

IAES International Journal of Artificial Intelligence (IJ-AI)

Vol 11, No 4: December 2022

IAES International Journal of Artificial Intelligence (IJ-AI)

IAES International Journal of Artificial Intelligence (IJ-AI), ISSN/e-ISSN 2089-4872/2252-8938 publishes articles in the field of artificial intelligence (AI). The scope covers all artificial intelligence (AI) and machine learning (ML) areas and their applications in the following topics: neural networks; fuzzy logic; simulated biological evolution algorithms (like genetic algorithm, ant colony optimization, etc); reasoning and evolution; intelligence applications; computer vision and speech understanding; multimedia and cognitive informatics, data mining and machine learning tools, heuristic and AI planning strategies and tools, computational theories of learning; technology and computing (like particle swarm optimization); intelligent system architectures; knowledge representation; bioinformatics; natural language processing; multiagent systems; supervised learning; unsupervised learning; deep learning; big data and AI approaches; reinforcement learning; and learning with generative adversarial networks; etc. This journal is indexed in Scopus and all published papers since 2018 issues were included in scopus.com.

Focus and Scope

The IAES International Journal of Artificial Intelligence (IJ-AI), ISSN/e-ISSN 2089-4872/2252-8938 covers all topics of artificial intelligence and soft computing and their applications, including but not limited to:

- Neural networks
- Reasoning and evolution
- Intelligent search
- Intelligent planning
- Intelligence applications
- Computer vision and speech understanding
- Multimedia and cognitive informatics
- Data mining and machine learning tools, heuristic and AI planning strategies and tools, computational theories of learning
- Technology and computing (like particle swarm optimization); intelligent system architectures
- Knowledge representation
- Bioinformatics
- Natural language processing
- Automated reasoning
- Logic programming
- Machine learning
- Visual/linguistic perception
- Evolutionary and swarm algorithms
- Derivative-free optimisation algorithms
- Fuzzy sets and logic
- Rough sets
- Simulated biological evolution algorithms (like genetic algorithm, ant colony optimization, etc)
- Multi-agent systems
- Data and web mining
- Emotional intelligence
- Hybridisation of intelligent models/algorithms
- Parallel and distributed realisation of intelligent algorithms/systems
- Application in pattern recognition, image understanding, control, robotics and bioinformatics
- Application in system design, system identification, prediction, scheduling and game playing
- Application in VLSI algorithms and mobile communication/computing systems

Principal Contact

Prof. Dr. Eugene Yu-Dong Zhang

Editor-in-Chief, IJ-AI

Chair in Knowledge Discovery and Machine Learning

Associate Fellow of Higher Education Academy

IEEE Senior Member

ACM Senior Member

Contact

F26 Informatics Building

Department of Informatics

University of Leicester, University Road,

Leicester, LE1 7RH, UK

Email: ijai@iaesjournal.com

IAES International Journal of Artificial Intelligence (IJ-AI)

Editorial Team

Editor-in-Chief

Prof. Dr. Eugene Yu-Dong Zhang
University of Leicester, United Kingdom

Managing Editor

Assoc. Prof. Dr. Tole Sutikno
Universitas Ahmad Dahlan, Indonesia

Associate Editors

Prof. Dr. Cheng-Wu Chen
National Kaohsiung Marine University, Taiwan, Province of China

Prof. Dr. Kiran Sree Pokkuluri
Shri Vishnu Engineering College for Women, India

Prof. Dr. Odiel Estrada Molina
University of Informatics Science, Cuba

Prof. Francesca Guerriero
University of Calabria, Italy

Prof. Francisco Torrens
Universitat de Valencia, Spain

Prof. George A. Papakostas
International Hellenic University, Greece

Prof. Hongyang Chen
Zhejiang Lab, China

Prof. Ioannis Chatzigiannakis
Sapienza University of Rome, Italy

Prof. Jianbing Shen
Beijing Institute of Technology, China

Prof. Panlong Yang
University of Science and Technology of China, China

Prof. Pingyi Fan
Tsinghua University, China

Assoc. Prof. Dr. Kamil Dimililer
Near East University, Turkey

Assoc. Prof. Dr. Wudhichai Assawinchaichote
King Mongkut's University of Technology Thonburi, Thailand

Assoc. Prof. Ts. Dr. Muhammad Zaini Ahmad
Universiti Malaysia Perlis, Malaysia

Dr. Ahmed Toaha Mobashsher
University of Queensland, Australia

Dr. Ahnaf Hassan
North South University, Bangladesh

Dr. Aida Mustapha
Universiti Tun Hussein Onn Malaysia, Malaysia

Dr. Choong Seon Hong
Kyung Hee University, Korea, Republic of

Dr. Chunguo Li
Henan University of Science and Technology, China

Dr. D. Jude Hemanth
Karunya University, India

Dr. Dhiya Al-Jumeily
Liverpool John Moores University, United Kingdom

Dr. Farhad Soleimani Gharehchopogh
Hacettepe University, Turkey

Dr. Floriano De Rango
University of Calabria, Italy

Dr. Gloria Bordogna
Institute for Electromagnetic Sensing of the Environment, Italy

Dr. Honghai Liu
University of Portsmouth, United Kingdom

Dr. Ibrahim Kucukkoc
Balikesir University, Turkey

Dr. Igor Kotenko
Saint-Petersburg Institute for Informatics and Automation of the Russian Academy of Sciences, Russian Federation

Dr. Iickho Song
Korea, Republic of

Dr. Imam Much Ibnu Subroto
Universitas Islam Sultan Agung, Indonesia

Dr. Iztok Fister Jr.
University of Maribor, Slovenia

Dr. Javier Gozalvez
Miguel Hernandez University of Elche, Spain

Dr. Jingjing Wang
Tsinghua University, China

Dr. John S. Vardakas
Iquadrat Informatica S.L., Spain

Dr. Karan Veer
DR BR Ambedkar National Institute of Technology, India

Dr. Liang Yang
Hunan University, China

Dr. Lin X. Cai
Illinois Institute of Technology, United States

Dr. Magdi S. Mahmoud
King Fahd University of Petroleum and Minerals, Saudi Arabia

Dr. Miroslav Voznak
VSB-Technical University of Ostrava, Czech Republic

Dr. Mortaza Zolfpour Arokhlo
Sepidan Branch, Islamic Azad University, Iran, Islamic Republic of

Dr. Mufti Mahmud
Nottingham Trent University, United Kingdom

Dr. Muhammad Shahid Farid
University of the Punjab, Pakistan

Dr. Nasimuddin Nasimuddin
Institute for Infocomm Research, Singapore

Dr. Rashid Ali
Aligarh Muslim University, India

Dr. Saeed Jafarzadeh
California State University Bakersfield, United States

Dr. Saleh Mirheidari
Navistar Inc., United States

Dr. Shahaboddin Shamshirband
University of Malaya, Malaysia

Dr. Shaikh Abdul Hannan Abdul Mannan
, Vivekanand College, India

Dr. Sherali Zeadally
Lunghwa University of Science and Technology, Taiwan, Province of China

Dr. Syamsiah Mashohor
Universiti Putra Malaysia, Malaysia

Dr. Tomasz M. Rutkowski
RIKEN AIP, Japan

IAES International Journal of Artificial Intelligence (IJ-AI)

Vol 11, No 4 December 2022

Table of Contents

Smart pools of data with ensembles for adaptive learning in dynamic data streams with class imbalance Radhika Vikas Kulkarni, S. Revathy, Suhas Haribhau Patil	310-318
Wiki sense bag creation using multilingual word sense disambiguation Shreya Patankar, Madhura Phadke, Satish Devane	319-326
Automatic face recording system based on quick response code using multicam Julham Julham, Muharman Lubis, Arif Ridho Lubis, Al-Khowarizmi Al-Khowarizmi, Idham Kamil	327-335
Indonesian part of speech tagging using maximum entropy markov model on Indonesian manually tagged corpus Denis Eka Cahyani, Winda Mustikaningtyas	336-344
Effective predictive modelling for coronary artery diseases using support vector machine Kuncahyo Setyo Nugroho, Anantha Yullian Sukmadewa, Angga Vidiyanto, Wayan Firdaus Mahmudy	345-355
An efficient machine learning-based COVID-19 identification utilizing chest X-ray images Mahmoud Masadeh, Ayah Masadeh, Omar Alshorman, Falak H Khasawneh, Mahmoud Ali Masadeh	356-366
An efficient resource utilization technique for scheduling scientific workload in cloud computing environment Nagendra Prasad Sodinapalli, Subhash Kulkarni, Nawaz Ahmed Sharief, Prasanth Venkatareddy	367-378
AraBERT transformer model for Arabic comments and reviews analysis Hicham EL Moubtahij, Hajar Abdelali, El Bachir Tazi	379-387
Implementation of FaceNet and support vector machine in a real-time web-based timekeeping application Ly Quang Vu, Phan Thanh Trieu, Hoang-Sy Nguyen	388-396
Identify tooth cone beam computed tomography based on contourlet particle swarm optimization Hiba Adreese Younis, Dhafar Sami Hammadi, Ansam Nazar Younis	397-404

Smart pools of data with ensembles for adaptive learning in dynamic data streams with class imbalance

Radhika Vikas Kulkarni¹, S. Revathy¹, Suhas Haribhau Patil²

¹Department of Computer Science Engineering, Sathyabama Institute of Science and Technology, Chennai, India.

²Department of Computer Science Engineering, Bharati Vidyapeeth's College of Engineering, Pune, India

Article Info

Article history:

Received Jun 3, 2021

Revised Dec 16, 2021

Accepted Dec 28, 2021

Keywords:

Adaptive learning

Class imbalance

Concept drift

Data streams classification

Ensemble

Online learning

ABSTRACT

Streaming data incorporates dynamicity due to a nonstationary environment where data samples may endure class imbalance and change in data distribution over the period causing concept drifts. In real-life applications learning in dynamic data streams, is vitally important and challenging. A combined solution to adapt to class imbalance and concept drifts in dynamic data streams is rarely addressed by researchers. With this motivation, the current communication presents the online ensemble model smart pools of data with ensembles for class imbalance adaptive learning (SPECIAL) to learn in skewed and drifting data streams. It employs an ageing-based G-mean maximization strategy to adapt to dynamicity in data streams. It employs smart data-pools with the local expertise ensemble to classify samples lying in the same data-pool. The empirical and statistical study on different evaluation metrics exhibits that SPECIAL is more adaptive to class imbalanced dynamic data streams than the state-of-the-art algorithms.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Radhika Vikas Kulkarni

Department of Computer Science Engineering, Sathyabama Institute of Science and Technology

Chennai, India

Email: radhikavikaskulkarni@gmail.com

1. INTRODUCTION

The current era of information technology demands predictive models for numerous real-life applications like social media sentiments extraction [1], financial risk prediction [2], network intrusion identification [3], and so on. Such applications produce a continuous stream of endless data with high volume and speed [4]-[6]. Data streams possess dynamicity due to the varying distribution of data in them. The change in data distribution is referred to as “concept drift” [5], [7]. The unequal number of class instances of a data stream cause skewness [8]-[9]. Learning in nonstationary data streams with the class imbalance and drifting concepts is a challenging task in computational intelligence [10].

Learning in nonstationary data streams has attracted many researchers [7]. Ensemble learning employs a combination of single classifiers to augment the generalization capability and is more suitable to handle dynamicity in data streams [4], [5], [11]-[13]. Block-based ensembles like accuracy weighted ensemble (AWE) [14], accuracy updated ensemble (AUE) [15], dynamic weighted majority for imbalance learning (DWMIL) [16], adaptive chunk-based dynamic weighted majority (ACDWM) [17] tackle to concept drifts by inserting a new learner trained on a new block of data. Online ensembles like ADWIN Bagging (BagADWIN) [18], online accuracy updated ensemble (OAUE) [19] train the model on each incoming data sample avoiding waiting for a block or whole training dataset. The implicit adaptation to concept drifts in data streams leads to passive learning algorithms like dynamic weighted majority (DWM) [20], anticipative

dynamic adaptation to concept change (ADACC) [21], dynamic classifier selection (DCS) [22]. However, explicit drift detection leads to an active learning approach as narrated in [23]-[26].

