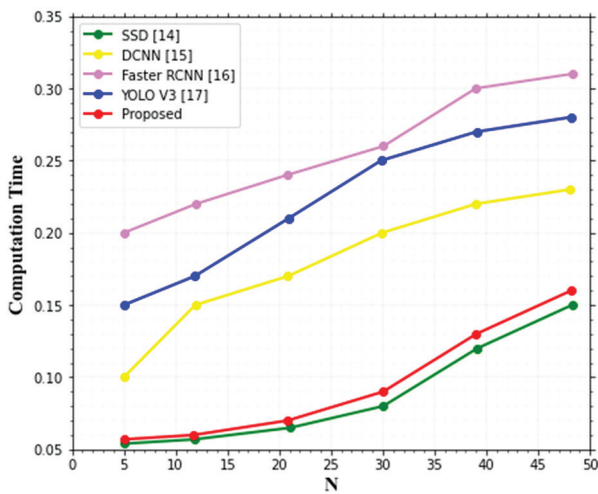


Fig. 14. Comparing the time complexity of the suggested approach to the existing techniques

From Figs. 10-14, it can be shown that the proposed strategy exceeded the other techniques and displayed the highest Accuracy, F-Score, Recall, and Precision with less consumption time.

Table 3. Human face emotions accuracy

Face Emotions	Accuracy (%)
Happy	95.6
Sad	96.9
Disgust	93.4
Neutral	99.5
Anger	97.8
Fear	98.2
Surprise	98.8

Various human facial emotions accuracy can be depicted in table 3. The various person faces emotions like happiness, sadness, surprise, anger, disgust, fear, and neutrality. Happy can yield 95.6% accuracy, sad obtain 96.9% of accuracy, disgust gain 93.4% of accuracy, neutral yield 99.5% of accuracy, anger has 97.8% of accuracy, fear obtains 98.2% accuracy and surprise gain 98.8% of accuracy. Among the seven human face emotion accuracy, the neutral emotion achieves greater accuracy. Figure 15 shows the graphical representation of the human face emotion classification.

Emotions like anger, fear, happiness, surprise, sadness, disgust, and neutrality are the ones that the model was having less accurate precision scores which are shown in Fig. 16. The most used technique for assessing classification errors is the confusion matrix. Based on the provided confusion matrix explanations,

developed the confusion matrix for the DenseNet proposed model. The diagram shows that the DenseNet model can classify facial emotions with the RGB-D-T database having anger, fear, happiness, surprise, sadness, disgust, and neutral images. This shows that the proper categorization of the two statuses has been carried out. The obtained confusion matrix for the cross-validation test of classification is shown in Figure 16. Our proposed approach provides higher classification accuracy with less computation times as compared to previous techniques for classifying emotions.

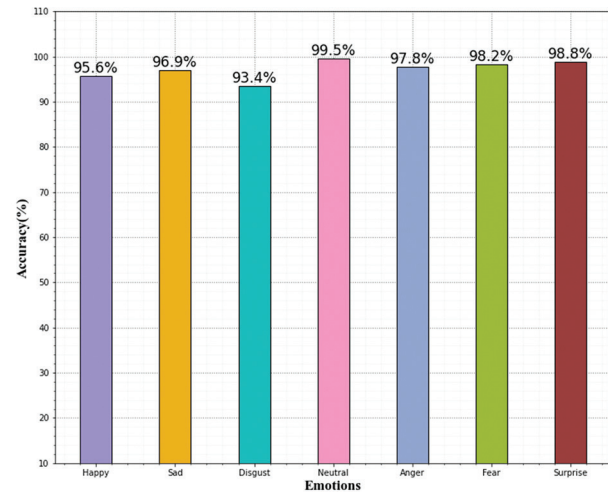


Fig. 15. Evaluation of the classifier's performance in analyzing emotions in terms of Accuracy

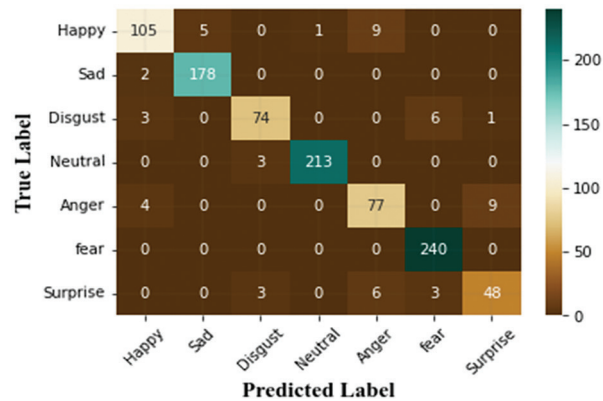


Fig. 16. Confusion matrix of facial emotions

6. CONCLUSION

In this paper, a novel DenseNet technique is proposed and introduced to thermal face emotion identification. Initially, the Difference of the Gaussian filter is used to crop the input images and then the median filter is used to normalize the input images in pre-processing step. An EfficientNet technique is used to extract the multi-scale features. The YOLOv4 technique is used for detecting the human face. Then the proposed model DenseNet is used to classify the images by extracted features. Our model performs optimally for thermal facial emotions classification, according to

the RGB-D-T results. This experiment gives 95.97% classification accuracy using the DenseNet classification technique. Our algorithm continues to perform well in general. In the future, we will gradually improve the model and further improve the detection accuracy of this algorithm based on considering the improvement of high-level features.

Conflict of interests

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We declare that this manuscript is original, has not been published before and is not currently being considered for publication elsewhere.

Availability of data and material: Not applicable

7. REFERENCES

- [1] Y. Bi, M. Lv, Y. Wei, N. Guan, W. Yi, "Multi-feature fusion for thermal face recognition", *Infrared Physics & Technology*, Vol. 77, 2016, pp. 366-374.
- [2] P. Saha, D. Bhattacharjee, B. K. De, M. Nasipuri, "Characterization and recognition of mixed emotional expressions in thermal face image", *Infrared imaging systems: Design, analysis, modeling, and testing xxvii*, International Society for Optics and Photonics, Vol. 9820, 2016, p. 982005.
- [3] Y. M. Elbarawy, R. S. El-Sayed, N. I. Ghali, "Local entropy and standard deviation for facial expressions recognition in thermal imaging", *Bulletin of Electrical Engineering and Informatics*, Vol. 7, No. 4, 2018, pp. 580-586.
- [4] A. Sancen-Plaza et al. "Facial recognition for drunk people using thermal imaging", *Mathematical Problems in Engineering*, Vol. 2020, 2020.
- [5] P. Saha, D. Bhattacharjee, B. K. De, M. Nasipuri, "A Thermal Blended Facial Expression Analysis and Recognition System Using Deformed Thermal Facial Areas", *International Journal of Image and Graphics*, 2021, p. 2250049.
- [6] U. Atila, M. Uçar, K. Akyol, E. Uçar, "Plant leaf disease classification using EfficientNet deep learning model", *Ecological Informatics*, Vol. 61, 2021, p. 101182.
- [7] Y. H. Lai et al. "Data fusion analysis for attention-deficit hyperactivity disorder emotion recognition with thermal image and Internet of Things devices", *Software: Practice and Experience*, Vol. 51, No. 3, 2021, pp. 595-606.
- [8] A. K. Prabhakaran, J. J. Nair, S. Sarath, "Thermal facial expression recognition using modified resnet152", *Advances in Computing and Network Communications*, Springer, Singapore, 2021, pp. 389-396.
- [9] A. I. Middya, B. Nag, S. Roy, "Deep learning based multimodal emotion recognition using model-level fusion of audio-visual modalities", *Knowledge-Based Systems*, Vol. 244, 2022, p. 108580.
- [10] D. Jiang et al. "A probability and integrated learning based classification algorithm for high-level human emotion recognition problems", *Measurement*, Vol. 150, 2020, p. 107049.
- [11] A. Bhattacharyya, S. Saha, S. Sen, S. Mirjalili, R. Sarkar, "Deep Feature Selection Using Moth-Flame Optimization for Facial Expression Recognition from Thermal Images", *Handbook of Moth-Flame Optimization Algorithm*, CRC Press, 2022, pp. 281-312.
- [12] M. K. Chowdary, T. N. Nguyen, D. J. Hemanth, "Deep learning-based facial emotion recognition for human-computer interaction applications", *Neural Computing and Applications*, 2021, pp. 1-18.
- [13] S. K. Km, R. Rajendran, Q. Wan, K. Panetta, S. S. Agaian, "TERNet: A deep learning approach for thermal face emotion recognition", *Mobile Multimedia/Image Processing, Security, and Applications*, International Society for Optics and Photonics, Vol. 10993, 2019, p. 1099309.
- [14] K. R. Akshatha et al. "Human Detection in Aerial Thermal Images Using Faster R-CNN and SSD Algorithms", *Electronics*, Vol. 11, No. 7, 2022, p. 1151.
- [15] A. Bhattacharyya, S. Chatterjee, S. Sen, A. Sinitca, D. Kaplun, R. Sarkar, "A deep learning model for classifying human facial expressions from infrared thermal images", *Scientific Reports*, Vol. 11, No. 1, 2021, pp. 1-17.
- [16] S. Nayak, B. Nagesh, A. Routray, M. A. Sarma, "Human-Computer Interaction framework for emotion recognition through time-series thermal video sequences", *Computers & Electrical Engineering*, Vol. 93, 2021, p. 107280.

- [17] N. K. Benamara, E. Zigh, T. B. Stambouli, M. Keche, "Towards a Robust Thermal-Visible Heterogeneous Face Recognition Approach Based on a Cycle Generative Adversarial Network", *International Journal of Interactive Multimedia and Artificial Intelligence*, Vol. 7, No. 4, 2022.
- [18] M. F. Siddiqui, A. Y. Javaid, "A multimodal facial emotion recognition framework through the fusion of speech with visible and infrared images", *Multimodal Technologies and Interaction*, Vol. 4, No. 3, 2020, p. 46.
- [19] S. Pachade, P. Porwal, M. Kokare, L. Giancardo, F. Mériaudeau, "NENet: Nested EfficientNet and adversarial learning for joint optic disc and cup segmentation", *Medical Image Analysis*, Vol. 74, 2021, p. 102253.
- [20] T. Liu, B. Pang, L. Zhang, W. Yang, X. Sun, "Sea Surface Object Detection Algorithm Based on YOLO v4 Fused with Reverse Depthwise Separable Convolution (RDSC) for USV", *Journal of Marine Science and Engineering*, Vol. 9, No. 7, 2021, p. 753.
- [21] I. Sim, J. H. Lim, Y. W. Jang, J. You, S. Oh, Y. K. Kim, "Developing a Compressed Object Detection Model based on YOLOv4 for Deployment on Embedded GPU Platform of Autonomous System", arXiv:2108.00392, 2021.
- [22] J. H. Sejr, P. Schneider-Kamp, N. Ayoub, "Surrogate Object Detection Explainer (SODEx) with YOLOv4 and LIME", *Machine Learning and Knowledge Extraction*, Vol. 3, No. 3, 2021, pp. 662-671.
- [23] X. Sun, T. Liu, X. Yu, B. Pang, "Unmanned Surface Vessel Visual Object Detection Under All-Weather Conditions with Optimized Feature Fusion Network in YOLOv4", *Journal of Intelligent & Robotic Systems*, Vol. 103, No. 3, 2021, pp. 1-16.
- [24] D. Wu, S. Lv, M. Jiang, H. Song, "Using channel pruning-based YOLO v4 deep learning algorithm for the real-time and accurate detection of apple flowers in natural environments", *Computers and Electronics in Agriculture*, Vol. 178, 2020, p. 105742.
- [25] N. Kumari, V. Ruf, S. Mukhametov, A. Schmidt, J. Kuhn, S. Küchemann, "Mobile Eye-Tracking Data Analysis Using Object Detection via YOLO v4", *Sensors*, Vol. 21, No. 22, 2021, p. 7668.
- [26] M. Zhang, S. Xu, W. Song, Q. He, Q. Wei, "Lightweight Underwater Object Detection Based on YOLO v4 and Multi-Scale Attentional Feature Fusion", *Remote Sensing*, Vol. 13, No. 22, 2021, p. 4706.
- [27] X. Li, X. Shen, Y. Zhou, X. Wang, T. Q. Li, "Classification of breast cancer histopathological images using interleaved DenseNet with SENet (IDSNet)", *PloS One*, Vol. 15, No. 5, 2020, p. e0232127.
- [28] D. Mahouachi, M. A. Akhloufi, "Adaptive deep convolutional neural network for thermal face recognition", *Thermosense: Thermal Infrared Applications XLIII*, Vol. 11743, SPIE, 2021, pp. 15-22.

A Performance Enhancement of Deepfake Video Detection through the use of a Hybrid CNN Deep Learning Model

Original Scientific Paper

Sumaiya Thaseen Ikram

School of Information Technology and Engineering
Vellore Institute of Technology,
Vellore, Tamil Nadu, India.
isumaiyathaseen@vit.ac.in

Priya V

School of Information Technology and Engineering
Vellore Institute of Technology, Vellore,
Tamil Nadu, India.
vpriyacse@vit.ac.in

Shourya Chambial

School of Information Technology and Engineering
Vellore Institute of Technology, Vellore,
Tamil Nadu, India.
shourya39@gmail.com

Dhruv Sood

School of Information Technology and Engineering
Vellore Institute of Technology, Vellore,
Tamil Nadu, India.
sooddhruv2@gmail.com

Arulkumar V

School of Computer Science and Engineering
Vellore Institute of Technology, Vellore,
Tamil Nadu, India.
arulkumar.v@vit.ac.in

Abstract – In the current era, many fake videos and images are created with the help of various software and new AI (Artificial Intelligence) technologies, which leave a few hints of manipulation. There are many unethical ways videos can be used to threaten, fight, or create panic among people. It is important to ensure that such methods are not used to create fake videos. An AI-based technique for the synthesis of human images is called Deep Fake. They are created by combining and superimposing existing videos onto the source videos. In this paper, a system is developed that uses a hybrid Convolutional Neural Network (CNN) consisting of InceptionResnet v2 and Xception to extract frame-level features. Experimental analysis is performed using the DFDC deep fake detection challenge on Kaggle. These deep learning-based methods are optimized to increase accuracy and decrease training time by using this dataset for training and testing. We achieved a precision of 0.985, a recall of 0.96, an f1-score of 0.98, and support of 0.968.

Keywords: Deepfake, Machine learning, Deep learning, Inception, Xception

1. INTRODUCTION

Information sharing and broadcasting are now much easier and faster, thanks to the growth of social media platforms. With only one click, people may now access knowledge from around the globe. Regarding news consumption, social media platforms can be utilized for two different purposes: to alert the public of breaking news or, conversely, to disseminate false information [1].

DeepFakes is a popular concept with widespread application. Deepfakes ("fake") are synthetic media (AI-generated media) in which an existing image or clip of a person is superimposed with another person's image [2] [3].

To damage the character's reputation, deepfake technology is used to replace performers' faces in pornography, revenge porn, fake news, hoaxes, and financial fraud with the faces of celebrities. This has spurred business and government actions to identify and forbid their use. The three most risky ways to apply face-swapping algorithms identified are as follows: (i) Face-swap, in which one face is automatically superimposed on another; (ii) Lipsync, a technique in which only a portion of a person's face is altered, forcing them to utter things they have never said before; and (iii) puppet master, in which the face of the target individual is animated by a person sitting in front of the camera [4]. FakeApp, created by a Reddit user using the auto encoder-decoder pairing structure, was

the first deepfake generation attempt. The face images are broken down into their parts in this manner by the autoencoder, which also extracts latent properties from the face images. Two encoder-decoder pairs, each trained on a different image set, are required to swap faces between the source and target images. The two network pairs share the encoder's parameters. Alternatively, two pairs share a common encoder network [5]. Many businesses, including Facebook Inc., Google, and the United States Defense Advanced Research Projects Agency (DARPA), have launched a research initiative to find and eliminate deep fakes [6] and [7]. Numerous deep learning methods, including long short-term memory (LSTM), recurrent neural networks (RNN), and even hybrid approaches, have been created to detect deep fakes in images and videos, and additional research has been conducted in this area [8] and [9].

Many studies have been done on deep-fake detection due to the quick development of face swaps and other video manipulation technologies. Various attempts have been made to find a solution to this problem. Visual artifacts, common among deep fakes [10], have been used frequently in solution strategies. The Deepfake Detection Challenge was developed in collaboration with META, Microsoft, and AWS on AI's Media Integrity Steering Committee and academics (DFDC). The challenge's purpose is to persuade scholars worldwide to create successful new techniques for detecting deep fakes and controlling the media. In another instance, Google researchers announced the AI Principles, stating that they are committed to developing AI models that reduce the risk of harm and misuse [11]. The researchers contributed a synthetic speech dataset in 2018 to aid in a big competition to build very effective fake audio detectors. In 2019, they contributed a sizable collection of visual deepfakes.

The primary goal of this paper is to examine the available approaches, highlight trends, and address the current issues in the investigations. Finally, the performance of the various techniques is analyzed. This paper proposes a new hybrid technique based on Inception Resnet V2 and Xception. Multiple input samples, such as positive, negative, and generated samples, are used to train the Xception and InceptionResnet v2 networks for classification. During the training process, a regularization loss is implemented to ensure the embedding space's inter-class proximity and intra-class regularity.

Spreading deep fakes over social media platforms has grown increasingly common, resulting in spamming and speculation based on inaccurate information. These deep lies will be terrifying and deceiving to the general population. Deep fake detection is essential for fixing this issue. As a result, we describe a unique deep learning-based technique for distinguishing between AI-generated false films (deep fake Movies) and true videos. It is vital to develop technologies capable of identifying forgeries to detect and prevent deep fakes from spreading over the internet.

Our work aims to develop a robust and efficient model to help reduce the threat posed by malicious users who try to exploit online and open-source images for unethical purposes and to malign a person's image. It also aims to reduce the false information spread by these fake videos.

Section 2 describes existing techniques for deepfake detection; Section 3 describes the background; Section 4 describes the methodology, which also includes dataset description, data preprocessing, the proposed technique, and the technology used; Section 5 includes results and analysis, and Section 6 presents conclusions and future work.

2. RELATED WORK

2.1 DETECTION BASED ON ML

Xin et al. [12] developed a system against exposing AI-generated fraudulent face pictures or videos and compared head locations computed using all visual indicators to those judged using only the center area. Li et al. [13] identified blinking of eyes in films, a behavioral indication poorly represented in the bogus film. Falko et al. [14] presented a collection of simple characteristics for recognizing produced faces, deep fakes, and Face2Face pictures in the eyes, teeth, and facial contours. Guarnera et al. [15] examined bogus videos of human faces to develop a novel discernment approach capable of detecting a forensics trail buried in photos.

2.2 DETECTION BASED ON CNN

A. Facial Tampering

Guera and Delp [16] devised a solution consisting of key components of a convolutional neural network and long short-term memory. After combining the attributes of many consecutive frames, CNN creates a collection of features for each frame in a particular picture sequence and provides them to the LSTM for analysis. The suggested model underwent training on 600 videos and attained an accuracy of 97.1 percent.

Li and Lyu [17] established a technique for identifying distorted images in manipulated films with an accuracy of up to 99 percent when trained with four distinct deep-learning models on legitimate and modified photos. Zhou et al. [18] proposed a multi-stream network for facial recognition modification in the deepfake. A deep learning face classification model is being trained in the first stream to collect evidence of tampering with artifacts. In the second stream, a steganographic model-based multi-layer network is trained to regulate functions that collect leftover noise evidence nearby. Afshar et al. [19] built MesoNet, a CNN, to distinguish between the actual and Deepfake-modified faces. Meso-4 and MesoInception-4 are two models based on inception used in the network, along with

layers linked with the max-pooling function. Khalid and Simon [20] developed a one-class approach for identifying deepfakes and achieved 97.5% accuracy on the face forensics++ dataset without having any fake images in the training samples. The authors in [22] developed a strategy for creating a Deepfake detector dubbed FakeCatcher (FC), which emphasizes using features derived from face regions to recognize synthetic portrait films. Missing reflections and minute features in the facial areas are exploited, and characteristics from the face are retrieved from the essential facial features and supplied into machine learning classifying models for identifying them as fake or real films.

B. Digital Media Forensics

Oza and Patel [23] developed a One-class convolutional Neural Network as an instance of a one-class-based technique (OC-CNN). The primary notion behind OC-CNN is to employ a negative class of zero-centered Gaussian noise in the hidden space and train the network using cross-entropy loss. Cozzolino et al. [24] proposed ForensicTransfer (FT), an architecture based on autoencoders that distinguish legitimate from tampered photos. The ForensicTransfer contacts multiple tests and results with an accuracy rate of up to 80% to 85%. Nguyen et al. [25] suggested an aggregate deep-learning method for simultaneously detecting and dividing altered pictures and clips. The suggested system includes an encoder that encodes binary classification characteristics and a Y-shaped decoder that adopts the results from one of its sub-branch to partition the modified areas. The authors in [26] reported a deep learning model that detects Deepfake using a capsule network (CN). Furthermore, it detects replay assaults and computer-generated images.

3. BACKGROUND

A. Generic Overview

This section attempts different ways to determine

whether videos are fake. The annotations are saved in a JSON file in the train sample videos folder, and a video dataset is used for analysis.

The stages involved are as follows:-

- Reading and collecting images from the videos.
- The image is placed in the correct folder after reading the label from the JSON file.
- After converting the image to an array, the data is divided into train and test groups.
- Using InceptionResNetV2 and Xception to train data and customize them.
- Testing for accuracy and outcomes.

The dataset is pre-processed first in the low-level design, and then the model is trained, tested, and results in predictions. The DFDC dataset is used for experimental analysis. The dataset is pre-processed. The model's initial state consists of frames of real and fake images being generated under the real and fake folders, respectively, and these images will be the input for the model. Finally, the model is tested using test videos and produces the desired output.

High level: This is the overview of the system. In the proposed approach, both real and fake images are considered. Fake images are generated using a generator and a discriminator to discriminate between fake and real images. The low-level and high-level diagrams of the proposed approach are shown in Fig. 1 and Fig. 2.

The primary contribution of Inception-V3 and Xception is that they mix numerous convolution filters, such as Conv (1-1), Conv (3-3), and Conv (5-5), in a multi-extractor. Typically, the Inception-V3 design has 22 convolutional layers and 5 pooling layers. Because of the variety of Inception and its high memory requirements in V3, a more optimized version of the creation family known as Xception has been proposed to reduce computational complexity. Separable convolutions have been suggested in this variant [27].

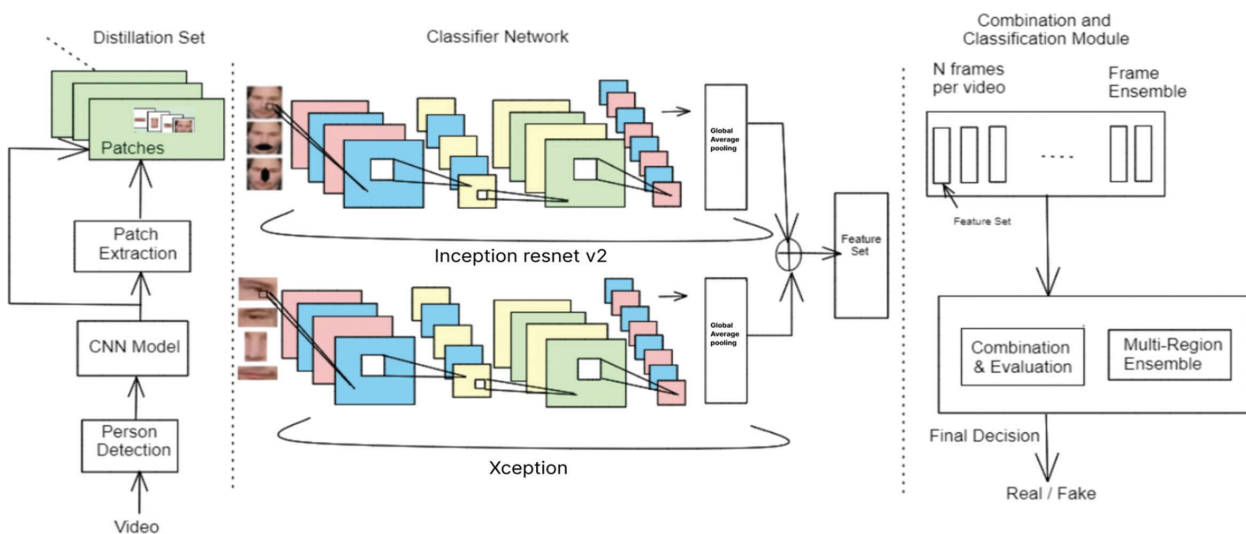


Fig. 1. Low-level diagram of modules

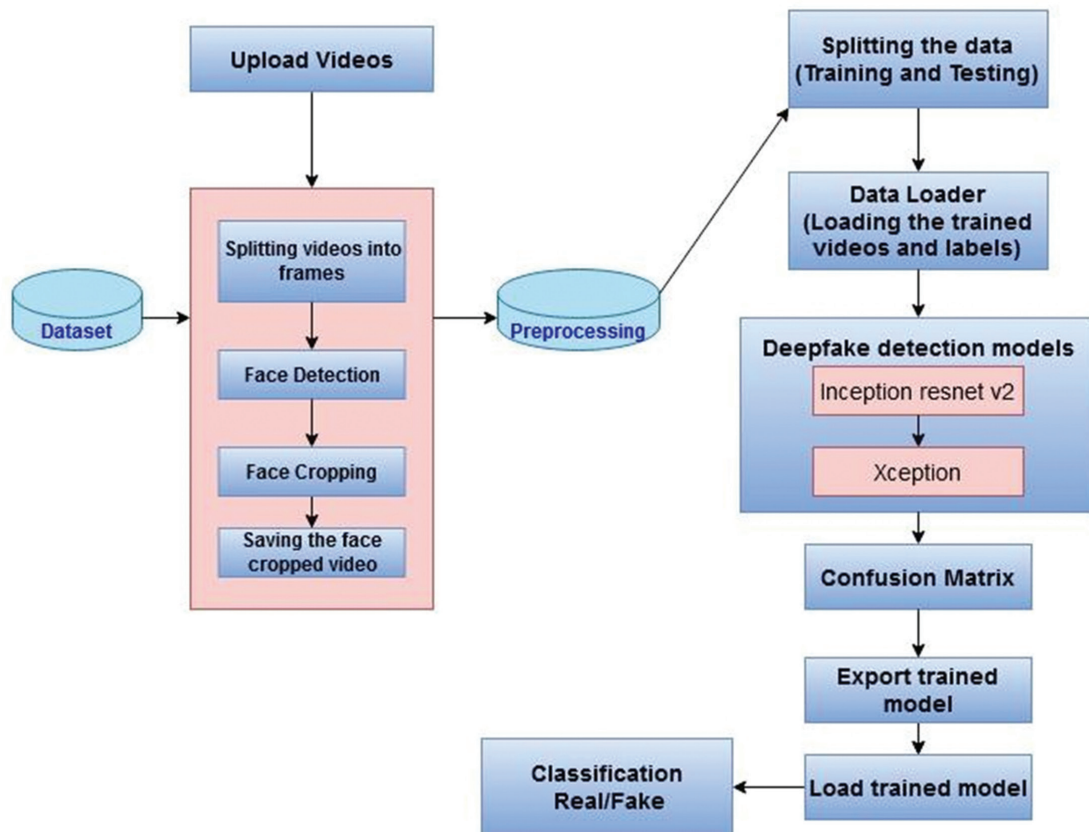


Fig. 2. High-level diagram

B. About Inception ResNet v2

Inception-ResNet-v2 is a convolutional neural network trained on millions of images from the image net collection. It has a network of over one hundred and fifty layers and has been used to classify pictures into thousand distinct items such as flowers, food, aeroplanes, etc. Therefore, as a consequence, the framework has learned a wide range of rich feature interpretations for a wide range of pictures. Consider an input image with dimensions of around 300 by 300 pixels to generate a list of anticipated class probabilities.

C. Working of Inception ResNet V2

The basis of the model is based on the structure of Inception, ensembled with the residual connection. Several combinations are made between residual connections and convolutional filters of various sizes. The residual connections have been utilized to handle the degradation problem and have even helped reduce the training time by fifty percent. The output from the inception model is added to the current input connections of the residual network.

The input and output dimensions must be in sync to perform the residual addition. One by one, convolution has been utilized to match the depth size. The Inception network has three modules, A, B, and C, to form the entire network. The pooling layer will be replaced by a one-by-one convolution layer along with the residual connection network.

D. Pseudocode for Inception Network followed by ResNet

This pseudocode is used to identify and classify images.

Input: clips and frames of images.

Output: The face is detected using a boundary-based box.

- A one-by-one convolution non-activation layer is added to the network to match the depth.
- Summation layers are not a part of the batch normalization process; apart from them, normalizations are used everywhere.
- Residual values were then scaled down before being added to the prior layer activation, which helped to stabilize the training. Scaling values of 0.1 to 0.3 were chosen to scale the residuals.
- A combination of residual connections with several-sized filters happens in the network block. This brings down the training time by fifty percent.
- Factorize five-dimensional convolution into two three-dimensional convolution processes to increase computing speed. A five-dimensional convolution costs around three times as much as a three-dimensional convolution, which may look contradictory. Stacking two three-dimensional convolutions enhances performance as a consequence.

- The dimensions would be considerably decreased if the module was made deeper instead, resulting in information loss. The filter banks of the module were thus expanded to remove the above factor.

The architecture of Inception ResNet V2 is shown in Figure 3. Inception Architecture ResNet V2, The architecture of Inception ResNet V2, is shown in Figure 3.

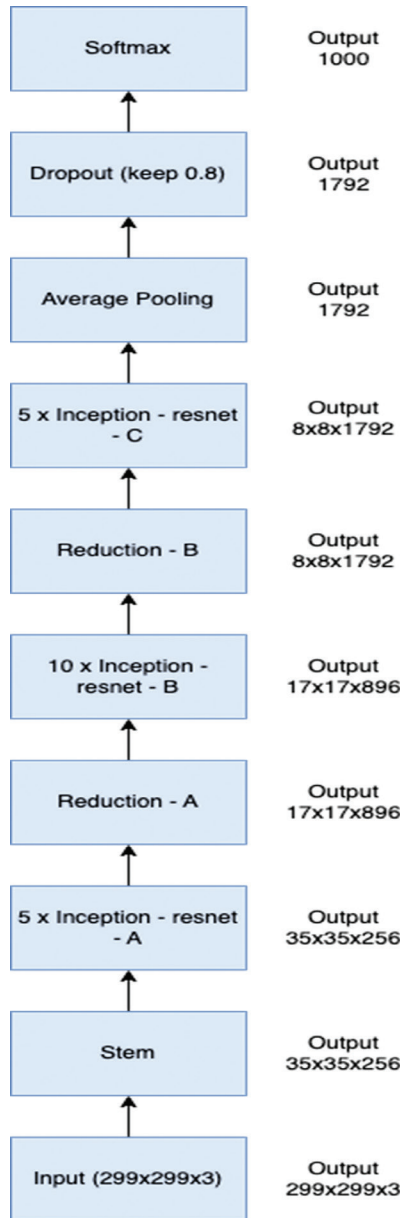


Fig. 3. Architecture of InceptionResNet V2

E. About Xception

Firstly, the data is passed through the input flow, through the middle flow eight times, and finally, through the exit flow. Every layer of separable convolution and convolution is subject to batch normalization, and the architecture is shown in Fig 4.

F. Working of Xception

Xception is a very efficient deep-learning model that depends on the following:

- Depth-wise Separable Convolution
- As in ResNet, there are shortcuts between Convolution blocks.

The architecture of Xception is made up of Depth wise separable convolution blocks and max-pooling, all of which are coupled via shortcuts in the same way as ResNet implementations. A Pointwise convolution does not follow the Depth wise convolution in Xception; instead, the sequence is inverted.

G. Pseudocode for Xception

- All the necessary layers must be imported Necessary functions must be written for
- Conv-BatchNorm block
- SeparableConv- BatchNorm block
- For each of the three flows (Entry, Middle, and Exit), write a separate function.
- Utilize these features to create the full model.

4. PROPOSED METHODOLOGY

A. Dataset

The DFDC dataset is used for the experiments. Many deepfake or face swap datasets include films shot in non-natural environments like news or briefing rooms. Worse, the people in these films may not have consented to have their faces modified. With over 100,000 total clips collected from 3,426 paid actors and produced using a variety of Deepfake, GAN-based, and non-learned algorithms, the DFDC dataset is by far the largest currently and publicly available face swap video dataset.

Each of the 100,000 forged videos in the DFDC Dataset is a one-of-a-kind target/source switch. DF-1.0 consists of 1,000 distinct bogus videos, despite the disruptions. The DFDC dataset includes movies of people in indoor and outdoor situations, with a wide range of lighting situations.

The various datasets available for Deepfake detection have been tabulated in Table 1.

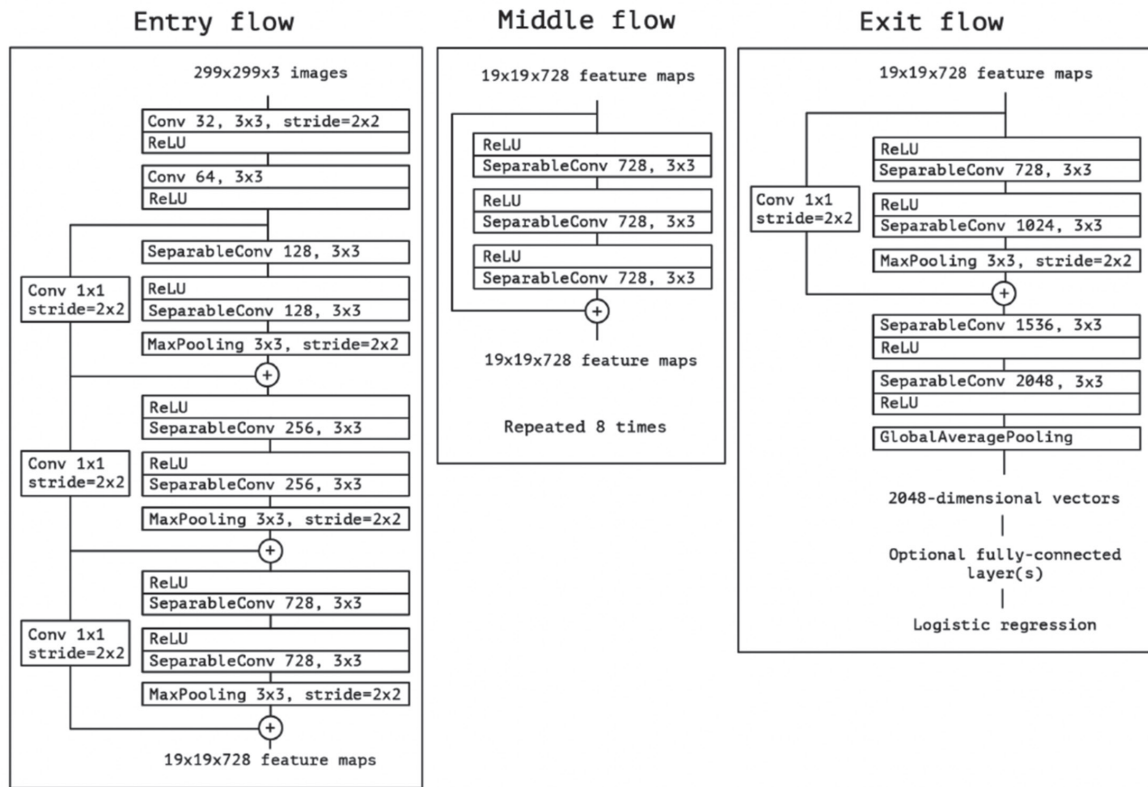
B. Our Approach

Deepfakes news is influencing the globe because individuals worldwide use it for various purposes, including face swapping, reproducing pornographic movies with someone's face or body, and manufacturing and disseminating fake news.

Deep Fakes are increasingly harming democracy, privacy, security, religion, and people's cultures. Deep Fakes are becoming more common, yet there is no standard for evaluating deep fake detection systems. Since 2018, the number of deep fake movies and photos discovered online has nearly doubled. For more than ten years, the Massachusetts Institute of Technology (MIT) evaluated 126,000 news stories shared by 3,000,000 individuals. Finally, they determined that bogus news travels 1,500 times faster than accurate news. Deepfakes create fake news, photos, videos, and

terrorist events. Deepfake undermines public faith in the media and contributes to social and financial fraud. Religions, organizations, politicians, artists, and voters

are all affected by deepfake. People will disregard the truth as deepfake videos and pictures proliferate on social media.



Overall Architecture of Xception (Entry Flow > Middle Flow > Exit Flow)

Fig. 4. Architecture of Xception [28]

	Train Data		Test Data	
	Real	Fake	Real	Fake
UADFV	35 videos (13976 frames)	35 videos (13638 frames)	14 videos (3353 frames)	14 videos (3353 frames)
Celeb-DF	370 videos (158992 frames)	733 videos (290043 frames)	38 videos (16409 frames)	62 videos (22834 frames)
Deepfake Detection	254 videos (202723 frames)	2148 videos (1678558 frames)	109 videos (94437 frames)	920 videos (681550 frames)

For learning temporal aspects of facial data from training films, a hybrid deep learning model employing CNN (Convolutional Neural Network) models consisting of Inception Resnet v2 followed by Xception is proposed. We have suggested a CNN-based model that learns different patterns between Deep-Fake and actual videos. Pixel distortion, discrepancies with facial superimposition, skin color variances, blurring, and other visual aberrations are among these distinguishing characteristics. Using a frame-based technique based on the aforementioned different properties, the suggested approach has successfully trained a CNN (convolutional neural network) to discern DeepFake films. The proposed work, which involves an ensemble of Inception and xception, shows the viability of our model's ability to identify deep fake faces in a specific video source accurately. This will help security applications used by social media platforms combat the growing threat of "deepfakes" by ac-

curately determining the authenticity of videos, allowing them to be flagged or removed before they cause harm that cannot be repaired.

The dataset is imported and converted based on metadata training and labeling in a JSON file. All face frames were cropped, aligned, and reduced to 256x256 pixels after internal face tracking and alignment were utilized to preprocess the source videos. 5,000 face frames were used to train models. The Inception ResNet v2 model feeds temporal features to the Xception model. The Xception model's feedback architecture may learn from consecutive inputs. We trained our model with 10 epochs and 25 batches. An "epoch" is a machine learning term that describes how many rounds the algorithm did across the full training dataset. Once educated, the ".h5" file can be downloaded. Hybridization successfully leverages many model layers to boost learning performance.

Table 2. Accuracy Achieved

Model	Accuracy
Inception_ResNet_v2	89.41%
Xception	93.85%
Hybrid Model	95.75%

C. Our Methodology

In the proposed approach, both real and fake images are considered. Fake images are generated using a gen-

erator, and then a discriminator is used to differentiate between fake and real images. In the low-level design, the dataset is pre-processed first, and then the model is trained, tested, and the results are determined. The DFDC dataset is used for experimental analysis. The dataset is pre-processed. The model's initial state consists of frames of real and fake images generated under the real and fake folders, respectively, and these images will be the input for the model. Finally, the model is tested using test videos and produces the desired output.

The low-level diagram depicting the flow of events is shown in Fig. 5.

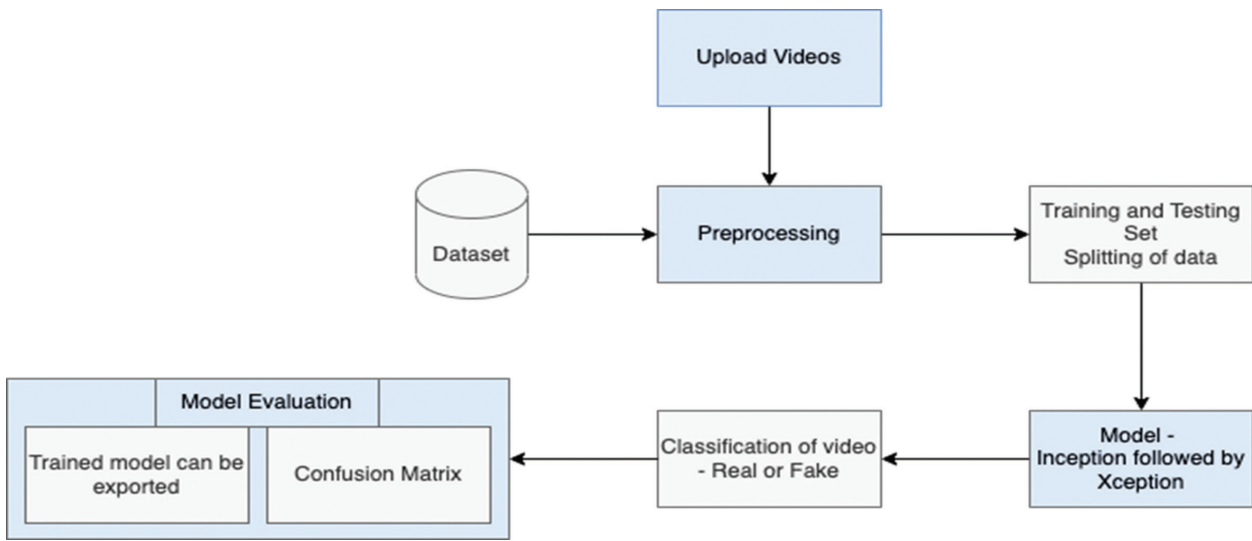


Fig. 5. Low-level diagram depicting the flow of events

The flow diagram is shown in Fig. 6.

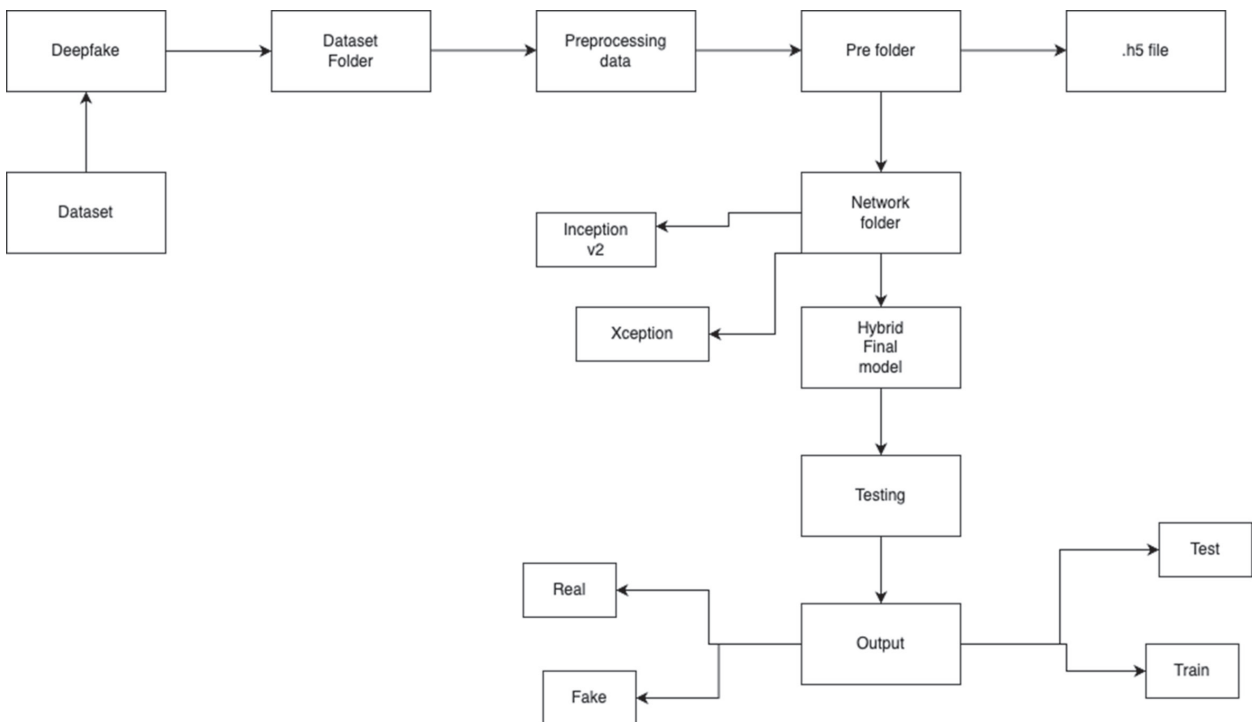


Fig. 6. Flow diagram

The pipeline design of our architecture is shown in Fig. 7.

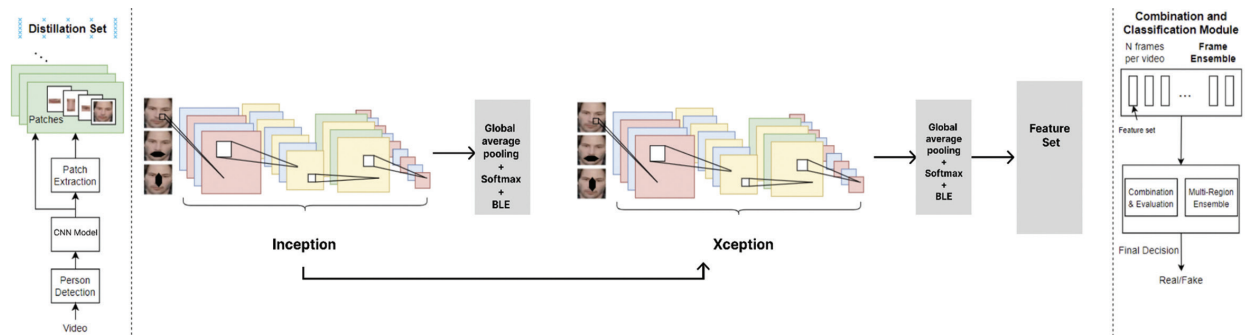


Fig. 7. Architecture of the proposed method

5. RESULTS

The method described in this paper is a way to identify deepfake pictures by utilizing the sign of the source to highlight irregularities inside the manufactured pictures. It is based on the theory that images distinguished by source features can be protected and removed after going through best-in-class deep-learning processes. The work presented here presents a smart

portrayal of the learning approach, known as pairwise self-consistency learning (PCL), used for preparing convolutional networks to separate the origin highlights and distinguish bogus pictures. It is combined with an irregularity picture generator (I2G) method to produce clear information for PCL. The ROC curve for the proposed work is shown in Fig. 8. It is a plot of the true positive rate as a function of the false positive rate for various cut-off points of a parameter.

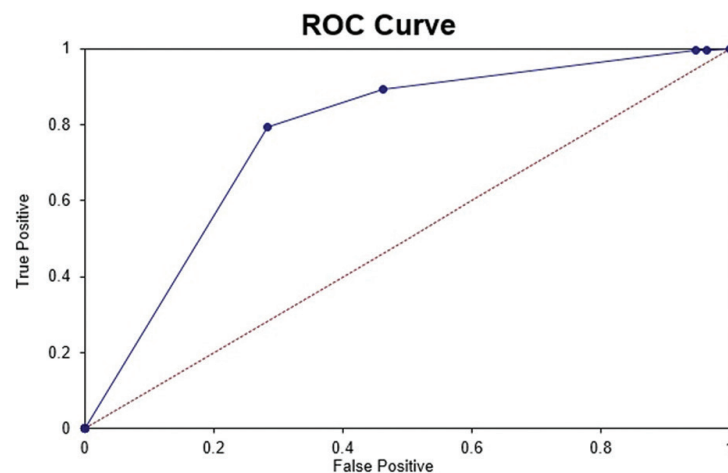


Fig. 8. ROC Curve for Training and Validation

Exploratory outcomes on Inception_resnet_v2, Xception, and hybrid are tabulated below. The evaluation metrics of the Inception, Xception, and Hybrid models are shown in Tables 3(a), 3(b), and 3(c), respectively. The hybrid model received a training accuracy of 0.98 and a validation accuracy of 0.93, respectively.

Table 3. Performance of Base and Hybrid Model

(a) Evaluation metrics for the Inception

	Inception			
	precision	recall	f1-score	support
0(FAKE)	0.98	0.97	0.97	712
1(REAL)	0.88	0.91	0.89	185
Accuracy			0.96	897
Macro avg	0.93	0.94	0.93	897
Weighted avg	0.96	0.96	0.96	897

(b) Performance of Inception Model

	Xception			
	precision	recall	f1-score	support
0(FAKE)	0.98	0.99	0.98	712
1(REAL)	0.97	0.91	0.94	185
Accuracy			0.98	897
Macro avg	0.97	0.95	0.96	897
Weighted avg	0.98	0.98	0.98	897

(c) Performance of Hybrid Model

	Hybrid			
	precision	recall	f1-score	support
0(FAKE)	0.98	1.00	0.99	1486
1(REAL)	0.99	0.92	0.96	451
Accuracy			0.98	1937
Macro avg	0.99	0.96	0.97	1937
Weighted avg	0.98	0.98	0.98	1937

6. CONCLUSION

This approach uses a CNN-based model to uncover the bogus clips. The model performed well on the DFDC dataset, including low- and high-quality movies. The outputs of tampered videos highlight that by adopting a hybrid network of Inception Resnet v2 and Xception, it can identify whether a clip has ever been deceived. This work is an effective first line of defense in detecting bogus media made with online technologies. In addition, the model can attain competitive output by adopting a pipeline design, which is also demonstrated. In the future, we can use subtle tactics during training to see how we can strengthen the system against false accusations. The experimental analysis demonstrates that the enhancements have greatly improved deepfake detection results, with maximum precision, recall, and f1-score of 0.98, respectively. Simultaneously, because video forgery technology and the caliber of video are still developing, it will be possible to facilitate the proposed model.

7. REFERENCES

- [1] S. Senhadji, R. A. San Ahmed, "Fake news detection using naïve Bayes and long short term memory algorithms", *IAES International Journal of Artificial Intelligence*, Vol. 11, No. 2, 2022, pp. 748-754.
- [2] K. N. Ramadhani, R. Munir, "A Comparative Study of Deepfake Video Detection Method", *Proceedings of the 3rd International Conference on Information and Communications Technology*, November 2020, pp. 394-399.
- [3] D. Pan, L. Sun, R. Wang, X. Zhang, R. O. Sinnott, "Deepfake Detection through Deep Learning", *Proceedings of the IEEE/ACM International Conference on Big Data Computing, Applications and Technologies*, December 2020, pp. 134-143.
- [4] A. A. Maksutov, V. O. Morozov, A. A. Lavrenov, A. S. Smirnov, "Methods of deepfake detection based on machine learning", *Proceedings of the IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering*, January 2020, pp. 408-411.
- [5] T. T. Nguyen, Q. V. Nguyen, D. T. Nguyen, "Deep learning for deep fakes creation and detection: A survey", *Computer Vision and Image Understanding*, Vol. 223, 2022, pp. 1-19.
- [6] A. O. Kwok, S. G. Koh, "Deepfake: A Social Construction of Technology Perspective", *Current Issues in Tourism*, Vol. 24, No. 13, 2020, pp. 1798-1802.
- [7] M. Westerlund, "The Emergence of Deepfake Technology: A Review", *Technology Innovation Management Review*, Vol. 9, No. 11, 2019, pp. 40-53.
- [8] Y. Li, S. Lyu, "Exposing Deepfake Videos by Detecting Face Warping Artifacts", arXiv:1811.00656, 2018.
- [9] A. M. Almars, "Deepfakes detection techniques using deep learning: a survey", *Journal of Computer and Communications*, Vol. 9, No. 5, 2021, pp. 20-35.
- [10] N. S. Ivanov, Arzhakova, V. G. Ivanenko, "Combining deep learning and super-resolution algorithms for deep fake detection", *Proceedings of the IEEE Conference of Russian Young Researchers in Electrical and Electronic Engineering*, January 2020, pp. 326-328.
- [11] K. Zhu, B. Wu, B. Wang, "Deepfake detection with clustering-based embedding regularization", *Proceedings of the IEEE fifth International Conference on Data Science in Cyberspace*, July 2020, pp. 257-264.
- [12] F. Matern, C. Riess, M. Stamminger, "Exploiting visual artifacts to expose deep fakes and face manipulations", *Proceedings of the IEEE Winter Applications of Computer Vision Workshops*, January 2019, pp. 83-92.
- [13] E. Sabir, J. Cheng, A. Jaiswal, W. AbdElmageed, I. Masi, P. Natarajan, "Recurrent convolutional strategies for face manipulation detection in videos", *Interfaces (GUI)*, Vol. 3, No. 1, 2019, pp. 80-87.
- [14] D. Güera, E. J. Delp, "Deepfake video detection using recurrent neural networks", *Proceedings of the 15th IEEE International Conference on Advanced Video and Signal Based Surveillance*, November 2018, pp. 1-6.
- [15] Y. Li, S. Lyu, "Exposing deepfake videos by detecting face warping artifacts", arXiv:1811.00656, 2018.
- [16] Y. Li, M. C. Chang, S. Lyu, "In ictu oculi: Exposing ai created fake videos by detecting eye blinking", *Proceedings of the IEEE International Workshop on Information Forensics and Security*, December 2018, pp. 1-7.

- [17] D. Afchar, V. Nozick, J. Yamagishi, I. Echizen. Me-sonet, "A compact facial video forgery detection network", Proceedings of the IEEE International Workshop on Information Forensics and Security, December 2018, pp. 1-7.
- [18] P. Zhou, X. Han, Morariu, L. S. Davis, "Two-stream neural networks for tampered face detection", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, July 2017, pp. 1831-1839.
- [19] B. Schölkopf, J. C. Platt, J. Shawe-Taylor, A. J. Smola, R. C. Williamson, "Estimating the support of a high-dimensional distribution", Neural Computation, Vol. 13, No. 7, 2001, pp. 1443-71.
- [20] H. Khalid, S. S. Woo, "OC-FakeDect: Classifying deepfakes using one-class variational autoencoder", Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops, 2020, pp. 656-657.
- [21] M. F. Ahmed, M. S. Miah, A. Bhowmik, J. B. Sulaiman, "Awareness to Deepfake: A resistance mechanism to Deepfake", International Congress of Advanced Technology and Engineering, July 2021, pp. 1-5.
- [22] U. A. Ciftci, I. Demir, L. Yin Fakecatcher, "Detection of synthetic portrait videos using biological signals", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020, pp. 1-17.
- [23] P. Oza, V. M. Patel, "One-class convolutional neural network", IEEE Signal Processing Letters, Vol. 26, No. 2, 2018, pp. 277-81.
- [24] D. Cozzolino, J. Thies, A. Rössler, C. Riess, M. Nießner, L. Verdoliva, "Forensictransfer: Weakly-supervised domain adaptation for forgery detection", arXiv:1812.02510, 2018.
- [25] H. H. Nguyen, F. Fang, J. Yamagishi, I. Echizen, "Multi-task learning for detecting and segmenting manipulated facial images and videos", Proceedings of the 10th International Conference on Biometrics Theory, Applications and Systems, September 2019, pp. 1-8.
- [26] H. H. Nguyen, J. Yamagishi, I. Echizen, "Capsule-forensics: Using capsule networks to detect forged images and videos", Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing, May 2019, pp. 2307-2311.
- [27] A. Kherraki, R. El Ouazzani, "Deep convolutional neural networks architecture for an efficient emergency vehicle classification in real-time traffic monitoring", IAES International Journal of Artificial Intelligence, Vol. 11, No. 1, 2022, pp. 110-120.
- [28] S. H. Tsang, "Review: Xception-with depthwise separable convolution, better than inception-v3 (image classification)", <https://towardsdatascience.com/review-xception-with-depthwise-separable-convolution-better-than-inception-v3-image-dc967dd42568> (accessed: 2022)

Feature Selection Model using Naive Bayes ML Algorithm for WSN Intrusion Detection System

Original Scientific Paper

Deepa Jeevaraj

Department of ECE, Bharath Institute of Higher Education and Research, India
jdeepainbox@gmail.com

B. Karthik

Department of ECE, Bharath Institute of Higher Education and Research, India.
karthikguru33@gmail.com*, karthik.ece@bharathuniv.ac.in

T. Vijayan

Department of ECE, Bharath Institute of Higher Education and Research, India.
tvij16@gmail.com

M. Sriram

Department of CSE, Bharath Institute of Higher Education and Research, India.
msr1sriram@gmail.com

Abstract – Intrusion detection models using machine-learning algorithms are used for intrusion prediction and prevention purposes. Wireless sensor network has a possibility of being attacked by various kinds of threats that will de-promise the performance of any network. These WSN are also affected by the sensor networks that send wrong information because of some environmental causes in-built disturbances misaligned management of the sensors in creating intrusion to the wireless sensor networks. Even though signified routing protocols cannot assure the required security in wireless sensor networks. The idea system provides a key solution for this kind of problem that arises in the network and predicts the abnormal behavior of the sensor nodes as well. But built model by the proposed system various approaches in detecting these kinds of intrusions in any wireless sensor networks in the past few years. The proposed system methodology gives a phenomenon control over the wireless sensor network in detecting the inclusions in its early stages itself. The Data set pre-processing is done by a method of applying the minimum number of features for intrusion detection systems using a machine learning algorithm. The main scope of this article is to improve the prediction of intrusion in a wireless sensor network using AI-based algorithms. This also includes the finest feature selection methodologies to increase the performance of the built model using the selected classifier, which is the Bayes category algorithm. Performance accuracy in the prediction of different attacks in wireless sensor networks is attained at nearly 95.8% for six selected attributes, a Precision level of 0.958, and the receiver operating characteristics or the area under the curve is equal to 0.989.

Keywords: IDS, WSN, Machine learning, ROC, Precision, Naïve Bayes

1. INTRODUCTION

Trending widespread application of wireless sensor networks in getting the solution for intrusions cum detection for these networks has become a challenging task nowadays. There is an urgent need to develop a system that detects intrusions, malicious node that breaks down the wireless sensor networks [1-3]. Anomaly-based intrusion detection systems are the trending demand in this widespread application. According to the behavior of some effective detection methods using a machine-learning algorithm in this article. The main concern of this article is to build a model using the Bayes category ML algorithm to predict and prevent various types of intrusion attacks that create a breakdown of wireless sensor network applications [4-8]. This is deployed by monitoring various parameters of the wireless sensor Network and the output based on their weights

and concentrations and energy consumed at various nodes. The built model is used to identify the intrusions that create attacks on the WSN as well as increase energy consumption or loss of energy consumption. The model built provides a higher rate of intrusion detection rate and reduces the loss of energy.

WSN is used by Defence Services, biotelemetry health care, automation Industries so on. The physical attributes [9] where human activity finds it difficult to supervise these wireless sensor networks the sensor nodes that are deployed in specific areas. Nodes transmit and receive data continuously through the base stations. There are many issues that come in contact with these wireless sensor networks including the attacks and energy consumption in not identifying the malicious node. The specific protocol used for routing, their efficiency in energy consumption the cluster head

selection, and the Novelty of the wireless sensor network, etc [9-11]. Trending constraints in this article is to build a model to optimize and detect all the intrusion cum attacks created to these wireless sensor networks using a machine learning Framework. Figure 1 shows the sensor nodes that monitor all the physical parameters like temperature, pollution, and Connected devices. The data collected all are synced through the internet and get the information from the nodes is shown in Fig.1.

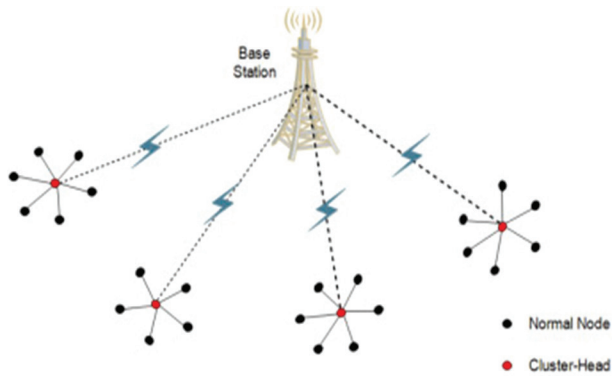


Fig. 1. WSN architecture with a base station

Organization of the paper: The article is composed of five sections such as the related works that support greatly in including the machine learning concepts in the research article. Secondly the data type and its description of all the dataset parameters. The next section is about the materials and the experiment methodology handled in the research proposal. The next section is the experimental investigations and inferences discussed. The next section is the conclusion part, the future scope, and enhancements.

2. RELATED WORKS

In [12] authors presented the present day Scenario integrated internet using a wireless sensor network is presently having a great impact on today's life. The privacy and security of a network in preventing and detecting intrusion in WSN is a challenging issue. The different types of threats which are prominent and very hard to detect the device attacks are taken into the act. Protection in communication connected with wireless sensor networks used encryption-based techniques traditionally. Which has proved to be inefficient in recent days. This proposal has given an intrusion detection cum protection of WSN using a new direction towards internet integrated wireless sensor network.

In [13] authors prescribe Wireless sensor network comes in contact with compact size and inexpensive sensor nodes. The place of usage of WSN with a sensor undergoes arbitrary Placement in open areas. In this kind of situation, there is a higher rate of attacks. The innovative idea behind the intrusion detection system in this proposal is building a model using machine learning algorithms like support vector machines to detect

intrusion in the wireless sensor network. The result of this portrayed high results of accuracy nearly 94.09% and a detection rate of 95%.

In [14] authors monitor wireless sensor network has a wide range of applications in the environment, health, military, industries, etc. WSN has Limited source and energy concerns. A challenging task that is designed in such a way that it utilizes minimum energy consumption and gives a maximum lifetime of the network. In most of the Daniel of service attacks that destroys the network and loss its energy rapidly is identified using a novel approach. An efficient intrusion detection system or scheme is designed in such a way that malicious node is identified with very little energy conservation. All the nodes are continuously monitored whose energy consumption is monitored and by comparing the actual and the predicted energy the malicious node is identified. This malicious node is identified by using a bayesian approach of a machine learning algorithm.

In [15] authors propose WSN is a key object in any cyber-physical system. It is composed of many stationary as well as mobile parts like sensors that transmit and receives information through WSN. The intrusion that affects the WSN has to prevent using a special mechanism in a smart environment. The novel approach is a sequence backward selection algorithm that detects the attacks at a faster rate. The experiment results based on this approach have given an efficient F-measure of 0.96 0.99 for all kinds of network attacks.

In [16] authors present a new protocol developed in WSN integrated into IoT deployment. In today's activities, every common man has an advancement in information communication and Technology ICT. Advancement is also suffering from various attacks that occur in WSN and IoT. Because of trending progress in this fast-moving environment and more vulnerable security threats. In the future, everyone is connected to the internet with numerous smart objects and for a smooth progression there is a need for IDS and IPS. This article gives an emerging intrusion detection system with a new approach. A privacy preservation protocol is integrated with WSN and IoT to address the intrusion detection Protocol in wireless sensor networks that is integrated into the internet-of-things.

In [17] authors prescribe Network security as an unavoidable event in our daily interactions and networks. Intrusions are also developing more and more critical as Technology also grows. Techniques employed using machine learning algorithms to detect intrusions. However, there is an advantage of deep learning algorithms and AI to generate special features that automatically detect attacks without any human intervention. Long short term memory network with spatial features is employed to detect a hybrid intrusion detection system with a model that is built using this deep learning methodology. The investigational report specifies high accuracy, Precision, and detection rate as very high and effective.

In [18] authors propose a secured energy-efficient barrier coverage schedule that has been developed using a machine learning algorithm to maintain the quality of service. A barrier coverage schedule is also energy conserving scheme. In spite of a wide range of areas called the barriers and a subset of sensor nodes overlapped to meet all the quality of service requirements. Expected node failures due to barrier security attacks such as Daniel of service is a challenging in maintaining the quality of service levels. A smart proposal using a machine learning algorithm is proposed to detect The Attacks in an efficient way. WSN-based IoT applications that utilize kNN machine learning algorithms to detect malicious attacks.

In [19,20] authors present the Wireless sensor network as one of the third Millennium technology that had a wide range of applications in the surrounding medium or environment. The main reason for the application of WSN application is the low production cost, the installation, unattended operations, anonymous and longtime operations that occur. WSN integrated with IoT in sensor nodes and sensing ability using internet-connected devices is a recent advancement taking place in WSN. The absence of physical in-line security defense gateways that comes in contact with network security with IoT is a big concern to the scientific community. A novel technique for the prevention, detection, and mitigation of all the attacks is proposed in this article. Recent integration and collaboration of WSN and IoT are facing open challenges in terms of security. A system should be developed with security administrators and network managers to predict all the threats and attacks to detect the malicious nodes Machine learning tool is a powerful tool in predicting the intrusion caused in a WSN in less time. However, the prediction is accurate with the only parameter being the perfect dataset with the required attributes. So that the trained model will very well perform in predicting the intrusions timely.

3. DATA DESCRIPTION

The data set WSN is collected from public platforms like kaggle.com [21]. Table 1 shows the description of the data set which consists of nearly three lakh seventy-four thousand and six hundred and sixty-two instances with 18 attributes. The attributes are namely the ID the channel is present or absent, the Signal strength indicator, the average distance of the channel and the energy consumption of the nodes, and the number of messages and advertisers that are received from the nodes.

That acknowledgment of the number of advertisers using time division multiplexing broadcast messages to the nodes. Data is transmitted and received from the nodes. The packet is sent to the base stations and the distance between the channel and the base station. The code is finally in the cluster where there are nearly five output classes that designate the type of attack in the WSN undergone.

Table 1. Experimental dataset for WSN attack prediction

S.No	Attributes	Instances count	Class description of Level
1	18	374662	5, TDMA, Black hole, Flooding, Grey hole, and Normal

The selected machine learning algorithm for this article is a Naive Bayes classifier. The basic principle of this algorithm is it works on the probability of the events X and Y. It also comprises some text sentiments and opinions for processing. The Bayes Theorem basically gives the hypothesis with the prior acquired knowledge available from the previous experiences as given in the formula (1).

$$P\left(\frac{X}{Y}\right) = \frac{P\left(\frac{Y}{X}\right)P(X)}{P(Y)} \quad (1)$$

Where P(X) and P(Y) are the probability of events X and Y.

4. METHODOLOGY

The experiment proceeds in the following ways as shown in Fig. 2 in attaining the optimum model.

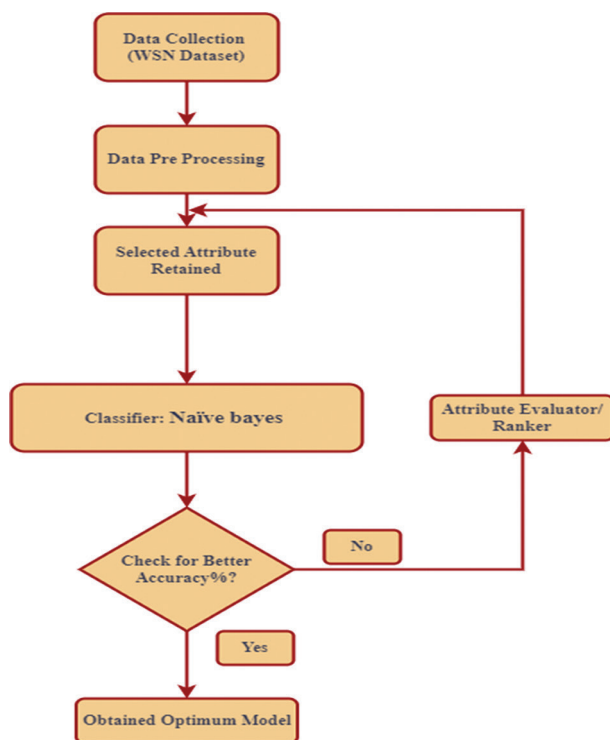


Fig. 2. workflow diagram of WSN attack prediction

The work show illustrates the data collection in the initial state from a public platform like kaggle.com. The data format is changed according to the usable format in the attribute relation file format. Once it is accessible in a tool used like Weka 3.8.5 [22] The selected attributes are retained for classification purposes nearly 18 attributes are retained for the first classification procedures. After the classification by done using the select-

ed machine learning algorithm called Naive Bayes. The next step is to check for the accuracy percentage of the f-measure, the receiver operating characteristics, and the Precision tabulated. The procedure is the full data set process tree process some of the attributes according to the information present in the attributes using and ranker attribute in the Weka tool. Once the ranks are obtained according to the information gain some of the attributes with high again and go with the classification process. Again the procedure is repeated for selected attributes and removing the attributes with less information.

The final optimum model is then concluded with maximum accuracy, precision, and receiver operator characteristics. The investigation was supported by giv-

ing maximum efficiency for the build model and output accuracy with nearly 95.854% with 6 attributes itself and yielding maximum area under the curve of 0.979.

5. EXPERIMENTS RESULTS AND DISCUSSION

The experimental results are as follows: The attribute accessor search methodology is applied. The retained attributes are classified whose accuracy, precision, Recall, F-Measure, and receiver operator characteristics are analyzed and tabulated. Nearly five different types of attribute are selected which is classified from retaining all the attributes to removing the attributes [9] according to their rank. Table 1 Investigation results after classifying all attributes to retained attributes.

Table 1. Experimental Results of Attribute Selection method

S.NO	Attribute Selection	Attribute selection session (Attribute assessor/Search method)	Classifier	Accuracy	Precision	Recall	F-Measure	ROC
1.	Present all attributes(18), ADV_S, Is_CH, Expanded Energy, DATA_S, Rank, send code, JOIN_S, Dist_To_CH, ADV_R, SCH_R, SCH_S, who CH, Data_Sent_To_BS, id, JOIN_R, dist_CH_To_BS, Time, DATA_R	InfoGain AttributeEval / Ranker	Naïve Bayes	95.3734 %	0.967	0.954	0.958	0.980
2.	selected attributes (15), ADV_S, Is_CH, Expanded Energy, DATA_S, Rank, send code, JOIN_S, Dist_To_CH, ADV_R, SCH_R, SCH_S, who CH, Data_Sent_To_BS, id, JOIN_R, (Removed last 3 attributes)	InfoGain AttributeEval / Ranker	Naïve Bayes	95.3216 %	0.965	0.953	0.957	0.980
3.	selected attributes (12), ADV_S, Is_CH, Expanded Energy, DATA_S, Rank, send code, JOIN_S, Dist_To_CH, ADV_R, SCH_R, SCH_S, who CH, (Removed last 6 attributes)	InfoGain AttributeEval / Ranker	Naïve Bayes	94.2433 %	0.958	0.942	0.949	0.983
4.	selected attributes (9), ADV_S, Is_CH, Expanded Energy, DATA_S, Rank, send code, JOIN_S, Dist_To_CH, ADV_R, (Removed last 9 attributes)	InfoGain AttributeEval / Ranker	Naïve Bayes	92.328 %	0.933	0.923	0.925	0.979
5.	selected attributes (6), ADV_S, Is_CH, Expanded Energy, DATA_S, Rank, send code, (Removed last 12 attributes)	InfoGain AttributeEval / Ranker	Naïve Bayes	95.8154 %	0.968	0.958	0.961	0.989

Table 1 shows the experimental results of the built model by applying the information gain cum attribute evaluator which ranks for attributes in the dataset. The ranked attributes with maximum information are retained attributes. 6th attribute ADV_S is ranked first according to the information gain. Attribute 3 is_CH to channel is present or not and the energy conservation

is the third attribute and so on. The Second attribute (time) and the fourteenth attribute data_R are attributes with very less information as per the attribute information evaluator.

Figure 3 shows the graph for the number of attributes retains to the percentage of accuracy performance. The percentage of accuracy is 95.82%stage for 6 attributes

itself. The performance accuracy was 95.37% stage for retaining all the attributes. Therefore, feature selection using the selected attributes gives A Remarkable performance using this information gain evaluator.

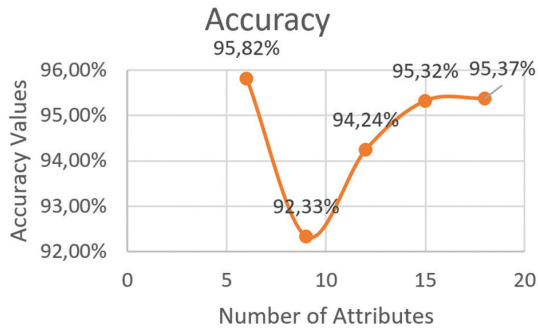


Fig. 3. No of the retained attribute to % of Accuracy

Attribute versus Precision characteristics clearly gives the built model has given good precise values for retaining 6 attributes out of 18 as shown in Fig. 4. The precision value for retaining all 18 attributes has given 0.967 and for retaining six different attributes as given 0.968. Good precision value characteristics for using the feature selection option in the Weka tool.

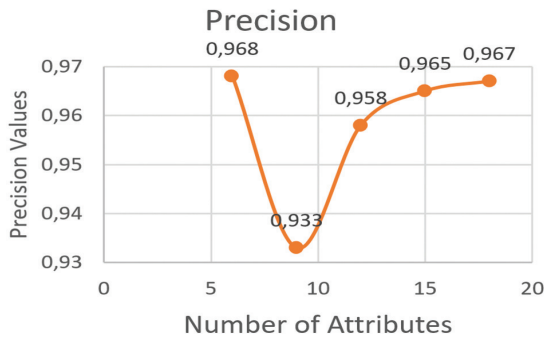


Fig. 4. Attributes versus Precision

The retained attribute versus recall characteristics also gives a good performance by attaining a value of 0.954 for 18 attributes as shown in Fig. 5. Just for six attributes the recall value of 0.958 using the information gain evaluator.

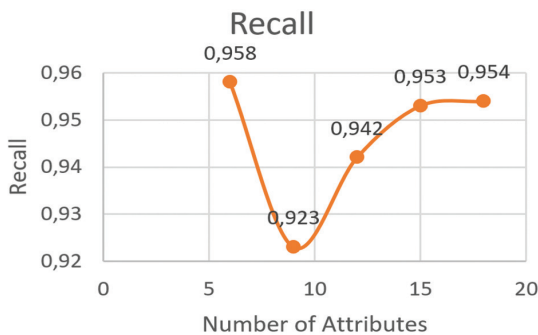


Fig. 5. Attributes versus Recall

The consolidated measure of both Precision and recall is done using F-measure. The attribute versus F-measure

characteristics are shown in Fig. 6. The consolidated effect of both Precision and recall values of the built model. This specifies the model classifying the output has given a remarkable output for retaining 6 attributes.

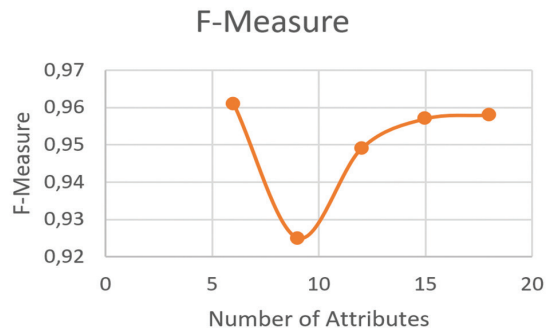


Fig. 6. Attributes versus F -Measure

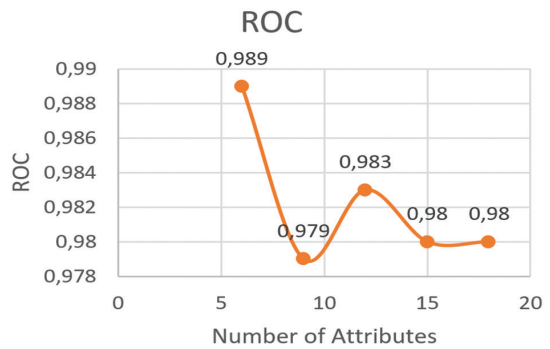


Fig. 7. Attributes versus ROC

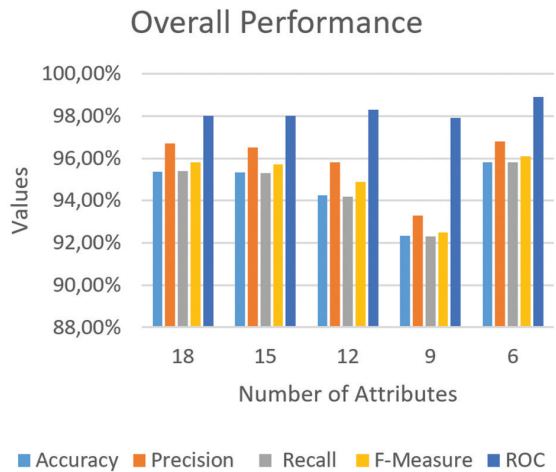


Fig. 8. Overall performance characteristics of a built model

The overall performance characteristics of the built model using the Naive Bayes algorithm are shown in Fig. 8. The characteristics clearly mark that the built model gives a remarkable output in classifying the output intrusion in WSN. The difference between retaining all the attributes and retaining six attributes. The performance characteristics support predicting the attacks in a WSN in less time and preventing the WSN from intrusions. The feature selection method using an information gain evaluator really works in an efficient manner.

6. CONCLUSION AND FUTURE SCOPE

In the first iteration, the accuracy percentage with maximum performance was attained to be 95.37 percent with eighteen attributes. The result obtained with six attributes was 95.82 percent using feature selection methods. These findings are the first of their type in this structure for intrusion prediction utilizing the WSN dataset that is based on real-time data acquisition. In the future, this can be implemented in finding the intrusion cum preventing system in WSN with deep learning methodologies. The future scope of the proposed methodology is a fast-growing field. The intrusion detection in WSN faces greater demand in the future and the proposed method can be established in deep learning technique in meeting the above-mentioned demand in the future.