A rich literature on the class imbalance problem is available [8], [27] that describes three broad approaches to handle skewness in data: i) sampling-based, ii) algorithm-based, and iii) cost-sensitive. Most of the researchers have handled the problem of class imbalance and dynamicity in data streams separately. The combined solution to both problems is presented by comparatively fewer researchers [28]-[29].

Through this correspondence, we provide a novel combined solution to the issue of class imbalance and concept drifts in dynamic data streams. We propose an online ensemble classifier smart pools of data with ensembles for class imbalance adaptive learning (SPECIAL) to classify binary dynamic data streams. The contributions of the proposed work are:

- It provides a passive drift detection model using an online ensemble with a test-then-train approach.
- It employs an ageing-based strategy to adapt to the dynamicity in data.
- It deals with the class imbalance in streaming data with the objective of G-mean maximization.
- It investigates empirical and statistical test results and compares the performance of the proposed algorithm SPECIAL with state-of-the-art ensembles on different performance measures using various real and synthetic benchmark datasets.

2. RESEARCH METHOD

The current section describes the problem of this research and the proposed methodology to solve it. The proposed methodology elaborates the training and testing phase of the proposed algorithm SPECIAL. The section further presents the experimental framework to test the performance of the algorithm SPECIAL.

2.1. Problem formulation

The data stream DS arrives as a sequence of massive data samples $\{d_1, d_2, d_3, \dots\}$ where d_t is a data sample in the m -dimensional feature space arrived at time step $t = 1, 2, \dots$. Associated with each of the data sample d_t there is a class label $c_t \in C = \{c_1, c_2, \dots, c_L\}$ where L is the number of class labels. As our work focuses on a binary classification problem, we consider $C = \{0, 1\}$ where '0' defines a negative class, and '1' defines a positive class. Let DS^0 be the set of negative data samples and DS^1 be the set of positive data samples. We consider class imbalance in the data stream where $|DS^0| \gg |DS^1|$. A concept in the data stream is defined by a joint distribution $P_t(DS, C)$. At time step t , it generates a tuple (d_t, c_t) . The dynamicity in the data stream may cause concept drift where its joint distribution changes over the period i.e. $P_{t-1}(DS, C) \neq P_t(DS, C)$ [7]. As an online learner progresses with the recently received data samples, it is suitable for handling the sequential incoming flow of data samples of the data streams [7], [30]. Hence, for the prediction of a class c_t of an input data sample d_t of a dynamic data stream, we propose an online classifier $f: DS \rightarrow C$.

2.2. Proposed methodology

Through this communication, we present an online adaptive ensemble SPECIAL. It is an ensemble of ensembles $E = \{e_1, e_2, \dots, e_s\}$ where s is the number of sub ensembles to deal with dynamically drifting imbalanced data streams. We focus on the local expertise of each of the sub ensembles on the area of the m -dimensional feature space where the recent data sample has appeared. Thus, it prefers the predictions by the sub ensemble which exhibited better classification accuracy in this area.

The data samples in the same area of the m -dimensional feature space are mapped to a single data-pool which represents the m -dimensional hypersphere. The data samples in a data-pool point to their expertise sub ensemble e_i where $i \in \{1, 2, \dots, s\}$ which gives the lowest classification error for the data samples in that data-pool. Each data-pool p_j , $j \in \{1, 2, \dots, z\}$ where z is the number of existing data-pools is characterized by the following metadata at time step t :

- a. A pool-master (PM_t^j): It is a representative data sample of the j^{th} data-pool based on the minimum prediction error in its classification. At time step t , a tie with the same prediction errors, if any, is resolved by selecting the most recently mapped data sample to that pool as its pool-master.
- b. Pool size (PS_t^j): It is the number of data samples mapped to the j^{th} data-pool by the time step t .
- c. Average prediction error (APE_t^j): It is the average prediction error incurred in the classification of all data samples in the j^{th} data-pool by the time step t .
- d. Prediction factor (PF_t^j): It is given as $(PF_t^j) = 1 - (APE_t^j)$. The higher value of (PF_t^j) indicates that the expertise sub ensemble associated with the j^{th} data-pool shows better classification results.
- e. Representative factor (RF_t^j): It is a ratio of pool size of the j^{th} data-pool to the total number of data samples received by the time step t .

- f. Weight factor (WF_t^j): It is a ratio of prediction factor (PF_t^j) to representative factor (RF_t^j) of the j^{th} data-pool by the time step t . The pool with a higher weight factor is preferred as it gives better prediction with a lesser number of data samples mapped to it. And, as it has a smaller pool-size we can map more suitable data samples to it.

The proposed SPECIAL classifier employs a test-then-train approach. Through this approach, it first predicts a class of each sample (d_t, c_t) received at time step t and then uses the same for the training of the model. The detailed description of the testing phase and training phase of the learning model is given below.

2.2.1. Model testing phase

For each incoming data sample at time t , the proposed algorithm identifies its K nearest pools. It assigns the highest priority to one of these K nearest pool-masters if it is the most recently defined pool-master and has the highest weight factor (WF_t^j). Then the expertise learner associated with the selected highest priority j^{th} data-pool classifies the newly arrived data sample. On correct classification, the new data sample is mapped to the selected data-pool, and metadata of that data-pool is updated.

At the start when no data-pool is created and no pool-master is available, the first incoming data sample itself forms a new data-pool and becomes a pool-master. It is tested on each sub ensemble $e_i \in E$ where $i \in \{1, 2, \dots, s\}$. This newly formed data-pool points to the sub ensemble with the least prediction error. Accordingly, its metadata is updated. The prediction results are empirically and statistically examined on various evaluation metrics. The performance of the SPECIAL algorithm is compared with other state-of-the-art algorithms. Algorithm 1 describes the testing phase of the SPECIAL model.

Algorithm 1: SPECIAL model testing

Input: (d_t, c_t) is an incoming data instance at time $t=\{1, 2, \dots\}$ of data stream DS ; $E=\{e_1, e_2, \dots, e_s\}$ is an ensemble of s ensembles; $P=\{p_1, p_2, \dots, p_z\}$ is a collection of existing z data-pools.

Output: \hat{c}_t : A predicted class of data instance d_t .

```

1.  $P_{near} = \text{Search\_}K\text{-nearest-pools}(d_t, K, P)$ ;
2. if ( $P_{near} == \emptyset$ ) {
3.    $\hat{c}_t = \text{Get\_Prediction}(d_t, E)$ ;
4.    $p_t = \text{Create\_New\_Data-pool}(d_t)$ ; Update_Metadata( $p_t$ );
5. } else {
6.    $p = \text{Select\_Highest-priority\_Pool}(P_{near})$ ;
7.    $xe = \text{Get\_Associated\_Expertise\_Subensemble}(p)$ ;
8.    $\hat{c}_t = \text{Get\_Prediction}(d_t, xe)$ ;
9.   if ( $c_t == \hat{c}_t$ ) {
10.    Map_to_Data-pool( $d_t, p$ ); Update_Metadata( $p$ );
13. Evaluate the performance of the SPECIAL model on various evaluation metrics using ( $\hat{c}_t$ );

```

2.2.2. Model training phase

The proposed SPECIAL classifier employs online learning as presented by Oza N. [31]. Generally, the bootstrapping samples follow a normal distribution. Due to the unavailability of all training instances at the start and a huge volume of incoming streaming data, the normal distribution of bootstrapping samples is approximated by a Poisson (1) distribution, if the number of training instances $n \rightarrow \infty$.

The SPECIAL algorithm adapts to dynamicity in the data stream by assigning more weightage to the recently arrived data instances. It incorporates two predefined ageing metrics: 1) data-ageing metric β , ($0 < \beta < 1$) and 2) sensitivity-ageing metric γ , ($0 < \gamma < 1$) to give more weightage to the latest data and sensitivity (i.e. true positive rate). The function $f(c_t, c_L)$ returns 1 if the class label c_t of incoming data instance is c_L and 0 otherwise, where $L = \{0, 1\}$. The function $g(c_t, \hat{c}_t)$ returns 1 if the class label of incoming data instance is correctly predicted and 0 otherwise. Let (DS_t^L) where $L = \{0, 1\}$ be the metrics defining the percentages of negative class (if $L=0$) or positive class (if $L=1$) by time step t . Let (Se_t) be the metric defining sensitivity by time step t . The ageing-based computations of (DS_t^L) where $L = \{0, 1\}$ and (Se_t) are given by (1) and (2) respectively. Referring to these ageing-based computations shown in (1) and (2) it can be noticed that the ageing metrics β and γ force the old samples to contribute less to computations of metrics of class percentage and sensitivity, respectively.

$$DS_t^L = \beta \cdot DS_{t-1}^L + (1 - \beta) \cdot f(c_t, c_L); \text{ where } L = \{0, 1\} \quad (1)$$

$$Se_t = \gamma \cdot Se_{t-1} + (1 - \gamma) \cdot g(c_t, \hat{c}_t) \quad (2)$$

When the difference between class percentages $|(DS_t^0 - DS_t^1)|$ crosses the predefined threshold Ψ , the proposed SPECIAL algorithm considers it as a scenario of class imbalance at time step t . To handle a class imbalance in binary data streams, it focuses on G-mean improvisation as described in [32]. It focuses on both positive (minority) and negative (majority) classes and deemphasises negative class only when the class imbalance results in a poor true positive rate. Accordingly, when data size $n \rightarrow \infty$, and there is a class imbalance in an incoming data stream, the number of copies B of the positive and negative class samples by Poisson (λ) approximated bootstrapping at time step t is given by (3). In the training phase, the sub ensembles associated with data-pools are trained by using a B number of copies of the data sample (d_t, c_t) . The trained model of the SPECIAL classifier at time step t is used to test the unseen incoming data sample (d_{t+1}, c_{t+1}) at the next time step $t+1$. Algorithm 2 describes the training phase of the SPECIAL model.

$$B \sim \begin{cases} \text{Poisson}(Se_t); & \text{if } |(DS_t^0 - DS_t^1)| > \Psi \text{ and } c_t = 0 \\ \text{Poisson}\left(\frac{DS_t^0}{DS_t^1}\right); & \text{if } |(DS_t^0 - DS_t^1)| > \Psi \text{ and } c_t = 1 \\ \text{Poisson}(1); & \text{if } |(DS_t^0 - DS_t^1)| \leq \Psi \end{cases} \quad (3)$$

Algorithm 2: SPECIAL Model Training

Input: (d_t, c_t) is an incoming data instance at time $t=\{1, 2, \dots\}$ of data stream DS ; $E=\{e_1, e_2, \dots, e_s\}$ is an ensemble of s ensembles; $P=\{p_1, p_2, \dots, p_z\}$ is a collection of z data-pools; \hat{c}_t is a predicted class of instance d_t in the testing phase; Ψ is the imbalance threshold.

Output: An updated model of SPECIAL.

1. **Initialize:** $(DS_t^0)=(DS_t^1)=0$; $(Se_t)=0$;
2. **if** $(c_t \neq \hat{c}_t)$ {
3. $p=\text{Select_Highest-priority_Pool}(P)$;
4. $xe=\text{Get_Associated_Expertise_Subensemble}(p)$;
5. $\hat{y}_t=\text{Get_Prediction}(d_t, xe)$;
6. **if** $(c_t == \hat{y}_t)$ {
7. $\text{Map_to_Data-pool}(d_t, p)$; $\text{Update_Metadata}(p)$;
8. **else** $\{p_{new}=\text{Create_New_Data-pool}(d_t)$; $\text{Update_Metadata}(p_{new})$; $P=P \cup p_{new}$;}
9. Update metrics (DS_t^0) , (DS_t^1) using equation (1) and (Se_t) using equation (2);
10. **if** $(|(DS_t^0 - DS_t^1)| > \Psi \text{ and } c_t == 0)$ {
11. $B \sim \text{Poisson}(Se_t)$; Repeat B times training of the ensemble $E=\{e_1, e_2, \dots, e_s\}$;
12. **else if** $(|(DS_t^0 - DS_t^1)| > \Psi \text{ and } c_t == 1)$ {
13. $B \sim \text{Poisson}\left(\frac{DS_t^0}{DS_t^1}\right)$; Repeat B times training of the ensemble $E=\{e_1, e_2, \dots, e_s\}$;
14. **else** $\{B \sim \text{Poisson}(1)$; Repeat B times training of the ensemble $E=\{e_1, e_2, \dots, e_s\}$;

2.3. Experimental framework

This section describes the experimentation framework used to assess the performance of SPECIAL. It provides the details of various datasets used for the experimentation. It also specifies the performance metrics used and the necessary experimental setup for the current study.