7. REFERENCES

- [1] L. Zhiqiang, G. Mohiuddin, Z. Jiangbin, M. Asim, W. Sifei, "Intrusion detection in wireless sensor network using enhanced empirical based component analysis", *Future Generation Computer Systems*, Vol. 135, 2022, pp. 181-193.
- [2] G. Creech, J. Hu, "A Semantic Approach to Host-Based Intrusion Detection Systems Using Contiguous and Discontiguous System Call Patterns", *IEEE Transactions on Computers*, Vol. 63, 2014, pp. 807-819.
- [3] L. Vokorokos, A. Baláz, "Host-Based Intrusion Detection System", *Proceedings of the IEEE 14th International Conference on Intelligent Engineering Systems*, Las Palmas, Spain, 5-7 May 2010, pp. 43-47
- [4] A. H. Farooqi, F. A. Khan, "A Survey of Intrusion Detection Systems for Wireless Sensor Networks", *International Journal of Ad Hoc and Ubiquitous Computing*, Vol. 9, 2012, pp. 69-83.
- [5] G. Liu, H. Zhao, F. Fan, G. Liu, Q. Xu, S. Nazir, "An Enhanced Intrusion Detection Model Based on Improved kNN in WSNs", *Sensors*, Vol. 22, 2022, p. 1407.
- [6] Y. Canbay, S. Sagiroglu, "A hybrid method for intrusion detection", *Proceedings of the IEEE 14th International Conference on Machine Learning and Applications*, Miami, FL, USA, 9-11 December 2015, pp. 156-161.
- [7] S. Mohammadi, H. Mirvaziri, M. Ghazizadeh-Ahsaei, H. Karimipour, "Cyber intrusion detection by combined feature selection algorithm", *Journal of Information Security and Applications*, Vol. 44, 2017, pp. 80-88.
- [8] M. S. Koli, M. K. Chavan, "An Advanced Method for Detection of Botnet Traffic using Intrusion Detection System", *Proceedings of the IEEE International Conference on Inventive Communication and Computational Technologies*, Coimbatore, India, 10-11 March 2017, pp. 481-485.
- [9] T. Vijayan, M. Sangeetha, A. Kumaravel, B. Karthik, "Feature selection for Simple Color Histogram Filter based on Retinal Fundus Images for Diabetic Retinopathy recognition", *IETE Journal of Research*, 2020.
- [10] Y. Sun, F. Liu, "SMOTE-NCL: A Re-Sampling Method with Filter for Network Intrusion Detection", *Proceedings of the IEEE International Conference on Computer and Communications*, Chengdu, 14-17 October 2016, pp. 1157-1161.
- [11] H. Elbahadır, E. Erdem, "Modeling Intrusion Detection System Using Machine Learning Algorithms in Wireless Sensor Networks", *Proceedings of the 6th International Conference on Computer Science and Engineering*, Ankara, Turkey, 15-17 September 2021, pp. 401-406.
- [12] B. J. S. Kumar, S. Sinha, "An Intrusion Detection and Prevention System against DOS Attacks for Internet-Integrated WSN", *Proceedings of the 7th International Conference on Communication and Electronics Systems*, Coimbatore, India, 22-24 June 2022, pp. 793-797.
- [13] S. Amaran, R. M. Mohan, "Intrusion Detection System using Optimal Support Vector Machine for Wireless Sensor Networks", *Proceedings of the International Conference on Artificial Intelligence and Smart Systems*, 2021, pp. 1100-1104.
- [14] S. S. Shivaji, A. B. Patil, "Energy Efficient Intrusion Detection Scheme Based on Bayesian Energy Prediction in WSN", *Proceedings of the Fifth International Conference on Advances in Computing and Communications*, Kochi, India, 2-4 September 2015, pp. 114-117.
- [15] S. Jiang, J. Zhao, X. Xu, "SLGBM: An Intrusion Detection Mechanism for Wireless Sensor Networks in Smart Environments", *IEEE Access*, Vol. 8, 2020, pp. 169548-169558.

- [16] S. Pundir, M. Wazid, D. P. Singh, A. K. Das, J. J. P. C. Rodrigues, Y. Park, "Intrusion Detection Protocols in Wireless Sensor Networks Integrated to Internet of Things Deployment: Survey and Future Challenges", *IEEE Access*, Vol. 8, 2020, pp. 3343-3363.
- [17] Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, R. Ahmad, "CNN-LSTM: Hybrid Deep Neural Network for Network Intrusion Detection System", *IEEE Access*, Vol. 10, 2022, pp. 99837-99849.
- [18] D. Thomas, R. Shankaran, M. A. Orgun, S. C. Mukhopadhyay, "SEC2: A Secure and Energy Efficient Barrier Coverage Scheduling for WSN-Based IoT Applications", *IEEE Transactions on Green Communications and Networking*, Vol. 5, No. 2, 2021, pp. 622-634.
- [19] Butun, P. Österberg and H. Song, "Security of the Internet of Things: Vulnerabilities, Attacks, and Countermeasures", *IEEE Communications Surveys & Tutorials*, Vol. 22, No. 1, 2020, pp. 616-644.
- [20] D. Hemanand, G. V. Reddy, S. S. Babu, K. R. Balmuri, T. Chitra, S. Gopalakrishnan, "An intelligent intrusion detection and classification system using CSGO-LSVM model for wireless sensor networks (WSNS)", *International Journal of Intelligent Systems and Applications in Engineering*, Vol.10, No. 3, 2022, pp. 285-293.
- [21] Kaggle, <https://www.kaggle.com/datasets/bas-samkasasbeh/5-1-wsnds> (accessed: 2022)
- [22] Weka, <https://www.cs.waikato.ac.nz/ml/weka/> (accessed: 2022)

Ensemble Deep Learning Network Model for Dropout Prediction in MOOCs

Original Scientific Paper

Gaurav Kumar

Lovely Professional University,
Jalandhar-Delhi, G.T. Road, Phagwara,
Punjab, India -144411.
gaurav.20323@lpu.co.in

Amar Singh

Lovely Professional University,
Jalandhar-Delhi, G.T. Road, Phagwara,
Punjab, India -144411.
amar.23318@lpu.co.in

Ashok Sharma

University of Jammu, Bhatnagar Campus,
Jammu and Kashmir -180006.
ashoksharma@jammuuniversity.ac.in

Abstract – In the online education field, Massive open online courses (MOOCs) have become popular in recent years. Educational institutions and Universities provide a variety of specialized online courses that helps the students to adapt with various needs and learning preferences. Because of this, institutional repositories creates and preserve a lot of data about students' demographics, behavioral trends, and academic achievement every day. Moreover, a significant problem impeding their future advancement is the high dropout rate. For solving this problem, the dropout rate is predicted by proposing an Ensemble Deep Learning Network (EDLN) model depending on the behavior data characteristics of learners. The local features are extracted by using ResNet-50 and then a kernel strategy is used for building feature relations. After feature extraction, the high-dimensional vector features are sent to a Faster RCNN for obtaining the vector representation that incorporates time series data. Then an attention weight is obtained for each dimension by applying a static attention mechanism to the vector. Extensive experiments on a public data set have shown that the proposed model can achieve comparable results with other dropout prediction methods in terms of precision, recall, F1 score, and accuracy.

Keywords: Deep learning; MOOC; feature extraction; dropout prediction; activity patterns.

1. INTRODUCTION

With the assistance of big data technology and artificial intelligence, an innovative and rapidly growing educational strategy is MOOCs [1]. Through online courses, MOOCs connect participants in global education and give students, instructors, and academic institutions access to an interactive Internet platform [2]. MOOCs now have a significantly larger student population, particularly in the current pandemic with their affordability and convenient features [3]. The high dropout rate currently in place, however, is severely impeding the growth of MOOCs. According to numerous research, less than 10% of MOOC courses are completed [4]. Only 7% of students finish the University of California's courses offered on the Coursera platform, according to statistical data [5]. Significant possibili-

ties for early reversal of the alarming student dropout and higher retention rates are predicted by the MOOC dropout prediction models [6]. These predictions are used to keep students motivated to learn and stop students from dropping out of course instructors through interventions [7].

Depending on the current learning behavior of the students, the chances of course dropout are examined by the MOOC dropout prediction [8]. For MOOC dropout prediction, traditional deep learning and machine learning methods are currently used [9] [10]. Most machine learning-based classification techniques are used in traditional machine learning research [11]. A large amount of time and effort must be expended manually for extracting features [12] [13]. Moreover, the lack of large-scale datasets for training these tech-

niques restricts their application in the MOOC present context [14]]. Higher predictive results are produced by deep learning models than the traditional machine learning models [15] [16]. The feature information is automatically extracted from input data by using the convolutional neural network (CNN), which is the most popular current dropout prediction model. However, it is unable to utilize the data from the time series [17]. The dropout prediction is effectively improved by using Faster RCNN models in certain researchers, and the time information is also captured by this network [18]. Furthermore, several recent research discovered that various characteristics should be handled differently because they have various consequences on the decision to drop out. So, to accomplish this concept, attention becomes a useful focus [19] [20].

The innovative MOOC dropout prediction model is proposed in this research based on previous research. The proposed model is called Ensemble deep learning network (EDLN) model. Faster RCNN and attention mechanisms are integrated with this proposed model. Automatic local feature extraction from the source data is done by the proposed model. Then these features are combined with time series information and predicted by multiplying the combined features by feature-wise weights. At the end of the course, in contrast to existing models, the proposed model's advantage is that it also predicts students' status. The learners' status is predicted by additionally fully exploiting the learner's key feature information and the learner's time series information during every week of the learning process. The essential information is provided by the proposed network model for instructors at risk of dropping out to select when and how to deliver personalized instruction to students.

The main contributions of the research are

- The input for the MOOC dropout model is a time series matrix in two dimensions. The original data's time series state is efficiently preserved while the weekly learner's input features are recorded in this matrix. During the course learning process, the learners' weekly status is predicted by this approach and makes timely interventions and it provides instructors intervention in time.
- The temporal relation between student behavior characteristics weekly is examined using the Faster RCNN. To weigh the characteristics, a static attention method is used by the significance of the behavioral characteristics. In dropout prediction, the effective features are extracted by using the ResNet-50. The efficiency of dropout prediction is effectively improved by the proposed model.
- Comparison experiments established the EDLN model's validity. While compared to the existing models, the proposed EDLN model predicts dropout effectively in the KDD CUP 2015 dataset.

2. LITERATURE REVIEW

This section, review some existing DL techniques for dropout prediction of MOOC learners.

A new feature extraction method is proposed by Jin et al [21] for behavior data of students for learning in this paper. The weekly characteristics of student learning behaviors are used for the experiment analysis. Then, the student dropout is predicted by developing the new support vector regression (SVR). An improved quantum particle swarm optimization (IQPSO) algorithm is used for optimizing the parameters in this paper.

A different integrated structure for MOOCs dropout prediction is proposed by Qiu et al [22]. A feature selection (FSPred) is proposed in this paper. Feature generation, feature selection, and dropout prediction are included in the proposed model. The features are generated by applying a fine-grained feature-generation method and then the valid features are selected by using the hybrid feature selection method. After the generation and selection of features, the logistic regression model is used for the dropout prediction.

A novel supervised ML algorithm is proposed by Panagiotakopoulos et al [23] to predict the dropout of students in MOOC. Six well-known metrics were used to evaluate several predictive models. The learning algorithm's performance is improved by using random search to automatically optimize the hyperparameter. The classification performance is further improved by applying stacked generalization approach was applied to further improve the classification performance.

The novel dropout prediction model is proposed by Xing et al [24]. The intervention personalization was examined for improving the effectiveness of the model in MOOCs. The dropout prediction model is constructed by developing the deep learning model in this research. After that individual student dropout probability is predicated on a temporal prediction mechanism. For at-risk students in MOOCs, individual dropout rates to personalize and prioritize intervention are examined.

In online short courses, a new methodology is examined by Chen et al [25] for dropout prediction of students. The creation of predictive learning analytics is complicated due to the limited enrollment in this course and the absence of intermediate assessments. Only behavior-based machine learning features that have been processed from measurements gathered throughout the learning process are used in this method.

A novel feature extraction method is done by Wan et al [26] for predicting the effectiveness of the students. Then, a model for transfer learning based on TrAdaBoost was proposed. It was applied to the current course iteration's pre-trained model using the data from the previous iteration of the course. In addition, this research contrasted how latecomers changed their learning behavior between the controlled group and the experimental group.

Deep learning is used for increasing the model's performance from the investigation of the above studies across many fields. A MOOC dropout prediction model based on this study's concept to combine the static attention mechanism with Faster RCNN is presented. By assigning the extracted features weights based on static attention, the model's accuracy is increased while identifying important features.

3. PROPOSED METHODOLOGY

3.1. PROBLEM STATEMENT

For five weeks, the analysis was done on the students' records for this research. Whether the learners dropped out is accessed by using the five-week activity records. If there were ten consecutive days without any learning activities, students were classified as dropouts; otherwise, they were classified as non-dropouts. Problems with categorical prediction were established from the dropout problems in this research, with those who had not dropped out represented by 1 and who had dropped out represented by 0.

3.2. PROPOSED EDLN MODEL

The Faster RCNN and static attention mechanisms are combined in the MOOC dropout prediction model based on EDLN. First, the original data is sent to the model as a two-dimensional temporal matrix. For the behaviors of the learners, the two-dimensional convolu-

tion kernel of ResNet-50 is used to extract the local high-dimensional feature information automatically. Then the computational load of the model and dimension of invalid features is reduced by adding a max pooling layer.

Using the local feature data, the time series' hidden long memory features are then retrieved by Faster RCNN and it uses a time series encoding algorithm to encode the data. The feature information is assigned by weight using a static attention mechanism. The key feature information is also highlighted by the static attention method and also the model's effectiveness is further enhanced by this mechanism. Finally, a sigmoid function representing the results of the MOOC dropout classification prediction is output. Fig. 1 shows the structure of the EDLN model. 1 fully connected layer, 7 pooling layers, 7 convolutional layers, 1 RCNN layer, and 1 Static Attention layer are presented in the proposed detection network model.

An EDLN-based model for predicting MOOC dropouts is proposed in this research. In Fig. 2, the model-based prediction process is described. Preprocessing of data, prediction, and evaluation of the model are the main three parts of the proposed model. The KDD 2015 dataset's clickstream data is first processed, and the weekly data on the behavioral characteristics of every student is used as the original data. The time series information and local feature learning of the source data is then automatically extracted and learned using the EDLN MOOC dropout prediction model. Finally, the model's performance was assessed using precision, recall, F1-score, and accuracy.

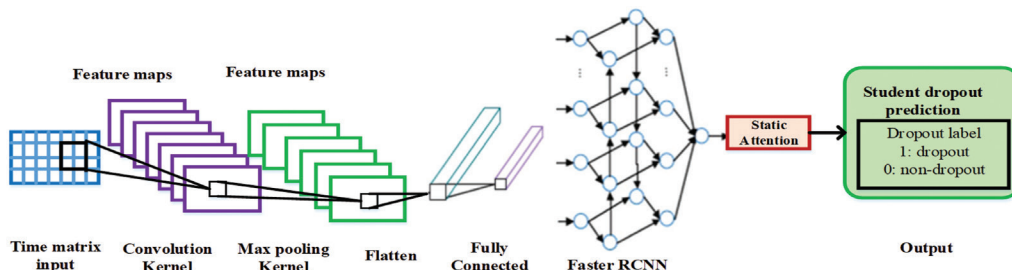


Fig.1. EDLN model structure.

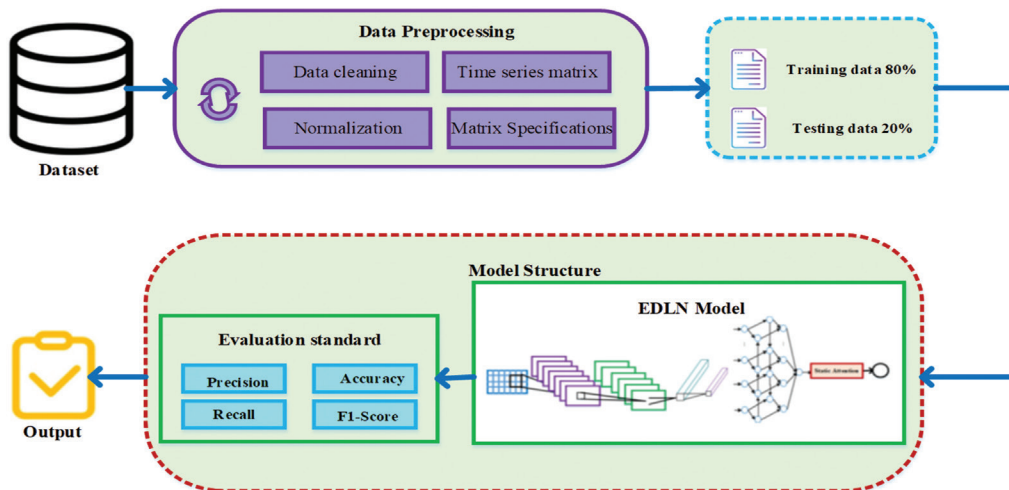


Fig. 2. Schematic diagram of the proposed methodology

3.2.1. Preprocessing

The clickstream data in this data set provides a behavior log that specifically records the course ID, student ID, occurrence time, and click event. In this research, the student data is first cleaned, and removed dropout labels from the dataset. From 12,000 students, the original data is selected randomly using student ID numbers during course learning which consisted of 7 different feature data types totaling 60,000 pieces during course learning. The multivariate time series data is presented in the characteristic behavior of

learner's datasets with many behavioral features like wiki, discussion, video, page close, traverse, access, and problem. Therefore, a two-dimensional temporal matrix is used to preprocess the data to properly utilize the time series and the information about hidden features between different behavioral variables. Data on a student's d behavioral characteristics over s weeks are contained in the time matrix, starting from week t . Each model input's time matrix is described in Eq. (1).

$$X_t = \begin{bmatrix} x_t^1 & x_t^2 & \dots & x_t^d \\ x_{t+1}^1 & x_{t+1}^2 & \dots & x_{t+1}^d \\ \dots & \dots & \dots & \dots \\ x_{t+s-1}^1 & x_{t+s-1}^2 & \dots & x_{t+s-1}^d \end{bmatrix} \quad (1)$$

The frequency of the d distinctive behaviors is shown for each row in the matrix for the corresponding week, according to the equation above.

For MOOCs with various temporal dimensions, this research generated five separate time series matrices with varying specifications as input to the EDLN model by successively segmenting the datasets and using the data from various weeks' models to examine the dropout prediction performance of the EDLN model. The week dataset utilized and each specification time series matrix are shown in Table 1.

Table 1. Specifications for time series matrices.

Datasets	Specifications for time series matrices
Week 1 data	1 x 7
Week 1-2 data	2 x 7
Week 1-3 data	3 x 7
Week 1-4 data	4 x 7
Week 1-5 data	5 x 7

The data on the characteristics of the student's behavior is divided into weeks in this experiment. $n \times 7$ -dimensional input matrices are generated from the combined weekly data, where $n = 1, 2, 3, 4, 5$. The EDLN MOOC dropout prediction model employed the normalized time series matrix as its input.

3.2.2. Feature extraction using ResNet-50 network

The process of feature extraction is carried out manually and it needs researchers with specialized knowledge. The labor-intensive and time-consuming process of manual feature extraction occurs due to the very low frequency of behavioral characteristics of students and very complex potential dropout patterns in the course. The ResNet-50 algorithm is used for solving the limitations of manual extraction. To extract local feature information, in which the number of input features is equal to the number of convolution kernels.

Full connection, pooling, and convolution are the three levels of a ResNet-50 network's neural architecture. In the convolutional layer, the time matrix's input features are used to calculate the k convolutional kernels. The convolution kernels are enabled for extracting each input feature's characteristic information for each dimension, and the convolution kernel's size is set at (u, v) , where Eq. (2) shows the convolution kernel calculation formula.

$$CF = q(w^T X^{(i+u-1, j+v-1)} + b) \quad (2)$$

Where the activation function is represented by q , how much behavioral feature data for learners overlap with the convolution kernel is denoted and the bias term is represented by b .

Convolutional and fully connected layers comprise the proposed ResNet-50 feature extraction network. There are trainable weights set for each layer. The time series matrix of size is the model's input. Using a kernel method, the combination of both behavioral dimension features (7) and temporal dimension features (n weeks) are extracted by the proposed EDLN model.

The convolutional layer is the first layer of the feature extraction network. After the last convolutional layer, a flattening operation is performed. Pooling, activation, and convolution are present in the convolutional layer. From the original data, only seven behavioral features were taken into consideration during the convolution process. The convolutional operations are performed with a 3×3 kernel. While focusing on the neighbors, using a smaller kernel navigates the input fields more than large kernels. It allows the model to fully utilize the limited information. Testing each layer with a range of 5 to 10 kernels, the different information is captured by employing the 7 distinct kernels finally for providing the highest performance.

After each convolution, a function of Rectified linear unit (Relu) is selected. We utilize a max-pooling technique in the pooling process. Then the feature maps in the original data are less distinct and close to zero. Because data on the behavior of the dropout students are typically 0 (meaning none). This observation leads to the process of max-pooling, which retrieves the greatest value. This highest value is better suitable for the proposed analysis. After that, the model performs the flattening operation of the generated feature maps.

The fully connected layer is the second element of the CNN module. The flattened convolutional outputs are represented more densely by the fully connected layer. For the next Faster RCNN module, these representations are used as the inputs.

However, learners' learning behavior features are automatically extracted by CNN. These features are in the form of time series data, and the significant time series correlation information range is present in the data. As a result, time series data cannot be extracted only with ResNet-50. Moreover, the time series relationship between the features of learners' behavior learners' time series behavior feature's relationship is extracted by using the Faster RCNN model in this research.

3.2.3. Dropout prediction using Faster RCNN

Faster RCNN is a development of CNN-based RCNN and fast RCNN networks. Several object detection processes are performed by this network. How regions are chosen for processing is the significant difference between them. A region selection algorithm is used by both RCNN and fast RCNN for object detection like Selective Search or Edge Boxes that are different from the CNN network. When training and detecting CNN, the region selection is performed by faster RCNN. The dataset related to this application is used to train the last fully connected layer.

The gradient disappearance problem is improved by developing the Faster RCNN model in recurrent neural networks (RNNs) caused by long input sequences. The input gates, output gates, forget gates, input layer, and output layer are all components of a faster RCNN. The "gate" control mechanism is used by the Faster RCNN for adding or discarding part of the information. Then the memory cell state is updated by combining the current input, historical memory, and historical state.

The neural unit's input information is currently controlled by the input gate. The neural unit's output information is currently controlled by the output gate. The historical data previously stored by the neural unit is controlled by the forget gate.

The information is selectively filtered by the "gate" structure and it consists of the dot product operation's sigmoid function. The sigmoid function produces an output in the range [0, 1], where complete passing is represented by 1 and complete rounding is represented by 0. The below equation is used to determine the dropout prediction,

$$IG_t = \sigma(WT_i \cdot [h_{t-1}, x_t] + BS_i) \quad (3)$$

$$FG_t = \sigma(WT_f \cdot [h_{t-1}, x_t] + BS_f) \quad (4)$$

$$OG_t = \sigma(WT_o \cdot [h_{t-1}, x_t] + BS_o) \quad (5)$$

$$G_t = \sigma(WT_c \cdot [h_{t-1}, x_t] + BS_c) \quad (6)$$

$$CS_t = FG_c \cdot CS_{t-1} + IG_t \cdot G_t \quad (7)$$

Where the sigmoid function is represented by $\sigma(\cdot)$, the weights and biases of the input gate IG_t , forget gate FG_t , and output gate OG_t is represented by WT_i , WT_f , WT_o , BS_i , BS_f , and BS_o from which output h_t at current moment t and cell state C_t at current moment t is calculated.

3.2.4. Attention Mechanism

In several deep-learning fields, the attention mechanism has been extensively employed in recent years. Assigning larger weights to information by using the attention mechanism, is more important for the proposed model. The attention method used in this research is implemented using static attention. Faster RCNN uses the simple, efficient, and typically designed static attention method. While compared to soft attention, the model efficiency is improved by achieving a data vector representation with only one calculation.

Evaluating important features and ignoring unimportant features are done by using the static attention mechanism. Information on each feature's weight is determined by the static attention mechanism for the time series-based generation of local feature information. For adaptive learning, the weights are multiplied by the input feature data. The first hidden state of the RCNN's first layer typically uses the effective light-weight attention module known as static attention. Combining feature information with its output provides a weight value calculation. The following formula represents the main calculation.

$$O_d = \vec{O}_d(t) \parallel \vec{O}_d(t) \quad (8)$$

$$q(t) = \tanh(w_{om} O_d(t) + w_{om}^o) \quad (9)$$

$$p(t) = \exp(w_{qp}^T q(t)) \quad (10)$$

$$Z = O_d p \quad (11)$$

At t the moment, the output of the Faster RCNN for each feature information is $O_d(t)$ for the input feature information d , and after static attention processing, the feature information's weighted vector is represented by q , where the weights represent the static attention network's level of attention provided to feature information.

4. RESULTS AND DISCUSSIONS

This section presents the environment settings, experimental datasets, and relevant software and hardware. The criteria for both the evaluations and performance analysis are also both clearly described.

4.1. DATASET DESCRIPTION

In this research, the EDLN dropout prediction model's effectiveness is assessed using data from the Cup 2015 KDD, which evolved from "XuetangX," China's largest MOOC platform. Over five months in 2013–2014, the 120,542 clickstream data points are recorded by the dataset in 39 courses from 79,186 students, with each

course lasting five weeks. Seven behavioral factors were identified in this dataset that describes students' behavior such as page close, navigate, discussion, wiki, access, video, and problem. A label was given to each chosen student indicating whether or not they had dropped out.

4.2. EXPERIMENTAL PARAMETERS

For this experiment, the training and test sets were divided into the dataset at an 8:2 ratio. In the model parameters, the empirical values are chosen for the optimum hyperparameters. The hyperparameters are tuned by using the Adam optimizer. The proper model training is ensured by setting the model's dropout is 0.2 to address the overfitting issue. 200 batches are being processed, with a learning rate of 0.0025. The parameter training is done by using the logarithmic loss function and adaptive learning rate optimization is performed by using the Adam optimization function. The best results are achieved by setting the model's epoch to 20.

4.3. EVALUATION METRICS

The most used metric for evaluating the effectiveness of deep learning models is accuracy. The larger teaching accidents are obtained by misclassifying dropout samples as non-dropout samples could result in more significant adverse effects. Therefore misclassifying non-dropout samples as dropout samples is preferable. Precision, recall, F1-score, and accuracy are used for the evaluation of the proposed model for predicting the MOOC student samples. Table 5 displays the confusion matrix used to define the MOOC dropout prediction model.

Higher priority is given to the precision and recall metrics for predicting the dropout samples because of the MOOC dropout prediction problem's cost-sensitive nature. A model with higher precision will accurately predict more samples. A model with a higher recall misses fewer data when making predictions. The model's higher accuracy demonstrates that it avoids making inaccurate predictions. The symmetrical mean of precision and recall is the F1 score.

4.4. EXPERIMENTAL RESULTS

Table 2 shows the five-time matrices that are compared in terms of evaluation values. When the student behavioral features within the first five weeks are used as input, the proposed model performs at its best. 97.5% accuracy can be attained. When the EDLN model uses the input data as a time series matrix of 1×7 , the poorest classification performance is achieved than other inputs, it has obtained 87.7% accuracy. This is because fewer course tasks are in for learners, causes producing relatively few behavioral features. The more the course continues, the more behaviors the students produce.

Five-time matrices are compared in terms of evaluation values, as shown in Fig. 3. The model performed better than the time series matrix of 1×7 by about 1.7%

when the time series matrix of 2×7 was used as input data. While comparing the inputs of the 5×7 time series matrix and the 4×7 -time series matrix, there is just about a 2.9% difference between the two experiments. The five-week length of each course in the sample provides for this result. In the last week, several learners decided for learning offline. In the fifth week, it results in learners displaying behavioral characteristics significantly less commonly. This affects the model's assessment results. Fig. 4 displays the proposed model's loss and accuracy and Fig. 5 displays the proposed model's confusion matrix.

Table 2. The performance result comparison over five-time matrices

Model	F1-Score (%)	Recall (%)	Precision (%)	Accuracy (%)
EDLN with 1×7 matrix	86.2	83.5	85.2	87.7
EDLN with 2×7 matrix	88.6	85.2	86.6	89.4
EDLN with 3×7 matrix	92.3	90.9	90.2	92.9
EDLN with 4×7 matrix	94.3	94.2	94.6	94.6
EDLN with 5×7 matrix	97.2	96.1	97.3	97.5

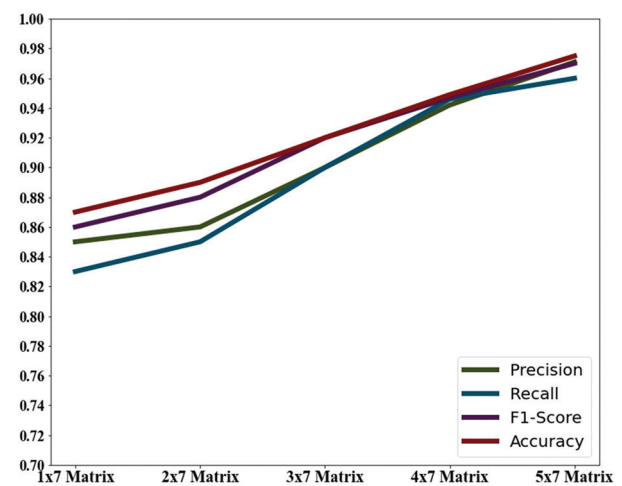
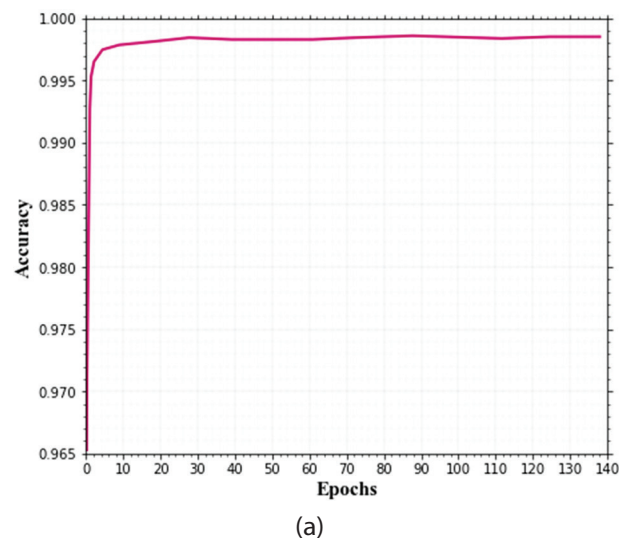


Fig. 3. Performance result comparison over five-time matrices



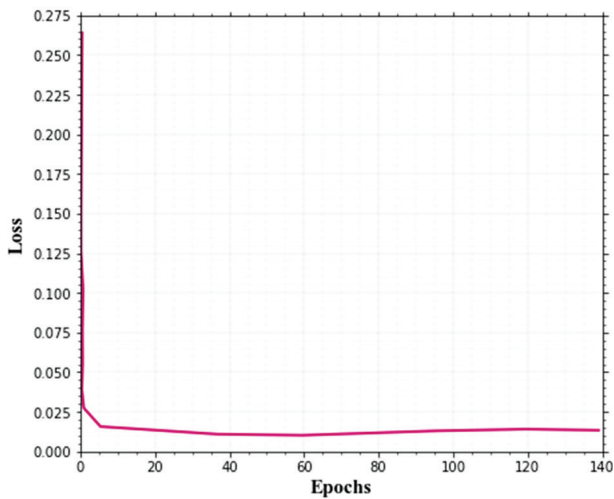


Fig. 4. Proposed model (a) accuracy (b) loss during the training epochs

Confusion Matrix: $\begin{bmatrix} 2131 & 1647 \\ 669 & 13652 \end{bmatrix}$

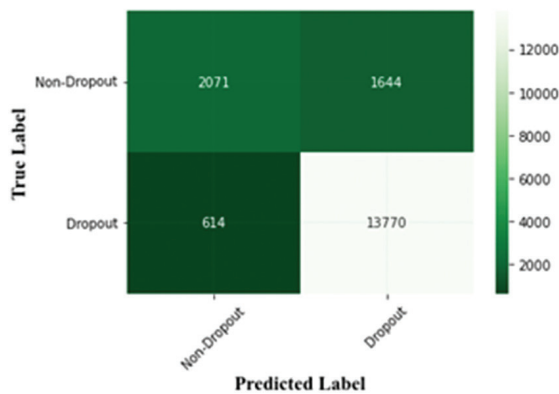


Fig. 5. Confusion matrix of the proposed model

The user's dataset was used to test the system for 5 weeks, and it successfully predicted how many students would drop out over that time. The model produces two classes of output: finished learners, and unfinished learners.

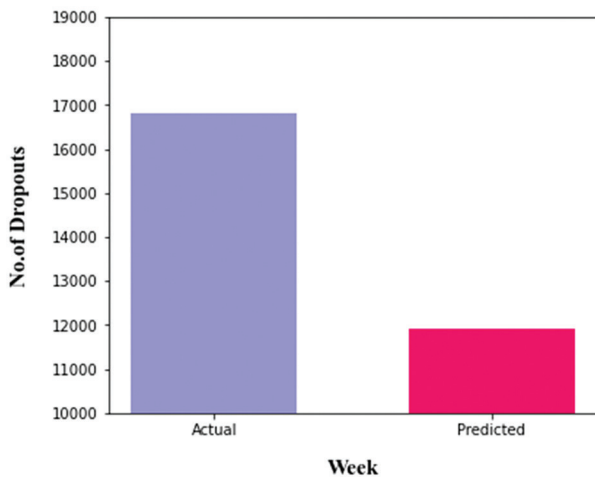


Fig. 6. Actual vs Predicted Dropout in a week

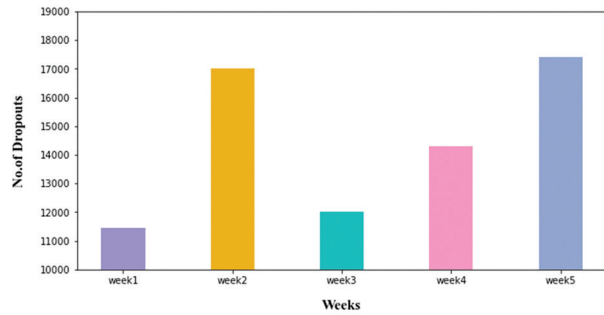


Fig. 7. The performance of Week-wise dropout

The performance of each student is also given a weekly rank. A steady gain will encourage the learner to finish the course early, whereas a consistent decrease will warn the learner about his or her likelihood of leaving the course in the approaching weeks. From the proposed model's performance, we have achieved a reduced dropout rate and this proposed method improves the engagement level of MOOC learners.

Week: 1 - Rank: 11306
Buck up buddy

Week: 2 - Rank: 918
That was a great improvement

Week: 3 - Rank: 6036
Watchout! You are slipping

Week: 4 - Rank: 11419
Watchout! You are slipping

Week: 5 - Rank: 11454
Watchout! You are slipping

At this rate you are more likely to dropout

Fig. 8. Student Intervention after every week

4.5. QUANTITATIVE EVALUATION

While comparing the baseline model with the EDLN model, the proposed model's advantages and efficiency are validated. Table 3 shows the performance effectiveness of the several existing models and the proposed model.

Table 3. Performance comparison of various models

Model	F1-Score (%)	Recall (%)	Precision (%)	Accuracy (%)
LSTM	78.7	78.8	78.7	79.2
CNN	81	81.2	80.9	81.3
MMSE	92.6	89.5	86.3	87.7
SVM-SGD	93.1	93.1	93.7	91
AdaBoost	94.2	93.8	94.9	92.9
The proposed model (EDLN)	97.2	96.5	97.1	97.4

The proposed technique achieved an accuracy, precision, recall, and F1 measure of 97.4%, 97.1%, 96.5%, and 97.2%, respectively, which indicates that the pro-

posed technique outperforms all other state-of-the-art methods. When compared to the SVM-SGD model, the proposed model accuracy is 6.4% higher. From this analysis, the large-scale MOOC dropout prediction is performed more effectively by the proposed model than the SVM model.

The MMSE, LSTM, and CNN deep learning models are also included in the baseline models. The accuracy of the proposed model is higher than LSTM and CNN by approximately 18.2% and 16.1%. The local receptive field feature is used for performing the feature learning for CNN models in such disordered data. When comparing the proposed EDLN model with the CNN model in terms of all performance analysis values, the proposed EDLN model performs better than the standard CNN model. The performance analysis graph is shown in Fig. 8. Using the time matrix to predict MOOC dropouts, excellent performance is achieved by the proposed model in this research.

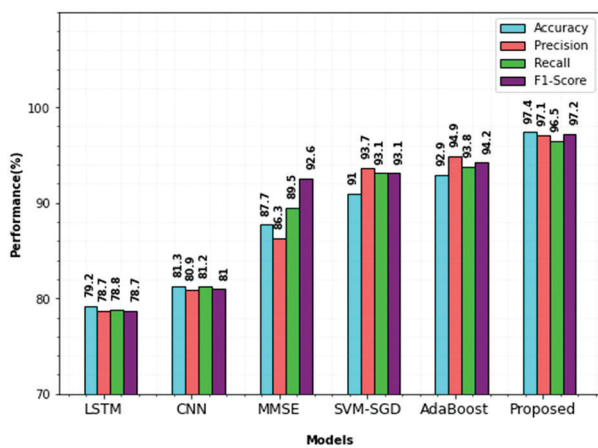


Fig. 9. Performance analysis

The source data for the five models were taken from the KDD CUP 2015 dataset. For dropout prediction, when comparing the existing deep learning model, the proposed EDLN model outperformed. It is shown that the proposed EDLN model, which incorporates the attention process and model input as a temporal matrix, is successful in increasing the dropout prediction accuracy in MOOCs. The proposed EDLN MOOC dropout prediction model more effectively extracts and learns the time series information and local feature learning of the source data automatically and also the proposed model successfully predicted how many students would drop out over that time.

In [21], the SVRQ model is proposed for student dropout prediction, the parameters are optimized by the IQPSO algorithm. The optimization algorithm improves the performance of the student's dropout prediction. An accuracy of 92% and an F1-score of 95% are achieved by this SVRQ model. Qiu et al [22] utilize the FSPred model for student dropout prediction and it achieves an accuracy of 86.34%. Panagiotakopoulos et al [23] proposed an early dropout prediction model

and it achieve an accuracy of 91.00%. In comparison to this, Xing et al [24] and Chen et al [25] achieve an accuracy of 95.01% and 92.5% for dropout prediction. This approach is used to personalize and prioritize intervention for at-risk students in MOOCs by using individual dropout probabilities. While compared to the literature review dropout prediction models, the proposed EDLNet model achieves better results and it effectively predicts the dropout of the student at an early stage. The proposed EDLNet model has the main benefit of preventing overfitting and having no negative effects on network performance due to the classification and segmentation process.

The following conclusions are obtained by empirical analysis. First, when it comes to the length of the course, shorter courses have lower dropout rates than longer ones. To lower the online learning dropout rate, instructors should employ particular subjects. Second, the dropout rate is reduced by social-interactive engagement. To reduce students' feelings of isolation and disconnection, it is important to encourage students to engage in more online activities. For dropout rate reduction, additionally, learner experience is extremely beneficial. The dropout rate is also reduced by increasing experience because collecting experiences is the process of engagement. Lastly, the course itself has an impact on the dropout rate. Less difficult courses have lower dropout rates than more challenging ones.

5. CONCLUSION

For MOOC dropout prediction, this research proposed an EDLN model. The process of training and testing is performed successively after being converted into a two-dimensional temporal matrix form from the public dataset for the 2015 KDD Cup. The EDLN model was compared to five baseline models, and tests with varying time matrix specifications and attention mechanism-based ablation were carried out. During the first five weeks, the dropout situation is accurately predicted by the EDLN model based on learner behavior and characteristic data. The results of the experiment demonstrate that with significant information. Based on better predictive performance, less time required, F1 values, recall, precision, and accuracy, the proposed EDLN model outperforms other baseline models.

While comparing to the baseline models, in addition to creating dropout prediction models that are more accurate, the deep learning methodology will also provide a reliable method to help with intervention design for reducing the dropout rate. The research object for this study was a MOOC dropout prediction problem and ResNet-50, Faster RCNN, and static attention are combined in the proposed model for the prediction of dropout in MOOCs to effectively resolve online learning platforms with a high dropout rate.

The interpretability of the MOOC dropout prediction will be the main area of research in the future. We also

examine the model's basis further to produce predictions that match the learner's states, specifically, the correlation between the learner behavior features and the model's prediction results as well as the learner behavior features. Furthermore, based on a more detailed search of inaccurate predictions, we will try to significantly enhance the current model.

6. REFERENCE

- [1] R. B. Basnet, C. Johnson, T. Doleck, "Dropout prediction in Moocs using deep learning and machine learning", *Education and Information Technologies*, Vol. 27, 2022, pp.1-15.
- [2] H. Aldowah, H. Al-Samarraie, A. I. Alzahrani, N. Alalwan, "Factors affecting student dropout in MOOCs: a cause and effect decision-making model", *Journal of Computing in Higher Education*, Vol. 32, No. 2, 2020, pp. 429-454.
- [3] W. Wang, L. Guo, L. He, Y.J. Wu, "Effects of social-interactive engagement on the dropout ratio in online learning: insights from MOOC", *Behaviour & Information Technology*, Vol. 38, No. 6, 2019, pp. 621-636.
- [4] Y. Mourdi, M. Sadgal, H. El Kabtane, W. B. Fathi, "A machine learning-based methodology to predict learners' dropout, success, or failure in MOOCs", *International Journal of Web Information Systems*, Vol. 15, No. 5, 2019, pp. 12-24.
- [5] S. Ardchir, M. A. Talhaoui, H. Jihal, M. Azzouazi, "Predicting MOOC dropout based on learner's activity", *International Journal of Engineering & Technology*, Vol. 7, No. 32, 2018, pp.124-126.
- [6] J. Chen, J. Feng, X. Sun, N. Wu, Z. Yang, S. Chen, "MOOC dropout prediction using a hybrid algorithm based on a decision tree and extreme learning machine", *Mathematical Problems in Engineering*, 2019, pp. 20-39.
- [7] S. Dass, K. Gary, J. Cunningham, "Predicting student dropout in self-paced MOOC course using random forest model", *Information*, Vol. 12, No. 11, 2021, pp.476.
- [8] Y. Zheng, Z. Gao, Y. Wang, Q. Fu, "MOOC dropout prediction using FWTS-CNN model based on fused feature weighting and time series", *IEEE Access*, Vol. 8, 2020, pp. 225324-225335.
- [9] A. A. Mubarak, H. Cao, I. M. Hezam, "A deep analytic model for student dropout prediction in massive open online courses", *Computers & Electrical Engineering*, Vol. 93, 2021, p.107271.
- [10] T. Y. Yang, C. G. Brinton, C. Joe-Wong, M. Chiang, "Behavior-based grade prediction for MOOCs via time series neural networks", *IEEE Journal of Selected Topics in Signal Processing*, Vol.11, No. 5, 2017, pp.716-728.
- [11] C. Jin, "Dropout prediction model in MOOC based on clickstream data and student sample weight", *Soft Computing*, Vol. 25, No. 14, 2021, pp. 8971-8988.
- [12] M. Youssef, S. Mohammed, E. K. Hamada, B. F. Wafaa, "A predictive approach based on efficient feature selection and learning algorithms' competition: Case of learners' dropout in MOOCs", *Education and Information Technologies*, Vol. 24, No. 6, 2019, pp. 3591-3618.
- [13] M. Şahin, "A comparative analysis of dropout prediction in massive open online courses. Arabian Journal for Science and Engineering", Vol. 46, No. 2, 2021, pp. 1845-1861.
- [14] J. Kabathova, M. Drlik, "Towards predicting student dropout in university courses using different machine learning techniques", *Applied Sciences*, Vol. 11, No. 7, 2021, p.3130.
- [15] K. Coussement, M. Phan, A. De Caigny, D. F. Benoit, A. Raes, "Predicting student dropout in subscription-based online learning environments: The beneficial impact of the logit leaf model", *Decision Support Systems*, Vol. 135, 2020, p.113325.
- [16] S. Sood, M. Saini, "Hybridization of cluster-based LDA and ANN for student performance prediction and comments evaluation", *Education and Information Technologies*, Vol. 26, No. 3, 2021, pp. 2863-2878.
- [17] S. Lee, J. Y. Chung, "The machine learning-based dropout early warning system for improving the performance of dropout prediction", *Applied Sciences*, Vol. 9, No. 15, 2019, p. 3093.
- [18] A. A. Mubarak, H. Cao, W. Zhang, "Prediction of students' early dropout based on their interaction logs in the online learning environment", *Interactive Learning Environments*, Vol. 30, No. 8, 2020, pp. 1-20.

- [19] P.M. Moreno-Marcos, P.J. Muñoz-Merino, J. Maldonado-Mahauad, M. Perez-Sanagustin, C. Alario-Hoyos, C. D. Kloos, "Temporal analysis for dropout prediction using self-regulated learning strategies in self-paced MOOCs", *Computers & Education*, Vol. 145, 2020, p. 103728.
- [20] C. Ye, G. Biswas, "Early prediction of student dropout and performance in MOOCs using higher granularity temporal information", *Journal of Learning Analytics*, Vol. 1, No. 3, 2014, pp. 169-172.
- [21] C. Jin, "MOOC student dropout prediction model based on learning behavior features and parameter optimization", *Interactive Learning Environments*, 2020, pp. 1-19.
- [22] L. Qiu, Y. Liu, Y. Liu, "An integrated framework with feature selection for dropout prediction in massive open online courses", *IEEE Access*, Vol. 6, 2018, pp. 71474-71484.
- [23] T. Panagiotakopoulos, S. Kotsiantis, G. Kostopoulos, O. Iatrellis, A. Kameas, "Early dropout prediction in MOOCs through supervised learning and hyperparameter optimization", *Electronics*, Vol. 10, No. 14, 2021, p. 1701.
- [24] W. Xing, D. Du, "Dropout prediction in MOOCs: Using deep learning for personalized intervention", *Journal of Educational Computing Research*, Vol. 57, No. 3, 2019, pp. 547-570.
- [25] W. Chen, C. G. Brinton, D. Cao, A. Mason-Singh, C. Lu, M. Chiang, "Early detection prediction of learning outcomes in online short-courses via learning behaviors", *IEEE Transactions on Learning Technologies*, Vol. 12, No.1, 2018, pp.44-58.
- [26] H. Wan, K. Liu, Q. Yu, X. Gao, "Pedagogical intervention practices: Improving learning engagement based on early prediction", *IEEE Transactions on Learning Technologies*, Vol. 12, No. 2, 2019, pp. 278-289.

Iterative Feature Selection-Based DDoS attack Prevention Approach in Cloud

Original Scientific Paper

Sarah Naiem

Helwan University,
Faculty of Computers and Artificial Intelligence
Cairo, Egypt
SarahNaiem@fci.helwan.edu.eg

Amira M. Idrees

Fayoum University
Faculty of Computers and Artificial Intelligence
Cairo, Egypt
ami04@fayoum.edu.eg

Ayman E. Khedr

Future University in Egypt
Faculty of Computers and Information Technology
Cairo, Egypt
ayman.khedr@fue.edu.eg

Mohamed Marie

Helwan University,
Faculty of Computers and Artificial Intelligence
Cairo, Egypt
Dr.mmariam@fci.helwan.edu.eg

Abstract – Distributed Denial of Service (DDoS) attacks aim to exploit the capacity and performance of a network's infrastructure, making the cloud environment one of the biggest targets for attackers. Many efforts are being made in the field of technology to prevent them from disrupting the services provided. Machine Learning techniques are a means to protect against DDoS attacks. Data preprocessing, feature selection, and classifiers are the main components of any prevention framework. The focus of this study is to find and enhance the feature selection approach for increasing the accuracy of the classifiers in detecting DDoS attacks from regular traffic. We used four different techniques, including Pearson Correlation Coefficient (PCC), Random Forest Feature Importance (RFFI), Mutual information (MI), and Chi-squared(X2) measure which we tested on different classifiers. The first selection approach was based on the feature's independency level then the second iteration was based on the feature's importance. We also examined the claim of dropping attacks from the dataset for better accuracy. The best performing set of features was from using PCC and RFFI together for feature selection with average accuracy and precision of 99.27% and 97.60%, which is higher than the use of PCC for both measures by almost 2%. The accuracy is also higher by nearly 12% from the same approach dropping 50% of the attacks.

Keywords: DDoS attacks; cloud environment; machine learning; feature selection; random forest; Pearson correlation coefficient; mutual information; chi-square.

1. INTRODUCTION

Distributed Denials of services (DDoS) is on the list of top attacks jeopardizing the cloud environment, messing with the cloud traffic, and denying benefits to a legitimate user [1]. Recently the cloud computing environment gained massive popularity due to the variety of services it provides, including education, networking, storage, security, elasticity, and migration flexibility, making it a target for cybercrime. [2] [3]. DDoS aims to disrupt a specific server, service, or network's usual traffic by saturating the target or its surrounding infrastructure with non-legitimate traffic. The attacker's DDoS attempts are usually successful because they use several compromised computers as attack traffic sources. Where DDoS attacks operate under the theory of using numerous machines to produce high-intensity-based attack traffic to compromise the integrity of the network [4]. These machines are unaware

that the attacker is employing them to harm and are usually referred to as "zombies" who are challenging to detect. [5]. Many efforts have been made to protect the cloud and internet from these attacks with the help of Machine Learning techniques, deep learning, count-base filtering, resource usage, data mining, and other methods. [6]. On the other hand, feature selection techniques are proposed in different research which tackles various issues such as in the education field [7], [8], and construction field [9]. Moreover, features selection techniques also proved their effectiveness in other research directions such as in Recommendation systems sentimental analysis [10], classification [11] [12] [13], query answering [14] [8], decision support systems [15] [16], and Internet of Things (IoT) [17].

This research focuses on preventing and detecting using machine learning (ML) techniques, including feature engineering and selection, data normalization, and ML classification algorithms. In addition to that, it has been

claimed by Tan et al. in [18] that dropping part of the attacks from the dataset would improve the detection accuracy, and we are testing this claim throughout our work. The rest of this paper will include a literature review of the previous related work, a description of the framework applied, including the dataset, the feature engineering approaches followed, including Pearson Correlation, Information Gain, Chi2, Random Forest Feature Importance RFFI, and a comparison between different machine learning classifiers including Random Forest, Decision Tree, and Gaussian Naïve Bayes.

2. LITERATURE REVIEW

In [19] the authors focused on applying the K-fold cross-validation on the CIC-IDS2017 dataset. They tested the model on Random Forest (RF), K- nearest neighbor (KNN), Decision Tree (DT), Gaussian Naïve Bayes (GNB), Support Vector Machine (SVM) on all three kernels, including sigmoid, polynomial, and RBF), and logistic regression. They set the K value equal to 5, and each k was split into 5 sets, one for testing and 4 for training. The results of the classifier's accuracy, precision, and recall were compared, and they concluded that the one with the best results was the DT. It is clear that even though the DT is the best classifier with an accuracy of 99.94, the other 6 classifiers, except for the DT, showed an accuracy between 99.78 and 99.88, while the GNB had an accuracy of 61.22.

The authors in [20], used 3 combined datasets, including CSE-CIC-IDS2018-AWS, CICID2017, and CIC DOS 2017 datasets, and selected 7 features. The resulting dataset was tested using RF, DT, GNB, and multilayer perception neural network approaches using different training and testing split sets from 90/10 to 50/50. The results showed that the RF had the highest accuracy from the average of the 5 train test splits at 99.969%, followed by DT at 99.951%, MLP at 98.87,6%, and the least accurate was the GNB at 78.45%. It was also concluded that the different Train-test splits don't affect the model's accuracy feature selection approach discussed was neither mentioned nor the normalization technique followed. Also, the author was not sure about the accuracy of the model due to the selected features or the combination of different datasets. While in [1], in the dataset CICDDOS2019, the authors applied the chi-squared (χ^2) feature selection approach to select the top 10 features after cleaning the data. The preprocessed dataset was used for the detection using the deep learning technique mixing BILSTM and CNN using different filter size, filter count, and unit size. After using 10 models with varying filter size and count and unit size, it was clear that reducing the unit size increases the accurate model's accuracy while lowering the filter size decreases training time. But even though the hybrid model was compared to other machine learning models, including RF, SVM, KNN, LR, and XBG, proving that it has higher accuracy, the accuracy rate of these models was significantly lower than the average, where

it ranged from 64 to 78%. This indicated that the set of selected features using the χ^2 doesn't represent this dataset's best features.

To improve the feature selection of DDOS in the cloud, the authors of [21] used the CICDDOS2019 dataset and applied the chi-squared (χ^2) feature selection approach to select the top 10 features after cleaning the data. The preprocessed dataset was used to detect DDOS through a hybrid deep learning technique mixing BILSTM and CNN using different filter size, filter count, and unit size. After using 10 models with varying filter size and count and unit size, it was clear that reducing the unit size increases the accuracy of models and the filter size decreased the training time. Even though the authors compared their hybrid model to other machine learning models, including RF, SVM, KNN, LR, and XBG, proving that it has higher accuracy, the accuracy rate of these models was significantly lower than the average, where it ranged from 64 to 78%.

The authors in [5] focused on explaining and providing a detailed background for DDOS detection, including the different conceptual ML techniques and types of DDOS attacks. The random forest feature importance (RFFI) technique was used for selecting the critical features resulting in 13 features according to their score. In [21], the authors used 2 different feature selection approaches, including Mutual Information (MI) and RF approach, and compared the accuracy by training the dataset using logistic regression (LR), KNN, Gradient Boosting (GB), and RF, and weighted voted ensemble (WVE). They used 3 sets and tested them with the 5 classifiers—16 features with MI, 19 with RFFI, and 23 with MI. The accuracy of the 16 selected features was 99.993%, and the 19 features was 99.997%. Even though the set of features chosen had very high accuracy, the authors only stated that their paper proved that MI and RFFI work well with the selected ML classifiers.

The authors in [22], focused on enhancing the accuracy of GNB. The dataset selected for the research was KDD 99, which was cleaned before deciding the essential features using correlation-based feature selection (CSF). The GNB classifier was enhanced using 2 approaches. The first was the elimination of the zeroes from the dataset, and the second was changing the GNB statistical equation from its multiplication form to its addition form, increasing the accuracy by about 4 %. In [23], the author used the CICIDOS-2019 data, where she selected the top 20 features using the Extra Tree Classifier approach. After that, she used Rf, DT, SVM, and NB classifiers for DDOS detection focusing on LDAP and MSSQL DDOS attacks. The accuracy of RF and SVM classifiers was 99.99%, while DT was 99.89%, and NB was 99.98%.

The authors in [24] focused on the slow rate DDOS attacks where they integrated CICID2017 and CSE-CIC-IDS2018, extracted the top 30 features using Information gain, and then selected the top 10 using

the Chi-square approach. The model was trained on J48, bagging technique, MLP, and KNN classifiers. The attacks under testing included DOS GoldenEye, Slowloris, Slowhttpstest, Hulk, DDoS HOIC, and DDOS LOIC-HTTP. The F1 score, True positive, and true negative were calculated, showing that the feature selection approaches high results for all classifiers. The authors stated that the model's accuracy was almost 99%, with a meager false negative rate.

In [25], the DT model for feature ranking was applied on CICDDOS2019 resulting in a list of the top 30

features, in addition to the use of the person correlation coefficient (PCC) approach resulting in a list of 20 features. They tested the selected features on different ML models, including RF, Light Gradient Boosting, Cat Boost, and CNN. The results of the 20 selected features with the RF and GB and the 30 features with the CatBoost and the CNN were the highest-performing classifiers. Table 1 summarizes the previously mentioned studies in the literature review and their limitations when it comes to the techniques and approaches followed in feature selection which we are trying to overcome in our research.

Table 1. Literature review summary

Reference	Dataset	ML techniques applied	limitation
(Nalayinil and Katiravan 2022) [19]	CIC-IDS2017	K-fold cross-validation on Rf, Knn, GNB, SVM, and LR	The researchers did not focus on the feature selection phase, which would have an impact on the results if considered
(Coelho 2022) [20]	3 combined datasets CSE-CIC-IDS2018-AWS, CICID2017, and CIC DOS 2017 datasets	RF, DT, GNB, and multilayer perception neural network	The feature selection approach discussed was not mentioned, nor was the normalization technique followed. The author needed clarification about the accuracy of the models, whether it was due to the selected features or the combination of different datasets.
(Praveen and Rimal 2020) [1]	CCIDS2017	SVM and NB	The authors only mentioned using 20 features without stating the feature selection criteria.
(Alghazzawi , et al. 2021) [26]	CICDDOS2019	hybrid deep learning technique mixing BILSTM and CNN	The set of selected features using the X2 does not represent the best set of features for this Dataset
(Narote, Zutshi and Potdar 2022) [5]	CCIDS2017	Used RFFI for feature selection	The selected set of features was not tested on any ML techniques, and no accuracy or results were provided to show the success of the feature selection approach chosen.
(Alduailij , et al. 2022) [21]	Not specified	MI and RF for feature selection and compared results using LR, KNN, GB, RF, and WVE)	It should have stated which approach is better and how to choose the appropriate one.
(Kurniawan, et al. 2021) [22]	KDD 99	GNB	The improvement in the GNB classifier was not tested on an up-to-date dataset.
(Mishra 2022) [23]	CICIDOS-2019	Extra Tree Classifier for feature selection and RF, DT, SVM, and NB	Only 2 types of DDOS attacks (LDAP-DDOS MSSQL) were taken into consideration
(Swe, Aung and Hlaing March 7-11, 2021) [24]	CICID2017 and CSE-CIC-IDS2018	RF, DT, SVM	The feature selection approach chosen was only tested for the detection of slow-rate DDOS, and it does not show if it would work for other DDOS attack types
(Alghoson and Abbass 2021) [25]	CICDDOS2019	DT model for feature selection Rf, Light Gradient Boosting, Cat Boost, and CNN	Even though the authors tested the different 2 sets of features on 4 classifiers, they only displayed the results of 4 ML algorithms, and the accuracy of the 8 sets of classifiers should have been provided.

3. PROPOSED FRAMEWORK

This section will describe the components of our framework and methodology. Our Main purpose is to use this dataset most efficiently to be able to detect anomalies in the traffic. An overview of the proposed framework is displayed in Fig 1.

3.1. DATASET

Throughout our research, we targeted some of the available datasets regarding IDs and DDOS attacks. Our focus is on CSE-CIC-ID2018, an open-source data-

set made available by the University of New Brunswick UNB. The dataset has 80 features presenting seven attacks, including DDOS, DOS, Web-attacks, infiltration, Brute force, and Botnet attacks, and benign traffic generated through the CICFlowmeter-V3, which we presented in table 2. The data distribution shows that the total number of attacks given in the dataset is less than 20% of the traffic flow. The dataset's traffic is captured and delivered in 7 CSV formatted files classified according to the dates of their occurrences, including Wednesday 14/2/2018, Thursday 15/2/2018, Friday 16/2/2018, Tuesday 20/2, Wednesday 21/2/2018, Thursday 22/2/2018, Friday 23/2/2018 [27] [28] [29]

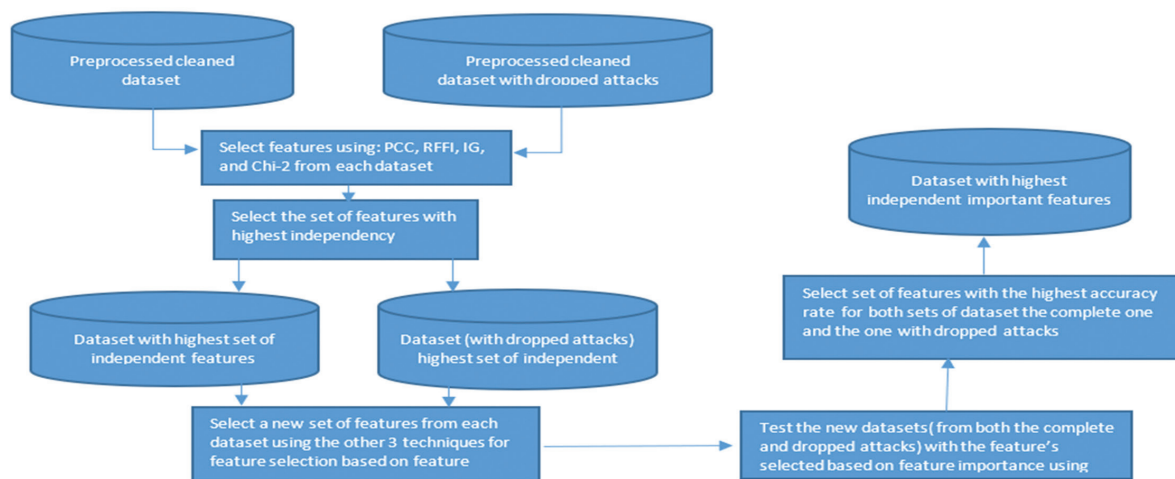


Fig 1. framework overview

Table 2. dataset traffic distribution

Type of traffic	Distribution of traffic within the dataset
Benign	83.07%
DDoS	7.79%
DoS	4.03%
Web attacks	0.006 %
Infiltration	0.997%
Brute-force	2.35%
Botnet	1.76 %

3.2. DATA PREPROCESSING

The first step we took was data preprocessing, which is cleaning the data. After that, the feature selection and normalization of the data is conducted using Machine learning techniques to predict and prevent attacks. Without this phase, any model won't perform as intended, as no machine learning algorithm can handle it to produce predictions and insights.

The most vital part of the data preprocessing is the data cleaning performance, which removes any missing and incomplete data that would result in inaccurate results if considered. The dataset used included some columns with a handful of Zero values, infinite and null values which would have highly impacted the framework's accuracy, so we replaced all the null and infinite values with zeros, and the rows with more than 40% missing values were dropped from the data.

Another thing applied in this phase was dropping 50 percent of the attack traffic presented to represent a more realistic attack. According to their study, this approach was conducted by M.Tan 2019 in their attempts to create a deep learning approach for real-time network intrusion detection and according to their research. Even though they dropped 90% of the attacks, not 50%, to represent a more realistic traffic environment, it still resulted in more accurate results [18]. Through this study, we will present how applying this affects the accuracy of the feature extraction and different supervised and unsupervised ML models to not drop any attack from the dataset.

3.3. FEATURE EXTRACTION

Several different feature selection approaches were conducted to reach the optimum method of generating the best-fitted list of essential features from the 80 features presented through the dataset, including PCC, MI, RFFI, and chi2.

- **Pearson Correlation Coefficient**

The PCC score algorithm calculates each feature's grades regarding the label feature. The features with a threshold higher than or equivalent to 0.08 are selected, and the rest of the 80 features are dropped before the data is used in further steps. PCC score calculates the force of the linear relationship between variables in a correlative matter [30].

- **Mutual Information**

MI feature selection is based on MI obtained from getting the information gain (IG) and entropy to get the top features with the most IG. The mutual information for variables X and Y and the entropy are represented as follows [31]:

$$I(X; Y) = H(X) - H(X | Y) \text{ Where } I(X; Y) \text{ is the mutual information for X and Y, } H(X) \text{ is the entropy for X and } H(X | Y) \text{ is the conditional entropy for X given Y.}$$

It is conducted in python with the help of some sklearn libraries, including the "mutual_info_classif" and "SelectKBest" along with the "train_test_split." After the data is cleaned and split into train and test, it is passed to the mutual information class, and the features are sorted from the ones with the highest information gain to the least. The top 20 features of the dataset with dropped and without attacks turned out to be the same.

- **Random Forest Feature Importance**

RFFI is one of the most vital feature selections in data science regarding selecting the most significant features. Its approach is based on random forest trees to reduce the Gini impurity. It uses the data after it is cleaned to train using a random forest classifier to se-

lect the most vital features. As a result, it creates a subset dataset that is trained again using the RF classifier to compare the accuracy of both datasets. The result of this approach is a list of the essential features.

- **Chi-Squared**

Chi2 followed for feature selection was the Chi2, a statistical approach used to evaluate the correlation between independent categorical variables of a dataset by calculating the p-value and selecting the ones with high correlation reflected through the best chi-square score. It is calculated by subtracting the expected frequency from the observed frequency and divided by the predicted frequency [32].

In our efforts to find the best set of features for the CID2018 dataset, the RFFI was conducted on four data sets. The first two sets were based on the original dataset with 80 features, once without dropping the attacks and once with dropping 50 % of the attacks. The second two sets were based on the features selected from the first phase result based on the feature dependency. The set of selected features with the highest level of independence was the PCC, which resulted in 24 features dropping 50% of the attacks and 29 without that. We chose a subset of features from the 24 and 29 features based on feature importance achieved from applying the RFFI approach.

Out of all the feature selection methods and approaches, we had 10 sets of derived features, and one of the two sets of selected features using IG was eliminated since they gave the same set of features.

The resulting 9 sets of features are displayed in table 3 and table 4.

Table 3. Set of selected features without dropping attacks

FEATURE SELECTION MODEL	FEATURES
PCC	Dst Port, Protocol, Fwd Pkt Len Max, Fwd Pkt Len Min, Fwd Pkt Len Mean, Fwd Pkt Len Std, Bwd Pkt Len Min, Bwd Pkt Len Mean, Flow Pkts/s, Bwd IAT Tot, Fwd Pkts/s, Bwd Pkts/s, Idle Std, Pkt Len Max, Pkt Len Mean, Pkt Len Std, ACK Flag Cnt, Fwd Seg Size Min, Pkt Size Avg, Fwd Seg Size Avg, Bwd Seg Size Avg, Init Fwd Win Byts, Active Max, TotLen Fwd Pkts, Flow Duration
RFFI	Subflow Fwd Byts, Flow Pkts/s, Init Fwd Win Byts, Flow IAT Std, Active Mean, Fwd IAT Max, Active Max, Bwd Blk Rate Avg, SYN Flag Cnt, Fwd Pkt Len Max, Fwd Blk Rate Avg, Fwd IAT Std, Active Std, Fwd Pkt Len Min, Idle Max
CHI2	Flow Byts/s, Flow IAT Std, Bwd Pkts/s, Fwd IAT Std, Flow Duration, Fwd IAT Tot, Flow IAT Max, Fwd IAT Max, Bwd IAT Mean, Bwd IAT Tot, Flow IAT Mean, Fwd IAT Mean, Bwd IAT Max, Fwd Pkts/s, Bwd IAT Min, Bwd IAT Std, Dst Port, Flow Pkts/s, Fwd Pkt Len Std, Active Mean, Idle Mean, Active Max, Idle Min, Fwd Seg Size Avg, Fwd Pkt Len Mean, Idle Max, Idle Std, Pkt Size Avg, Pkt Len Var, Pkt Len Std
IG	Fwd Seg Size Min, Init Fwd Win Byts, Dst Port, Bwd Pkts/s, Fwd Pkts/s, Flow Pkts/s, Flow IAT Mean, Flow Duration, Init Bwd Win Byts, Flow IAT Max, Fwd Pkt Len Max, Pkt Len Max, Subflow Fwd Byts, TotLen Fwd Pkts, Fwd Seg Size Avg, Fwd Pkt Len Mean, Pkt Len Mean, Pkt Size Avg, Pkt Len Std, Pkt Len Var
RFFI-PCC	Subflow Fwd Byts, Flow Pkts/s, Flow IAT Std, Init Fwd Win Byts, Active Mean, Fwd IAT Max, Active Max, Bwd Blk Rate Avg, SYN Flag Cnt, Fwd Pkt Len Max, Subflow Bwd Byts, Bwd Pkt Len Mean, Fwd URG Flags, Bwd IAT Std, Bwd Pkt Len Std, Fwd Blk Rate Avg

Table 4. Set of selected features with dropping 50% of the attacks

Feature selection Model Name with dropping attacks	Features
PCC-D	Dst Port, Protocol, Fwd Pkt Len Max, Fwd Pkt Len Min, Fwd Pkt Len Mean, Fwd Pkt Len Std, Bwd, Pkt Len Min, Bwd Pkt Len Mean, Flow Pkts/s, Bwd IAT Tot, Fwd Pkts/s, Bwd Pkts/s, Pkt Len Min, Pkt Len Max, Pkt Len Mean, Pkt Len Std, ACK Flag Cnt, URG Flag Cnt, Pkt Size Avg, Fwd Seg Size Avg, Bwd Seg Size Avg, Init Fwd Win Byts, Init Bwd Win Byts, Fwd Seg Size Min
RFFI-D	Subflow Fwd Byts, Flow Pkts/s, Flow IAT Std, Init Fwd Win Byts, Active Mean, Fwd IAT Max, Active Max, Bwd Blk Rate Avg, SYN Flag Cnt, Fwd Pkt Len Max, Subflow Bwd Byts, Bwd Pkt Len Mean, Fwd URG Flags, Bwd IAT Std, Bwd Pkt Len Std, Fwd Blk Rate Avg
Chi2-D	Flow Byts/s, Flow IAT Std, Bwd Pkts/s, IAT Std, Flow Duration, Fwd IAT Tot, Flow IAT Max, Fwd IAT Max, Bwd IAT Mean, Bwd IAT Tot, Flow IAT Mean, Fwd IAT Mean, Bwd IAT Max, Fwd Pkts/s, Bwd IAT Min, Bwd IAT Std, Dst Port, Flow Pkts/s, Fwd Pkt Len Std, Active Mean, Idle Mean, Active Max, Idle Min, Fwd Seg Size Avg, Fwd Pkt Len Mean, Idle Max, Idle Std, Pkt Size Avg, Pkt Len Var, Pkt Len Std
RFFI-PCC-D	Dst Port, Pkt Len Max, ACK Flag Cnt, Init Fwd Win Byts, Pkt Len Mean, Protocol, Bwd IAT Tot, Pkt Len Std, Pkt Len Min, Bwd Seg Size Avg, Bwd Pkts/s, Bwd Pkt Len Mean, Init Bwd Win Byts, Fwd Seg Size Min, Fwd Pkt Len Std

3.4. NORMALIZATION

This phase is vital to transfer the data into a format that could be used in the training and testing phase without affecting its essence or performance [25]. It is conducted because most of the data represent different types and formats, making it nearly impossible to handle, making it an important step to standardize the data before using it. Several techniques could be used for data normalization, including MinMaxScaler and StandardScaler.

MinMax Scaler is based on representing the maximum value as 1 and the minimum value as 0. Accordingly, it represents all the data between 1 and zero, while the standard scaler scales the data within the maximum and minimum values range. The idea of using the MinMax scaler is based on maintaining the actual distribution and representation of the data [33]

The equation for the MinMax Scaler is

$$X' = \frac{x - \min(x)}{(x) - \min(x)} \quad (1)$$

while the Standard Scaler equation is

$$z = \frac{x - \mu}{\sigma} \quad (2)$$

where x is the score μ the men, and sigma is the standard deviation [33].

3.5. DDOS ATTACKS DETECTION AND DATA CLASSIFICATION

After the data had been preprocessed and normalized, the classification using different machine learning classifiers was applied. The first step in this process is splitting the data into train and testing, done through the

train-t testing. For our set of data, we selected three different supervised machine learning algorithms, including the Decision Tree (DT), Random Forest (RF), and Gaussian Naïve Bayes (GNB).

- Decision Tree (DT) is very similar in structure to a flow chart based on a graph with nodes descending from the root or central node. It creates branches in efforts to model the relation between the features of a dataset and its targeted potential output. It is one easy and understandable structure, and normalization of the data is not a necessary step in data preprocessing. DT classifiers use the “Bagging Technique” which trains more DTs in parallel through bootstrapping data samples where the final prediction is based on the results of the trees that are running in parallel [25] [6].
- Random Forest is also a supervised ML classification and regression algorithm based on bagging techniques. It builds several DTs from different illustrations from the datasets and uses their results together. The RF uses the bootstrap data sample and calculates each node’s split by subdivision of the features. Using RF ML classifiers results in very accurate results even with imbalanced and missing data. It is also very flexible, has less variance than a single DT, works perfectly with a largamountsnt of data, and resolves the overfitting of data by averaging the results of several DTs. Unfortunately, the main problem with RF is its complexity which results in high computational time and the need for higher computational resources [25] [23].
- Gaussian Naïve Bayes supports continuous data derived from the Gaussian normal distribution, which is also based on Naive Bayes (NB) derived from the Bayes theorem. The NB is based on the hypothesis that features are independent. This classifier is considered one of the simple and easily implementable techniques for supervised machine learning classification [22] [20].

4. PERFORMANCE EVALUATION AND RESULTS DISCUSSION

Several metrics were calculated to evaluate the perfor-

mance of the model. These metrics support the model analysis and reflect the specific machine learning algorithms’ attack detection quality. The metrics mentioned are defined as where TP, TN, FP, and FN are True Positive, True Negative, False Positive, and False Negative [25] [20]:

- Accuracy: represents the overall performance concerning the actual correct predictions calculated by using equation 4

$$\frac{TP+TN}{TP+FP+TN+FN} \quad (3)$$

$$\frac{TP}{TP+FP} \quad (4)$$

We examined the accuracy of each model with the different classifiers; the results are displayed in Fig. 2. The model with the highest accuracy for the RF classifiers is the set of features from the PCC-RFFI iterative approach without dropping any attacks from the dataset, followed by the chi-2 model. The DT classifier with the highest accuracy is the chi-2 model, and for the GNB, it’s the PCC-RFFI without dropping any attacks.

The accuracy of the GNB is not similar to the other classifiers due to its probabilistic nature, which leads us to calculate the average for all classifiers. Table 5 and Fig. 3 show the average accuracy and precision for each model to be able to identify the best-fitting iterative feature selection approach.

Table 5. Average Accuracy and Precision for the sets of selected features

	accuracy	precision
RFFI-D	71.72%	61.38%
CHI2-D	85.98%	75.67%
PCC-D-RFFI	86.60%	94.67%
RFFI	86.86%	96.00%
CHI2	91.79%	89.64%
IG	92.32%	84.40%
PCC-D	95.90%	83.33%
PCC	96.95%	95.67%
PCC-RFFI	99.27%	97.60%

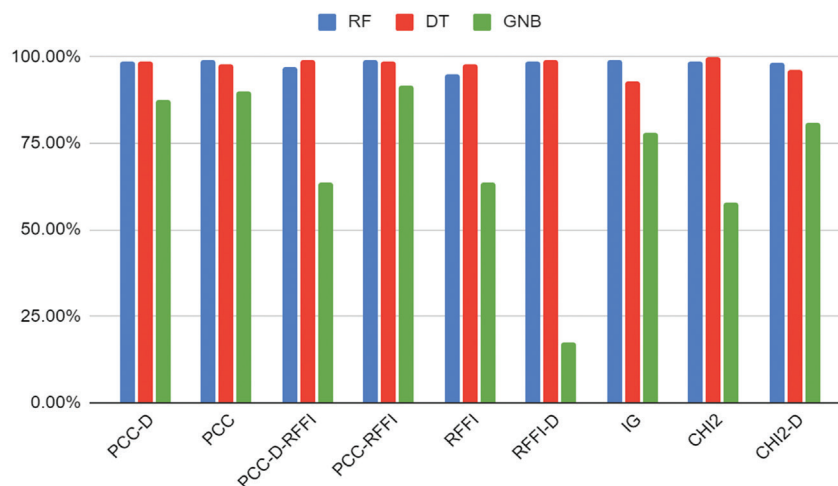


Fig. 2. Accuracy for different models and classifiers

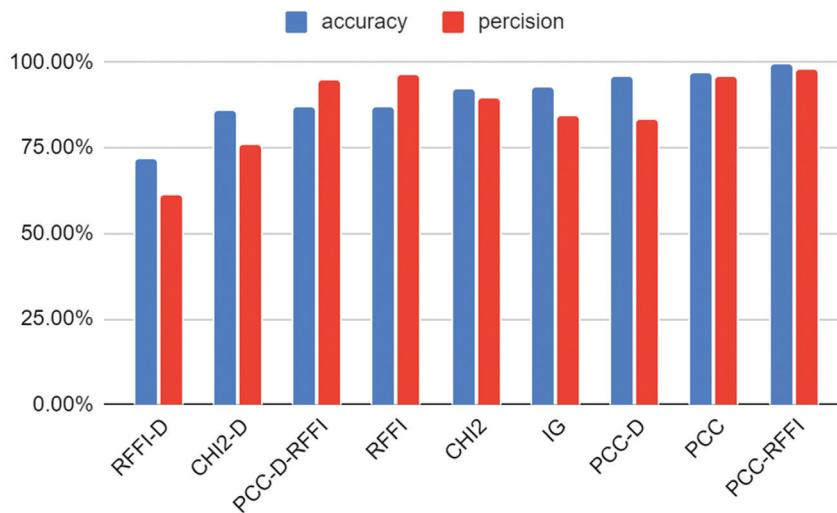


Fig. 3. Average accuracy and precision

5. CONCLUSION AND FUTURE WORK

Distributed Denial of services is one of the biggest problems we face nowadays when it comes to cloud computing and distributed environments in general, where machine learning techniques are considered one of the top ways to protect against them. The prevention and detection process of DDOS attacks using a machine-learning approach is divided into two main stages. The first is the feature selection stage, and the second is the ML classifiers trained to detect these attacks. Our focus through this framework was trying to find a more efficient way for selecting the essential features to increase the accuracy of any classifier and figuring out if the claim of dropping a percentage from the attacks in the datasets would improve the accuracy of selected classifiers or not. We created 9 sets of features from the available features in the dataset CICID2018 using 4 techniques, including PCC, RFFI, MI, and Chi-squared once, dropping 50% of the attacks and once without. This results in creating 2 sets of features for the PCC, 2 for the RFFI, 2 for the Chi2, and one for the MI, as dropping the attacks did not affect the resulting features. In the first round of feature selection, we based our selection process on the dependency of the features selecting the sets of features with the highest independence levels, which resulted from using the PCC approach. After that, the second selection iteration was based on the feature importance. We then compared the accuracy and precision of all the models on the DT, RF, and GNB classifiers. We calculated the average accuracy and precision for the three classifiers. In both cases, the highest average for the selected features was for the model RFFI proving that our iterative feature selection approach resulted in higher prediction accuracy. Our results also showed that, on the contrary, dropping the attacks didn't significantly impact the accuracy of the different classifiers with the other models. In our future work, we aim to test our iterative approach on more classifiers using more feature selection approaches and improve their performance.