2.3.1. Datasets

A variety of datasets are widely used in the study on dynamic data streams as mentioned in [33]. The current study uses six popular binary datasets. Table 1 summarizes the description of these datasets. The used benchmark data streams have variations in the number of samples, the number of attributes, imbalance ratio, and types of drifts. Electricity pricing, agrawal and rotating hyperplane (RotHyperplane) datasets are taken from massive online analysis (MOA) framework [34]. The datasets like weather, SEA and rotating checker board with constant drift rate (RotChBoard-CDR) are available in the repository by [35].

2.3.2. Performance metrics

The presented research focuses on the classification of dynamic streams which may possess concept drift and skewness. Let TP , FP , TN , and FN be the number of true positive, false positive, true negative, and false negative data samples resulting in the binary classification, respectively. For skewed data, the only accuracy $((TP + TN)/(TP + TN + FP + FN))$ cannot justify the performance of the classifier as it gets influenced by the majority samples [7]-[8]. However, G-mean $\left(\sqrt{(TP/(TP + FN)) \cdot (TN/(TN + FP))}\right)$ focuses on the classification of both positive and negative samples as it gives a geometric mean of sensitivity and specificity. Hence, in addition to accuracy, the current study investigates the predictive capabilities of the SPECIAL using the metrics like G-mean, and F1-measure $\left(\frac{2 \cdot (TP/(TP + FN)) \cdot (TP/(TP + FP))}{(TP/(TP + FN)) + (TP/(TP + FP))}\right)$.

Table 1. The summary of datasets used

Dataset	Samples	Classes	Attributes	Positive Samples	Negative Samples	Type of Drifts
Weather (R)	18159	2	8	5698	12461	N/A
Electricity Pricing (R)	45312	2	8	19237	26075	N/A
SEA (S)	50000	2	3	18581	31419	Real, Abrupt
Agrawal (S)	100000	2	9	32656	67344	N/A
RotHyperplane (S)	200000	2	10	99935	100065	Real, Gradual
RotChBoard-CDR(S)	409600	2	2	204758	204842	Real, Gradual

(R)-Real, (S)-Synthetic, N/A-It has a drift, but its type is not available.

2.3.3. Experimental setup

The proposed SPECIAL algorithm builds an ensemble of ensembles. For the same, it blends five seminal ensembles as its base learners. The list of these sub ensembles used to build a learning model of SPECIAL is:

- Hoeffding tree (HT) [36]: It is an incremental decision tree to learn in a streaming environment.
- Dynamic weighted majority (DWM) [20]: It is a chunk-based ensemble that continuously evaluates the weights of its base learners considering the prediction results. It replaces the poor-performing base learner and updates the ensemble.
- ADWIN bagging (BagADWIN) [23]: It is an extension of the online bagging algorithm [31] by integrating the ADWIN (ADaptive sliding WINdowing) algorithm [37] in it.
- ADWIN boosting (BoostADWIN) [23]: It is an extension of the online boosting algorithm [31] by combining the ADWIN (ADaptive sliding WINdowing) algorithm [37].
- Anticipative dynamic adaptation to concept change (ADACC) [21]: It is an incremental ensemble capable of handling recurring concept drifts using Kappa statistics.

The performance of SPECIAL is compared with seven state-of-the-art algorithms used in data stream classification. It is compared with its five sub ensembles as mentioned in the above list. In addition to that it is also compared with the following two classifiers:

- Hierarchical linear four rates (HLFR) [26]: It is an online method for concept drift detection in the dynamic data stream.
- Adaptive chunk-based dynamic weighted majority (ACDWM) [17]: It is a block-based ensemble method for concept drift detection in the streaming environment.

All algorithms are evaluated using the test-then-train approach. We define both data-ageing metric β and sensitivity-ageing metric γ as 0.9. Also, the threshold Ψ used for the detection of class imbalance scenario is set to 0.6. In the testing phase, we search for $K=20$ nearest data-pools.

3. RESULTS AND DISCUSSION

This part evaluates the performance of the proposed SPECIAL algorithm on a variety of datasets. It presents the empirical results of the experimentation. It also describes the statistical analysis of the current work.

3.1. Empirical results

We empirically test the performance of SPECIAL on three evaluation metrics: i) accuracy, ii) G-mean, and iii) F1-measure and compare it with seven state-of-the-art algorithms used to learn in the streaming environment. Tables 2 to 4 present the experimental results of all algorithms on three metrics. All results are given in percentages and the values in the parenthesis indicate the rank of the algorithm when tested on a specific dataset. The minimum value of average rank indicates the best performance of the algorithm.

Table 2 presents the accuracy results of all algorithms. SPECIAL gives the highest accuracy on real datasets weather and electricity. With the least value of the average rank, it shows the overall best performance on accuracy metric. The G-mean results of all algorithms are given in Table 3. Considering the overall average rank value on all datasets, SPECIAL provides the best G-mean results. Table 4 summarizes the F1-measure results of all algorithms. SPECIAL is the best performer with the minimum average ranking on F1-measure. Figure 1 depicts the overall average of ranks of all algorithms. It is noticed that the proposed algorithm SPECIAL with the least value of the overall mean of ranks beats all other state-of-the-art classifiers. The ageing factors used in SPECIAL gives more significance to the recent data that helps to adapt to the latest changes in the incoming data. SPECIAL incorporates the G-mean improvisation strategy with the Poisson (λ) approximated bootstrapping that focuses on the recalls of both positive and negative classes. Also, the employment of the ensemble of locally expertise sub ensembles in SPECIAL alleviates the

limitations of an individual classifier and gives improved classification results. Thus, the SPECIAL provides better results for adaptive online learning in skewed dynamic data streams than other state-of-the-art learners.

Table 2. Accuracy percentage and ranking of all algorithms on all datasets

Dataset	Weather	Electricity	SEA	Agrawal	RotHyperplane	RotChBoard-CDR	Avg. Rank
HT	73.43(5)	80.33 (5)	94.1 (2)	85.32 (5)	84.09 (4)	60.21 (8)	(4.83)
DWM	70.13(7)	78.98 (6)	87.55(5)	87.87 (2)	89.86 (1)	71.95 (6)	(4.5)
OzaBagAdwin	75.01(2)	83.67 (4)	94.58(1)	87.01 (3)	88.01 (2)	85.37 (3)	(2.5)
OzaBoostAdwin	74.39(3)	88.13 (3)	88.17(4)	83.45 (7)	76.57 (7)	94.34 (1)	(4.17)
ADACC	73.57(4)	89.66 (2)	82.83(7)	84.78 (6)	82.77 (6)	80.03 (5)	(5)
HLFR	61.61(8)	60.82 (8)	61.44(8)	60.17 (8)	55.68 (8)	66.57 (7)	(7.83)
ACDWM	71.09(6)	76.48 (7)	87.21(6)	94.35 (1)	83.28 (5)	81.64 (4)	(4.83)
SPECIAL	75.08(1)	90.26 (1)	93.03(3)	86.05 (4)	86.31 (3)	93.1 (2)	(2.33)

Table 3. G-mean percentage and ranking of all algorithms on all datasets

Dataset	Weather	Electricity	SEA	Agrawal	RotHyperplane	RotChBoard-CDR	Avg. Rank
HT	63.16(6)	80.03 (5)	93.51(2)	82.66 (5)	84.08 (4)	60 (8)	(5)
DWM	70.45(2)	77.16 (6)	79.03(6)	85.09 (2)	89.86 (1)	71.95 (6)	(3.83)
OzaBagAdwin	60.66(7)	82.45 (4)	94.3 (1)	84.54 (3)	88.01 (2)	85.37 (3)	(3.33)
OzaBoostAdwin	67.85(4)	87.73 (3)	85.32(5)	81.16 (7)	76.57 (7)	94.34 (1)	(4.5)
ADACC	67.81(5)	89.29 (2)	73.59(7)	81.9 (6)	82.77 (6)	80.03 (5)	(5.17)
HLFR	43.74(8)	59.79 (8)	53.79(8)	46.02 (8)	55.67 (8)	66.57 (7)	(7.83)
ACDWM	72.23(1)	75.74 (7)	85.86(4)	94.86 (1)	83.28 (5)	81.64 (4)	(3.67)
SPECIAL	68.84(3)	89.81 (1)	91.72(3)	83.64 (4)	86.31 (3)	93.1 (2)	(2.67)

Table 4. F1-measure percentage and ranking of all algorithms on all datasets

Dataset	Weather	Electricity	SEA	Agrawal	RotHyperplane	RotChBoard-CDR	Avg. Rank
HT	52.36 (6)	77.15 (5)	91.04(2)	79.04 (5)	84.15 (4)	62.13 (8)	(5)
DWM	59.98 (2)	73.56 (6)	76.69(6)	82.43 (2)	89.86 (1)	71.87 (7)	(4)
OzaBagAdwin	50.45 (7)	79.86 (4)	91.85(1)	81.46 (3)	87.98 (2)	85.38 (3)	(3.33)
OzaBoostAdwin	57.59 (4)	85.92 (3)	81.22(5)	76.86 (7)	76.58 (7)	94.33 (1)	(4.5)
ADACC	57.29 (5)	87.73 (2)	68.33(7)	78.14 (6)	82.74 (6)	79.95 (6)	(5.33)
HLFR	28.45 (8)	54.25 (8)	42.67(8)	31.41 (8)	55.27 (8)	84.49 (4)	(7.33)
ACDWM	62.14 (1)	72.14 (7)	82.53(4)	91.77 (1)	83.28 (5)	80.51 (5)	(3.83)
SPECIAL	58.85 (3)	88.37 (1)	89.21(3)	80.21 (4)	86.28 (3)	93.1 (2)	(2.67)

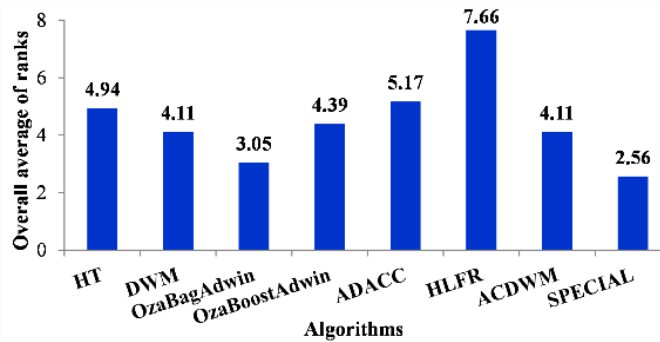


Figure 1. Overall average of ranks of all algorithms

3.2. Statistical results

The empirical results in Tables 2 to 4 show varying performances of different algorithms on different data sets. Hence, to rigorously assess the performances of all studied algorithms we carry out the nonparametric statistical tests as indorsed in [38]. We perform the Iman-Davenport test with a confidence of 95% ($\alpha=0.05$) on all evaluation metrics of above mentioned eight algorithms. It rejects the null hypothesis (H_0 : Ranks of all algorithms are equivalent) for each evaluation metric and infers that at least one of the studied algorithms shows better performance than others on each measure. As the empirical result in Figure 1 claims that the proposed algorithm SPECIAL is the overall best performer, we conduct the pairwise Friedman posthoc test with Finner’s correction [38] to statistically analyse whether SPECIAL is the best

performer among the other seven state-of-the-art classifiers on each metric. Table 5 summarizes the results of the posthoc test with a confidence of 95% ($\alpha=0.05$). The bold-faced values indicate the noteworthy performance improvement of SPECIAL as compared to the other seven classifiers.