6. REFERENCES

- [1] D. R. Praveen, A. N. Rimal, "DDOS Attack Detection Using Machine Learning", *International Journal of Emerging Technologies and Innovative Research*, Vol. 7, No. 6, 2020, pp. 185-188.
- [2] A. E. Kheder, A. M. Idress, "Adapting Load Balancing Techniques for Improving the Performance of e-Learning Educational process", *Journal of Computers*, Vol. 12, No. 3, 2017, pp. 250-257.
- [3] A. E. Khedr, A. M. Idrees, "Enhanced e-Learning System for e-Courses Based on cloud computing", *Journal of computers*, Vol. 12, No. 1, 2017.
- [4] S. Naiem, M. MARIE, A. E. Khedr, A. M. Idrees, "Distributed Denial Of Services Attacks And Their Prevention In Cloud Services", *Journal of Theoretical and Applied Information Technology*, Vol. 100, No. 4, 2022, pp. 1170-1181.
- [5] P. A. Narote, V. Zutshi, A. Potdar, "Detection of DDos Attacks using Concepts of Machine Learning", *International Journal for Research in Applied Science & Engineering Technology*, Vol. 10, No. VI, 2022, pp. 390-403.
- [6] S. Naiem, A. M. Idress, M. Marie, A. E. Khedr, "DDOS Attacks Defense Approaches And Mechanism In Cloud Environment", *Journal of Theoretical and Applied Information Technology*, Vol. 100, No. 13, 2022, pp. 4632-4642.
- [7] A. M. Idrees, M. H. Ibrahim, "A Proposed Framework Targeting the Enhancement of Students' Performance in Fayoum University", *International*

- Journal of Scientific & Engineering Research, Vol. 9, No. 11, 2018.
- [8] A. M. Mostafa, Y. M. Helmy, A. E. Khedr, A. M. Idrees, "A Proposed Architectural Framework For Generating Personalized Users' Query Response", Journal Of Southwest Jiaotong University, Vol. 55, No. 5, 2020.
- [9] A. M. Idrees, A. I. ElSeddawy, M. O. Zeidan, "Knowledge Discovery based Framework for Enhancing the House of Quality", International Journal of Advanced Computer Science and Applications, Vol. 10, No. 7, 2019, pp. 324-331.
- [10] A. M. Mohsen, H. A. Hassan, A. M. Idrees, "A Proposed Approach for Emotion Lexicon Enrichment.", International Journal of Computer, Electrical, Automation, Control, and Information Engineering, Vol. 10, No. 1, 2016.
- [11] H. A. Hassan, M. Y. Dahab, K. Bahnassy, A. M. Idrees, F. Gamal, "Arabic Documents Classification Method a Step towards Efficient Documents Summarization", International Journal on Recent and Innovation Trends in Computing and Communication, Vol. 3, No. 1, 2015, pp. 351-359.
- [12] A. E. Khedr, A. M. Idrees, A. Elseddawy, "Adaptive Classification Method Based on Data Decomposition", Journal of Computer Science, Vol. 12, No. 1, 2016, pp. 31-38.
- [13] A. E. Khedr, A. M. Idrees, A. I. El Seddawy, "Enhancing Iterative Dichotomiser 3 algorithm for the classification decision tree", WIREs Data Mining Knowledge Discovery, Vol. 6, 2016, pp. 70-79.
- [14] H. A. Hassan, M. Y. Dahab, K. Bahnassy, A. M. Idrees, F. Gamal, "Query answering approach based on document summarization", International Open Access Journal of Modern Engineering Research, Vol. 4, No. 12, 2014.
- [15] A. M. Idrees, M. H. Ibrahim, A. I. El Seddawy, "Applying spatial intelligence for decision support systems", Future Computing and Informatics Journal, Vol. 3, 2018, pp. e384-e390.
- [16] A. M. Idrees, "Towards an Automated Evaluation Approach for E-Procurement", Proceedings of the 13th International Conference on ICT and Knowledge Engineering, Bangkok, Thailand, 18-20 November 2015.
- [17] A. M. Idrees, A. E. Khedr, A. A. Almazroi, "Utilizing Data Mining Techniques for Attributes' Intra-Relationship Detection in a Higher Collaborative Environment", International Journal of Human-Computer Interaction, 2022.
- [18] M. Tan, A. Iacovazzi, N.-M. Cheung, Y. Elovici, "A Neural Attention Model for Real-Time Network Intrusion Detection", Proceedings of the IEEE 44th Conference on Local Computer Networks, Osnabrueck, Germany, 14-17 October 2019, pp. 291-299.
- [19] C. M. Nalayinil, D. J. Katiravan, "Detection of DDoS Attacks using Machine Learning Algorithms", Journal of Engineering technologies and innovative Research, 2022, pp. 223-232.
- [20] E. A. R. Coelho, "DDoS Detection using Machine Learning Techniques", Advanced information security, 2022, pp. 1-8,
- [21] M. Alduailij , Q. W. Khan, M. Tahir , M. Sardaraz, M. Alduailij, F. Malik, "Machine-Learning-Based DDoS Attack Detection Using Mutual Information and Random Forest Feature Importance Method", symmetry, Vol. 14, No. 1095, 2022.
- [22] Y.I. Kurniawan, F. Razi, B. Wijayanto, M. L. Hidayat, "Naive Bayes modification for intrusion detection system classification with zero probability", Bulletin of Electrical Engineering and Informatics, Vol. 10, No. 5, 2021, pp. 2751-2758.
- [23] A. Mishra, "Prediction approach against DDOS Attacks Based on Machine Learning Multiclassfier", arXiv:2204.12855, 2022.
- [24] Y. M. Swe, P. P. Aung, Hlaing, "A slow DDOS attack Detection Mechanism using Feature Weighing and Ranking", Proceedings of the 11th annual International Conference on Industrial Engineering and Operation Managment, Singapore, 7-11 March, 2021.
- [25] E. S. Alghoson, O. Abbass, "Detecting Distributed Denial of Service Attacks using Machine Learning Models", International Journal of Advanced Computer Science and Applications, Vol. 12, No. 12, 2021, pp. 616-622.
- [26] D. Alghazzawi , O. Bamasag, H. Ullah, M. Z. Asghar, "Efficient Detection of DDoS Attacks Using a Hybrid Deep Learning Model with Improved Feature

- Selection", *Applied Science*, Vol. 11, No. 11634, 2021, pp. 1-22.
- [27] C. I. f. Cybersecurity, "<http://www.unb.ca/cic/datasets/ids-2018.html>" (accessed: 2018)
- [28] I. Sharafaldin, A. H. Lashkari, A. A. Ghorbani, "Toward Generating a New Intrusion Detection Dataset and Intrusion Traffic Characterization", *Proceedings of the 4th International Conference on Information Systems Security and Privacy*, Portugal, 2018. pp 108-116.
- [29] A Realistic Cyber Defense Dataset (CSE-CIC-IDS2018), <https://registry.opendata.aws/cse-cic-ids2018> (accessed: 2022)
- [30] I. M. Nasir, M. A. Khan, M. Yasmin, J. H. Shah, M. Gabryel, R. Scherer, R. Damaševičius, "Pearson Correlation-Based Feature Selection for Document Classification Using Balanced Training", *Sensors*, Vol. 20, No. 6793, 2020.
- [31] J. R. Vergara, P. A. Estevez, "A review of feature selection methods based", *Neural Computing and Applications*, Vol. 24, 2014, pp. 175-186.
- [32] O. S. Bachri, "Feature selection based on Chi Square in Artificial Neural Network to Predict the Accuracy of Student study Period", *International Journal of Civil Engineering and Technology*, Vol. 8, No. 8, 2017, pp. 731-739.
- [33] G. Karatas, "The Effects of Normalization and Standardization an Internet of Things Attack Detection", *European Journal of Science and Technology*, No. 29, 2021, pp. 187-192,.
- [34] A. E. Khedr, A. M. Idrees, R. Salem, "Enhancing the e-learning system based on a novel tasks' classification load-balancing algorithm", *PeerJ Computer Science*, Vol. 7, 2021, p. e669.

NoSQL Databases: Modern Data Systems for Big Data Analytics - Features, Categorization and Comparison

Original Scientific Paper

Atul O. Thakare

CSE Department, Koneru Lakshmaiah Education Foundation,
Guntur District, Andhra Pradesh, India
aothakare@kluniversity.in

Omprakash W. Tembhurne

School of Computing, MIT Art Design and Technology University,
Pune, Maharashtra, INDIA
omprakash.tembhurne@mituniversity.edu.in

Abhijeet R. Thakare

MCA Department, Shri Ramdeobaba College of Engineering and Management,
Nagpur, Maharashtra, India.
thakarear@rkneec.edu

Soora Narasimha Reddy

CSE Department, Kakatiya Institute of Technology and Science,
Hasanparhty, Warangal, Telangana, India
snreddy75@gmail.com

Abstract – Because of the massive utilization of the world wide web and the drastic use of electronic gadgets to access the online world, there is an exponential growth in the information produced by these hardware gadgets. The data produced by different sources, such as smart transportation, healthcare, and e-commerce, are large, complex, and heterogeneous. Therefore, storing and querying this data, coined "Big Data," is challenging. This paper compares relational databases with a few of the popular NoSQL databases. The performance of various databases in executing join queries, filter queries, and aggregate queries on large datasets are compared on a single node and multinode clusters. The experimental results demonstrate the suitability of NoSQL databases for Big Data Analytics and for supporting large userbase interactive web applications.

Keywords: Unstructured Data, NoSQL Database, Horizontal Scaling, Vertical Scaling, CAP Theorem, Weak Consistency

1. INTRODUCTION

Traditional Relational and Object-Relational database approaches are ineffective at delivering the flexibility and scalability needed when processing enormous amounts of fast-moving structured, unstructured, and semi-structured data and supporting a large number of concurrent users. NoSQL databases fully handle these problems. The following two subsections go into further detail about these ideas.

1.1. LIMITATIONS OF RELATIONAL DATABASES IN SUPPORTING MODERN WEB APPLICATIONS

Since the beginning of computers, the relational model has been the standard for storing and retrieving data. The limits of relational databases were made clear by the exponential increase in internet users and the widespread adoption of new-generation online applications, which ultimately called for the developing of a new generation of No-SQL databases.

The reason why relational DBMSs are not well suited to support the requirements of modern web applications is their complete dependence on the fixed pre-defined schema, strict consistency rules (adherence to ACID properties) [1], need for joining several tables for query processing (which is difficult and slow when data grows huge and is stored distributedly across multiple servers), poor performance in terms of availability and scalability while dealing with large volumes of unstructured or semi-structured workloads.

1.2. PROBLEM STATEMENT AND SOLUTION FEATURES OF NOSQL DATABASES

In order to handle massive amounts of data and a very large user base, modern web-based applications have incorporated scale factors that have never been used before. Modern web applications must be able to serve vast numbers of concurrent users, respond quickly to a considerably large user base scattered throughout the globe, provides constant availability, manages a wide range of data, and update swiftly for new up-

dates and features. The advent of these new challenges posed by the term coined as "big data" (made up of structured, semi-structured, and unstructured data and described by its volume, variety, and velocity features) created a need for out-of-the-box horizontal scalability for today's data management systems. Additionally, we are dealing with a new community of applications where flexibility, scalability, availability, and cost are more crucial to application success than consistency in read-and-write operations. To ensure that our application exhibits the features mentioned above, the first requirement is that the data management systems be designed to run on distributed systems.

The last ten years have seen the rise of cloud computing as the most practical and economical alternative for meeting the ever-increasing storage and processing demands of modern online applications. As a result, most web applications nowadays are hosted in the cloud, where the available resources can change size in response to the application's needs. Additional servers can be provided to the website to handle increased demand in the event of a sudden rise in traffic (scaling out) [2], thus ensuring the system's constant availability and good performance. When the traffic returns to normal, the additional servers can be removed. NoSQL systems are faster and better equipped as they can take advantage of "scaling out," i.e., adding more nodes to the distributed system and distributing the additional load over newly added nodes.

The focus of this paper is to present the differences between relational databases and No-SQL databases as well as broad categorization of No-SQL data stores and to compare and analyze three popular No-SQL solutions – Cassandra, MongoDB, and CouchBase, and to highlight their differences with relational databases and other No-SQL data stores.

2. SQL VS. NOSQL

In order to tackle Big Data, the world has moved away from the "one size fits all" philosophy of RDBMS and toward the more flexible approach of No-SQL systems [3]. NoSQL databases give more freedom to design systems by studying the application's features and the type and volume of data it is dealing with, which is the cause for the shift in emphasis from traditional RDBMS to NoSQL database systems for handling Big Data.

2.1. NO-SQL SILENT FEATURES

No-SQL data stores are devised to scale horizontally and run on commodity hardware. The term "horizontal scalability" means the ability to distribute the data and the processing workload of database operations over many servers with no shared memory. On the other side, "vertical scalability"—the only way of expanding the capabilities of centralized RDBMS systems—means boosting the resources of the specialized server, which comes at a significant cost. Moreover, No-SQL data

stores can benefit significantly from cloud infrastructures from an implementation standpoint since they can be scaled and made available according to the application's needs [4].

2.2. DEVIATION FROM RELATIONAL DATABASES

The transactions in relational databases obey the ACID principle, which stands for Atomic, Consistent, Isolated, and Durability [5]. Many rows spanning many tables are updated as a single operation. This operation either succeeds or fails in its entirety, and concurrent operations are isolated from each other, so they cannot see any partial updates. Following these rigid consistency requirements could needlessly limit databases' ability to address the performance issues for a specific application. In distributed databases, following ACID properties strictly for all the transactions where many nodes handle the operations within the transaction creates complications. In relational databases, it is essential not to allow any application to view the inconsistent state of the data. Satisfying strict consistency requirements in distributed databases will force the system to ensure that communication channels maintain strict consistency of data and total synchronization of replica copies with the consistent data across the clusters of nodes. Performing this complicated task at each transaction without compromising on the availability of the system is highly difficult.

Because for each transaction, the system will have to wait till the updates inflicted by the transaction are propagated to all the other nodes hosting some portion of the affected data. Unlike relational databases, many NoSQL systems are not ACID compliant [6]. In some NoSQL databases which bypass the strict consistency rule of RDBMS, within the cluster, all writes are received by the MASTER, and synchronization of the SLAVE replicas with that of the MASTER data is carried out periodically. Hence the updates are eventually propagated. SLAVE nodes are always available to respond to read queries, because of which there are reduced guarantees of returning consistent results for all the read requests. The idea is that by relaxing on the strict ACID constraints, we can enable the database systems to improve on the other desirable characteristics like availability, scalability & fault tolerance. As already discussed, many of today's modern applications won't mind not following strict ACID properties. But, its performance will be badly affected if the uptime for all the requesting clients is not maintained. In replacement to ACID, No-SQL databases follow BASE [7] semantics which are explained below:

- (BA) Basically Available: an application is ready to accept read/write requests all the time.
- (S) Soft state: Results may not always be based on consistent data (no consistency guarantee).
- (E) Eventual consistency: The system assures that data will become consistent at some later point.

When a distributed database system is installed on a group of server computers, high availability may be achieved by maintaining replica copies of the data on several machines within the group and updating those copies whenever a write operation is carried out on the MASTER machine [8]. Consider a scenario in which the network links between the cluster nodes are broken, and the network is divided into several fragments that cannot be reached by one another. In this situation, the database system is kept accessible for clients by making one network segment active and disconnecting the others. Additionally, this stops the nodes of disconnected segments from responding to client queries, preventing the delivery of inconsistent results.

After receiving the missed writes (updates) from the active cluster nodes, the inaccessible cluster segments begin serving the directed traffic of client requests. In a distributed database, the capacity of the clusters to continue operating despite communication breakdowns is referred to as partition tolerance [9]. The CAP theorem, which argues that in partition tolerance, one must choose between consistency and availability, explains the complicated trade-off between consistency and availability in distributed databases [10]. Another way around, the system cannot have all the following three properties at any given time: consistency (all servers having a consistent version of the data), availability (each request receives a timely response), and partition-tolerance (as the information is distributed and replicated, even if there is a failure in a part of the system, the system continues to work). No-SQL systems frequently compromise consistency to some degree to achieve high availability [11].

Table 1. Comparison of RDBMS and NoSQL

Feature	RDBMS	NoSQL
Data	Structured	Structured, Unstructured, Semi-structured
Schemas	Fixed	Dynamic
Scalability	Vertical	Horizontal
Compliance	ACID properties	BASE properties
Architecture	Centralized	Distributed
Consistency	Strict	Eventual
Query Language	SQL	OO API, SQL like
Performance	Slow	Fast
Best suited for	banking, financial transactions	large-scale web applications, Sensor data

2.3. NO-SQL DATA MODELS

Most No-SQL databases run on distributed systems and fall into four categories.

1. Key-Value Stores

- a. The data is stored in the form of key-value pairs.
- b. Keys are identifiers (unique in the namespace), and values are data associated with the keys. Keys are used for looking up data.

- c. For fast lookups, keys are hashed [12].
- d. In key-value pair, the value may be data or another key.
- e. Supports querying, modifying data through primary key, and mass storage.
- f. Provides higher concurrency and higher query speed.
- g. Best suited when high-speed and highly scalable caches are needed.
- h. e.g., Amazon Dynamo, Azure Cosmos, Riak, Redis, etc.
- i. Suitable for applications that use a single key to access data, e.g., online shopping cart.

2. Document Stores

- a. Documents contain contents as well as formatting information (JSON, XML).
- b. Documents contain information in key-value pairs.
- c. Keys are a string of characters, and values can be any basic data type or structure.
- d. Collections are the list of documents. Documents in the same collection can follow different structures.
- e. A document can have embedded documents or arrays inside it.
- f. Supports indexing, designed for scalability and high performance.
- g. In addition to CRUD (create, read, update, delete) operations, it supports filtering collections, joining multiple collections, performing groupings, aggregations, etc.
- h. Best suited when fast and constantly growing data.
- i. e.g., MongoDB, Couchbase, CouchDB, RavenDB etc.
- j. Suitable for content management systems, e.g., social networking sites, blogging platforms, etc.

3. Wide Column-Databases

- a. In Column- Databases basic unit of storage is a column, i.e., data is organized in column families.
- b. Column data is stored one after the another. Hence, the last item of the first column is followed by the first item of the second (next) column, and so on.
- c. A query language for column-family databases supports CRUD and creates column-family operations [13].
- d. An index is created on columns, reducing the I/O cost for the queries accessing columns of data.
- e. More suitable for data warehouses where most of the analytical queries involve aggregations that need to scan data from a few columns but for all the rows.
- f. e.g., HBase, Cassandra, Accumulo, Hypertable, etc.

4. Graph Databases (GD)

- a. A collection of nodes (vertices) and relationships (edges) tagged with the information forms a graph [14].
- b. A node is an object that has an identifier and a set of attributes. A relationship is a link between two nodes that contain features about that relation.
- c. It provides fast operations as it models adjacency between objects.
- d. Convenient to use for representing social networking media, creating recommendation systems, and pattern mining.
- e. It is best suited when the relationship between entities is more important than the entities themselves.
- f. The data of popular applications like Facebook, Twitter, LinkedIn, etc., are modeled using graphs.
- g. Neo4J, Infinite Graph, InfoGrid, HyperGraphDB, etc.

3. POPULAR NOSQL DATABASES

A few of the popularly used NoSQL databases are described below:

1. MongoDB is developed in the cloud. It is a scalable, open-source No-SQL database that is document-oriented, schema-free, and simple to use. It aims to fulfill the needs of expansive web applications by implementing highly parallel and globally scattered database systems. MongoDB supports auto sharding, where it splits the data collections and stores the different data chunks among the available servers [15]. Additionally, it provides features like high performance, partition tolerance, automatic scalability, and replication (uses Master-Slave replication) [16].
2. Cassandra is a distributed storage system for structured data management that can handle large volumes of data. Scalability, high performance, high availability, high reliability, applicability, and replication are a few of its essential characteristics [17]. Its feature of executing map-reduce jobs in hadoop clusters is ideally suitable for mission-critical applications [18]. It is incrementally scalable, and data is partitioned and distributed among the nodes of a cluster in a fashion that allows repartitioning and redistribution.
3. CouchDB: CouchDB is sometimes called a "Cluster of unreliable commodity hardware." Initially, CouchDB was implemented in C++ but later ported to the Erlang OTP for implementing thoroughly lock-free concurrency of read-write requests. CouchDB databases consist of documents made up of fields with a key, i.e., name and value. The value may be a number, boolean, date, string, ordered list, or associative map [19]. Documents may contain references to other documents (embedded documents). CouchDB is distributed; its other essential concepts are

schema-free, views, distribution, replication, map-reduce, etc. The cluster is conveniently scaled horizontally and has no single point of failure. Clusters are designed to allow live changes means there is no downtime during database updates and software-hardware upgrades [20]. If you're scaling reads over numerous servers, a write must happen on them. It also offers incremental replication with bidirectional conflict detection and resolution.

4. HBase is a Hadoop-based open-source system that runs on the fault-tolerant Hadoop Distributed File System (HDFS). HDFS uses a master-slave architecture that consists of name nodes (manages the file system) and data nodes (stores and replicate data) [21]. In a hadoop cluster, coordination between different nodes is maintained by one type of node called Zookeeper (single point of failure for HBase). The zookeeper stores the location of the META table. The client will query the META table to get the region server corresponding to the row key it wants to access. The client caches this information along with the META table location. HBase has two types: read cache (BlockCache) and write cache (MemStore). In HBase, data is stored in tables, but it is schema-less. Tables have lexicographically indexed multidimensional row keys and several column families, each having a set of column qualifiers, which stores the fundamental data element [22]. Hence a combination of the table name, row key, column family, and column qualifier define the access. The system stores different versions of a data item, each assigned with an individual timestamp. This feature of HBase attributes to its high write performance. In addition to hadoop services, HBase also has servers for managing metadata about the distribution of table data. The primary storage unit (shard) in HBase is Region which is managed by RegionServers (responsible for administrative activities). As the size of the data grows, new regions are created. The subset of the rows of a column family, ordered by the row key, are assigned to a particular region. An asynchronous write-ahead log (WAL) is used in HBase cluster replication which eventually targets consistency. The META table is an HBase table that lists all regions in the system. This unique HBase Catalog table holds the location of all areas in the cluster.

3.1 COMMON KEY DESIGN CHARACTERISTICS OF NO-SQL DATABASES

Following are the critical characteristics of NoSQL databases that support Big Data Analytics.

1. Scalability: ability to efficiently meet the needs for varying workloads in terms of resources and performance. A Spike in the number of users triggers the application running on a cloud to acquire additional servers. As the spike subsides and traffic becomes normal, additional servers are released [23].

2. **Flexibility:** Before starting a project, RDBMS database designers must design all the tables, table relationships, and schemas required to support an application. Unlike relational databases, which demand the schema to stand defined before adding any data, No-SQL is schema-less and hence more capable of handling significant variations in the data structures.
3. **Availability:** Most of the No-SQL systems are almost always ready to accept new read or write requests. In the back end, No-SQL databases are usually deployed on a distributed system consisting of multiple low-cost servers, each having an identical copy (replica) of the database. The large-scale web application also runs on a cluster of servers. As the backup servers keep replicated copies of data from the primary server, if a primary server goes down for maintenance or fails, the secondary servers elect the new primary, and thus the system remains available. In case of a sudden increase in the number of users, cluster expands in terms of compute, storage & bandwidth capacity to maintain system performance.
4. **Replication:** To prevent automatic fail-over caused by events like server or network failures, MongoDB utilizes an architecture called replica set to distribute copies of data among computers in the cluster. Scaling the number of database reads is another benefit of replication. Database reads in read-intensive applications can be distributed among the computers in the replica set cluster. Replica sets typically include a primary server and a backup server. If a master-slave arrangement is used, the primary server can handle both read and write requests, while the secondary servers can only handle read requests. Each write on the primary will be transmitted to all the secondary servers. In other words, reads from an alternative location will succeed only when they receive all the changes made to the primary. If the primary server fails, one of the secondary servers will be elected as a new primary [24].
5. **Cost:** Most of the No-SQL databases are available as open source and hence are free, thereby avoiding the issues like licensing, charging the users, etc.
6. **Sharding (Partitioning) Data splitting (per-collection basis) over many servers with an emphasis on order preservation will increase performance.** To do this, we partition the dataset into various servers and replicate each portion over many servers. We can significantly increase the read and write speed of the system since different users are accessing distinct portions (shards) of the dataset. A server can be a slave for a few shards and a master for few others.
7. **Map-Reduce:** MapReduce is a programming model in which computations are expressed as a map and reduce functions, which are impulsively parallelized across numerous machines within a cluster-based computing environment. It can perform calculations on large volumes of data in a reasonable amount of

time by distributing and parallelizing it across multiple devices in the cluster. The computations take a set of input key-value pairs and produce a set of intermediate key-value pairs accepted by the set of reduced functions. The reduced functions merge the values by grouping using the intermediate keys (using operations like counting, summing, or averaging) to produce possibly smaller values.

3.2 HADOOP BIG DATA FRAMEWORK

- Hadoop is a valuable open-source framework for developing distributed applications that process large amounts of data [25]. Hadoop is intended to run on clusters of commodity hardware (or cloud services such as Amazon's EC2), hence is capable of handling hardware malfunctions and failures. In hadoop, large data sets are divided into more number of smaller (64 MB) blocks which are spread to live in the cluster of several machines using the Hadoop Distributed File System (HDFS).
 - HDFS achieves this using its two components, NameNode (stores metadata) and DataNodes (stores portion of actual data). Depending on the degree of replication, replica copies of each block will be maintained in the hadoop. If NameNode is the master daemon, Data Nodes are slave daemons. Hadoop NameNode uses the rack information to distribute replicas across racks (avoiding multiple copies of the same block on the same rack) to ensure fault tolerance in case of rack failure. The other side of this rack awareness replication policy is the increased I/O cost due to the movement of blocks across racks.
 - Cluster machines can access the distributed dataset in parallel, thus providing high throughput [26]. For computations, hadoop uses distributed data processing framework called MapReduce, which uses the move code to data principle [27]. Hence, a portion of data is computed on the same node where it resides. MapReduce has two phases: the map phase and the reduce phase. The map phase uses one or more mappers to process the input data, and the reduce phase uses zero or more reducers to process the data output during the map phase.
- 1. Map phase**
 - a. Split the input data into several data segments.
 - b. Generate and assign a separate map task for each data segment.
 - 2. Distribute the map tasks across the clusters of nodes.**
 - 3. Run the map tasks in the distributed framework.**
 - a. Each map task runs on the disjoint set of input key-value pairs.
 - b. Each map task outputs partially consolidated data in output key-value pairs.

- c. Output key-value pairs are also called intermediate key-value pairs.

4. Reduce phase

- a. The outputs of map tasks (intermediate data set) are sorted and segmented.
- b. Segmentation is done so that values associated with the same key belong to the same segment and are sent to the same reducer.
- c. Reducers reduce the number of values associated with a particular key.
- A hadoop cluster can have only one JobTracker and several Task-Trackers. JobTracker accepts the client's MapReduce job submission, creates, and coordinates job tasks to the TaskTrakers, finds the location of data from the NameNode, schedules, and monitors the functions on the TaskTracker, and handles the failed TaskTracker tasks [28, 29].

4. EXPERIMENTATION

The experiment is performed by installing the Hadoop cluster on three machines with 11th Gen Intel(R) Core (TM) i5 processor, 8 GB RAM, and SSD. The Hadoop ecosystem is installed with Hbase, Cassandra, MongoDB, and CouchDB as NoSQL databases and MySQL as a relational database on the ubuntu platform on VM VirtualBox. Experimentation involves the execution of a few basic CRUD (Create, Read, Update, Delete) queries along with complex join, filter, and aggregate queries using a single node and the multinode cluster. The task of creating big datasets is automated through PL/SQL scripts. The generated datasets are imported into multiple database systems, and semantically equivalent queries are executed on each. The results are used to gain insights into the performance efficiency of different database systems in different scenarios. MySQL and HBASE experimentations are performed on the employee database, whose schema is shown in following Fig. 1.

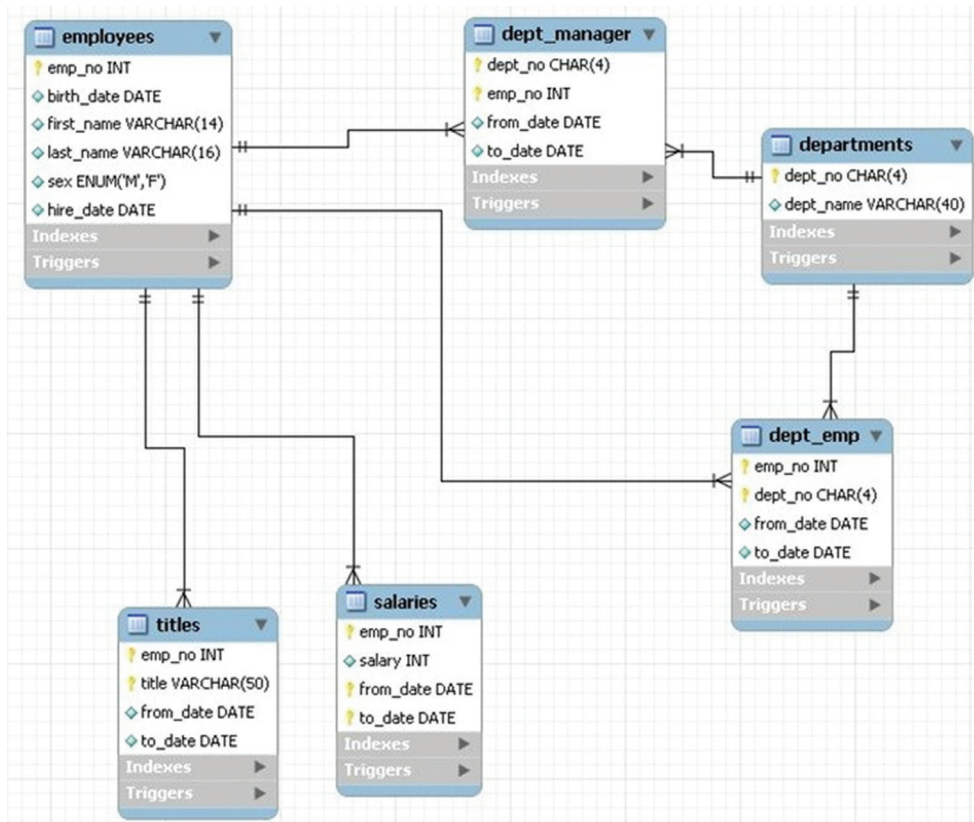


Fig. 1. Employee Dataset schema

A few examples of select and update queries executed on MySQL and HBASE are,

Find the total salary dispensed by each department.

- List the employees who joined after 2019.
- List all employees in terms of increasing or decreasing salaries.
- Increment the salary of each employee by 10 % of the current salary.

MongoDB, Cassandra, and CouchDB-based experimentations are performed on the Movie database (structure derived from <https://www.kaggle.com/harshitshankhdhar/imdb-dataset-of-top-1000-movies-and-tv-shows>). Data is generated using PL/SQL script, which generates random values (from the predefined set of values extracted from the web-crawled data) having attributes: Title, ReleaseDate, Runtime, Director, Star1, Star2, Star3, and GrossIncome.

A few examples of select, join, and aggregate queries executed on the Movie database are as follows:

- Report the director-wise total income from the short movies (maximum runtime 2 hours).
- Display the year-wise topmost movies in terms of Gross Income.
- Find out the number of movies in which the director is also a leading actor (Star1).
- List all the movies released in the year 2019 in ascending order of their release date.

4.1 EXPERIMENTATION RESULTS

As shown in Tables 2 & 3, although the insertion performance of HBASE (using HIVE) is less than MySQL for the examined datasets, the batches of filter, join, and aggregate queries run several times faster on HBASE multinode cluster in comparison to MySQL. With the increase in the number of nodes in the HBase cluster, the performance difference in further processing the batch of queries is expected to increase substantially with the scaling up of HBASE cluster performance. The HBASE execution time diminishes by approximately half when the cluster size doubles.

Table 2. Table-wise average Insertion time of MySQL and HBASE (* - Single Node, # - Multinode Cluster)

Database	No of rows	employees	salary	titles	dept_emp
MySQL	100000	1375 ms	1328 ms	1015 ms	3024 ms
HBASE*	100000	40256 ms	40307 ms	40187 ms	40078 ms
MySQL	200000	3284 ms	3313 ms	2671 ms	4684 ms
HBASE*	200000	77461 ms	77465 ms	76554 ms	77123 ms
HBASE#	100000	2 m 51 s	2 m 25 s	2 m 45 s	2 m 55 s

Table 3. MySQL versus HBASE: Average execution time for the filter, join and aggregate queries. (* - Single Node, # - Multinode Cluster)

Database	No of rows	Where Queries (avg)	Join Queries (5000)	Aggregate Queries (5000)
MySQL	100000	5 m 3 s	7 m 35 s	9 m 3 s
HBASE*	100000	12 s	8 m 45 s	10 m 35 s
MySQL	200000	9 m 52 s	18 m 38 s	14 m 4 s
HBASE*	200000	22 s	16 m 40 s	21 m 52 s
HBASE#	100000	1 s	2 m 45 s	3 m 35 s

Table 4. HBASE Performance -- Static vs. Dynamic Cluster (SSN – Static Single Node, DSN – Dynamic Single Node)

Database	No of rows	Where Queries (avg)	Join Queries (5000)	Aggregate Queries (5000)
HBASESSN	100000	12 s	8 m 45 s	10 m 35 s
HBASESSN	200000	22 s	16 m 40 s	21 m 52 s
HBASEDSN	200K–300K	36 s	25m 47s	32m 25s

With HBASE, for the same number of nodes, static cluster performance in terms of scale-up and resource utilization is better than dynamic cluster (shown in Table 4), where the changes in the configuration of the

cluster are applied automatically in the online mode. MongoDB accomplishes a higher throughput on dynamic clusters for all three types of queries (shown in Table 5).

Table 5. MongoDB Performance -- Static vs. Dynamic Cluster (SSN – Static Single Node, DSN – Dynamic Single Node, SC – Static Cluster, DC – Dynamic Cluster)

Database	No of rows	Insertion	Where Queries (avg)	Aggregate Queries
MongoDBSSN	100000	13.8 s	0.0250 s	11.84 m
MongoDBSSN	200000	16.05 s	0.0104 s	24.43 m
MongoDBSSN	400000	26.13 s	0.0103 s	49.73 m
MongoDBDSN	400000	28.5 s	0.0137 s	52.06 m
MongoDBSC	400000	13.93 m	13.31 m	1.81 h
MongoDBDC	400000	11.7 m	0.012 s	52.28 m

Table 6. MySQL vs CouchDB

Database	No of rows	Where Queries (avg)	Join Queries (5000)	Aggregate Queries (5000)
MySQL	100000	4.7 m	38.7 m	44.8 m
CouchDB*	100000	15.3 m	7.2 m	13.9 m
MySQL	200000	5.4 m	8.8 m	87.2 m
CouchDB*	200000	15.3 m	8.8 m	12.0 m
MySQL	300000	5.7 m	109.8 m	127.2 m
CouchDB*	300000	15.6 m	9.9 m	12.1 m
MySQL	400000	5.6 m	147.1 m	170.1 m
CouchDB*	400000	15.9 m	11.3 m	11.8 m

The experimental observations (shown in Table 6) show that filter queries take more time in CouchDB whereas join and aggregate queries run faster in CouchDB than in MySQL database. With the inbuilt cache and use of map-reduce, CouchDB read-write

performance is good and is suitable for interactive applications (Table 6).

MongoDB with Apache Storm cluster with static/dynamic data is far more time efficient than MySQL, and its performance scales up nicely with the expansion in the number of nodes in the cluster. Because of this, MongoDB has high success in running a large number of queries in a short time. MongoDB's performance for join queries (complex queries in general) is far superior in comparison to its SQL as well as NoSQL counterparts because of its extensive use of subdocuments and embedded lists, thereby avoiding the computations for searching the matched documents in the other collections (results are shown in Table 7). With update-heavy workloads, MongoDB performance dips, but on read-heavy workloads, its performance remains leading compared to other databases.

MongoDB's scaling performance is better than Cassandra's insertion time and retrieval time (total and average time) for various data sizes on single and multinode clusters are shown in Tables 8 and 9.

Table 7. MySQL vs. MongoDB

Database	No of rows	Insertion	Where Clause	Aggregate Clause	Join Clause
MySQL	100000	1.45 h	2.11 m	31.5 s	4.2 m
MySQL	200000	2.88 h	7.99 m	51.1 s	14.91 m
MySQL	400000	6.01 h	14.78 m	1.95 s	41 m
MongoDB*	400000	303.11 s	0.023 s	1.408 h	@
MongoDB#	400000	292.008 s	1.009 s	2.405 h	@

The number of queries per clause= 10000,

*: MongoDB performance using Apache storm Single Node with static data

#: MongoDB performance using Apache storm Cluster with dynamic data.

@: Data Stored in one collection only with embedded documents and lists of other papers. All related data are in a single collection only. No joins of multiple collections were performed.

For the Cassandra database, in tables 8 and 9, the total and average insertion time and total and average retrieval time for a number of rows (in the 2nd column) are shown (in the 3rd and 4th column [from the left], respectively) for both single node and multinode cluster.

Experimentation shows on importing the Movie dataset and running the batches of queries (read-intensive, write-intensive, and read-write mixed) in HBase, MongoDB, and Cassandra that, mostly HBASE is more efficient in handling write-intensive workloads.

In contrast, Cassandra is more efficient while dealing with read-intensive workloads. For read-intensive workloads (above 80% reads), MongoDB gives better performance than all the other databases. HBase performance is better on workloads having mixed read and write requests. On balanced read-write workloads (50 % each), MongoDB shows better scaling behavior when compared to Cassandra (linear). When tried on read-intensive workloads, Cassandra shows significantly high disk I/O compared to CouchDB and MongoDB.