Table 5. Results of pairwise Friedman posthoc test ($\alpha=0.05$) to compare SPECIAL on all metrics

Metric	HT	DWM	OzaBagAdwin	OzaBoostAdwin	ADACC	HLFR	ACDWM
Accuracy	0.2	0.2	0.9	0.2	0.2	0	0.2
G-mean	0	0	0	0.1	0	0	0.2
F1-measure	0.1	0.4	0.6	0.1	0.1	0	0.1

4. CONCLUSION

The proposed algorithm SPECIAL provides a novel joint solution to the challenging problem of learning in dynamic data streams with skewness and concept drifts. SPECIAL is a passive drift detection ensemble with the smartness of G-mean maximization and ageing-based adaptive learning. It integrates five seminal ensembles as its base learners. It forms the smart pools of data mapping to the same area of the feature space. These pools point to the local expertise sub ensembles which are likely to give the best classification results in that feature space. SPECIAL follows online learning with a test-then-train approach. It adapts to the dynamicity in data by employing an ageing-based strategy to forget the historic data and to emphasize the recent data. It handles skewness in data streams with the objective of G-mean maximization. The performance of SPECIAL is compared with seven state-of-the-art ensembles on three performance metrics—i) accuracy, ii) G-mean, and iii) F1-measure using a variety of benchmark datasets. Based on the empirical analysis of these metrics the overall average ranking of SPECIAL indicates that it outperforms the other state-of-the-art ensembles in adaptive learning of dynamic data streams. The statistical analysis underpins that the proposed online ensemble model shows noteworthy performance improvement. The current research presents a passive drift detection model of an ensemble of ensembles. The future study will explore the active drift detection model to handle different types of drifts. Also, the presented study assesses the performance of the proposed model empirically and statistically. So, in the future, we would like to focus on the theoretical performance guarantees of the SPECIAL algorithm.




REFERENCES

- [1] N. Seman and N. Atiqah Razmi, "Machine learning-based technique for big data sentiments extraction," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 9, no. 3, pp. 473–479, Sep. 2020, doi: 10.11591/ijai.v9.i3.pp473-479.
- [2] J. Sun, H. Li, H. Fujita, B. Fu, and W. Ai, "Class-imbalanced dynamic financial distress prediction based on Adaboost-SVM ensemble combined with SMOTE and time weighting," *Information Fusion*, vol. 54, pp. 128–144, Feb. 2020, doi: 10.1016/j.inffus.2019.07.006.
- [3] M. A. Rezvi, S. Moontaha, K. A. Trisha, S. T. Cynthia, and S. Ripon, "Data mining approach to analyzing intrusion detection of wireless sensor network," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 516–523, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp516-523.
- [4] H. M. Gomes, J. P. Barddal, F. Enembreck, and A. Bifet, "A survey on ensemble learning for data stream classification," *ACM Computing Surveys*, vol. 50, no. 2, pp. 1–36, Jun. 2017, doi: 10.1145/3054925.
- [5] L. Deshpande and M. N. Rao, "Concept drift identification using classifier ensemble approach," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 1, pp. 19–25, Feb. 2018, doi: 10.11591/ijece.v8i1.pp19-25.
- [6] E. Alothali, H. Alashwal, and S. Harous, "Data stream mining techniques: a review," *TELKOMNIKA*, vol. 17, no. 2, pp. 728–737, 2019, doi: 10.12928/TELKOMNIKA.v17i2.11752.
- [7] I. Khamassi, M. Sayed-Mouchaweh, M. Hammami, and K. Ghédira, "Discussion and review on evolving data streams and concept drift adapting," *Evolving Systems*, vol. 9, no. 1, pp. 1–23, Mar. 2018, doi: 10.1007/s12530-016-9168-2.
- [8] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue, and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Systems with Applications*, vol. 73, pp. 220–239, May 2017, doi: 10.1016/j.eswa.2016.12.035.
- [9] S. Ancy and D. Paulraj, "Handling imbalanced data with concept drift by applying dynamic sampling and ensemble classification model," *Computer Communications*, vol. 153, pp. 553–560, 2020, doi: 10.1016/j.comcom.2020.01.061.
- [10] S. Ren *et al.*, "Selection-based resampling ensemble algorithm for nonstationary imbalanced stream data learning," *Knowledge-Based Systems*, vol. 163, pp. 705–722, Jan. 2019, doi: 10.1016/j.knosys.2018.09.032.
- [11] B. Krawczyk, L. L. Minku, J. Gama, J. Stefanowski, and M. Woźniak, "Ensemble learning for data stream analysis: A survey," *Information Fusion*, vol. 37, pp. 132–156, Sep. 2017, doi: 10.1016/j.inffus.2017.02.004.
- [12] H. Ghomeshi, M. M. Gaber, and Y. Kovalchuk, "Ensemble dynamics in non-stationary data stream classification," in *Learning from Data Streams in Evolving Environments, Studies in Big Data*, vol. 41, M. Sayed-Mouchaweh, Ed. Springer, Cham, pp. 123–153, 2019.
- [13] R. S. M. de Barros and S. G. T. de C. Santos, "An overview and comprehensive comparison of ensembles for concept drift," *Information Fusion*, vol. 52, pp. 213–244, Dec. 2019, doi: 10.1016/j.inffus.2019.03.006.
- [14] H. Wang, W. Fan, P. S. Yu, and J. Han, "Mining concept-drifting data streams using ensemble classifiers," in *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining-KDD '03*, 2003, Art. no. 226, doi: 10.1145/956750.956778.
- [15] D. Brzezinski and J. Stefanowski, "Reacting to different types of concept drift: The accuracy updated ensemble algorithm," *IEEE*





- Transactions on Neural Networks and Learning Systems*, vol. 25, no. 1, pp. 81–94, Jan. 2014, doi: 10.1109/TNNLS.2013.2251352.
- [16] Y. Lu, Y. Cheung, and Y. Y. Tang, “Dynamic weighted majority for incremental learning of imbalanced data streams with concept drift,” in *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI-17)*, 2017, pp. 2393–2399, doi: 10.24963/ijcai.2017/333.
- [17] Y. Lu, Y.-M. Cheung, and Y. Yan Tang, “Adaptive chunk-based dynamic weighted majority for imbalanced data streams with concept drift,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 8, pp. 2764–2778, Aug. 2020, doi: 10.1109/TNNLS.2019.2951814.
- [18] A. Bifet, G. Holmes, B. Pfahringer, R. Kirkby, and R. Gavaldà, “New ensemble methods for evolving data streams,” in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining-KDD '09*, 2009, Art. no. 139, doi: 10.1145/1557019.1557041.
- [19] D. Brzezinski and J. Stefanowski, “Combining block-based and online methods in learning ensembles from concept drifting data streams,” *Information Sciences*, vol. 265, pp. 50–67, May 2014, doi: 10.1016/j.ins.2013.12.011.
- [20] J. Kolter and M. Maloof, “Dynamic Weighted Majority: An ensemble method for drifting concepts,” *Journal of Machine Learning Research*, vol. 8, pp. 2755–2790, 2007.
- [21] G. Jaber, A. Cornuéjols, and P. Tarroux, “A new on-line learning method for coping with recurring concepts: The ADACC system,” in *Neural Information Processing (ICONIP 2013), Lecture Notes in Computer Science*, vol. 8227, M. Lee, A. Hirose, Z. Hou, and R. M. Kil, Eds. Daegu, Korea (Republic of): Springer, Berlin, Heidelberg, pp. 595–604, 2013.
- [22] P. R. L. Almeida, L. S. Oliveira, A. S. Britto, and R. Sabourin, “Adapting dynamic classifier selection for concept drift,” *Expert Systems with Applications*, vol. 104, pp. 67–85, Aug. 2018, doi: 10.1016/j.eswa.2018.03.021.
- [23] A. Bifet, G. Holmes, B. Pfahringer, and R. Gavaldà, “Improving adaptive bagging methods for evolving data streams,” in *Advances in Machine Learning, ACML 2009, Lecture Notes in Computer Science*, vol. 5828, Z. Zhou and T. Washio, Eds. Springer, Berlin, Heidelberg, pp. 23–37, 2009.
- [24] R. S. M. Barros and S. G. T. C. Santos, “A large-scale comparison of concept drift detectors,” *Information Sciences*, vol. 451–452, pp. 348–370, Jul. 2018, doi: 10.1016/j.ins.2018.04.014.
- [25] D. R. de L. Cabral and R. S. M. de Barros, “Concept drift detection based on Fisher’s Exact test,” *Information Sciences*, vol. 442–443, pp. 220–234, May 2018, doi: 10.1016/j.ins.2018.02.054.
- [26] S. Yu, Z. Abraham, H. Wang, M. Shah, Y. Wei, and J. C. Príncipe, “Concept drift detection and adaptation with hierarchical hypothesis testing,” *Journal of the Franklin Institute*, vol. 356, no. 5, pp. 3187–3215, Mar. 2019, doi: 10.1016/j.jfranklin.2019.01.043.
- [27] H. Ali, M. N. Mohd Salleh, R. Saedudin, K. Hussain, and M. F. Mushtaq, “Imbalance class problems in data mining: a review,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 14, no. 3, p. 1552, Jun. 2019, doi: 10.11591/ijeecs.v14.i3.pp1552-1563.
- [28] W. Zhang and J. Wang, “A hybrid learning framework for imbalanced stream classification,” in *2017 IEEE International Congress on Big Data (BigData Congress)*, Jun. 2017, pp. 480–487, doi: 10.1109/BigDataCongress.2017.70.
- [29] Y. Sun, “A novel ensemble classification for data streams with class imbalance and concept drift,” *International Journal of Performability Engineering*, vol. 13, no. 6, pp. 945–955, 2017, doi: 10.23940/ijpe.17.06.p15.945955.
- [30] V. Losing, B. Hammer, and H. Wersing, “Incremental on-line learning: A review and comparison of state of the art algorithms,” *Neurocomputing*, vol. 275, pp. 1261–1274, Jan. 2018, doi: 10.1016/j.neucom.2017.06.084.
- [31] N. C. Oza, “Online ensemble learning,” University of California, Berkeley, 2001.
- [32] R. Kulkarni, S. Revathy, and S. Patil, “A novel approach to maximize G-mean in nonstationary data with recurrent imbalance shifts,” *The International Arab Journal of Information Technology*, vol. 18, no. 1, pp. 103–113, Dec. 2020, doi: 10.34028/iajit/18/1/12.
- [33] J. Lu, A. Liu, F. Dong, F. Gu, J. Gama, and G. Zhang, “Learning under concept drift: A review,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 12, pp. 1–1, Dec. 2018, doi: 10.1109/TKDE.2018.2876857.
- [34] A. Bifet, G. Holmes, R. Kirkby, and B. Pfahringer, “MOA: Massive online analysis,” *Journal of Machine Learning Research*, vol. 11, pp. 1601–1604, 2010.
- [35] R. Elwell and R. Polikar, “Incremental learning of concept drift in nonstationary environments,” *IEEE Transactions on Neural Networks*, vol. 22, no. 10, pp. 1517–1531, Oct. 2011, doi: 10.1109/TNN.2011.2160459.
- [36] P. Domingos and G. Hulten, “Mining high-speed data streams,” in *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '00*, 2000, pp. 71–80, doi: 10.1145/347090.347107.
- [37] A. Bifet and R. Gavaldà, “Learning from time-changing data with adaptive windowing,” in *Proceedings of the 2007 SIAM International Conference on Data Mining*, Apr. 2007, pp. 443–448, doi: 10.1137/1.9781611972771.42.
- [38] S. García, A. Fernández, J. Luengo, and F. Herrera, “Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power,” *Information Sciences*, vol. 180, no. 10, pp. 2044–2064, May 2010, doi: 10.1016/j.ins.2009.12.010.