Table 8. Cassandra read-write performance statistics on a single node cluster

Performance statistics on Cassandra			
Retrieval type	Number of Rows	Insertion time total) (avg) ms	Retrieval time (1000 queries) (total)/(avg) ms
static	100000	79152 (0.26384)	72047 (72)
static	200000	164342 (0.8217)	21873 (81)
static	300000	242731 (0.829)	83212 (83)
dynamic	700001-800000	110947 (1.1)	138176 (138)
dynamic	800001-1000000	218140 (1.0)	156368 (156)
dynamic	1000001-1300000	315755 (1.0)	198109 (198)

Table 9. Cassandra read write performance statistics on three node cluster

Cassandra Cluster 3-node cluster (replication factor: 3)			
Retrieval type	Number of Rows	Insertion time (total)/(avg) ms	Retrieval time (1000 queries) (total)/(avg) ms
static	1-100000	159034ms/1.590ms	54591ms/54.591 ms
static	100001-300000	229374ms/1.147ms	54903ms/54.903ms
static	300001-600000	346375ms/1.1546ms	66778ms/66ms
dynamic	600001-700000	154942ms/1.549ms	100469ms/100.469ms
dynamic	700001-900000	271857ms/1.359ms	77959ms/77.959ms
dynamic	900001-1200000	470318ms/1.568ms	73065ms/73.065ms

In general, using the traditional optimization techniques in the distributed databases, and deploying the databases on scalable distributed frameworks like the cloud, makes the application more efficient in query execution as well as in maintaining system performance in the presence of a fluctuating number of application users which eventually leads to fluctuating sizes of the query workloads. It is observed that within the available scope of further decomposition of the workload and workload queries, increasing the number of cores on a single node or the number of nodes in the cluster directly impacts the throughput and speed of query execution. However, the relative performances of different types of databases vary with the change in the read-write latencies, which is somewhat characterized by patterns of workload queries (read-intensive, write-intensive, read-write-mixed).

Experimentation shows that HBASE is efficient in handling write-intensive workloads and read-write mixed workloads, whereas Cassandra, MongoDB is more efficient when dealing with read-intensive workloads. From the performance behavior, it can be safely concluded that MongoDB is best suitable for read-intensive workloads. In contrast, HBASE is a better choice for writing dominant and balanced read-write workloads. With the inbuilt cache supporting the map-reduce framework, CouchDB read-write performance is suitable for interactive applications.

5. CONCLUSION

This paper contains a comparison of SQL databases and few NoSQL databases concerning their architectures, underlying working principles, advantages and disadvantages, and suitability in different application domains. Also, it demonstrates the characteristics of popular NoSQL databases using experimental results. NoSQL databases are designed to store and analyze big data generated by sources like social media, e-commerce websites, sensors, etc. They are instrumental in several IOT-based smart systems (e.g., Health Monitoring Systems, Traffic Monitoring Systems, etc.). To handle large volumes, velocity, and variety of data and to cater to advanced requirements like scalability, availability, and fault tolerance, databases need to have the capacity to compute, store, access, and analyze data in a distributed fashion. As per our experimental results,

it is evident that NoSQL databases fit these demands. Compared to relational databases, they are more suitable for dealing with big data tasks on scalable, elastic and fault-tolerant platforms like the cloud.

6. REFERENCES

1. E. Lotfy, A. I Saleh, H. A. El-Ghareeb, H. A. Ali, "A Middle Layer Solution to Support ACID properties for NoSQL Databases", *Journal of King Saud University-Computer and Information Sciences*, Vol. 28, No. 1, 2016, pp. 133-145.
2. R. Cattell, Scalable, "SQL and NoSQL data stores", *ACM Sigmod Record*, Vol. 39, No. 4, 2011, pp. 12-27.
3. R. Hecht, S. Jablonski, "NoSQL evaluation: A use case-oriented survey", *Proceedings of the International Conference on Cloud and Service Computing*, Hong Kong, China, 12-14 December 2011, pp. 336-341. 10.1109/CSC.2011.6138544.
4. C. Bazar, C. S. Iosif, "The transition from Rdbms to NOSQL. A Comparative Analysis of three Popular Non-Relational Solutions: Cassandra, Mongoddb and Couchbase", *Database Systems Journal*, Vol. 5, No. 2, 2014, pp. 49-59.
5. J. Sivakumaran, S. Z. Ali, "RDBMS Current Challenges and Opportunities with NoSQL to NewSQL", *Proceedings of the 3rd Middle East College Student Conference*, Muscat, Sultanate of Oman, 31 December 2017.
6. M. A Mohamed, O. G. Altrafi, M. O. Ismail, "Relational vs. NoSQL Databases: A Survey", *International Journal of Computer and Information Technology*, Vol. 3, No. 3, 2014, pp. 598-601.
7. J. Guia, V. G. Soares, J. Bernardino, "Graph Databases: Neo4j Analysis", *Proceedings of the 19th International Conference on Enterprise Information Systems*, Porto, Portugal, 26-29 April 2017, pp. 351-356.

8. M. Houcine, G. Belalem, K. Bouamrane, "Comparative study between the MySQL relational database and the MongoDB NoSQL database", *International Journal of Software Science and Computational Intelligence*, Vol. 13, No. 3, 2021, pp. 38-63.
9. M. Stonebraker, "SQL databases v. NoSQL databases", *Communications of the ACM*, Vol. 53, No. 4, 2010, pp. 10-11.
10. D. G. Chandra, "BASE analysis of NoSQL database", *Future Generation Computer Systems*, Vol. 52, 2015, pp. 13-21.
11. D. Glushkova, P. Jovanovic, A. Abelló, "Mapreduce performance model for Hadoop 2. x.", *Information Systems*, Vol. 79, 2019, pp. 32-43.
12. C. Candel, D. S. Ruiz, J. J. Garcia-Molina, "A Unified Metamodel for NoSQL and Relational Databases", *Information Systems*, Vol. 104, 2021, p. 101898.
13. R. Sellami, B. Defude, "Complex queries optimization and evaluation over relational and NoSQL data stores in cloud environments", *IEEE Transactions on Big Data*, Vol. 4, No. 2, 2017, pp. 217-230.
14. M. A. Elsabagh, "NO SQL Database: Graph database", *Egyptian Journal of Artificial Intelligence*, Vol 1, No. 1, 2022, pp. 1-7.
15. R. Gyorodi, G. Gyorodi, A. Pecherle, "A comparative study: MongoDB vs. MySQL", *Proceedings of the 13th International Conference on Engineering of Modern Electric Systems*, Oradea, Romania, 11-12 June 2016, pp. 1-6.
16. K. Orend, "Analysis and Classification of NoSQL databases and Evaluation of their Ability to Replace an Object-Relational Persistence Layer", *Architecture*, Vol. 1, 2010, pp. 1-100.
17. Apache Cassandra (TM) 4.0, The Documentation, <https://cassandra.apache.org/doc/latest/> (accessed: 2021)
18. Chebotko, A. Kashlev, S. Lu, "A Big Data Modeling Methodology for Apache Cassandra", *Proceedings of the IEEE International Congress on Big Data*, New York, NY, USA, 27 June - 2 July 2015, pp. 238-245.
19. N. D. Bhardwaj, "Comparative Study of Couchdb and Mongoddb–NoSQL Document-Oriented Databases", *International Journal of Computer Applications*, Vol. 136, No. 3, 2016, pp. 24-26.
20. R. Zafar, E. Yafi, M. F. Zuhairi, H. Dao, "Big Data: The NoSQL and RDBMS review", *Proceedings of the International Conference on Information and Communication Technology*, Bandung, Indonesia, 25-27 May 2016, pp. 120-126.
21. Apache HBase, The Documentation, <https://hbase.apache.org/> (accessed: 2021)
22. A. Gupta, S. Tyagi, N. Panwar, S. Sachdeva, U. Saxena, "NoSQL databases: Critical analysis and comparison", *Proceedings of the International conference on computing and communication technologies for smart nation*, New York, NY, USA, 12-14 October 2017, pp. 293-299.
23. A. Krechowicz, S. Deniziak, G. Łukawski, "Highly scalable distributed architecture for NoSQL datatore supporting strong consistency", *IEEE Access*, Vol. 9, 2021, pp. 69027-69043.
24. J. K. Chen, W. Z. Lee, "An Introduction of NoSQL Databases based on their categories and application industries", *Algorithms*, Vol. 12, No. 5, 2019, pp. 106-123
25. J. Anuradha, "A Brief Introduction on Big Data 5Vs Characteristics and Hadoop Technology", *Procedia computer science*, Vol. 48, No. 1, 2015, pp. 319-324.
26. J. Zakir, T. Seymour, K Berg, "Big Data Analytics", *Issues in Information Systems*, Vol. 16, No. 2, 2015.
27. J. Pokorny, "NoSQL databases: a step to database scalability in web environment", *Proceedings of the 13th International Conference on Information Integration and Web-based Applications and Services*, New York, NY, USA, 5-7 December 2011, pp. 278-283.
28. J. Dittrich, J. A. Quiane-Ruiz, "Efficient Big Data Processing in Hadoop MapReduce", *Proceedings of the VLDB Endowment*, Vol. 5, No. 12, 2012, pp. 2014-2015.
29. B. Jose, S. Abraham, "Performance analysis of NoSQL and relational databases with MongoDB and MySQL", *Materials today: Proceedings*, Vol. 24, No. 3, 2020, pp. 2036-2043.

Design and Implementation of a Simulator for Precise WCET Estimation of Multithreaded Programs

Original Scientific Paper

P. Padma Priya Dharishini

Department of Computer Science and Engineering
Ramaiah University of Applied Sciences
Bangalore, India
padmapriya.cs.et@msruas.ac.in

P. V. R. Murthy

Department of Artificial Intelligence and Data Science
Nitte Meenakshi Institute of Technology
Bangalore, India
pvr.murthy@nmit.ac.in

Abstract – Significant attention is paid to static analysis methods for Worst Case Execution Time Analysis of programs. However, major effort has been focused on WCET analysis of sequential programs and only a little work is performed on that of multithreaded programs. Shared computer architectural units such as shared instruction cache pose a special challenge in WCET analysis of multithreaded programs. The principle used to improve the precision of shared instruction cache analysis is to shrink the set of interferences, from competing threads to an instruction in a thread that may be accessed from shared instruction cache, using static analysis extended to barriers. An Algorithm that address barrier synchronization and used by the simulator is designed and benchmark programs consisting of both barrier synchronization and computation task synchronization are presented. Improvements in precision upto 20 % are observed while performing the proposed WCET analysis on benchmark programs.

Keywords: Worst Case Execution Time; Shared Instruction Cache Analysis; Multithreaded Program, Multicore Architecture

1. INTRODUCTION

Today Real Time Embedded Systems (RTES) are vastly used in avionics, automotive and tele-communications domains. In RTES, correctness of a system not only depends on its logical behavior but also computation time. In hard real time systems, missing deadlines can cause catastrophic damage. Multicore architectures are used in RTES domain due to their high processing power and concurrency in applications. For example, in night view assist multi-threaded in automotive environment, reading data from sensors, processing video streams and raising warning when an obstacle is detected on road happen concurrently.

Schedulability analysis is used to verify the capabilities of RTES to meet deadlines. All schedulability analyses assume that upper bound of execution of each program i.e., Worst Case Execution Time (WCET) is known. However, deriving tight and safe WCET of a program on multicore architecture is difficult because

of shared hardware resources such as cache memory, buses and Input/Output. There may be unpredictable delays in the execution of the program due to contention at shared resources. One of the main factors of unpredictability is due to cache memory. There are two or three levels of cache memory placed between core and main memory to bridge the gap of high-speed processor with low-speed main memory. L1 cache memory is the smallest cache memory close to the processor and it is private for each core, while larger L2 is shared between cores. In the case of an L1 cache miss (requested memory block not in L1 cache memory) then, memory block may be fetched from higher level of memory hierarchy(L2 cache). The memory access latency to be computed due to conflicts from other cores resulting in the removal of memory block from shared L2 instruction cache plays a critical role in the estimation of precise WCET of a multithreaded program. The problem of estimating worst case latency in turn to estimate WCET of a multithreaded program is motivated in this paper for shared instruction caches.

A static analyzer is designed in [1] using Hoare's Communicating Sequential Processes (CSP) [2] to compute WCET of a multithreaded program and is based on synchronized parallel processes arising from synchronization calls to wait() and notify(). A reference thread is one for which WCET is being estimated and instruction accesses in it encounter competition for shared instruction cache from parallel processes in other threads. The conflicts for any instruction I in reference thread are encountered only from the identified parallel processes that run parallel with I . A gap identified in the static analyzer [1] is that it does not deal with barrier synchronization processes. This paper extends Interference Partitioning algorithm in [1] to address the class of programs using barrier synchronization as well. User defined abstractions are linked to the program code using PragMatics approach in [3]. The approach is based on an annotation language comprising of all features to address individual loops, application context and function calls with optimization awareness. However, pragMatics does not support recursive applications. The structure of parallel program along with its target platform is considered to obtain tight contention delays in [4]. The main drawback of the approach is that it is limited to blocking communication. Fork-join parallel model is employed in [5], in contrast, the proposed method employs fork-join, Single Program Multiple Data (SPMD), Multiple Program Multiple Data (MPMD), producer - consumer model in parallel programming. An Integer Linear Programming (ILP) based approach is proposed in [6] that maximizes the WCET of a program. It is also proposed that algorithmic approaches scale better for larger programs than ILP based approach. A parallel execution graph is employed in [7] to explore all possible execution interleavings of a parallel task and an exclusion criterion is proposed to prove that certain interleavings can never occur to make precise and feasible WCET analysis of parallel periodic tasks. Communication between the tasks in concurrent software is through message passing and life time estimates of concurrently executing tasks on multicore are improved progressively in [8]. Automatic timing analysis of parallel applications is performed in [9] by considering synchronization stall time associated with each instruction and each basic block in Control Flow Graph (CFG) for WCET estimation process. The approach considers a simple time predictable architecture to estimate WCET. Loop bounds are provided as user annotations to the WCET analyzer [10]. WCET computation of a multithreaded program is proposed in [11] and communication edges are introduced between threads in a multithreaded program in micro architectural modelling phase of WCET estimation.

The instructions that can cause or suffer from timing interferences are extracted in [12]. Based on the extracted instructions, the real time tasks are separated into a sequence of time intervals. The ILP solver uses the time intervals to minimize the WCET of the application. Concurrent execution of programs is simulated to cause conflicts resulting in the eviction of memory block from the

shared instruction cache, being accessed by program in reference thread in [13]. A hardware mechanism is proposed to reduce the number of interfering accesses by forcing certain accesses to bypass shared cache. WCET analysis of parallel code can be performed using UP-PAAL model checker [14]. The approach in [14] considers granularity at instruction level that increases the size of the state space compared to the basic block level granularity. Scheduling model for real-time tasks is presented in [15] and concurrency during task execution is not considered explicitly. In contrast, in the proposed approach in our paper, concurrency among the threads is considered explicitly. Interference Partition (IP) Algorithm [16], computes WCET of a multithreaded program by considering partial order information [17] of the multithreaded program based on wait and notify synchronization. IP Algorithm partitions the Control Flow Graph (CFG) each thread of a multithreaded program into parallel processes $P_{m,i}$ (m is process id and i is thread id) based on partial order information derived using wait/notify synchronization primitives. The partitioning enables computation of a precise WCET of the multithreaded program. The research question addressed in this paper are:

- What parameters need to be considered during WCET analysis of a multithreaded program to provide precise estimates of WCET to designers of Real-time embedded applications?
- How can shared instruction cache memory be modelled by a WCET analyser for precise WCET estimation?

The main contributions are

- Extension of the interference partitioning algorithm in [16] for multithreaded programs to incorporate barrier synchronization calls
- Investigation of the effectiveness of the extended interference partitioning algorithm on benchmark programs adapted from Malardalen [18] benchmark suite
- WCET estimates of multithreaded programs with barrier synchronization calls and computation task specific synchronization calls (using wait() and notify())
- Parameters such as Number of conflicts, Conflict ratio, Overestimation ratio, Precision Improvement in Number of conflicts and Precision Improvement in WCET are proposed for performance evaluations

2. WCET ESTIMATION OF MULTITHREADED PROGRAMS

Typical WCET estimation framework of sequential program mainly comprises of three phases [10]: program flow analysis to obtain structural and functional constraints from the control flow graph of the program, micro-architectural modelling to obtain WCET of each basic block by considering underlying architectural features like cache, pipeline, branch prediction etc. and WCET calculation phase to obtain WCET of the program by maxi-

mizing the objective function comprises of execution time and execution count of each basic block. The above WCET estimation framework is not quite appropriate for WCET estimation of a multithreaded program because of complex interactions between threads in the program, mapped to different cores. A novel method is proposed and incorporated into simulator to obtain WCET of a multithreaded program implemented to run on a multicore architecture with shared instruction cache, by reducing the set of interferences to be considered to a minimal safe subset, from an interacting thread during the execution of an instruction in a reference thread.

Existing IP algorithm [16] does not deal with barrier synchronization processes and it is extended to obtain minimal safe subset of interferences from interacting threads using barrier synchronization primitives. All the functions from n threads of a multithreaded program have to reach the barrier before they proceed and barrier node counts the arrival of all the threads and once all the threads arrived it issued the proceed messages [19]. Real time embedded applications perform computation in k phases. The requirement of the application is that all the threads need to begin the computation of the i^{th} phase, $i \leq k$, only if $i-1^{th}$ phase is completed by all threads. For such an application typically barrier synchronization is used and all K barriers will be designed and programmed. It may be considered that the computation processes, in threads following the $i-1^{th}$ barrier until the i^{th} barrier, are parallel to each other. In fact, the above-mentioned parallel processes are special kind of synchronized parallel processes [1]. The parallel processes inside a barrier may also perform a computation task specific synchronization using wait and notify synchronization primitives at a lower level while there is higher level parallelism between threads using barrier synchronization processes.

Definition of computation Processes

Computation processes arise when two threads interact using wait() and notify() synchronization calls. We refer to them as synchronization parallel processes in the paper. There is an order between processes imposed by the partial order wait<notify. There exist code regions (synchronized parallel processes) in the two threads that may be parallel to each other. Here synchronization calls are used in threads simply to wait and notify and not for barrier synchronization. What is important for the Interference Partitioning algorithm is that synchronized parallel processes in two threads compete with each other for a shared resource such as the shared instruction cache.

For example, $BSP_{2,2} \parallel BSP_{2,3}$ in Fig.1.b are identified as computation processes because they interact using wait() and notify() synchronization calls. There is order between processes imposed by the partial order wait < notify.

Definition of computation task specific synchronization calls

A computation task specific synchronization calls are defined with respect to two interacting threads. The interaction is based on synchronization using calls to wait ()

and notify (). It is assumed that there is no notification loss as the program is validated before WCET analysis is performed. Therefore, for corresponding synchronization calls, wait < notify. Computation task specific synchronization calls identify not only which computation in a thread happens before the computation in another thread but also which computations happen in parallel.

For example, $BSP_{2,2} \parallel BSP_{2,3}$ are identified as computation task specific synchronization calls in Fig.1.b. In general, a multithreaded program may contain multiple barriers as shown in Fig.1.a. for a group of threads to synchronize on completion of tasks one after the other. When it comes to barrier synchronization parallel processes, two or more threads may participate in barrier synchronization and they cross the barrier for further computation together irrespective of the relative speeds until they reach the barrier. From the perspective of the Interference Partitioning algorithm, if k threads participate in the barrier, for a reference thread, $(k-1)$ barrier synchronization parallel processes compete for the shared instruction cache along with the reference thread.

Definition of barrier synchronization parallel processes

Barrier synchronization parallel processes, two or more threads may participate in barrier synchronization and they cross the barrier for further computation together irrespective of the relative speeds until they reach the barrier. From the perspective of the Interference Partitioning algorithm, if k threads participate in barrier synchronization, for a reference thread participating in the barrier synchronization, $(k-1)$ barrier synchronization parallel processes compete for the shared instruction cache.

For example, $BSP_{2,1} \parallel BSP_{2,2} \parallel BSP_{2,3}$ are identified as barrier synchronization parallel processes in Fig.1.a. If there are no task specific synchronization calls between two barrier lines, entire code region between the two barrier lines for each thread is a competing process that can cause shared instruction cache interferences to code of other threads in between the barrier lines. All the functions from n threads of multithreaded program need to reach the barrier before they proceed further. The barrier nodes shall keep track of the arrival of all the participating threads and once all the threads arrive at the node or barrier line, the threads proceed beyond.

Definition of Parallel Processes

Parallelism may exist between threads that simply arises out of no particular order between executions of regions of code in the threads, which we in general term as parallel processes. Parallel processes also compete for the shared instruction cache.

Competing processes refer to barrier synchronized parallel processes or synchronized parallel processes or only computation processes that run parallel with a corresponding process in the reference thread T_r . Competing processes cause conflicts to an instruction I in the

reference thread. The IP algorithm creates the mapping of the competing processes for all barrier synchronization parallel processes in Fig.1.a. as shown in Table 1.

Table 1. Mapping of competing processes for each Barrier Synchronization Parallel Processes $BSP_{m,j}$

Barrier Synchronization Parallel Processes $BSP_{m,j}$	Competing processes
$BSP_{1,1}$	$BSP_{1,2} BSP_{1,3} BSP_{1,4}$
$BSP_{2,1}$	$BSP_{2,2} BSP_{2,3} BSP_{1,4}$
$BSP_{3,1}$	$BSP_{3,2} BSP_{3,3} BSP_{1,4}$

In general, a thread is a composition of computation processes interacting using wait() and notify() calls, barrier synchronization processes and simply parallel processes. The interference partitioning algorithm addresses the problem of determining competing processes for a reference thread for each of the three categories of processes. Micro-architectural modelling uses competing processes to identify conflicts to any barrier synchronization parallel processes $BSP_{n,j}$ to compute WCET of each basic block referred as node in this paper.

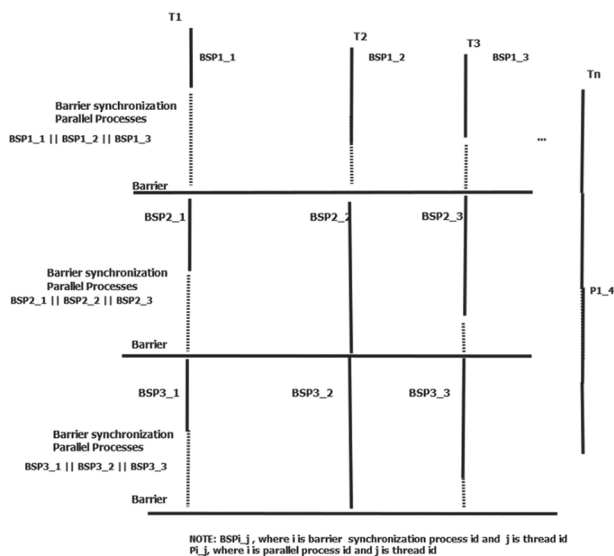


Fig. 1. a. Identified barrier synchronization parallel processes

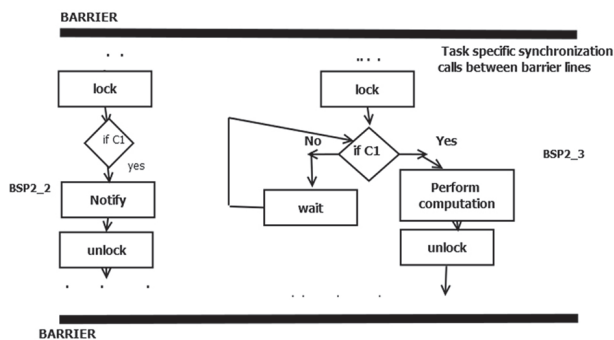


Fig. 1. b. Task specific synchronization calls between two barrier synchronization parallel processes

Fig.1. Multithreaded program with Barrier Synchronization Parallel Processes

WCET calculation uses Implicit Path Enumeration Technique (IPET) that combines program flow information along with its execution time of each basic block to compute WCET of each thread of multithreaded program. The subsection 2.1 explains mathematical representation of cache mapping function in detail.

2.1. CACHE MAPPING FUNCTION

The motivation for discussing Cache Mapping Function is to determine the instruction accesses, in shared instruction cache, made by a thread for which WCET is being estimated (reference thread) that are potentially evicted due to interferences from competing threads. The Cache Mapping Function is used by the Interference Partitioning algorithm. In this paper, barrier synchronization processes cause interferences that can evict instructions in shared instruction cache required by the reference thread. This paper extends Interference Partitioning algorithm to multithreaded programs using barrier synchronization. The three commonly used cache architectures are direct mapped cache, fully associative cache and A -way set associative caches [20]. An A -way set associative mapping architecture contains A cache lines for all S cache sets. Each cache line is capable of holding LS consecutive bytes of a memory block. Direct mapped cache, is a 1-way set associative cache where a cache block can appear in only one place in cache memory. Fully associative cache, is a A -way set associative cache where a cache block can be placed anywhere in the cache memory having only one set. A cache line may be valid (containing a memory block) or invalid (currently free).

This paper considers Harvard architecture i.e., $L1$ instruction cache is separated from $L1$ data cache and $L2$ shared instruction cache is a share resource for all cores.

- Cache size CS : Represents cache memory size in bytes
- Block size or Line Size LS : Represents number of bytes to be loaded in to cache for each memory access
- LS_L1 : Represents line size of level 1 cache memory
- LS_L2 : Represents line size of level 2 cache memory
- Associativity A : Accessed memory block can be placed in A cache lines in cache memory
- Cache Set S : $S = \{s_1, s_2, \dots, s_{(CS/LS)/A}\}$ A cache set s_i is a sequence of cache lines where memory blocks are stored
- S_L1 : Represents number of cache sets in level 1 cache memory
- S_L2 : Represents number of cache sets in level 2 cache memory
- Number of cache lines CL : $CL = \{l_1, l_2, \dots, l_A\}$
- Memory block: Sequence of consecutive instructions based on block or line size

The set of ages for A -way set associative caches are $A = \{0, 1, 2, \dots, A-1\}$. The block replacement method considers only the age of the memory block.

The most recently used memory block is given age 0 and least recently used memory block is given maximal age $A-1$. For each cache miss, the accessed block is placed in a particular cache set based on cache architecture with age as 0, age of all other memory blocks in particular cache set is increased by 1 and memory block with age $A-1$ is evicted from cache memory. For access to a memory block that is currently in cache memory with age a , its age is changed to 0 and the ages of memory blocks lesser than a are increased by 1 and ages of memory blocks greater than a remains the same. Instructions in each memory block are classified as Always Hit (AH), Always Miss (AM) or Not Classified (NC) based on Cache analysis using Abstract interpretation (AI) [20]. Abstract Interpretation based cache analysis does not require execution of the program to study cache behavior of the program; through appropriate abstraction, cache behavior for a program can be inferred using static analysis [20]. Each memory block is mapped to a particular cache set in L1 cache memory based on cache mapping function CMF_{L1} . The equation 1 and 2 shows the Cache Mapping Function of L1 and L2 cache memory respectively.

$$CMF_{L1} = ((Memory\ address / LS_{L1}) \% S_{L1}) \quad (1)$$

Similarly, mapping function of shared L2 instruction cache is,

$$CMF_{L2} = ((Memory\ address / LS_{L2}) \% S_{L2}) \quad (2)$$

The Instruction I mapped to cache set s_i having Cache Hit Miss Classification (CHMC) categorized as AH for shared L2 instruction cache is affected by a set of instructions $\{I_1, I_2, \dots, I_k\}$ from interacting threads that mapped to same cache set s_i [20][21][22]. To compute where to place the memory block in L1/L2 cache memory, the following notations are used. Suppose instruction $I_{\gamma,1}$: addiu \$29,\$29,-72 in Fig.3. of reference thread is stored at the memory address 0x400220. Each memory block is mapped to a particular cache set in L1 cache memory based on cache mapping function CMF_{L1} and it is used to compute the instruction's location in L1 cache memory having cache size CS as 256 bytes, line size LS_{L1} as 16 bytes and Associativity A as 1. Another parameter in CMF_{L1} , which is number of cache sets S_{L1} in L1 cache memory, is computed using $(CS/LS_{L1}/A)$ of L1 cache memory. The instruction in memory address 0x400220, is mapped to cache set number 2 of L1 cache memory.

$$CMF_{L1} = ((Memory\ address / LS_{L1}) \% S_{L1})$$

$$CMF_{L1} = ((0x400220 / 0x10) \% 0x4)$$

Interference Partition Algorithm for Barrier Synchronization

Each reference thread T_i is viewed as a composition of barrier synchronization parallel processes, communicating with other barrier synchronization parallel processes in interacting threads T_j in Fig.1.a. In general, a thread is a composition of computation processes interacting using wait() and notify() calls, barrier syn-

chronization processes and simply parallel processes. The interference partitioning algorithm addresses the problem of determining competing processes for a reference thread for each of the three categories of processes. The Interference Partitioning Algorithm accepts as input a Message Sequence Chart (MSC) representation of the Communicating Sequential Processes (CSP) specification of the multithreaded program. An MSC representation consists of lifelines for threads in the program as shown in Fig1.a. Interactions between threads are through computation task specific synchronization calls (wait(), notify()) or barrier synchronization calls. A partial order based on wait < notify is constructed from the multithreaded program while transforming it to an equivalent CSP specification. The Interference Partitioning Algorithm (addressing Barrier Synchronization) traverses through MSC representation looking for synchronization calls step by step. The WCET analyser identifies computation task synchronization parallel processes based on synchronization calls wait() and notify(). The WCET analyser identifies barrier synchronization region in each thread using barrier initialization and barrier related calls. Simply parallel processes are identified by the WCET analyser based on partial order through which neither less than nor greater than relation is observed for parallel regions. It may be noted that for uniformity, WCET analyser considers a sequential process in a thread to be parallel to an empty process in an interacting thread. In this way all processes are considered to be parallel.

Algorithm for Interference Partitioning identifying barrier synchronization processes

```

While there exists next set of syncCalls in Thread(T)
{
  listOfSyncCalls = getNextSetOfSynchronizationCalls(T);
  if listOfSyncCalls contains barrierSyncCalls
    barrierSyncProcesses=identifyNextBarrierSynchronizatio
    Processes(T);
  else if listOfSyncCalls contains computationTaskSyncCalls
    computationSyncProcesses=
    identifyNextSynchronizedParallelProcesses(T);
  else if (listOfSyncCalls is empty)
    onlyComputationProcesses=identifySolelyComputation
    Processes(T);
  CreateMappingOfCompetingProcessesToCurrentProcessIn
  Thread(T,barrierSyncProcesses,computationSyncProcesses,
  onlyComputationProcesses)
}

```

The Interference Partitioning Algorithm that identifies competing Barrier Synchronization processes is an extension of the basic Interference Partitioning Algorithm [16] that deals only with Computation Task Synchronization calls. The algorithm considers an abstract view of the multithreaded program as a Message Sequence Chart(MSC) as shown in Fig.1.a. The first thread T_1 may be considered a reference thread T for which

WCET is being estimated and with other threads competing for shared resources such as the shared instruction cache. The same procedure is applicable for each and every thread (as a reference thread). The Interference Partitioning algorithm uses thread T as the argument or input and the competing process set is determined for each process in T . The algorithm is applied on each thread to estimate the worst case latency in accessing shared instruction cache with competition of access from other threads. A benchmark program is considered to explain the estimation of worst case latency in accessing shared instruction cache which in turn is used in *WCET* estimation of each thread. Fig.1.a. shows generic structure of a benchmark multithreaded program with functions from Malardalen benchmark programs[18]. The Control Flow Graph (CFG) of each thread is constructed from the assembly code of the multithreaded program. After constructing individual CFGs of threads, the procedural call graph of the program is traversed to construct a global flow graph called Transformed Control Flow Graph (TCFG).

The existing approaches to determining conflict set in accessing shared instruction cache during the *WCET* analysis of a multi-threaded program are not quite exploiting the order and concurrency information between regions of code in threads [10] [13]. The Interference Partitioning algorithm uses the order and concurrency information inferred from partial order of execution of threads. As a consequence, a larger conflict set is used while accessing shared instruction cache when partial order between threads is not used. On the contrary, conflict set that Interference Partitioning algorithm uses, by exploiting partial order information between threads, is only a subset of the conflict set used without partial order information.

The instruction sequence of $BSP_{2,2}$ and $BSP_{2,3}$ in Fig.1.b is shown in Fig. 2. In the case of IP algorithm, conflicts arising for any instruction $I_{i,2}$ in $BSP_{2,2}$ are from any instruction in $BSP_{2,3}$, mapped to the same cache set S_i . In contrast, in existing approaches [10][13], conflicts are from all the instructions in the entire program region. In Fig. 2. the same is shown for instruction $I_{k,2}$ in $BSP_{2,2}$. The parallel process or code region $BSP_{2,3}$ is a subset of the entire code region and hence the conflict set generated using partial order information is smaller.

Let $BSP_{m,1}$ is the region of code in T_1 between barrier sync lines i and $i+1$. Fig. 3. shows CFG of a simple barrier synchronization parallel process $BSP_{m,1}$ along with Cache Hit and Miss Classification table (CHMC) of all instructions in $BSP_{m,1}$ following L2 cache analysis. The instruction $I_{2,1}$ in $BSP_{m,1}$ is categorized as AH in L1 instruction cache memory. Therefore, $I_{2,1}$ will never access shared L2 instruction cache. The instruction $I_{7,1}$ in $BSP_{m,1}$ is categorized as AH in L2 instruction cache memory. Therefore, $I_{7,1}$ will be affected by accesses to instructions made by threads running on other cores referred to as conflicts. Let $BSP_{m,1}$ is the region of code in T_i between barrier sync lines i and $i+1$.

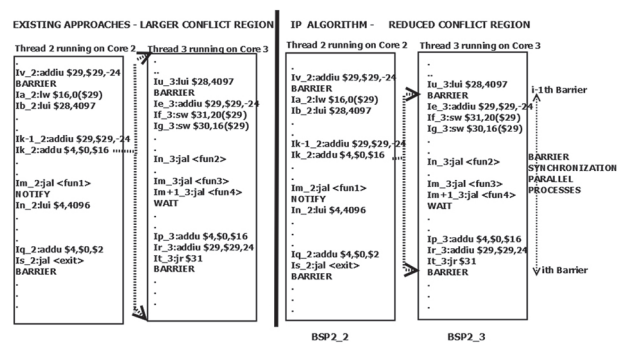


Fig. 2. Conflict Region for Existing approaches and IP Algorithm

Definition of conflicts in IP Algorithm

The conflicts for an instruction I accessed by thread T_i , mapped to cache set S_i , that belongs to any barrier synchronization parallel process $BSP_{m,1}$ are from the instruction set $\{I_1', I_2', \dots, I_p'\}$, mapped to same cache set S_i and that belongs to competing processes of $BSP_{m,1}$.

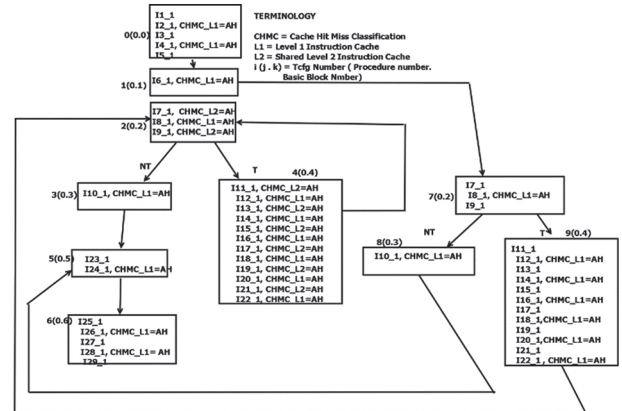


Fig. 3. Control Flow Graph of $BSP_{m,1}$

In the existing approaches, the conflicts for an instruction I in T_i mapped to cache set S_i is from entire program region of T_j mapped to same cache set S_i . For example, as shown in Table 2, the conflicts for instruction $I_{13,1}$ in T_1 mapped to cache set s_1 in shared L2 instruction cache are from all the instructions in T_2 mapped to same cache set s_1 . In IP Algorithm, the conflicts for any instruction are obtained based on partial order information derived using barrier synchronization primitives. Let $BSP_{m,1}$ and $BSP_{n,2}$ are barrier synchronization processes that belong to the same barrier region in threads T_1 and T_2 respectively. Therefore, the conflicts for instruction $I_{13,1}$ in Fig.3. of barrier synchronization parallel process $BSP_{m,1}$ of T_1 mapped to cache set s_1 in shared L2 instruction cache are from instructions $\{I_1', I_2', \dots, I_p'\}$ in $BSP_{n,2}$ that belongs to competing process in T_2 mapped to same cache set s_1 .

IP algorithm performs inter thread shared instruction cache analysis by considering conflicts only from the instructions in barrier synchronization parallel processes $BSP_{n,2}$ that run parallel with $BSP_{m,1}$. Based on those analyses, the age of each instruction is updated.

The Age Update Function (AUF) for any instruction I in shared instruction cache analysis is

$$AUF:Age(I) = Age(I) + conflicts$$

If age of instruction I is greater than or equal to associativity of shared $L2$ instruction cache memory, then instruction I that is currently in cache memory is categorized as NC for shared $L2$ instruction cache accesses. For example, the instruction $I_{7,1}$ in Fig. 3. is categorized as AH following the application of AUF and $I_{9,1}$ is categorized as Not Classified (NC) as shown in Table 2 for IP algorithm. The reduction of conflicts for each instruction

leads to reduction in number of consolidated number of conflicts for each node which in turn leads to reduction in number of consolidated conflicts for a parallel process and finally leading to a reduction in consolidated number of conflicts of a thread in a multithreaded program. The above leads to precision improvement in statically estimating $WCET$ for a multithreaded program that may use barrier synchronization. This is made possible as an abstract view of parallelism in threads is not at whole thread level in our $WCET$ analyser but at smaller process level which is arising from code in barrier synchronization regions in threads.

Table 2. Number of conflicts and worst case latency

Instruction: Address	Cache Set Number	Age	Number of Conflicts		Cache Hit/Miss Classification		Worst Case Latency in Clock Cycles	
			IP Algorithm	Existing Approaches	IP Algorithm	Existing Approaches	IP Algorithm	Existing Approaches
$I_{7,1}$:400220	2	1	2	13	AH	NC	7	37
$I_{9,1}$:400230	3	1	3	13	NC	NC	37	37
$I_{11,1}$:400240	0	1	2	14	AH	NC	7	37
$I_{13,1}$:400250	1	1	3	14	NC	NC	37	37
$I_{15,1}$:400260	2	1	2	13	AH	NC	7	37
$I_{17,1}$:400270	3	1	3	14	NC	NC	37	37
$I_{19,1}$:400280	0	1	3	14	NC	NC	37	37
$I_{21,1}$:400290	1	1	3	14	NC	NC	37	37

Number of Conflicts

As discussed, the conflicts for an instruction I in T_i mapped to cache set S_i that belongs to $BSP_{m,i}$ are from the set of instructions $\{I_1', I_2', \dots, I_p'\}$, mapped to the same cache set S_i which belongs to $BSP_{n,j}$ (i.e. competing process of $BSP_{m,i}$ in T_i).