BIOGRAPHIES OF AUTHORS







Radhika Vikas Kulkarni    received the M.Tech. degree in Computer Science and Technology from the Shivaji University, Kolhapur, India in 2011. She is currently pursuing a Ph.D. degree with the Department of Computer Science and Engineering, Sathyabama Institute of Science and Technology, Chennai India. She is working as an Assistant Professor in the Department of Information Technology, Pune Institute of Computer Technology, Pune, India. Her current research interests include Machine Learning, Data Analytics, and Big Data. Email: radhikavikaskulkarni@gmail.com.



S. Revathy     is presently working as an Associate Professor in the Department of Information Technology, Sathyabama Institute of Science and Technology, Chennai India. Her research interest includes Machine Learning, Data Analytics, and Big Data. She has published over 20 papers in refereed journals. Email: revathy.it@sathyabama.ac.in.



Suhas Haribhau Patil     received a Ph.D. degree in Computer Science and Engineering from Bharati Vidyapeeth Deemed University, Pune, India in 2009. He is currently working as a Professor in the Department of Computer Science and Engineering, Bharati Vidyapeeth Deemed University College of Engineering, Pune, India. His research area includes Machine Learning, Expert Systems, Computer Networks, Operating Systems, System Software. He has published over 65 papers in international journals, 36 papers in international conferences, and 42 papers in national conferences. Email: shpatil@bvucoep.edu.in.

Wiki sense bag creation using multilingual word sense disambiguation

Shreya Patankar¹, Madhura Phadke², Satish Devane³

^{1,2}Department of Computer Engineering, Datta Meghe College of Engineering, Navi Mumbai, India

³Department of Information Technology, Datta Meghe College of Engineering, Navi Mumbai, India

Article Info

Article history:

Received Jul 1, 2021

Revised Dec 22, 2021

Accepted Jan 2, 2022

Keywords:

Multilingual

Natural language processing

Word sense disambiguation

ABSTRACT

Performance of word sense disambiguation (WSD) is one of the challenging tasks in the area of natural language processing (NLP). Generation of sense annotated corpus for multilingual word sense disambiguation is out of reach for most languages even if resources are available. In this paper we propose an unsupervised method using word and sense embedding or improving the performance of these systems using untagged. Corpora and create two bags namely ontological bag and wiki sense bag to generate the senses with highest similarity. Wiki sense bag provides external knowledge to the system required to boost the disambiguation accuracy. We explore Word2Vec model to generate the sense

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Shreya Patankar

Department of Computer Engineering, Datta Meghe College of Engineering

Sector-3, Airoli, Opp Khandoba Temple Sri Sadguru Vanamrao Pai Marg, Navi Mumbai, Maharashtra, India

Email: snp.cm.dmce@gmail.com

1. INTRODUCTION

Increasing demands by the user to access text data in various languages opens up the doors of multilingual natural language processing (NLP) and word sense disambiguation (WSD) has proved to be a key step in performance improvement of many NLP systems. The accuracy of word sense disambiguation systems is far from being satisfactory and multilingual WSD has not achieved satisfactory results due to insufficient resource availability [1]. The availability of multilingual dictionaries has enhanced sense disambiguation using multilingual content which depicts the need for multilingual WSD [2]. It also opens up a different way of approaching multilingual WSD by making use of BabelNet, a wide ontological structure exploring semantic knowledge. This is the motivation for working on multilingual word sense disambiguation by exploring the available resources.

Relying only on multilingual knowledge-based system may hamper the growth of WSD systems and though multilingual dictionaries provide wide coverage exploring the interconnected ontology structure, various issues still remain to be seen such as proper nouns are not part of the dictionary and correlation between most frequent words and rare contextual words lack dictionary coverage. External knowledge in terms of raw text is needed which is provided using word and sense embedding [3]. Our research makes use of word and sense embeddings to create a semantic word cloud by designing a wiki bag in addition to the sense bag. Wiki bag is designed using Wikipedia as it is the largest encyclopedia which covers most of the database essential for disambiguation. The paper is organized being as: section 2 presents the literature review which highlights the research work of various researchers, section 3 describes the proposed

methodology used which includes working with multilingual input, multilingual dictionary BabelNet and the working of WSD engine. Section 4 focuses on results and discussions and section 5 sums up with conclusion.

2. LITERATURE REVIEW

Word2Vec model [4]–[13] provides an efficient tool for estimating vector model using the corpus. A sense bag was created [14] making use of dictionary resources such as synset members, example sentences, hypernymy and hyponymy subsets. A survey was presented on WSD [15] highlighting the motivation for solving the ambiguity of words and providing description of the task. The concept of Word sense disambiguation in multilingual setting [16] introduces by making use of large encyclopedic ontological network BabelNet. Precision achieved was 54.3% when tested on SemEval 2010 dataset. In 2013, Aziz and Specia [17] discusses expressing meanings in terms of paraphrases.

The role of WSD for multilingual scenario of NLP text was surveyed using English-Spanish languages [18]. WSD in multilingual machine translation (MT) is based on the concept that resource full language helps a resource low language by projecting parameters like sense distributions, and corpus co-occurrences [19]. The accuracy observed was 75% for three languages with domain specific corpus. WSD in NLP applications is also discussed [20]. Cross-lingual WSD systems was discussed [21], and evaluated on SemEval 2010 task. Machine translation is one of the important applications of WSD and is discussed [22], [23]. A survey of text classification of Kurdish language is beautifully presented [24]–[27] where they applied stemmer algorithm to find the stem to perform classification. WSD network approach, sentiment analysis and survey is explored [28]–[31]. It is observed that not much work is reported on WSD in multilingual setting to the best of our knowledge and it needs to be explored using various state of the art WSD methods.

3. PROPOSED METHODOLOGY

The proposed methodology is presented in the Figure 1 and we present the concept of representing multilingual input data in section 3.1. It includes accepting multilingual input which will benefit the engine. External knowledge is also provided to the system using sense embeddings.

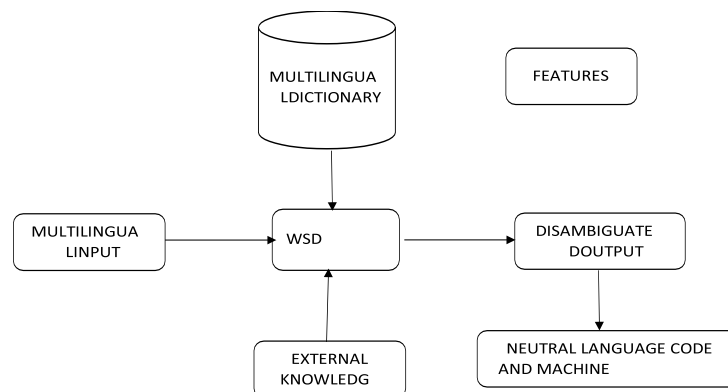


Figure 1. Proposed multilingual word sense disambiguation (WSD) framework

3.1. Multilingual input

We consider here input from various languages like German and French and make use of Babel Net multilingual dictionary described in section 3.2. This is done to explore various languages and taking help from other languages improves the system accuracy. Ambiguous word in one language may not be ambiguous in other language and this will benefit the system engine for improving the accuracy.

3.2. BabelNet

TheBabelNet is a huge multilingual ontological network incorporating lexical semantic and syntactic knowledge from various languages [1]. It represents a labelled graph specifying semantic relations between various nodes and edges. It combines the knowledge of various language WordNet and largest multilingual encyclopedia. Section 3.3 represents the working of WSD with thealgorithm for the same.

3.3. Word sense disambiguation (WSD) engine

WSD engine takes the multilingual input by exploring various languages altogether at the same time. It combines the translations of target word and other context words to produce more accurate sense predictions. Sense disambiguation begins by gathering the data required for disambiguation where the different senses of the ambiguous word are collected in S represented as synonymset from the BabelNet. Context words are collected in Ctx and the algorithm then proceeds by picking up the multilingual translations of the ambiguous and clue words stored in Tx and Ty respectively. Translations are considered in French and German languages as foreign languages are explored. The algorithm iterates through each synset $s \in S$ to collect the translations of each of its senses [7].

Algorithm also iterates through each context word $c_i \in Ctx$ to collect the translations in Ty in sense-specific German and French translations. Element t_i is selected from Tx and element t_j is selected from Ty and a multilingual context μ' is created by combining t_i and t_j with the Ctx. The variable μ' is used to build a graph $G = \{V, E\}$ by computing the paths in BabelNet which connects the synsets of t_i with those of other words in μ' as shown in Figure 2. By selecting at each step, a different element from T, a new graph is created where different sets of Babel synsets get activated by the context words in Ctx. The result of this procedure is a subgraph of BabelNet containing the senses of the words in the context and all edges and intermediate senses found in BabelNet along all paths connecting them. Figure 2 shows the disambiguation graph created to disambiguate the English language target word 'bank'. In the graph, some of the possible senses of this word are activated including the correct sense ($bank^2_{ENGLISH}$) but also related yet incorrect one is activated ($bank^9_{ENGLISH}$).

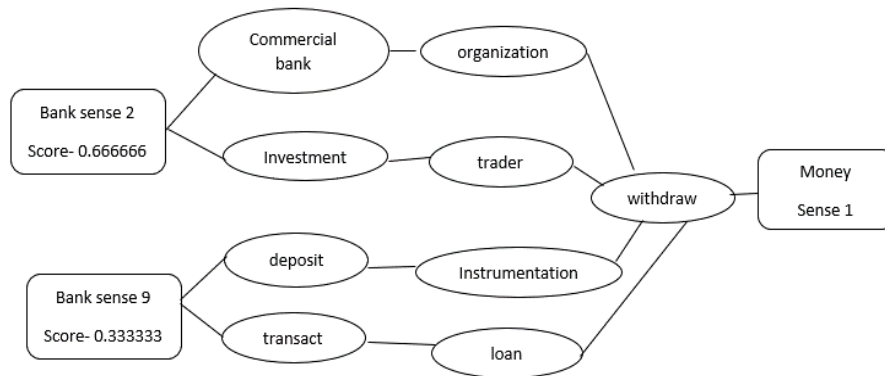


Figure 2. Disambiguation graph for English language

3.4. Scoring distribution

Scoring distribution is calculated using the Inverse path length sum measure. It scores each sense by summing over the inverse length of all paths which connect it to other senses in the graph. It is very useful for sense disambiguation and improves the accuracy.

$$score_j = \frac{1}{e^{length(p)-1}} \tag{1}$$

Where $paths(s_j)$ is the set of simple paths connecting s_j to the senses of other context words. Length (p) is the number of edges in the path p and each path is scored with the exponential inverse decay of the path length. Scores are calculated and stored in $\Delta score$ and in the final step; cosine distance similarity measure is calculated to find the maximum score which determines the closeness between the ambiguous word and the context words. The cosine distance formula is presented in (2):

$$Cos (S, SC (T)) = \frac{\sum_{i=1}^n S * SC(T)}{\sqrt{\sum_{i=0}^n S} * \sqrt{\sum_{i=1}^n SC(T)}} \tag{2}$$

where S is vector representing the score of ambiguous words, SC (T) is vector representing the score of context words. Global score consists of selecting the highest score represented and as a result of execution of algorithm; the scoring distribution which is maximum is returned to select the best disambiguation sense. Sections 3.5 and 3.6 represents the use of deep learning tools to represent the dictionary framework in

numeric representation. Table 1 represents the scoring distribution using the above formula for the two senses of bank namely building sense $\text{bank}^9_{\text{ENGLISH}}$ and financial institution $\text{bank}^2_{\text{ENGLISH}}$.