Number of conflicts Caused by a Competing Thread

The number of conflicts encountered by a node n in a barrier synchronization parallel process $BSP_{m,i}$ is the sum of number of conflicts encountered by each instruction $\{I_1', I_2', \dots, I_t'\}$ in n . Therefore, the number of conflicts of a barrier synchronization parallel process $BSP_{m,i}$ is the sum of consolidated number of conflicts of each node $\{B_1, B_2, \dots, B_k\}$ in $BSP_{m,i}$. Hence, the consolidated number of conflicts of a thread T_i is the sum of number of conflicts of each barrier synchronization parallel process $\{BSP_{1,i}, BSP_{2,i}, \dots, BSP_{q,i}\}$ in T_i . It may however be noted that precision improvement in Worst Case Latency in accessing shared instruction cache takes place as the worst case execution time of each instruction, as simulated, becomes more precise due to reduction in conflicts. Thus, reduction in consolidated number of conflicts caused by a competing thread using IP algorithm is just an indication of the superiority of the approach even when barrier synchronization is used.

As a consequence of reduced number of conflicts for instruction I , $CHMC$ of I remains AH in shared $L2$ instruction cache that leads to reduced worst case latency of instruction. There are a few instructions having reduced

number of conflicts with $CHMC$ categorized as NC due to its age in shared $L2$ instruction cache that leads to maximum worst case latency of instruction. Table 3 shows the number of conflicts of an instruction I , node n containing I , barrier synchronization parallel process $BSP_{m,i}$ containing node n and instruction I , thread T_i of I associated with its $WCET$ for both approaches.

Table 3. Number of Conflicts and $WCET$ of $I_{7,1}$ and $I_{9,1}$

Inst Id	Parameters	IP Algorithm	Existing Approaches	
$I_{7,1}$	Number of Conflicts	Instruction	2	13
		Node	5	26
		Barrier synchronization parallel process	21	109
	WCET in Clock Cycles	Thread	511	4365
		Instruction	7	37
		Node	45	75
$I_{9,1}$	Number of Conflicts	Barrier synchronization parallel process	2571	15300
		Thread	2910490	3506290
		Instruction	3	13
	WCET in Clock Cycles	Node	5	26
		Barrier synchronization parallel process	21	109
		Thread	511	4365
$I_{9,1}$	Number of Conflicts	Instruction	37	37
		Node	45	75
		Barrier synchronization parallel process	2571	15300
	WCET in Clock Cycles	Thread	2910490	3506290
		Instruction	3	13
		Node	5	26

Number of conflicts as is being talked about is only an indirect pointer to precision improvement. *WCET* estimate depends on better estimate of worst-case time for each instruction. Number of consolidated reductions in conflicts from a competing thread to shared instruction cache is an indication and explanation on why *WCET* estimate for a thread improves. It is also evident from Table 3 that reduction in Number of conflicts of an instruction *I* does not necessarily leads to reduction in *WCET* of *I*. In this paper, *L1* cache miss latency is assumed as 6 clock cycles and 30 clock cycles for *L2* cache miss latency.

Conflict ratio

The next parameter considered to evaluate the performance of IP algorithm is Conflict ratio. Conflict ratio of a node *n* in barrier synchronization parallel process $BSP_{m,i}$ computed for IP algorithm is always lesser than or equal to Conflict ratio of a node *n* in barrier synchronization parallel process $BSP_{m,i}$ of existing approaches.

Definition of Conflict ratio

Conflict ratio of a node *n* in barrier synchronization parallel process $BSP_{m,i}$ is calculated by dividing number of conflicts of a node *n* by the total number of instructions in *n*. Similarly, conflict ratio of a barrier synchronization parallel process $BSP_{m,i}$ in thread T_i is calculated by dividing number of conflicts of a barrier synchronization parallel process $BSP_{m,i}$ by the total number of instructions in barrier synchronization parallel process $BSP_{m,i}$. Likewise, conflict ratio for a thread T_i is calculated by dividing number of conflicts of a thread T_i by the total number of instructions in T_i .

Over Estimation Ratio of WCET

CMP-SIM simulator (a multi-core extension of simple scalar tool set [23]), used to evaluate the accuracy of the static analyzer experimentally. All the experiments are performed in 2-cores with different architectural parameters. The estimated *WCET* obtained using IP algorithm is compared with the simulated *WCET*.

The simulated *WCET* of the program is highly underestimated than actual *WCET*. The worst-case input of some benchmarks is difficult to obtain because of branching and other complex mathematical calculations. The over estimation ratio of existing approaches is computed as $WCET_{Existing Approaches} / WCET_{Observed WCET}$ similarly, overestimation ratio of IP algorithm is computed as $WCET_{Interference Partition algorithm} / WCET_{Observed WCET}$.

Precision Improvement in Number of conflicts

The reduction in number of conflicts is considered as one of the major parameters of performance evaluation. The precision improvement in Number of conflicts is computed as $((Number\ of\ conflicts_{Existing\ Approaches} - Number\ of\ conflicts_{Interference\ Partition\ algorithm}) / Number\ of\ conflicts_{Existing\ Approaches}) * 100$. The precision improvement in number of conflicts varies from 60-90%, this is mainly due to minimal safe subset of conflicts from an interacting thread during the execution of an instruction in a reference thread.

Precision Improvement in WCET

The precision improvement in *WCET* is computed as $((WCET_{Existing\ Approaches} - WCET_{Interference\ Partition\ algorithm}) / WCET_{Existing\ Approaches}) * 100$. The precision improvement in *WCET* varies from 15-20%, this is due to shared *L2* instruction cache hits inside loops. Though there is a huge precision improvement upto 90 % in number of conflicts, the precision improvement in *WCET* is 20% and the reason for the same is discussed in section 3.

3. RESULTS AND DISCUSSION

The simulator multi-core chronos [24] [25] is extended to incorporate Interference Partition Algorithm for barrier synchronization. Multi-core chronos tool is extended to make it aware of threads with synchronization information, that is, to identify barrier synchronization parallel processes to be used by the IP algorithm.

Design of Simulator

Multi-core Chronos Simulator [24][25] is extended to keep track of the code regions in other threads that compete for shared instruction cache through conflicts or interferences as an instruction in a thread *T* is being accessed from the shared instruction cache. *WCET* is being estimated for thread *T* and hence simulator needs to consider shared instruction cache misses encountered during the simulation of execution of thread *T* due to competing instruction accesses by other threads from shared instruction cache. The code regions in other threads that compete for shared instruction cache are Barrier Synchronization Processes, if the code regions along with the instruction under access in *T* are engaged in barrier synchronization. The code regions may be synchronizing processes(tasks) in two or more threads using calls to wait() and notify(). The code regions along with the instruction under access in thread *T* from shared instruction cache may be simply parallel processes without being engaged in any any form of synchronization. The Control Flow Graphs along with partial order information of the input multithreaded program are transformed into Hoare's *CSP* from which a Message Sequence Chart is visualized. Competing processes for each instruction in a thread are computed by the Interference Partitioning algorithm and are fed as input to the extended simulator. The Interference Partitioning Algorithm aids the simulator determine an abstract set of competing accesses to shared instruction cache as an instruction in thread *T* is accessed. The simulator can decide whether an access is a shared instruction cache miss based on the abstract set of competing accesses. A simulator that does not use Interference Partitioning algorithm handling barrier synchronization processes can only consider set of competing accesses to be the entire code regions of competing threads. On the contrary, our simulator based on Interference Partitioning Algorithm uses a much more precise set of competing accesses to shared instruction cache.

A simplified version of the typical multi-core architecture is assumed where each core has a small private *L1*

cache and comparatively larger $L2$ instruction cache, shared by all the cores. The access latency of shared $L2$ instruction cache is higher than that of $L1$ cache. The execution time of a multithreaded program is increased by the impact of the interfering shared cache accesses running on other cores. Performing cache analysis for a multithreaded program on a multicore architecture statically is a non-trivial task. It is essential for a real-time embedded application to obtain tighter $WCET$ estimates precise analysis of latency due to accesses to shared cache. To evaluate the performance of existing approaches and IP algorithm, a few parameters are proposed and considered for analysis. The proposed parameters are

- Number of conflicts

- Conflict ratio
- Overestimation ratio
- Precision Improvement in number of conflicts
- Precision Improvement in WCET

Number of Conflicts

Fig. 4. shows number of conflicts of benchmark program for both IP algorithm and existing approaches. It is evident that number of conflicts in IP algorithm is always lesser than or equal to number of conflicts in the existing approaches due to reduced minimal subset of conflicting region. This leads to more precise latency computation for an instruction accessing shared $L2$ instruction cache memory.

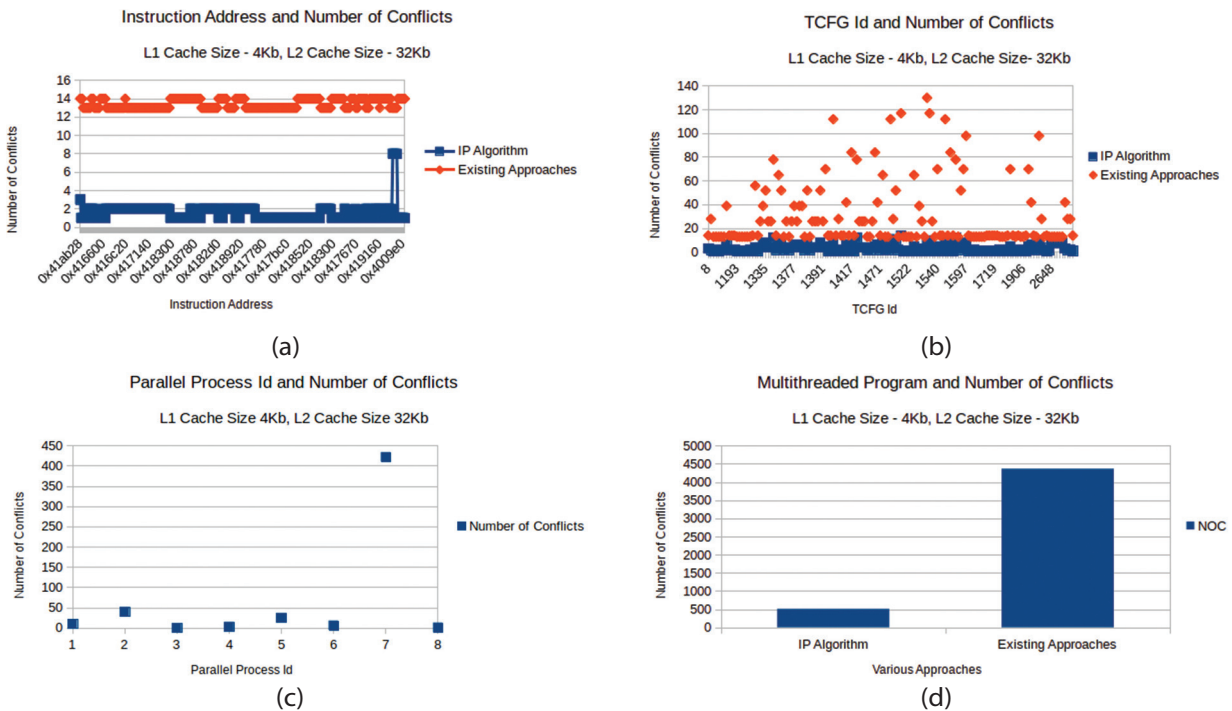
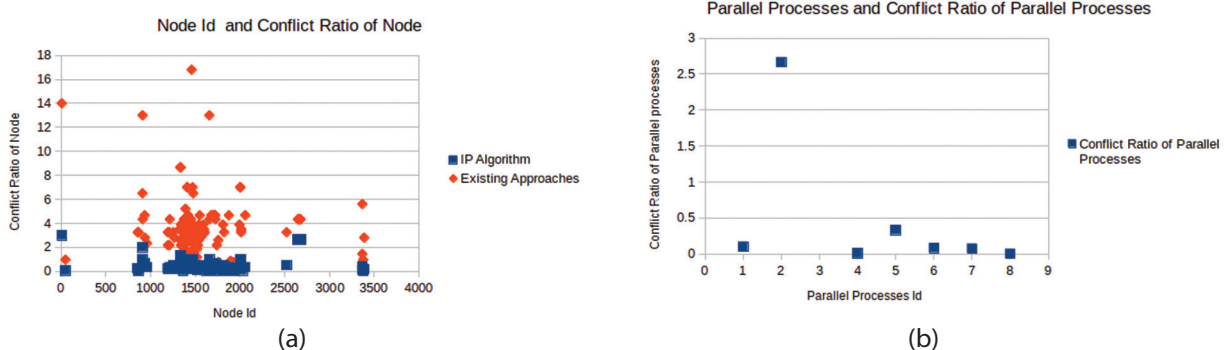


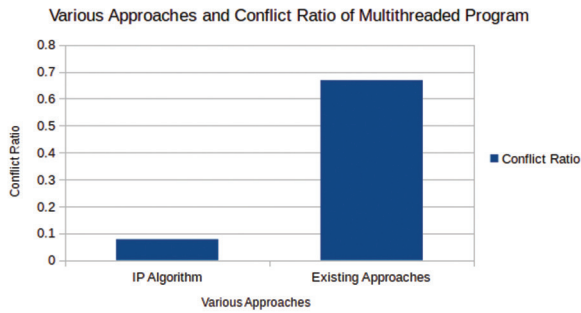
Fig. 4. Number of conflicts of benchmark program; a) of each instruction, b) of each node, c) for each Parallel Process, d) for various Approaches

Conflict Ratio

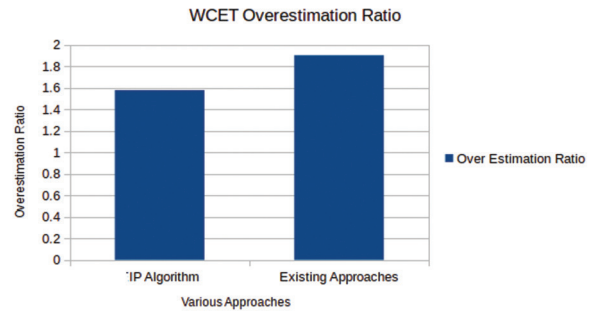
Consider that an instruction I in node n of barrier synchronization parallel process $BSP_{m,j}$ in T_i can have a maximum of x conflicts from interacting thread T_j for IP algorithm and let it be y for existing approaches and

it is proved experimentally that $x < y$. As shown in Fig. 5.b., the conflict ratio of $BSP_{2,1}$ is slightly higher than conflict ratio of other parallel process in T_i , this is due to the fact that the conflicts for $BSP_{2,1}$ are from parallel process having long calculation sequence and more number of branching statements.





(c)



(d)

Fig. 5. Conflict Ratio of multithreaded program; a) of a node, b) of a parallel process, c) of a multithreaded program, d) *WCET* of overestimation ratio

Overestimation Ratio of WCET

The main reason for reduction in overestimation ratio is due to impact of IP algorithm on architectural parameters of cache memory. IP algorithm reduces the number of conflicts that leads to a significant reduction in number of shared *L2* instruction cache misses. Compulsory misses remain misses even with an infinite cache memory and possible way to reduce compulsory misses is by larger block size, but larger block size increases conflict misses due to fewer cache lines/blocks.

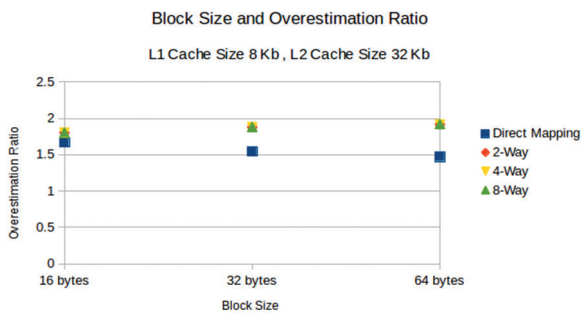


Fig. 6 .a. Block Size and Overestimation Ratio

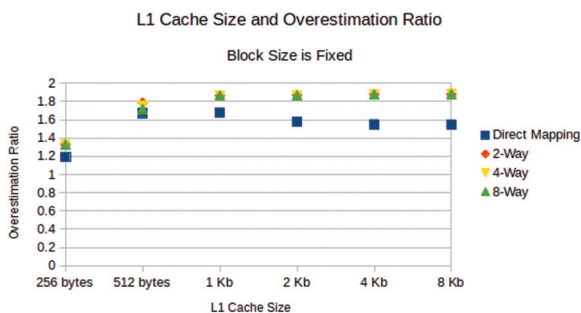


Fig. 6. b. Associativity and overestimation ratio

Fig. 6. Impact of block size and associativity on overestimation ratio

Fig. 6.a. shows the impact of various block sizes on overestimation ratio of *WCET*. One possible way to reduce conflict Misses is to have n -way set associative mapping. In n -way where $n > 1$, set associative mapping

cache memory, each set has n cache blocks so there are less chances of conflict between two addresses mapped to same cache set. It is evident from Fig. 6.b. that for n -way set associative mapping where $n > 1$, overestimation ratio is same. It is also observed that smaller block sizes do not take maximum advantage of spatial locality that results in a greater number of compulsory misses as shown in Fig. 6. a.

Precision Improvement in Number of conflicts and WCET

Though there is a huge precision improvement up to 90 % in reduction in number of conflicts, the precision improvement in *WCET* of a multithreaded program is only up to 20% which is still significant but not commensurate with the former. The reason for the same is discussed in this subsection. Significant improvements are observed when barrier synchronization parallel process size is considerably greater than that of *L1* cache size. This is because under the stated condition, interferences to shared instruction cache from competing processes are significantly less than those from interacting threads. This is a direct consequence of static identification of barrier synchronization parallel processes in interacting threads. If the size of a barrier synchronization parallel process in a thread is similar to the size of *L1* instruction cache, then the need to use shared instruction cache may be quite less for the execution of the barrier synchronization parallel process in the thread. *WCET* precision improvement varies based on cache architectural parameters and benchmark characteristics. In Table 4, a few more benchmark results are shown by varying *L1* cache size. Greater the number of threads, higher the number of conflicts, causing more cache misses, resulting in an increase in *WCET* estimate with greater imprecision. In contrast, the increase in *WCET* estimate using IP algorithm remains smaller by a fraction when compared to existing approaches. For the benchmarks when run on for 4-core architecture, IP algorithm gave lower *WCET* estimate over existing approaches, with the average precision improvement of 10%. It is noticed that, as the degree of parallelism in threads increases, there is reduction in percentage of precision improvement.

Table 4. Precision Improvement in Number of conflicts and WCET

Test Cases	Benchmarks Characteristics	L1 cache Size	Precision Improvement in Number of Conflicts (%)	WCET (Precision Improvement %)
TC1	Inner loop dependent on outer loop, Array and Matrix calculation	256 bytes	65.89%	19.58%
		512 bytes	60.1%	16.4%
		1 KB	55.8%	15.7%
TC2	Input dependent loops, Nested IF statement, Long calculation sequence, Automatically generated code	256 bytes	91.38%	21.8718%
		512 bytes	88.25%	18.1%
		1 KB	80.1%	16.8718%
TC3	Input dependent loops, Automatically generated code	256 bytes	92.69%	22.20%
		512 bytes	88.6%	19.7%
		1 KB	82.4%	16.8%
TC4	Multiple calls to same function, Nested Function calls	256 bytes	88.57%	24.6%
		512 bytes	83.9%	21.5%
		1 KB	78.2%	20.12%

4. CONCLUSION

Worst Case Execution Time Analysis of real-time embedded applications is a challenging task. In this paper, Interference partitioning (IP) algorithm is extended to obtain minimal safe subset of interferences from interacting threads using barrier synchronization primitives. Computation task specific synchronization inside barrier synchronization processes is also identified by IP algorithm. Investigation of the effectiveness of the extended interference partitioning algorithm on benchmark programs adapted from Malardalen benchmark suite is performed. Parameters such as Number of conflicts, Conflict ratio, Overestimation ratio, Precision Improvement in Number of conflicts and Precision Improvement in WCET are proposed for performance evaluations. There is a huge precision improvement upto 90 % in reduction in number of conflicts and the precision improvement in WCET is upto 20% due to IP algorithm.

5. REFERENCES:

- [1] P. P. P. Dharishini, P. V. R. Murthy, "Static Analyzer for Computing WCET of Multithreaded Programs using Hoare's CSP", Proceedings of the 15th Innovations in Software Engineering Conference, February 2022, pp. 1-12.
- [2] C. A. R. Hoare, "Communicating sequential processes", Proceedings of the Communications of the ACM, Vol. 21, No. 8, 1978, pp. 666-677.
- [3] S. Schuster, P. Wagemann, P. Ulbrich, W. Schröder-Preikschat, "Annotate once—analyze anywhere: context-aware WCET analysis by user-defined abstractions", Proceedings of the 22nd ACM SIGPLAN/SIGBED International Conference on Languages, Compilers, and Tools for Embedded Systems, 2021, pp. 54-66.
- [4] B. Rouxel, S. Derrien, I. Puaut, "Tightening contention delays while scheduling parallel applications on multi-core architectures", ACM Transactions on Embedded Computing Systems, Vol. 16, No. 5s, 2017, pp.1-20.
- [5] A. Alhammad, R. Pellizzoni, "Time-predictable execution of multithreaded applications on multicore systems", Proceedings of the Design, Automation & Test in Europe Conference & Exhibition, 24-28 March 2014, pp. 1-6.
- [6] K. Nagar, Y. N. Srikant, "Precise shared cache analysis using optimal interference placement", Proceedings of the IEEE 19th Real-Time and Embedded Technology and Applications Symposium, Berlin, Germany, 15-17 April 2014, pp. 125-134.
- [7] T. Kelter, P. Marwedel, "Parallelism analysis: Precise WCET values for complex multi-core systems", Science of Computer Programming, Vol. 133, 2017, pp. 175-193.
- [8] Y. Liang, H. Ding, T. Mitra, A. Roychoudhury, Y. Li, V. Suhendra, "Timing analysis of concurrent programs running on shared cache multi-cores", Real-Time Systems, Vol. 48, No. 6, 2012, pp. 638-680.
- [9] H. Ozaktas, C. Rochange, P. Sainrat, "Automatic WCET analysis of real-time parallel applications", Proceedings of the 13th International Workshop on Worst-Case Execution Time Analysis. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2013.
- [10] R. Wilhelm et al. "The worst-case execution-time problem—overview of methods and survey of tools", ACM Transactions on Embedded Computing Systems, Vol. 7, No. 3, 2008, pp. 1-53.

- [11] D. Potop-Butucaru, I. Puaut, "Integrated worst-case execution time estimation of multicore applications", Proceedings of the 13th international workshop on worst-case execution time analysis, 2013.
- [12] T. Carle, H. Cassé, "Reducing timing interferences in real-time applications running on multicore architectures", Proceedings of the 18th International Workshop on Worst-Case Execution Time Analysis, 2018, pp. 1-11.
- [13] D. Hardy, T. Piquet, I. Puaut, "Using bypass to tighten WCET estimates for multi-core processors with shared instruction caches", Proceedings of the 30th IEEE Real-Time Systems Symposium, Washington, DC, USA, 1-4 December 2009.
- [14] A. Gustavsson, A. Ermedahl, B. Lisper, P. Pettersson, "Towards WCET analysis of multicore architectures using UPPAAL", Proceedings of the 10th international workshop on worst-case execution time analysis, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik.
- [15] D. Casini, A. Biondi, G. Buttazzo, "Analyzing parallel real-time tasks implemented with thread pools", In Proceedings of the 56th Annual Design Automation Conference, Las Vegas, NV, USA, June 2019, pp. 1-6.
- [16] P. P. P. Dharishini, P. V. R. Murthy, "Precise Shared Instruction Cache Analysis to Estimate WCET of Multithreaded Programs", Proceedings of the IEEE 18th India Council International Conference, Guwahati, India, December 2021, pp. 1-7.
- [17] G. Coulouris, J. Dollimore, T. Kindberg, G. Blair, "Indirect Communication", Distributed systems: Concepts and Design, Fifth Edition, Addison-Wesley, 2011.
- [18] J. Gustafsson, A. Betts, A. Ermedahl, B. Lisper, "The Mälardalen WCET benchmarks: Past, present and future", Proceedings of the 10th International Workshop on Worst-Case Execution Time Analysis, Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2010.
- [19] K. S. Namjoshi, "Are concurrent programs that are easier to write also easier to check", Proceedings of the Workshop on Exploiting Concurrency Efficiently and Correctly, 2008.
- [20] H. Theiling, C. Ferdinand, R. Wilhelm, "Fast and precise WCET prediction by separated cache and path analyses", Real-Time Systems, Vol. 18, No. 2, 2000, pp. 157-179.
- [21] D. Hardy, I. Puaut, "WCET analysis of multi-level non-inclusive set-associative instruction caches", In Proceedings of the 29th IEEE Real-Time Systems Symposium, Barcelona, Spain, 30 November - 3 December 2008, pp. 456-466.
- [22] M. Lv, N. Guan, J. Reineke, R. Wilhelm, W. Yi, "A survey on static cache analysis for real-time systems", Leibniz Transactions on Embedded Systems, Vol. 3, No. 1, 2016.
- [23] T. Austin, E. Larson, D. Ernst, "SimpleScalar: an infrastructure for computer system modeling", Computer, Vol. 35, No. 2, 2002, pp. 59-67.
- [24] S. Chattopadhyay, L. K. Chong, A. Roychoudhury, T. Kelter, P. Marwedel, H. Falk, "A unified WCET analysis framework for multicore platforms", ACM Transactions on Embedded Computing Systems, Vol. 13, No. 4s, 2014.
- [25] X. Li, Y. Liang, T. Mitra, A. Roychoudhury, "Chronos: A timing analyzer for embedded 72 software", Science of Computer Programming, Vol. 69, No. 1, 2007, pp. 56-67.

Review of Loan Fraud Detection Process in the Banking Sector Using Data Mining Techniques

Review Paper

Fahd Sabry Esmail

Helwan University,
Faculty of Commerce & Business Administration, Department of Business Information Systems
Cairo, Egypt
Fahd.Sabry21@commerce.helwan.edu.eg

Fahad Kamal Alsheref

Beni-Suef University,
Faculty of Computers and Artificial Intelligence, Department of Information Systems
Beni-Suef, Egypt
drfahad@fcis.bsu.edu.eg

Amal Elsayed Aboutabl

Helwan University,
Faculty of Computers and Artificial Intelligence, Department of Computer Science
Cairo, Egypt
amal.aboutabl@fci.helwan.edu.eg

Abstract – At the era of digital transformation, fraud has dramatically increased, notably in the banking industry. Annually, it now costs the world's economies billions of dollars. Daily, news of financial fraud has a negative influence on the world economy. According to the harsh loss caused by fraud, effective strategies and methods for avoiding income statement fraud have to be implemented. Also, the procedure of identification should be applied. This is regarded as a result of the development of modern technology, modern invention, and the rapidity of global communications. Actually, deterrent technologies are most effective to reduce fraud and overcome cons. So, it is necessary to find ways to overcome such deterrence by depending on developed methods to identify fraud. Data mining techniques are currently the most widely used methods for the prevention and detection of financial fraud. The use of datasets for fraud detection complies with the norms of data mining, which include feature selection, representation, data gathering and management, pre-processing, comment, and summative evaluation. Methodologies for identifying fraud are essential if we want to catch criminals after fraud prevention has failed. The greatest fraud detection strategies for locating loan banking and financial fraud are compared in this article.

Keywords: Fraud, Loan Fraud, Loan Fraud detection, Data mining techniques.

1. INTRODUCTION

Banks handle massive amounts of client data. It requires sustainable work to functionalize the aggregated data to detect consumer behaviors. This wastes time and effort and enables top managers to make the best decision and avert any prospective losses.

Data science is currently not only a pattern but rather a must to stay competitive in the banking industry. Moreover, data mining is increasingly emerging as a strategically significant topic for many commercial organizations, including the banking sector. It involves compiling data from many aspects and turning it into useful information. In addition, banks employ data science for target marketing, forecasting, consumer sentiment research, fraud detection, and

customer support. Actually, it supports paying attention to the specifications presented to warn banks of fraud.

Furthermore, Fraud has a disastrous effect on economies all over the world, and several techniques have been applied but failed. However, machine learning is more dependable. Banks can find previously unidentified relationships in the data and look for hidden patterns in aggregations of data using data mining, which employs machine learning to make smart decisions based on the revealing insights. This enhances the capacity to identify resources, make better decisions, and perform better in loan appraisals and other categories of lending.

Obviously, many companies apply multifactor authentication as a competitive advantage. They evaluate their

success in terms of profitability partially by the amount of fraud they can keep out of the hands of their competitors. These companies frequently do not have the intention to discuss or reveal their fraud prevention techniques to competitors. Fraudsters may be scared away by rapid action. This obligatory change is a vital component for companies that consider fraud control as a source of comparative advantage. The fraudsters would target their competitors to apply the scheme because their main objective is to carry out ways before their competitors.

The data are divided into two or more categories using the support vector machine (SVM) classifier, a kernel-based supervised learning method. Specifically for binary classification, SVM was developed. Within the training phase, SVM builds a model, maps the decision boundaries for each class, and finds the hyperplane that separates the multiple classes. In supervised learning, a set of input properties, such as plasma metabolite or transcriptional levels, are used to predict a quantitative sampling distribution. For example, there is the identification of loan fraud, or a qualitative one, like healthy or ill individuals. Several supervised learning techniques were handled, including multiple linear regression and random forests, as well as their usual behaviors with various sample sizes and numbers of predictor variables. We look at two popular supervised machine learning methods, linear support vector machines (SVM) and k-nearest neighbors (kNN). Both have been functionalized successfully to overcome challenging issues.

For the sake of better customer targeting and acquisition, the banking industry may profit basically from data mining tools. Customer retention is very valuable and automatic. Credit approval is used for fraud protection, and real-time fraud detection. This provides with customer analysis, segment-based solutions, historical transaction patterns for improved relationship development and retention, risk management, and marketing.

The banking industry has benefited immensely from advances in digital technology [1]. The concept of data being stored at branches has been replaced by centralized databases. Today, there are a lot more options for accessing bank accounts. Financial systems have become more client-focused and technically advanced thanks to digital purchasing, automated wire transfers, ATMs, and cash and check deposit devices [2]. The number of channels has increased along with the number of transactions and the data stored about them. Major banks have huge digital data warehouses within their computational storage systems. The quantity and quality of data have been developed [3].

Thanks to advancements in data mining methods and skills, the organization's data mountain is now proving to be its most valuable asset [4]. These data have interesting patterns and informative content. There is a great deal of potential for banks to functionalize data mining within decision-making through areas including marketing, debt management, proceeds of crime identification, liquidity management, investment banking, and the prompt detection of fraud transactions. The disability to achieve success within these fields could have negative effects on the bank, such as client loss to competitors' companies,

monetary loss, reputational loss, and huge fines by the stakeholders.

In business, research, and many other fields, the need for commercial databases has expanded along with the requirement for content and retention. This increase in the amount of technologically held data can be explained by the growing acceptance of the link perspective of information preservation, as well as by the development and refinement of data access and generating contrast. In light of the need for data storage rose, this method was utilized. Recently, as previously, little consideration was given to creating software for data analysis. This changed when businesses found a resource hidden among these enormous data quantities. There is a wealth of information about their firm that has been kept and is simply waiting to be taken and used to improve a variety of elements assistance with company decisions. Functionalizing the database management systems that are used to manage these data sets, the user can presently only access information that is specifically contained in the databases. The amount of knowledge that exists is much greater than the size of a database, or the "ice shelf of knowledge," as it is called. Since this data unintentionally contains knowledge about numerous various aspects of their organization, it tends to be accessed and used for better decision-making.

Finding patterns within data represents the process of data mining. It can be beneficial in a variety of applications, including fraud detection. It combines complex data search techniques with statistical algorithms to uncover patterns and linkages. Inconsistent data, strange behavior, duplicate payments, missing invoices, abnormal transactions/vendors, and purchase and disbursement frauds, to name a few, are just a few examples of the abnormality and internal control holes that data mining can assist your firm find.

The aim of this research study is to contribute in literature concerning data mining methods used in banking to identify loan fraud. In order to categorize, extract relevant articles, and publish literature-based results, this work was completed in two parts. Stage one of classification involves locating applications, while stage two involves identifying fraud in the financial sector.

2. BACKGROUND

2.1. CONCEPT OF FRAUD

Fraud is defined as illegal deceit that is intentionally used by a person to get an unauthorized financial benefit. It may also take place with the express intent of misleading another person or organization, as in the case of making false assertions. Fraud is not a recent occurrence in contemporary life. For decades, fraudsters have been committing fraudulent acts [5]. This algorithm made somewhat more accurate predictions than the inspectors. Other reasoning systems [6] simulated the arguments of fraud experts by concentrating on two distinct tracks. The flexible anomaly classifier employed the Wang-Mendel technique to demonstrate how healthcare practitioners defrauded insurers. The search model uses an unsupervised network to discover links within data and to find clusters, after which patterns

inside the clusters are found. The research gap is investigated in [7-9]. The electronic fraud detection (EFD) system [10] functionalized statistical data with expertise and knowledge to identify those whose actions deviated from the norm. Since the other clustering algorithms are frequently prohibitively costly when the datasets are very large and visualization techniques are applied for rule analysis, building mathematical synopses of the entities associated with each rule, the hot spots method combines the k-means segmentation method for cluster detection. [11-13] expanded the spots technique by generating and exploring the rules using a learning algorithm.

Several factors, including assistance from bank employees, can facilitate fraudulent activities, such as access to client databases, personal information, and information technology (IT) systems of the bank.

One definition of clumping is the discovery of groups of items that share characteristics. This method combines transactions with comparable behavior together. Segmentation can be used as developed a reputation for deciding which feature subsets to classify [14-16]. For example, in the banking industry, consumers from the tier always request a policy that guarantees more security because they are not determined on taking risks.

Similarly, people in the same middle to upper class who reside in rural settings may have tastes for some name products that are different from those who live in urban areas. Instead of mass presenting one particular "hot" product, the organization will be able to bridge other products. The company's customer service agents will have access to customer profile pages that have been enhanced through data analysis, enabling them to determine which services and goods are most meaningful to consumers.

One of the recent advancement in parallel with data processing technologies are data mining and information extraction. It incorporates the disciplines of information science, system administration, machine learning, statistics, and visualization. This is a new field. Despite this, the industry becomes more effective as a tool to research its clients and take reasonable judgments [17-18]. The process of uncovering true, fresh, possibly helpful, and ultimately intelligible data patterns is known as information retrieval from datasets. Data mining is a key step in deep learning, and the two terms are frequently used interchangeably [19].

Finding useful information from vast data repositories to address important business concerns is a technique known as data mining. It reveals hidden human analysis correlations, trends, patterns, exceptions, and oddities. Customers have a wide range of options in today's fiercely competitive market climate. In order to maintain their customer base, banks must be proactive in analyzing customer inclinations and profiles and tailoring their offerings and services accordingly [20-22]. A bank can reduce losses before it's too late by classifying customers into problematic and excellent consumers [23]. A bank can identify credit card fraud by examining average demand before it has an impact on its earnings [24]. Data analysis could be useful for achieving these highly desired qualities.

Fraudulent activities vary by severity, sector, complexity, manner, and difficulty of discovery or prevention of fraud differently. The summary is a well-rounded, non-exhaustive collection of several fraud categorizations.

2.1.1. Credit Card Fraud

Credit Card Fraud is referred as unauthorized use of a credit card account. A methodology with a malfeasance property and a clustering time regular by a classifier without an embezzlement attribute were both recommended by the credit card theft model [25-27]. Automobile injury claims were divided into different categories based on the degree of deception suspicion using a soul feature map [28-31]. The correctness of the feature map was then tested using a back propagation method and recurrent neural network networks. This occurs if neither the cardholder nor the card issuers are aware that a third party is using the card. As a result, fraudsters are able to buy things for free or acquire access to money in an account [32-34].

2.1.2. Insurance Fraud

Insurance fraud is known as an effort to take advantage of or abuse insurance coverage. Insurance is designed to cover losses and guard against dangers. Fraud happens when an insured utilizes their insurance policy to gain an unauthorized profit [35-37].

2.1.3. Money Laundering

Money laundering is a technique criminals use to conceal the source and final location of money obtained illegally to make it appear genuine [8].

2.1.4. Telecommunication Fraud

As relevant to telecommunications, fraud is defined as the usage of any carrier service without the purpose of paying. Other motivations, such as political or personal motivations, could be available.

2.1.5. Financial Statement Fraud

Financial fraud, commonly referred to as accounting fraud, is the deliberate misrepresentation of financial information in order to deceive the reader. Specifically, they mislead lenders and investors about a business's strategic stability [9].

2.1.6. Monetary deception Investment fraud

Monetary deception Investment fraud, commonly referred to as financial markets fraud, describes dishonest actions when securities are offered and sold [38-39]. High-yield investment fraud is a frequent kind of securities fraud. Affinity fraud, pyramid scams, and Ponzi schemes are a few well-known examples.

2.2. LOAN FRAUD OVERVIEW

Loan fraud, also known as lending fraud, refers to any deceptive action taken to gain a financial advantage during the loan process. Loan fraud can take many forms,

including mortgage fraud, payday fraud, and loan scams. All of them will result in someone losing money, while the counterparty gains money and disappears. Fig.1 illustrates several types of secured and unsecured loans.

Secured loans have a pledge of something of value as security; if the borrower defaults on the loan, the bank or financial institution may sell the asset to recoup the loan balance. A good example of this kind of loan is a mortgage or home loan when the house or the property is used as collateral or security. If the borrower defaults, the bank may foreclose on the loan and change the loan balance.

Unsecured loans lack an underlying asset or another kind of security. The only thing the bank or other financial institution receives is the borrower's guarantee. Personal loans are an excellent example of an unsecured loan when the lending institution provides no security.