Table 1. Scoring distribution

Language	$\text{bank}^2_{\text{ENGLISH}}$	$\text{bank}^9_{\text{ENGLISH}}$
bankEN	0.6666666666	0.3333333333
bankGERMAN	0.3333333333	0
banqueFRENCH	0.4444444444	0

3.5. Synset dictionary framework

Our study explores the ontology of each sense definition from the dictionary namely hypernym, hyponym, holonymy, and gloss. as synset members alone are not sufficient for identifying the correct sense. Some of synsets have a very small number of synset members and the other reason is to bring down topic drift which may have occurred because of polysemous synset members. It is also observed that adding gloss of hypernym/hyponym gives better performance compared to synset members of hypernym/hyponym [5].

3.6. Word and sense embedding

There is a need to bring the clue words and ambiguous words together which is done using word embeddings. It represents embedding continuous vector space with lesser dimensions and word embedding are trained using word2Vec tool [4]. The training proceeds by presenting different context-target words pair from the corpus thus preparing an ensemble model for all the ambiguous words in the vocabulary as presented in Figure 3. The corpus ensemble model of vectors represents the closeness of the context-target pair for specific sense and to the best of our knowledge, this is the first of the kind attempt to generate sense specific word vector model which represents close proximity between the context words and ambiguous word in the vector space. Section 3.6 represents our contribution of sense bag creation.

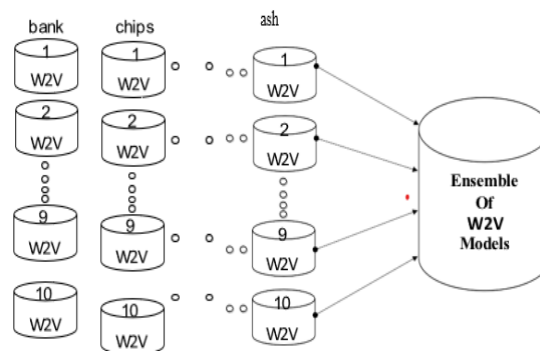


Figure 3. Corpus based ensemble vector model

3.7. Sense bag creation

Sense specific vector model is represented by extracting features from the lexical ontology as well as encyclopedic knowledge. Words are represented by retrieving the context words from the ontological structure of each sense such as synset members, gloss or example sentence, relations such as hypernym or hyponym. Word2Vec model is a layered neural network structure that processes the text by converting them into vectors; a numerical form which brings related words together. The input to the neural network is window of words, hidden layer comprises of weight matrix and output is vector representation of words. Wiki sense bag is also created which is vector representation of Wikipedia of ambiguous words. This is done so as to provide additional world knowledge to the Word sense disambiguation engine as Wiki sense bag covers maximum vocabulary needed to bring context-target pairs closure in the vector space. Wiki bag creation is represented in Figure 4 and Similarity measure is calculated in section 3.8.

3.8. Similarity measure

The similarity measure is calculated by considering the cosine similarity between the word representation of context vector and sense bag representation. It helps to generate a similarity score which

helps in the disambiguation process. Cosine similarity measure has proven to be more useful in the word sense disambiguation process.

$$\text{Cos}(\text{vec}(w), \text{vec}(SB)) = \frac{\sum_{i=1}^n w * SB}{\sqrt{\sum_{i=1}^n w^2} * \sqrt{\sum_{i=1}^n SB^2}} \quad (3)$$

Where $\text{vec}(w)$ is the word embedding for word w , SB represents the sense bag and $\text{vec}(SB)$ is the sense embedding representing the combined score of ontology bag and the wiki sense bag. Sense disambiguation (SD) is performed by summing the scores of (1)-(3) which represents multilingual Word sense disambiguation similarity score, word embedding and sense embedding scores of ontology bag and wiki sense bag to boost the disambiguation accuracy. The output of the WSD engine results in disambiguated sense which is converted into neutral language code to be used for MT. Section 3.9 represents the formation of neutral language code.

-3.1438863e-03	2.5703609e-03	2.3864100e-03	-1.6323227e-03
6.4692349e-04	2.6351425e-03	-4.3628053e-03	4.3827966e-03
-4.7720312e-03	-3.4716928e-03	3.4759669e-03	4.3763947e-03
3.2847153e-03	2.3355209e-03	2.8738815e-03	-2.2687481e-03
-4.8421333e-03	3.0184705e-03	-2.1880846e-03	1.8512266e-03
1.6703347e-03	-6.8748498e-04	-6.2847714e-04	-3.0067556e-03
3.0463885e-03	-3.5307638e-03	2.7850315e-03	3.9292048e-04
-2.6362720e-03	-3.6856441e-03	2.7092642e-04	1.5298135e-04
-4.8553180e-03	3.8366476e-03	-2.4513335e-03	3.6468427e-03
2.3314022e-03	1.7899536e-03	-4.3625557e-03	3.3640813e-03
-1.8001328e-03	1.4276117e-03	-1.1264355e-03	-4.4314810e-03
4.2599617e-03	1.2551763e-03	3.8926408e-03	2.4237178e-04
-4.3531498e-03	2.6536058e-03	-3.3246232e-03	4.0993919e-03

Figure 4. Wiki sense bag creation

3.9. Neutral language code

Words after disambiguation are converted into unique representation termed as neutral language code is formed using binary combination of 30-bit unique code where each bit represents significant information about the disambiguated polysemy noun represented in Table 2. Neutral language code is unique in the sense that it covers all the information other than sense identification and parts of speech. Results are presented in the next section.

Noun	Code
Bank	0001000110101100101011110011xx
	000-parts of speech
	0001-unique identification
	1101-Type of noun
	011-number
	001-gender
	11110000-tenses
	xx - reserved bits

4. RESULTS AND DISCUSSION

Word sense disambiguation framework comprises of multilingual input and evaluation is performed on a manually created corpus for English language consisting of 25 polysemous nouns, for English lexical sample task. Experiments were performed with 5000 instances out of which 70% was used for training and 30% for testing. Test instances were also collected from various search engines books and the accuracy observed for multilingual word sense disambiguation is 40% as compared to 25% observed for monolingual word sense disambiguation. Table 3 presents comparison of the two systems and results are presented for 10 polysemy nouns. For simplicity we consider two senses each for polysemy nouns. The system was tested using multilingual approach and observed accuracy was improved by 15 %. The overall accuracy observed was 40%. Observations and findings are presented in section 4.1.

Table 3. Monolingual versus multilingual word sense disambiguation

English	Sense	Accuracy in % for Monolingual word sense disambiguation	Accuracy in % for Multilingual word sense disambiguation
Chips	Silicon chip	25	45
	Wafers	24	40
Table	Furniture	30	35
	Row/column	32	43
Bat	Mammal	25	45
	Sports	27	45
Bank	Finance	32	45
	Riverbank	32	47
Tank	Military tank	25	44
Plant	Industry plant	35	44
	Tree	35	47
Stock	Capital	30	43
	Storage	29	40
Palm	Hand	28	44
	Name of tree	26	43
Account	Bank account	35	43
	Write up	35	45

4.1. Observations and findings

The problem of similar score faced in monolingual approach was eliminated using multilingual word sense disambiguation. Observed accuracy is 40% which is far less than the baseline accuracy observed for most frequent sense. It is also observed that proper nouns like Madhura, Shreyas from our instances were not part of the dictionary definitions which failed to generate proper scores. Also, dictionary definition being short lacks strong clues which fail the disambiguation accuracy.

Features of BabelNet senses are extracted from the synset (S), gloss of synset member (G), hypernymy (H), hyponymy (HP), synset gloss of hypernymy-hyponymy relation (HG), holonymy (HO) and gloss of holonymy (HOG). We tested these features on 2000 instances and results are represented by taking the maximum of the global scores received represented in Table 4. It is observed from the Table 4 that combining all the features of BabelNet senses together gives us an improved accuracy of 50%. It shows that combining all the features together yields significant improvement in the disambiguation process. Multilingual approach implements graph-based disambiguation and we observed that many clue words from the context were not in close proximity with the ambiguous words. Many words closely related are at distance from one another and this being one of the important findings results in less score which affects the disambiguation process. Words in similar context needs to come close for improve the accuracy. Word and sense embeddings are presented in section 4.2.

Table 4. Synset dictionary framework

Features	Global score	Accuracy in %
S	0.0869	24
S+G	0.1923	27
S+G+H	0.1666	33
S+G+H+HP	0.0588	38
S+G+H+HP+HG	0.3333	42
S+G+H+HP+HG+HO	0.0526	47
S+G+H+HP+HG+HO+HOG	0.5238	50

4.2. Word and sense embeddings

We evaluated our approach for testing the system on word and sense embeddings separately and then combining the two results for disambiguation process. Word embeddings are taken from the raw corpus and make use of gensim word2Vec model for our study. We compared our work with other state of the art methods in terms of precision and recall represented in Table 5. It is observed that our approach with word embeddings came close to baseline accuracy and unsupervised most frequent sense (UMFS) approach. Our approach gives a feasible way to extract predominant senses in an unsupervised setup. Our approach is domain independent so that it can be easily adapted to a domain specific corpus. To get the domain specific word and sense embeddings, we simply have to run the word2vec program on the domain specific corpus. Also, our approach is language independent and portable across mobile devices as smart phones being the most preferred mode of communication. Conclusion is summed up in the next section.

Table 5. Performance comparison of sense embeddings with other methods

System	Precision	Recall
Most frequent sense baseline	0.552	0.552
Lesk algorithm	0.097	0.053
Adapted lesk	0.240	0.234
UMFS (Bhingardive)	0.433	0.432
Multilingual WSD with word and sense embeddings	0.489	0.489

5. CONCLUSION

In this research work, we presented multilingual approach to word sense disambiguation and used BabelNet as multilingual lexicon for disambiguation. Multilingual word sense disambiguation exploits graph-based method to collect evidences from translations in various languages. We also explored the synset dictionary framework by making use of features from BabelNet dictionary. We created separate model for each ambiguous word sense and made an ensemble of the word2Vec models for disambiguation purpose using word embeddings. Our research contribution includes sense bag creation by using the ontological features of the BabelNet lexicon and encyclopedic knowledge from Wikipedia. It is observed that multilingual word sense disambiguation achieved good results in comparison to monolingual system as additional knowledge from various languages help to boost the accuracy. The results also show that our method of multilingual word sense disambiguation with sense embedding improves the accuracy of the system. The approach is open to explore other languages. We will explore our approach for other parts of speech and other languages especially Indian languages like Marathi, Hindi, and Bangla. We plan in the near future to create generalized sense representation for multiple languages so as to provide a general framework for knowledge rich multilingual word sense disambiguation.




REFERENCES

- [1] R. Navigli and S. P. Ponzetto, "Joining forces pays off: multilingual joint word sense disambiguation," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, Jul. 2012, pp. 1399–1410.
- [2] D. O, S. Kwon, K. Kim, and Y. Ko, "Word sense disambiguation based on word similarity calculation using word vector representation from a knowledge-based graph," in *Proceedings of the 27th International Conference on Computational Linguistics*, Aug. 2018, pp. 2704–2714.
- [3] Y. Wang, M. Wang, and H. Fujita, "Word Sense Disambiguation: A comprehensive knowledge exploitation framework," *Knowledge-Based Systems*, vol. 190, p. 105030, Feb. 2020, doi: 10.1016/j.knosys.2019.105030.
- [4] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, Jan. 2013.
- [5] S. Bhingardive, S. Shaikh, and P. Bhattacharyya, "Neighbors help: bilingual unsupervised WSD using context," 2013, vol. 2, pp. 538–542.
- [6] R. Fu, J. Guo, B. Qin, W. Che, H. Wang, and T. Liu, "Learning semantic hierarchies: a continuous vector space approach," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 23, no. 3, pp. 461–471, Mar. 2015, doi: 10.1109/TASLP.2014.2377580.
- [7] K. Taghipour and H. T. Ng, "Semi-supervised word sense disambiguation using word embeddings in general and specific domains," in *NAACL HLT 2015 - 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, 2015, pp. 314–323, doi: 10.3115/v1/n15-1035.
- [8] X. Chen, Z. Liu, and M. Sun, "A unified model for word sense representation and disambiguation," in *EMNLP 2014 - 2014 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2014, pp. 1025–1035, doi: 10.3115/v1/d14-1110.
- [9] I. Iacobacci, M. T. Pilehvar, and R. Navigli, "Embeddings for word sense disambiguation: an evaluation study," in *54th Annual Meeting of the Association for Computational Linguistics, ACL 2016 - Long Papers*, 2016, vol. 2, pp. 897–907, doi: 10.18653/v1/p16-1085.
- [10] A. Trask, P. Michalak, and J. Liu, "sense2vec - A fast and accurate method for word sense disambiguation in neural word embeddings," 2015.
- [11] H. Sugawara, H. Takamura, R. Sasano, and M. Okumura, "Context representation with word embeddings for WSD," in *Communications in Computer and Information Science*, vol. 593, Springer Singapore, 2016, pp. 108–119.
- [12] L. Specia, M. Stevenson, and M. Nunes, "Learning expressive models for word sense disambiguation," 2007.
- [13] S. Thater, H. Fürstenau, and M. Pinkal, "Word meaning in context: a simple and effective vector model," *AFNLP*. pp. 1134–1143, 2011.
- [14] S. Bhingardive, D. Singh, M. V Rudra, H. Redkar, and P. Bhattacharyya, "Unsupervised most frequent sense detection using word embeddings," in *NAACL HLT 2015 - 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference*, 2015, pp. 1238–1243, doi: 10.3115/v1/n15-1132.
- [15] R. Navigli, "Word sense disambiguation: a survey," *ACM Computing Surveys*, vol. 41, no. 2, pp. 1–69, Feb. 2009, doi: 10.1145/1459352.1459355.
- [16] R. Navigli and S. Ponzetto, "Multilingual WSD with just a few lines of code: the BabelNet API," 2012, pp. 67–72.
- [17] W. Aziz and L. Specia, "Multilingual WSD-like constraints for paraphrase extraction," 2013, pp. 202–211.
- [18] A. Montoyo, R. Romero, S. Vázquez, C. Calle, and S. Soler, "The role of WSD for multilingual natural language applications," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 2448, Springer Berlin Heidelberg, 2002, pp. 41–48.




- [19] M. M. Khapra, S. Shah, P. Kedia, and P. Bhattacharyya, "Projecting parameters for multilingual word sense disambiguation," in *EMNLP 2009 - Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing: A Meeting of SIGDAT, a Special Interest Group of ACL, Held in Conjunction with ACL-IJCNLP 2009*, 2009, pp. 459–467, doi: 10.3115/1699510.1699570.
- [20] P. Resnik, "WSD in NLP applications," Springer Netherlands, 2007, pp. 299–337.
- [21] C. Silberer and S. Ponzetto, "UHD: Cross-lingual word sense disambiguation using multilingual co-occurrence graphs," pp. 134–137, 2010.
- [22] D. Vickrey, L. Biewald, M. Teyssier, and D. Koller, "Word-sense disambiguation for machine translation," in *HLT/EMNLP 2005 - Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, 2005, pp. 771–778, doi: 10.3115/1220575.1220672.
- [23] R. Marvin and P. Koehn, "Exploring word sense disambiguation abilities of neural machine translation systems," *AMTA*, vol. 1, pp. 125–131, Mar. 2018, doi: 10.2/JQUERY.MIN.JS.
- [24] T. A. Rashid, A. M. Mustafa, and A. M. Saeed, "A Robust Categorization System for Kurdish Sorani Text Documents," *Information Technology Journal*, vol. 16, no. 1, pp. 27–34, Dec. 2016, doi: 10.3923/itj.2017.27.34.
- [25] A. M. Saeed, T. A. Rashid, A. M. Mustafa, R. A. A.-R. Agha, A. S. Shamsaldin, and N. K. Al-Salihi, "An evaluation of Reber stemmer with longest match stemmer technique in Kurdish Sorani text classification," *Iran Journal of Computer Science*, vol. 1, no. 2, pp. 99–107, Jan. 2018, doi: 10.1007/s42044-018-0007-4.
- [26] A. M. Mustafa and T. A. Rashid, "Kurdish stemmer pre-processing steps for improving information retrieval," *Journal of Information Science*, vol. 44, no. 1, pp. 15–27, Jan. 2018, doi: 10.1177/0165551516683617.
- [27] T. A. Rashid, A. M. Mustafa, and A. M. Saeed, "Automatic kurdish text classification using KDC 4007 dataset," in *Lecture Notes on Data Engineering and Communications Technologies*, vol. 6, Springer International Publishing, 2018, pp. 187–198.
- [28] Y. Choi, J. Wiebe, and R. Mihalcea, "Coarse-grained +/-effect word sense disambiguation for implicit sentiment analysis," *IEEE Transactions on Affective Computing*, vol. 8, no. 4, pp. 471–479, Oct. 2017, doi: 10.1109/TAFFC.2017.2734085.
- [29] R. R. Karwa and M. B. Chandak, "Word sense disambiguation: hybrid approach with annotation up to certain level – a review," *International Journal of Engineering Trends and Technology*, vol. 18, no. 7, pp. 328–330, Dec. 2014, doi: 10.14445/22315381/ijett-v18p267.
- [30] E. A. Corrêa, A. A. Lopes, and D. R. Amancio, "Word sense disambiguation: a complex network approach," *Information Sciences*, vol. 442–443, pp. 103–113, May 2018, doi: 10.1016/j.ins.2018.02.047.
- [31] A. H. Aliwy and H. A. Taher, "Word sense disambiguation: survey study," *Journal of Computer Science*, vol. 15, no. 7, pp. 1004–1011, Jul. 2019, doi: 10.3844/jcssp.2019.1004.1011.

BIOGRAPHIES OF AUTHORS






Shreya Nandkumar Patankar    is pursuing Ph.D. from Mumbai University in the area of Natural language processing and is currently working as Assistant Professor at Datta Meghe College of Engineering, Airoli, Navi Mumbai, India with 19 years of teaching experience. Her research is mainly focused on word sense disambiguation, Natural language processing, Artificial Intelligence and machine learning. She is good in programming languages like java and subjects such as algorithms, operating systems, data structures and database. She has published 10 papers in international conference, 7 international journals, 4 in national conferences and few papers are indexed in Scopus database and Elsevier. One of her paper featured as top 10 downloaded papers in SSRN digital library. One Masters candidate is actively involved under her guidance. She can be contacted at email: shreya.patankar@dmce.ac.in.



Madhura Mandar Phadke    is pursuing Ph.D. from Mumbai University in the area of Natural language processing and is currently working as Assistant Professor at Datta Meghe College of Engineering, Airoli, Navi Mumbai, India. She has 21 years of teaching experience. She is good in various subjects such as big data analysis, cryptography and system security, machine learning, security and database. Her research is mainly focused on machine translation using machine learning. She has published 11 papers in international conference, 4 international journals, 13 in national conferences. Her work was appreciated at one of the NLP conference. One Masters candidate has successfully completed her work under her guidance. She can be contacted at email: madhura.phadke@dmce.ac.in.



Dr. Satish R. Devane    is an Academician and completed his Ph.D. degree from Indian Institute of Technology (IIT). He is currently working as Principal at Karmaveer Baburao Ganpatrao Thakare College Of Engineering, Nashik, NaviMumbai. He is having 34 years of teaching experience, one year Industry and 4 years of Research experience and is proficient in many technical areas such as E-commerce, networking, Artificial Intelligence, Data Mining etc. His research area includes security, computer networks, natural language processing. He has published various research papers in international conferences and Journals out of which few papers are indexed in Scopus database. Four PhD awarded, and various candidates received their Masters degree under his guidance. He can be contacted at email: satish@dmce.ac.in.

Automatic face recording system based on quick response code using multicam

Julham¹, Muharman Lubis², Arif Ridho Lubis¹, Al-Khowarizmi³, Idham Kamil⁴

¹Department of Computer Engineering and Informatics, Politeknik Negeri Medan, Medan, Sumatera Utara, Indonesia

²School of Industrial Engineering, Telkom University, Bandung, Jawa Barat, Indonesia

³Department of Information Technology, Universitas Muhammadiyah Sumatera Utara, Medan, Sumatera Utara, Indonesia

⁴Department of Mechanical Engineering, Politeknik Negeri Medan, Medan, Sumatera Utara, Indonesia

Article Info

Article history:

Received May 10, 2021

Revised Dec 20, 2021

Accepted Jan 4, 2022

Keywords:

Digital camera

QR code

Server

Webcam

ABSTRACT

This research mainly talks about the use of quick response (QR) code reader in automating of recording the users' face. The applied QR code reader system is a dynamic type, which can be modified as required, such as adding a database, functioning to store or retrieve information in the QR code image. Since the QR code image is randomly based on its information, a QR code generator is required to display the image and store the information. While the face recorder uses a dataset available in the OpenCV library. Thus, only the registered QR code image can be used to record the user's face. To be able to work, the QR code reader should be 10 to 55 cm from the QR code image.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Julham

Department of Computer Engineering and Informatics, Politeknik Negeri Medan

Jl. Almamater No.1, Padang Bulan, Kec. Medan Baru, Kota Medan, Sumatera Utara 20155, Indonesia

Email: julham76@gmail.com

1. INTRODUCTION

The use of quick response (QR) codes is now widely found, such as for attendance systems, and validation systems [1], [2]. Incorporating a QR code application with a face detection application is important to produce a system that is able to automatically save faces and read QR codes in the workplace at the same time. This could be the forerunner of a system that can select who can read the QR code. The face detection application uses the OpenCV library which makes it easy to simplify programs related to digital images.

QR code was developed by Denso Corporation, a Japanese company that is mostly engaged in the automotive sector [3]-[5]. QR code is defined as the quick response code, which is a form of changing a barcode from a one-dimensional shape to a two-dimensional shape. There are two types of QR code, i.e. Static QR code and dynamic QR code [6], [7]. Static QR code is that containing information which is not required a specific application to translate the contents. Consequently, the static QR code can be used immediately with the help of the regular QR code reader application [8]. For example, a QR code, if scanned with an ordinary QR code reader application, it contains information that can directly link to a fixed web page [9]. QR code reader consists of 2 types, i.e. static and dynamic. The former is the QR code reader commonly used because the information can be directly understood by the users, while the latter is the QR code reader modified based on the QR code generator because of the required synchronization. Therefore, in the dynamic type, information will be more secure as it needs synchronization.

In other words, the information that appears can be directly read by the user. Static QR code is widely used today because the QR code generator application as a QR code generator is available free. Dynamic QR code is the opposite of static QR code, namely QR code which contains information which requires a specific application in translating its contents [10], [11]. The example of using a dynamic QR code is QR-code for <https://web.whatsapp.com>, when scanned using a regular QR scanner will generally produce an incomprehensible message, because it is an encrypted message from the WhatsApp application [12]. If scanned by using the WhatsApp application, it will do navigation to the WhatsApp web for an account. Therefore, this kind of QR code can only be used by scanning with certain applications.

This type of dynamic QR code is the object of research development, which is added a face detection process [13], [14]. With this face detection, it is expected that it will act as a trigger so that the QR code reader can work [15]. The built system consists of two parts, namely the first part is the generator or QR code generator side and the QR code image scanner (QR code reader).

2. RESEARCH METHODS

The method applied in this research is research and development (R&D), in which there are design, manufacture and system testing phases [16], [17]. Flowchart is used to make it easier for researchers to carry out the stages in research. In general, the flowchart of the research carried out is as shown in Figure 1.