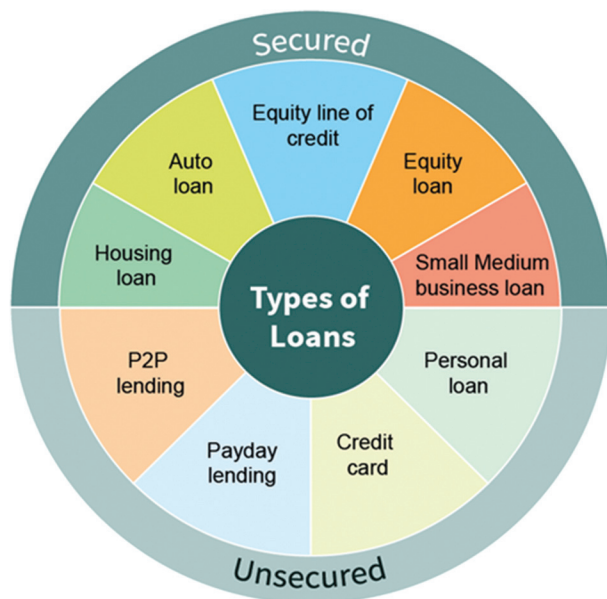


Fig. 1. Different types of secured and unsecured loans

2.3. LOAN APPLICATION FRAUD DEFINITION

Credit fraud, also known as loan application fraud, is a type of financial fraud in which the criminal obtains an illegal loan that they have no intention of repaying. They can take out the loan in their own name, which would be considered first-party fraud. However, it is more common for the line of credit to be opened in the name of a third party using forged or stolen identification documents. As shown in Table 1, there are two types of loan application fraud: first-party and third-party fraud.

Table 1. Types of Loan Application Fraud

First-Party Fraud	Third-party fraud
First-party fraud occurs when a criminal applies for a credit line or a personal loan using their legal name and documents. The criminal will then withdraw all of the funds from the account and vanish without leaving a trace.	Third-party loan fraud, on the other hand, involves receiving loans in another person's name. Criminals can accomplish this by using either stolen or forged identity documents. Read our article on document fraud to learn more about how they do it.

Personal loans, mortgages, commercial loans, and other sorts of loans are handled by banks and given to their customers. The bank will decide the customer's loan eligibility when the consumer submits an application [40-42]. Banks' main source of income comes from loans, but if borrowers don't make their payments on time, there will be bad loans [43-45]. To remain in business and earn the trust of their clients, banks must adhere to strict standards. In order to lower the risk that may hurt the bank, the most crucial criterion is to always investigate the behavior of the customer seeking a loan [13]. Numerous clients request bank loans, but because banks occasionally have limited resources and can only lend to a small number of customers, customers themselves are frequently forced to look for loans from third parties [14].

Naturally, fraudsters face a high-risk level because they must voluntarily hand over their personal information to the lender. While an evil actor could simply cross a state border and start a new life a century ago, this is nearly impossible in today's digital world. As a result, this method is becoming less popular by the year.

3. DATA MINING APPLICATIONS IN BANKING

As a method of identifying beneficial patterns and correlations, data mining has a specific place in financial modeling. Virtually all data mining techniques, like other computational methods, can be used in financial modeling. In the banking industry, data mining is beneficial.

Large volumes of data are explored through data analysis to enhance the market segment for businesses. You can develop a tailored loyalty promotion, particularly for that consumer segment by looking at the links between factors like user age, gender, etc. Additionally, it may be used to predict which people are most likely to pay to a company, what their search histories indicate about their preferences, or what subscribers should include on email lists to boost response times.

Banks can find patterns in a group and uncover hidden links in data using data mining. The bank fully profiles each customer of the bank. The customer data includes specific items regarding the client's financial circumstances and spending habits before and after the credit was approved.

Directed learning is another name for classifiers. A dependent attribute or aim that was previously understood guides the learning process. A set of independent qualities or predictors are used in directed data mining to explain the target's behavior. Predictive models are typically the result of supervised learning. In contrast, the aim of unsupervised classification is pattern recognition.

A supervised model must be trained, which entails having the computer examine numerous instances where the target value is already known. The model "learns" the reasoning behind the prediction during the training process. For instance, a model that aims to predict which customers are most likely to respond to an offer must be trained by studying the traits of several individuals who are known to have previously responded or not to a campaign.

Data mining techniques have been applied in the banking industry for a number of purposes, such as

Predicting bank collapse [15–16], identification of potential bank customer churns [17], fraudulent transaction detection [18], customer segmentation [19-20], bank telemarketing predictions [21-24], sentiment analysis for bank customers [25], and bank loan prediction [26-28]. Some

categorization studies in the banking sector are compared in Table 2. This table displays the aims of the previous researches, the years they were carried out, the ensemble learning techniques and the applied algorithms, the nationality of the bank, and the outcomes.

Table 2. Categorization studies in the banking sector

Reference	Year	Algorithms						Ensemble learning		Description	Country of the bank	Result
		DT	NN	SVM	KNN	NB	LR	Bagging (RF)	Boosting (AB, XGB)			
Rabihah Md. Sum et al.[30]	2022			✓				✓		Personal loan applicants	Malaysia	AUC 0.64
Zahra Faraji[31]	2022	✓			✓			✓	✓	Credit card fraud detection	-	ACC 99%
Dhanashri A. et al.[32]	2022	✓						✓	✓	Loan Approbation prediction	Afghanistan	ACC 88.53%
Doumpos M et al. [2]	2020			✓				✓	✓	Bank collapse expectation	USA	AUC 0.97
Khikmah L et al. [21]	2019	✓	✓	✓	✓	✓	✓	✓	✓	Long-term deposit forecast	Portugal	ACC 97.07%
Huang J et al. [18]	2019		✓							Discovery of bank account fraud	-	ACC 97.39%
Ravi V et al. [25]	2019	✓	✓	✓	✓	✓	✓	✓	✓	Analysis of customer satisfaction for banks	India	AUC 0.826
Farooqi et al. [22]	2019	✓	✓	✓	✓	✓				Prediction of the results of telemarketing by banks	Portugal	ACC 91.2%
Climent F et al. [15]	2019							✓	✓	Bank collapse expectation	USA	ACC 94.74%
Jing et al. [16]	2018		✓	✓				✓		Bank collapse expectation	USA	AUC 0.916
Lahmiri S. [23]	2017		✓							Prediction of the results of telemarketing by banks	Portugal	ACC 71%
Marinakos et al. [4]	2017	✓	✓	✓	✓	✓	✓			Classification of the bank's customers for marketing directly	Portugal	AUC 0.90
Ghaneei H et al. [17]	2016	✓								Prediction of customer attrition in banks	-	AUC 0.929
Yue et al. [29]	2016	✓		✓	✓			✓	✓	Forecasting loan defaults	China	AUC 0.965

All of the studies mentioned in the table above us only one method for categorization techniques in loan applications, rather than a combination of methods. Additionally, studies integrating the support vector machine (SVM) and the K-nearest neighbor algorithm (KNN) have also been conducted. This research aims to identify the algorithms used to detect fraud in the banking sector, specifically loan fraud.

Data mining may assist the banking sector in acquiring new customers and keeping hold of current ones. Customer recruitment and retention are issues that should concern any industry, but finance should be given special attention. Today's consumers have many different opinions regarding where to transact [17]. Executives in the banking industry must therefore be aware that if they do not give their consumers their entire attention, they can readily locate another bank that will. By delivering advantages tailored to each customer's needs and using data collection to identify target clients seeking services, a bank may be able to keep its existing clients and goods and learn about a customer's past purchasing habits. To prevent losing its profitable customers to other banks, Chase Madison Banks in New York started using data collection to review client accounts and alter its parameters for creating new accounts [25]. Medi-

cal centre Fleet Bank uses data mining to identify the most suitable new buyers for its equity investment offers. To determine which customers could be more likely to purchase a corporate bond, the bank analyses client demographics and financial accounts across several product lines. This data is then utilized to attract those people. With client profiles compiled through data mining, the contact centre for Bank of America is prepared to offer new services and offers that are most pertinent to each caller. Another difficulty is that the finances are client retention. Using slashing Web-based technologies, making predictions, and consumer advertising helps lenders attract new customers and retain existing ones.

According to the findings of this study, SVM and KNN may be used to forecast the likelihood that a customer will be approved for a loan.

4. FRAUD DETECTION IN BANKING SECTOR

In the Banking sector, data mining may be used to detect fraud. Since fraud detection is a priority for many businesses, data mining has increased the amount of fraud that is being identified and reported. Two separate methods have been developed by financial organizations to identify fraud trends.

In the first method, a bank employs data mining software and a third-party data warehouse to discover fraud trends. The bank can then check for indications of internal issues by comparing those patterns to its own database. The second method relies only on internal bank data to identify fraud patterns. The majority of banks take a hybrid approach [1]. The banking industry is putting in more effort to detect fraud. Fraud management requires a lot of expertise. Because it reveals which transactions were not allowed by the user, it is essential in the identification of fraud.

Advertising is one of the often-used data mining applications in the banking sector. In order to evaluate customer information and create statistically accurate portraits of customers' preferences for items and services, banks' marketing teams can utilize data mining [42]. By just offering the goods and services that customers genuinely want, banks can save a significant amount of funds on marketing and discounts that would be useless [32]. As a result, bank marketers must focus on their customers by learning more about them. Obviously, to increase sales and improve service quality, Citigroup uses marketing technology. Due to the unification of four years' supply of client history paperwork, the business was able to market to and provide consumers with tailored services.

A key use of data mining in the financial system is fraud detection. Too many businesses are concerned about being able to spot fraudulent conduct, and data analysis is helping to find and report more suspicious transactions. Two unique techniques have been developed by financial organizations to spot fraud patterns. The first technique involves a bank obtaining a third coalition's data center, which may include metadata from several organizations, and using information retrieval techniques to identify fraud patterns. The bank can then check for any signs of internal problems by comparing those characteristics to its own database. In the second method, only data that the bank already has is used to identify fraud trends. The vast majority of banks use a "hybrid" approach. One approach that has been successful in detecting fraud is Falcon's "fraud assessment." It examines the activity for 60% of the cards that consumers in the country have, and it is used by nine of the top ten banks that offer credit cards. Mellon Bank can better safeguard itself and its customers' assets from prospective fraudulent transactions by using data mining for fraud prevention.

This section provides a review of previous research on fraud detection in the loan banking sector.

Goyal et al. [1] provided an example of several extensively used DM and MLT for detecting credit card fraud. Investigations into credit card fraud have been done in several ways. It began by outlining the significance of the subject and the present limitations in customary practices. The danger associated with counterfeit transactions varies; hence it is important to develop efficient and precise methods for identifying high-risk transactions. Standard data mining techniques are insufficient to identify these transactions. To find the best solution, advanced algorithms should be used.

The examination of each feature's information gain ratio serves as the foundation for feature selection. The division and conquer technique is imitated in the construction of

the lower directions in data mining, which also follows the same process of information gain evaluation. Three essential components are included while creating a technique for solving a problem.

The technique of extracting knowledge from vast quantities of unstructured data is known as data mining. The learning must be fresh, obscure, applicable, and defensible in the field in which it was discovered.

Singh et al. [33] proposed a structure based on three-layer verification techniques. In order to detect and reduce fraudulent credit card applications and transactions, they used threshold values, a genetic algorithm, community detection, and spike detection to accomplish this. The outcomes demonstrate the superiority of the suggested methodology. This essay also covers an extensive list of security recommendations that have helped credit card customers avoid fraudulent actions. The framework can make it easier for a rookie researcher to spot credit card (financial) fraud.

Kavipriya et al. [34] developed a system that uses effective clustering and classification techniques like apriori and support vector machines to analyze, spot, and identify fraudulent transactions. The results show that the proposed method outperforms the existing hidden Markov model in terms of fraud coverage while also having a low false alarm rate. The credit card fraud detection system was discussed in this thesis. The suggested technique has involved extensive testing on several different kinds of transactions. The findings were encouraging; nearly all fraudulent transactions were successfully identified, and when the new technique was compared to the current method, the outcome showed that it outperformed the latter.

Suresh et al. [35] presented a survey that represents a systematic examination of data mining techniques and how they are used in the processing of credit cards. The primary focus of the study was on data mining techniques, especially as they are used in credit card processing, which helps to spotlight considerably bigger components. As a result, this survey should be highly helpful for both academics doing a thorough evaluation of the literature in their subject and credit card companies choosing an effective solution for their problem. This article served as an overview of methods for identifying fraudulent credit card use and credit card fraud. Missions for categorization and prediction are particularly important in the credit card procedure.

A reliable and serious approach for identifying accounting information was put out by Yao et al. [36]. Businesses with a fraudulent financial statement (FFS) and non-FFS cases between 2002 and 2013 are their research subjects. Support vector machine (SVM), decision tree (DT), artificial neural network (ANN), and bayesian belief network (BBN) were utilized to detect FFS. Conventional statistical methods, such as regression models, have a greater mistake rate than data mining methods.

Zhou et al. [37] surveyed the evidence on financial products and fraud detection methods based on ensemble, transfer, supervised, unsupervised, and semi-supervised techniques. The bulk of fraud detection systems has been found to employ at least one supervised learning technique. There are still challenges to be resolved in this area,

though. Beginning with feature engineering, parameter selection, and hyper parameter tweaking, data mining-based credit scoring and fraud detection encounter the same challenges as other categorization tasks. Second, it is practically hard to describe complicated financial situations, particularly those in China, and researchers do not have access to sufficient public data to train and evaluate their models.

Talavera et al. [38] developed a modern strategy for using computational intelligence approaches to identify fraudulent credit transactions. To train a classification and clustering algorithm, they divided a dataset of historical customer data from a financial institution. They train a radial basis function network to assess if a client engages in credit fraud. Then, to create customer profiles, they construct a fuzzy c-means clustering. This algorithm can give a degree of membership in the form of points outside of clusters and group data inside clusters.

Dushyant Singh et al. [39] focused on analyzing credit card transaction datasets to identify patterns that fraudulent credit card transactions follow to help implement and design a credit card fraud detection algorithm. The data is analyzed by creating histograms of the variables in the dataset and a correlation matrix of the variables. The data analysis also assists us in determining the best machine-learning techniques to use when implementing the algorithm. The algorithm is implemented using the local outlier factor machine learning technique, which shows that this technique is highly accurate but there is a low accuracy in detecting credit card fraud. The algorithm can detect fraudulent transactions that match the pattern.

A comprehensive program for detecting and preventing fraud must include quality as the key for managers and employees to fraud awareness. Employee tips are the primary source of professional fraud detection. Actually, my study demonstrates that companies with anti-fraud training programs for leaders, executives, and personnel have lower losses and shorter scams than companies without such initiatives. At the very least, staff members should be informed about what constitutes fraud, how it negatively affects everyone in the business, and how to report suspicious activity.

Kirkos E. et al. [40] investigated the efficiency of several classification methods of fraudulent financial statement detection (FFS) using data mining (DM).

Researchers also determined the main FFS potential risks. The authors used neural networks, decision trees, and bayesian belief networks for classification (BBN). Different ratios and variables derived from financial statements are used as input for their experiments, such as total assets, working capital, sales to total assets, net income, quick assets, liabilities, fixed assets to total assets, and earnings before interest and taxes. They also concentrated on management fraud, which is induced by managers in order to meet targets while concealing losses or debt. Financial distress, they claim, is also a motivator for management fraud.

Various forms of fraudulent activity were categorized by West et al. [41] as follows: 1) Economic fraud 2) Business fraud the theft of insurance. Bribery, financial crimes, and fraudulence were their additional three classifications for banking fraud. Corporate fraud also includes deception in-

volving financial statements, commodities, and securities. Fig.2 displays various variations in fraud detection methods. Additionally, they categorize financial crimes into two groups: healthcare fraud and motor insurance fraud. They said that standard audit committee fraud prevention is no longer feasible in the era of big data. In the given figure, the blue color represents health care fraud and the green color represents motor insurance fraud.

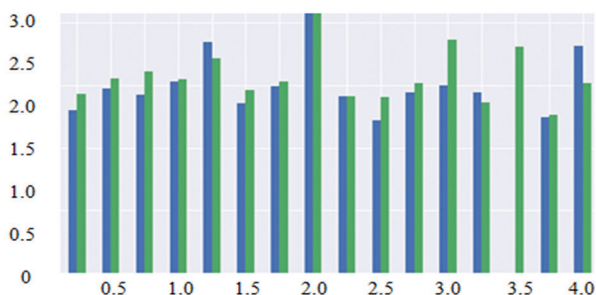


Fig. 2. Fluctuations in fraud detection tasks

Finally, we found out numerous challenges in the detection of financial fraud from table 3, as following:

- Financial fraud is an ever-changing field. Being one step ahead of the offenders is essential.
- Financial fraud detection methods vary depending on the situation.
 - There is no one-size-fits-all approach to data mining.
 - Sometimes hybrid methods are more effective.
 - Parameter tuning improves results. It took a lot of trial and error to find the best set of parameters.
- Privacy concerns led to corporate reluctance to share information, which resulted in various experimental limitations such as under-sampling.
- To avoid loss, financial fraud requires near-real-time detection.
- Misclassification has a financial and/or business cost. As a result, more emphasis should be placed on performance (accuracy and time) versus misclassification cost.
- Having a common guideline that can handle fraud cases across several domains could be advantageous.
- The financial dataset is highly skewed [a few fraudulent transactions among millions of transactions].

Besides the previously mentioned challenges, traditional approaches have limitations.

Every lending decision made by a bank involves some level of risk. This risk may be quantified to facilitate risk management and lower the possibility of financial loss for the bank. Credit management can make better judgments if they are aware of the repayment capacity of their customers.

Additionally, data mining may be used to identify whether clients would skip or default on loan installments. This increased information can help the bank make the necessary corrections to prevent losses. Consideration should be given to behavioral patterns, balance sheet numbers, turnover trends, limit use, and check return patterns when making such forecasts. Default patterns from the past can also be utilized to anticipate future defaults when comparable patterns are found.

To increase the accuracy of credit score predictions and decrease default probability, data mining techniques are applied. A borrower's creditworthiness is shown by their credit score. To predict a client's future behavior in a range of circumstances, behavioral scores are created

using probability models of customer behavior. By examining the borrower's previous debt repayment practices and the accessible credit history, data mining can determine this score, which will be used in our subsequent articles.

Table 3. Summary of fraud detection in banking sector

Author	Year	Proposed Work	Gap/ Future Work	Classification / Approach Used	Other Model / Techniques
Zahra Faraji[31]	2022	This study aims to present the commonly used supervised algorithms for fraud detection. In addition, this work aims to apply specific strategies, evaluate how well they work on actual data, and create an ensemble model as a viable solution to this problem.	The data is only one of the drawbacks of this study. The results of this study cannot be generalized to all banks or financial institutions because the data were limited to a single financial institution. Future studies could investigate machine learning methods with larger data sets. Another disadvantage is that unsupervised techniques were not used in this study.	decision tree, logistic regression, KNN, random forest, and XGBoost	Not Used
M Sathy et al [42]	2019	They suggested approach (the credit card expenditure model) is better at lowering the percentage of false alarms since it looks at the correlation between fraudulent transactions and those that are only suspected of fraud.	Future work on this study is By integrating more rules in the rule engine that models expand, the system's accuracy may be increased (Hidden Markov and K-mean clustering).	Not Used	Clustering, Hidden Markov Model, k-mean
Janaki K. et al [43]	2019	They suggested a mechanism to identify and stop fraudulent transactions and acts to lessen the economic industry's unit of drop.	The restrictions are that Web-based research models are utilized to reach out by taking these tactics into account. On the other hand, you may investigate the alternative models online. The location of extortion cases may be rapidly determined by the internet.	Naive Bayes, Decision Tree, Random Forest	Not Used
Aswathy et al [44]	2018	They frequently used techniques such as genetic algorithms, rule induction, SVM, ANN, decision tree, and logistic regression. The neural network is the most commonly used algorithm for detecting fraud. Furthermore, these algorithms can be used singly or in combination to create models.	Not Mentioned	SVM, Logistic Regression, Neural Network, Decision Tree.	Rule Induction, Genetic Algorithm
S.Vimala et al [45]	2017	They presented a research paper on using data mining to identify fraud in credit cards. They conclude that the optimum method for detecting fraud is to use a hidden Markov model, and decision tree.	The clustering algorithm and Markov chain model need to be enhanced further to prevent future fraud.	Neural Network, Decision Tree	Genetic Algorithm, Hidden Markov Model, k-mean

5. FINDINGS

The study's most recent findings investigate the fraud detection tools currently in use. We examined indicators for model performance that were based on the statistics. We demonstrated the application of data mining approaches to model evaluation. Considering evaluation, performance metrics including accuracy, recall, and sharpness are obtained. Analysis using data mining techniques offers a very visual summary of a model's performance. It is significant with regard to class skew, making it a trustworthy performance metric in numerous significant application domains for fraud detection. A comprehensive program for preventing and detecting fraud must include targeted training for managers and employees on fraud awareness. Employee tips are the primary source of occupational fraud detection, but my study also demonstrates that companies with anti-fraud training programs for managers, ceos, and personnel have lower losses and shorter scams than companies without such initiatives. Staff members ought to be informed at a minimum about what constitutes fraud, how it hurts the corporation as a whole, and how to report suspicious activity.

6. CONCLUSIONS

Data mining is a technique that supports the banking and retail industries for making better decisions by sifting through the vast amount of already available evidence to find particular information. Data processing condenses a variety of information into a manageable form so that it may be mined. The organization as a whole uses research methodology to gather data to support for decision-making. The financial industry can benefit significantly from the use of data mining techniques for better client acquisition, automatic lending for fraud detection, real-time fraud detection, segment product design, analysis of existing money transfer data for better client service and client retention, risk management, and advertising.

To conclude, loan fraud prevention became more and more challenging as research develops and the variety of commitment to giving rises. The conventional auditing approach for detecting loan fraud is no longer applicable because it is manual, labor-intensive, expensive, and inaccurate. Fraud is a serious issue in the financial industry. Fraudsters always devise new schemes. As they attempt to avoid detection, the plans become more intricate, making

it difficult to detect and stop fraud. This paper aims to provide a comprehensive analysis of fraud detection and data mining applications in the banking sector. A framework will soon be proposed as an improvement over the limitations offered by the approaches examined for the study.

7. REFERENCES:

- [1] R. Goyal, A. Kumar, "Review on Credit Card Fraud Detection using Data Mining Classification Techniques & Machine Learning Algorithms", *International Journal of Research and Analytical Reviews*, Vol. 7, No. 1, 2020, pp. 972-975.
- [2] M. Georgios, M. Doumpos, C. Zopounidis, E. Galariotis. "An Ordinal Classification Framework for Bank Failure Prediction: Methodology and Empirical Evidence for US Banks", *European Journal of Operational Research*, Vol. 282, No. 2, 2020, pp. 786-801.
- [3] A. Gupta, V. Pant, S. Kumar, P. K. Bansal, "Bank Loan Prediction System using Machine Learning", *Proceedings of the 9th International Conference on System Modeling and Advancement in Research Trends*, Moradabad, India, 4-5 December 2020, pp. 423- 426.
- [4] G. Marinakos, S. Daskalaki, "Imbalanced customer classification for bank direct marketing", *Journal of Marketing Analytics*, Vol. 5, No. 1, 2017, pp. 14-30.
- [5] B. Baesens, V. Van Vlasselaer, W. Verbeke, "Fraud Analytics Using Descriptive, Predictive, and Social Network Techniques: A Guide to Data Science for Fraud Detection", John Wiley & Sons, 2015.
- [6] S. Bhattacharyya, S. Jha, K. Tharakunnel, J. C. Westland, "Data mining for credit card fraud: A comparative study", *Decision support systems*, Vol. 50, No. 3, 2011, pp. 602-613.
- [7] A. Abdallah, M. A. Maarof, A. Zainal, "Fraud detection system: A survey", *Journal of Network and Computer Applications*, Vol. 68, 2016, pp. 90-113.
- [8] M. Kharote, V. P. Kshirsagar, "Data mining model for money laundering detection in financial domain", *International Journal of Computer Applications*, Vol. 85, No. 16, 2014.
- [9] O. Gottlieb, C. Salisbury, H. Shek, V. Vaidyanathan, "Detecting corporate fraud: An application of machine learning", *A publication of the American Institute of Computing*, 2006, pp. 100-215.
- [10] K. Golmohammadi, O. R. Zaiane, "Data mining applications for fraud detection in securities market", *Proceedings of the IEEE European Intelligence and Security Informatics Conference*, Odense, Denmark, 22-24 August 2012, pp. 107-114.
- [11] M. A. Sheikh, A. K. Goel, T. Kumar, "An approach for prediction of loan approval using machine learning algorithm", *Proceedings of the IEEE International Conference on Electronics and Sustainable Communication Systems*, 2-4 July 2020, pp. 490-494.
- [12] J. Sanjaya, E. Renata, V. E. Budiman, F. Anderson, M. Ayub, "Prediksi Kelalaian Pinjaman Bank Menggunakan Random Forest dan Adaptive Boosting", *Jurnal Teknik Informatika Dan Sistem Informasi*, Vol. 6, No. 1, 2020, pp. 50-60.
- [13] K. Gupta, B. Chakrabarti, A. A. Ansari, S. S. Rautaray, M. Pandey, "Loanification- Loan Approval Classification using Machine Learning Algorithms", *Proceedings of the International Conference on Innovative Computing & Communication*, 24 April 2021, pp. 1-4.
- [14] A. Kulothungan, "Loan Forecast by Using Machine Learning", *Turkish Journal of Computer and Mathematics Education*, Vol. 12, No. 7, 2021, pp. 894-900.
- [15] F. Climent, P. Carmona, A. Momparler, "Predicting failure in the US banking sector: An extreme gradient boosting approach", *International Review of Economics & Finance*, Vol. 61, 2019, pp. 304-323.
- [16] Z. Jing, Y. Fang, "Predicting US bank failures: A comparison of logit and data mining models", *Journal of Forecasting*, Vol. 37, No. 2, 2018, pp. 235-256.
- [17] H. Ghaneei, A. Keramati, S. M. Mirmohammadi, "Developing a prediction model for customer churn from electronic banking services using data mining", *Financial Innovation*, Vol. 2, No. 1, 2016, pp. 1-13.
- [18] J. Huang, W. Wang, Y. Wei, Y. Sun, "A two-route CNN model for bank account classification with heterogeneous data", *PlosOne*, Vol. 14, No. 8, 2019, p.e0220631.
- [19] I. Smeureanu, G. Ruxanda, L. M. Badea, "Customer segmentation in private banking sector using machine learning techniques", *Journal of Business Economics and Management*, Vol. 14, No. 5, 2013, pp. 923-939.
- [20] F. N. Ogwueleka, S. Misra, R. Colomo, L. Fernandez, "Neural network and classification approach in identifying customer behavior in the banking sector: A case study of an international bank", *Human factors and ergonomics in manufacturing & service industries*, Vol. 25, No. 1, 2015, pp. 28-42.

- [21] L. Khikmah, A. Ilham, A. Indra, "Long-term deposits prediction: a comparative framework of classification model for predict the success of bank telemarketing", *Journal of Physics: Conference Series*, Vol. 1175, No. 1, 2019, p. 12035.
- [22] R. Farooqi, N. Iqbal, "Performance evaluation for competency of bank telemarketing prediction using data mining techniques", *International Journal of Recent Technology and Engineering*, Vol. 8, No. 2, 2019, pp. 5666-5674.
- [23] S. Lahmiri, "A two-step system for direct bank telemarketing outcome classification", *Intelligent Systems in Accounting, Finance and Management*, Vol. 24, No. 1, 2017, pp. 49-55.
- [24] S. Moro, P. Cortez, P. Rita, "A data-driven approach to predict the success of bank telemarketing", *Decision Support Systems*, Vol. 62, 2014, pp. 22-31.
- [25] V. Ravi, G. Krishna, B. Reddy, M. Zaheeruddin, "Sentiment classification of Indian Banks' Customer Complaints", *Proceedings of the IEEE in 10th Annual International Conference*, Kochi, India, 17-20 October 2019, pp. 429-434.
- [26] H. Ramachandra, G. Balaraju, R. Divyashree, H. Patil, "Design and Simulation of Loan Approval Prediction Model using AWS Platform", *Proceedings of the IEEE International Conference on Emerging Smart Computing and Informatics*, 5 March 2021, pp. 53-56.
- [27] M. Alaradi, S. Hilal, "Tree-Based Methods for Loan Approval", *Proceedings of the International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy*, 2020, pp. 1-6.
- [28] M. A. Sheikh, A. K. Goel, T. Kumar, "An Approach for Prediction of Loan Approval using Machine Learning Algorithm", *Proceedings of the IEEE International Conference on Electronics and Sustainable Communication Systems*, Coimbatore, India, 2-4 July 2020, pp. 490-494.
- [29] Z. Yue, J. Wan, D. Yang, Y. Zhang, "Predicting non performing loan of business Bank with data mining techniques", *International Journal of Database Theory and Application*, Vol. 9, No. 12, 2016, pp. 23-34.
- [30] A. D. Wani, S. Adarsha, "In Banking Prediction of Loan Approbation Using Machine Learning", *International Journal of Research Publication and Reviews*, Vol. 3, No. 5, 2022, p. 7421.
- [31] Z. Faraji, "A Review of Machine Learning Applications for Credit Card Fraud Detection with A Case study", *SEISENSE Journal of Management*, Vol. 5, No. 1, 2022, pp. 49-59.
- [32] R. Sum, M. Rabihah, "A New Efficient Credit Scoring Model for Personal Loan using Data Mining Technique Toward for Sustainability Management", *Journal of Sustainability Science and Management*, Vol. 17, No. 5, 2022, pp. 60-76.
- [33] Singh, Ajeet, Anurag, "A Novel Framework for Credit Card Fraud Prevention and Detection (CCFPD) Based on Three Layer Verification Strategy", *Proceedings of the ICETIT: Emerging Trends in Information Technology*, 2020, pp. 935-948.
- [34] T. Kavipriya, N. Geetha, "An identification and detection of fraudulence in credit card fraud transaction system using data mining techniques", *International Research Journal of Engineering and Technology*, Vol. 5, No. 1, 2018.
- [35] G. Suresh, R. J. Raj, "A Study on Credit Card Fraud Detection using Data Mining Techniques", *International Journal of Data Mining Techniques and Applications*, Vol. 7, No. 1, 2018, pp. 21-24.
- [36] J. Yao, J. Zhang, L. Wang, "A financial statement fraud detection model based on hybrid data mining methods", *Proceedings of the IEEE International Conference on Artificial Intelligence and Big Data*, Chengdu, China, 26-28 May 2018, pp. 57-61.
- [37] X. Zhou, P. Xu, "A state of the art survey of data mining based fraud detection and credit scoring", *Proceedings of MATEC Web of Conferences*, EDP Sciences, Vol.189, 2018, p. 3002.
- [38] Talavera, Alvaro, "Data Mining Algorithms for Risk Detection in Bank Loans", *Proceedings of Springer Annual International Symposium on Information Management and Big Data*, 2018, pp. 151-159.
- [39] D. Singh, S. Vardhan, N. Agrawal, "Credit Card Fraud Detection Analysis", *International Research Journal of Engineering and Technology*, Vol. 5, No. 11, 2018, pp. 1600-1603.
- [40] E. Kirkos, C. Spathis, Y. Manolopoulos, "Data mining techniques for the detection of fraudulent financial statements", *Expert systems with applications*, Vol. 32, No. 4, 2007, pp. 995-1003.
- [41] J. West, M. Bhattacharya, "Intelligent financial fraud detection practices: an investigation", *Proceedings of the Springer International Conference on Security and Privacy in Communication Networks*, 2014, pp. 186-203.

- [42] M. Sathyapriya, V. Thiagarasu, "A Cluster Based Approach for Credit Card Fraud Detection System using HMM with the Implementation of Big Data Technology", *International Journal of Applied Engineering Research*, Vol. 14, No. 2, 2019, pp. 393-396.
- [43] K. Janaki, V. Harshitha, S. Keerthana, Y. Harshitha, "A Hybrid Method for Credit Card Fraud Detection Using Machine Learning Algorithm", *International Journal of Recent Technology and Engineering*, Vol. 7, No. 654, 2019, pp. 235-239.
- [44] M. S. Aswathy, L. Sameul, "Survey on Credit Card Fraud Detection", *International Research Journal of Engineering and Technology*, Vol. 05, No.11, Nov 2018, pp.1291-1294.
- [45] S. Vimala, K. C. Sharmili, "Survey Paper for Credit Card Fraud Detection Using Data Mining Techniques", *International Journal of Innovative Research in Science, Engineering and Technology*, Vol. 6, No. 11, 2017, pp.357-364.

INTERNATIONAL JOURNAL OF ELECTRICAL AND COMPUTER ENGINEERING SYSTEMS

Published by Faculty of Electrical Engineering, Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia.

About this Journal

The International Journal of Electrical and Computer Engineering Systems publishes original research in the form of full papers, case studies, reviews and surveys. It covers theory and application of electrical and computer engineering, synergy of computer systems and computational methods with electrical and electronic systems, as well as interdisciplinary research.

Topics of interest include, but are not limited to:

- Power systems
- Renewable electricity production
- Power electronics
- Electrical drives
- Industrial electronics
- Communication systems
- Advanced modulation techniques
- RFID devices and systems
- Signal and data processing
- Image processing
- Multimedia systems
- Microelectronics
- Instrumentation and measurement
- Control systems
- Robotics
- Modeling and simulation
- Modern computer architectures
- Computer networks
- Embedded systems
- High-performance computing
- Parallel and distributed computer systems
- Human-computer systems
- Intelligent systems
- Multi-agent and holonic systems
- Real-time systems
- Software engineering
- Internet and web applications and systems
- Applications of computer systems in engineering and related disciplines
- Mathematical models of engineering systems
- Engineering management
- Engineering education

Paper Submission

Authors are invited to submit original, unpublished research papers that are not being considered by another journal or any other publisher. Manuscripts must be submitted in doc, docx, rtf or pdf format, and limited to 30 one-column double-spaced pages. All figures and tables must be cited and placed in the body of the paper. Provide contact information of all authors and designate the corresponding author who should submit the manuscript to <https://ijeces.ferit.hr>. The corresponding author is responsible for ensuring that the article's publication has been approved by all coauthors and by the institutions of the authors if required. All enquiries concerning the publication of accepted papers should be sent to ijeces@ferit.hr.

The following information should be included in the submission:

- paper title;
- full name of each author;
- full institutional mailing addresses;
- e-mail addresses of each author;
- abstract (should be self-contained and not exceed 150 words). Introduction should have no subheadings;
- manuscript should contain one to five alphabetically ordered keywords;
- all abbreviations used in the manuscript should be explained by first appearance;
- all acknowledgments should be included at the end of the paper;
- authors are responsible for ensuring that the information in each reference is complete and accurate. All references must be numbered consecutively and citations of references in text should be identified using numbers in square brackets. All references should be cited within the text;
- each figure should be integrated in the text and cited in a consecutive order. Upon acceptance of the paper, each figure should be of high quality in one of the following formats: EPS, WMF, BMP and TIFF;
- corrected proofs must be returned to the publisher within 7 days of receipt.

Peer Review

All manuscripts are subject to peer review and must meet academic standards. Submissions will be first considered by an editor-

in-chief and if not rejected right away, then they will be reviewed by anonymous reviewers. The submitting author will be asked to provide the names of 5 proposed reviewers including their e-mail addresses. The proposed reviewers should be in the research field of the manuscript. They should not be affiliated to the same institution of the manuscript author(s) and should not have had any collaboration with any of the authors during the last 3 years.

Author Benefits

The corresponding author will be provided with a .pdf file of the article or alternatively one hardcopy of the journal free of charge.

Units of Measurement

Units of measurement should be presented simply and concisely using System International (SI) units.

Bibliographic Information

Commenced in 2010.
ISSN: 1847-6996
e-ISSN: 1847-7003

Published: semiannually

Copyright

Authors of the International Journal of Electrical and Computer Engineering Systems must transfer copyright to the publisher in written form.

Subscription Information

The annual subscription rate is 50€ for individuals, 25€ for students and 150€ for libraries.

Postal Address

Faculty of Electrical Engineering,
Computer Science and Information Technology Osijek,
Josip Juraj Strossmayer University of Osijek, Croatia
Kneza Trpimira 2b
31000 Osijek, Croatia

IJECES Copyright Transfer Form

(Please, read this carefully)

This form is intended for all accepted material submitted to the IJECES journal and must accompany any such material before publication.

TITLE OF ARTICLE (hereinafter referred to as "the Work"):

COMPLETE LIST OF AUTHORS:

The undersigned hereby assigns to the IJECES all rights under copyright that may exist in and to the above Work, and any revised or expanded works submitted to the IJECES by the undersigned based on the Work. The undersigned hereby warrants that the Work is original and that he/she is the author of the complete Work and all incorporated parts of the Work. Otherwise he/she warrants that necessary permissions have been obtained for those parts of works originating from other authors or publishers.

Authors retain all proprietary rights in any process or procedure described in the Work. Authors may reproduce or authorize others to reproduce the Work or derivative works for the author's personal use or for company use, provided that the source and the IJECES copyright notice are indicated, the copies are not used in any way that implies IJECES endorsement of a product or service of any author, and the copies themselves are not offered for sale. In the case of a Work performed under a special government contract or grant, the IJECES recognizes that the government has royalty-free permission to reproduce all or portions of the Work, and to authorize others to do so, for official government purposes only, if the contract/grant so requires. For all uses not covered previously, authors must ask for permission from the IJECES to reproduce or authorize the reproduction of the Work or material extracted from the Work. Although authors are permitted to re-use all or portions of the Work in other works, this excludes granting third-party requests for reprinting, republishing, or other types of re-use. The IJECES must handle all such third-party requests. The IJECES distributes its publication by various means and media. It also abstracts and may translate its publications, and articles contained therein, for inclusion in various collections, databases and other publications. The IJECES publisher requires that the consent of the first-named author be sought as a condition to granting reprint or republication rights to others or for permitting use of a Work for promotion or marketing purposes. If you are employed and prepared the Work on a subject within the scope of your employment, the copyright in the Work belongs to your employer as a work-for-hire. In that case, the IJECES publisher assumes that when you sign this Form, you are authorized to do so by your employer and that your employer has consented to the transfer of copyright, to the representation and warranty of publication rights, and to all other terms and conditions of this Form. If such authorization and consent has not been given to you, an authorized representative of your employer should sign this Form as the Author.

Authors of IJECES journal articles and other material must ensure that their Work meets originality, authorship, author responsibilities and author misconduct requirements. It is the responsibility of the authors, not the IJECES publisher, to determine whether disclosure of their material requires the prior consent of other parties and, if so, to obtain it.

- The undersigned represents that he/she has the authority to make and execute this assignment.
- For jointly authored Works, all joint authors should sign, or one of the authors should sign as authorized agent for the others.
- The undersigned agrees to indemnify and hold harmless the IJECES publisher from any damage or expense that may arise in the event of a breach of any of the warranties set forth above.

Author/Authorized Agent

Date

CONTACT

International Journal of Electrical and Computer Engineering Systems (IJECES)
Faculty of Electrical Engineering, Computer Science and Information Technology Osijek
Josip Juraj Strossmayer University of Osijek
Kneza Trpimira 2b
31000 Osijek, Croatia
Phone: +38531224600,
Fax: +38531224605,
e-mail: ijeces@ferit.hr