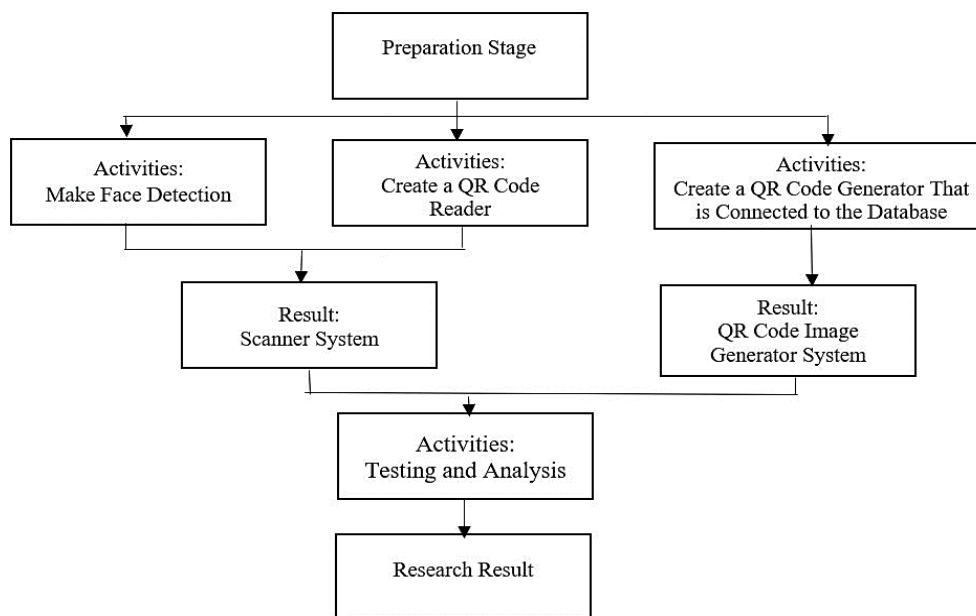


Figure 1. Research flowchart

From Figure 1, the research begins with the preparation stage, i.e., to determine the producing or generating side of the QR code image (QR code Generator) and the scanning side of the QR code image (QR code reader). On the QR code generator side, a connection to the database is required so that information can be stored, for example to store information on Final Project Student services to replace the original signature in the form of ID, name, Final Project Student service. While the QR code reader requires two webcams and a connection to the same database as the QR code Generator. So as to produce a QR code image maker system and a QR code image scanning system. The test is carried out in a room with light that can still capture normal images by the computer.

3. MATERIAL AND METHOD

In this research, supporting materials and methods are needed to be used in the research. In order to support each other, a block diagram is needed. The research begins with creating a system workflow described in a block diagram as shown in Figure 2.

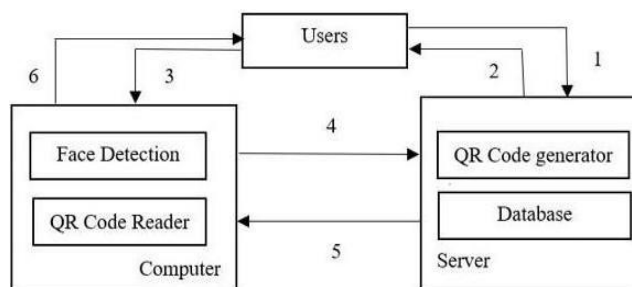


Figure 2. System block diagram

The explanation of Figure 2 is:

- Starting with the user accessing the server to get the QR code generator service (explanation to the arrow 1). Its function is to register information to be saved to a QR code image. Then the user fills in the available fields.
- Then the user notifies the server that the column is completed by pressing the available button. After that the server gives a reply in the form of a QR code image (explanation to the arrow 2).
- The explanation of the direction of arrow number 3, i.e. the QR code image received by the user, the contents of the information can be read if only using a certain scanner (reading device). To use the device, the user and the QR code image must be simultaneously scanned by the device. The discussion about these scanning devices is in the next section. If the user forces to use the regular QR code reader application to read the contents of the information in the QR code image, encrypted (random) text will appear.
- Then the computer will forward the data from the scanning results to the server (explanation to the arrow 4).
- Then the server responds to the data it receives by sending it to the sending computer (explanation to the arrow 5).
- After that the computer displays the information to the user through a screen based on the data it receives (explanation to the arrow 6).

From Figure 2, hardware and software support is needed so that the research results can be as optimal as possible.

3.1. Hardware and software

The information inserted in the QR code image is in the form of ID, name, and Final Project Student services. This entry is used to generate QR code images by the QR code generator application later. The next step is how to extract the information. In this study, the required hardware and software are:

- a. The intended server is a private server or known as a virtual private server (VPS). This server does not have a real physical form like a computer in general but has services that can be accessed and managed [18]. Besides, every VPS definitely has a public IP. And this VPS can be configured based on its users. In this study applies a VPS with the Linux operating system. After the VPS can be accessed, the next step is to fill in the service in the form of a webserver [19], [20], that is using the Apache application, server scripts using personal home page (PHP) and database applications [21]-[24]. All of these applications are intended to build web-based QR code generator services and database servers. The use of the database here is to store information that will be embedded into a QR code image. In this study, a QR code image will be formed if the users have already completed the data in the form provided and then presses the submit button, which is as shown in Figure 3. Figure 3 is an initial display that must be completed in order that a QR code image can be generated. The filled data will be stored in the database and followed by the process of generating a QR code image as shown in Figure 4. The data stored in the database are intended to be recalled. In this study, in order that data appear automatically, the detected QR code image must be appropriate. The process for generating these images uses the help of the PHP library created by Kentaro Fukuchi and is an opensource library.
- b. The intended computer is a computer that is used for the QR code image scanning system and face detection that is on the user's side. Two webcam units are the additional hardware directly connected to the computer. The important specifications on the webcam are that they have autofocus and high definition (HD) resolution of 720p. The goal is that the object captured by this webcam is more focused and sharper, so that the processing program can easily process it. Both webcams use a USB port to connect to a computer.

- c. The intended software is a programming language installed on the computer while the programming language used is the processing language [25], [26]. Processing is an open-source graphics library and integrated development environment (IDE) built for the electronic arts, new media arts, and visual design community with the aim of teaching the fundamentals of computer programming in a visual context. The Processing language uses the Java language, with additional simplifications such as additional classes and mathematical functions and operations. It also provides a graphical user interface to simplify the compile and execution stages. The program for processing QR codes requires additional program libraries, namely `zxing4processing`. This library was created by Salvatore Iaconesi and Rolf van Gelder who were released for its development. Moreover, the program for human face detection requires additional program libraries, i.e. `opencv processing`. This library is open source and includes a face detection algorithm called Haar like feature. Furthermore, this algorithm is supported by the cascade classifier to combine many features [27], [28].

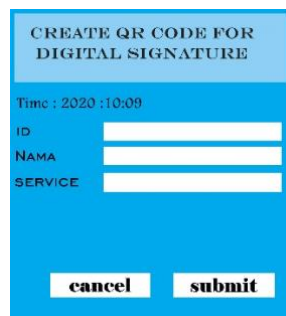


Figure 3. The example of QR code generator service

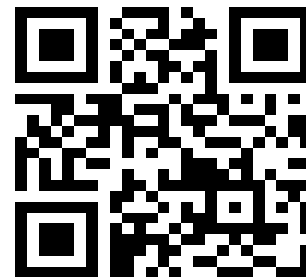


Figure 4. PHP QR code example

3.2. Data

There are two types of data taken in this study, i.e. the first is the reading distance by the scanning system (in this case the webcam) with the QR code image object only. The second is the reading distance between the scanner system and the QR code image object and the human face object simultaneously. Data collection is carried out alternately according to the stages.

- a. The data retrieving in the first form still uses a webcam unit to scan QR code images in real-time. The data collection procedure is:
 - First, make two QR code images through the application in Figure 3.
 - After getting the QR code image, make sure the image is saved on the smartphone. In this study, the screen size of the smartphone is 6.4 inches. Then zoom in to the maximum of the QR code image.
 - After successfully getting a QR code image as shown in Figure 5, then proceed by testing it to get back the information previously entered through the QR code Generator. With a webcam installed on the computer, then the QR code image is placed on the webcam, the information that appears is shown in Figure 6. And the result will be similar to in the information entered through the QR code generator as shown in Figure 3.
 - Then, carry out the measurement of reading distance in which every change in the reading distance, its QR code image will be changed, resulting in Table 1.

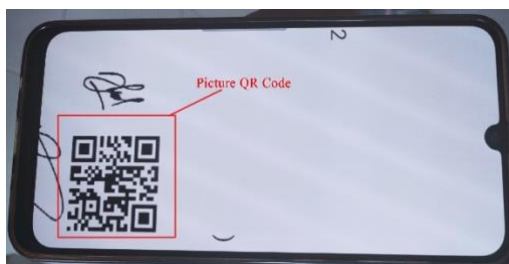


Figure 5. Display of QR code for testing



Figure 6. The system display while the expected information appear

Table 1. Data retrieving with one unit of webcam

No	Distance (cm)	Test 1	Test 2	Test 3	Test 4	Test 5
1	1	0	0	0	0	0
2	5	0	0	0	0	0
3	10	1	0	1	0	1
4	15	1	1	1	1	1
5	20	1	1	1	1	1
6	25	1	1	1	1	1
7	30	1	1	1	1	1
8	35	1	1	1	1	1
9	40	1	1	1	1	1
10	45	1	1	1	1	1
11	50	1	1	1	1	1
12	55	1	1	1	0	1
13	60	0	1	1	0	0
14	65	0	0	0	0	0
15	70	0	0	0	0	0
16	75	0	0	0	0	0

- b. Data retrieving in the second form uses two webcam units simultaneously, one webcam unit handles QR code images and the other handles human face detection. The data collection procedure is the same as data retrieving with a webcam unit, but there are some additions to the following points:
- The position of the QR code image and the face object faces each webcam, in which the position of the QR code image will be changed by the data, while the position of the user's face does not change (fixed). The position of the face must strictly be put in the green square line on the computer screen; in this case the distance is 1 meter from the face detection webcam. In Figure 7 is a display when the QR code image can provide the expected information. This is indicated by containing the data NAME, SERVICE and ACC TIME. Meanwhile, if the QR code image cannot provide the expected information, the system responds in the form of a sentence with no data.
 - Then, the measurement of the reading distance is carried out in which every change in the reading distance, its QR code image will be changed, resulting in Table 2 and Table 3.

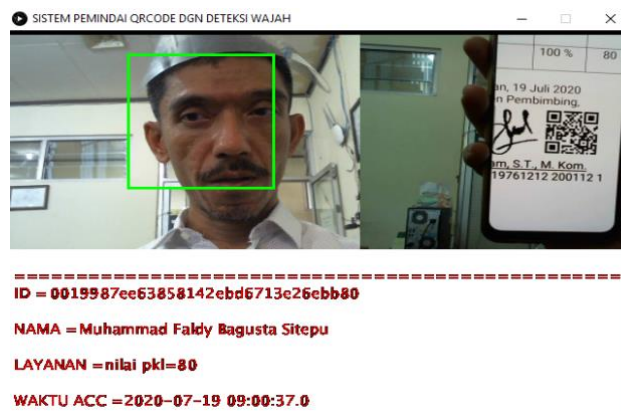


Figure 7. System view when expected information appears

During the experiment or testing, the resulted information is the same information as that entered at the beginning of getting the QR code image. This is because the information stored in the database can be retrieved based on the keywords contained in the image. Vice versa, information will not appear when it does not match. Interestingly, the utilization of QR code as two dimensional have been numerously deliver the benefit for various sectors due to the ability to store large information and extracting them with small amount of little distortion compare to one dimensional barcode [29], [30]. It can be said QR code as a fast readable way that can increase the process to scan, improving the degree of reading up to 360, support for more languages and durability against soil and damage such as several researches conducted for visual cryptography in ciphering messages [31], [32], securing the sharing process of personal confidential information either in term of forgery prevention or anti phishing attack [33]-[37]. In addition, there also several researches that focus on secure recognition process in login system using QR image [38], which mainly have challenges in term of

localization such as illumination variation and perspective distorting as well as the adaptation into embedded system platform for processing complexity and resource requirement [39]. On the other hand, in term of mobility and agility, the implementation of QR codes support various application such as smart attendance in the university [40] or even in broader level of digital education system [41] and smart city bus for the payment system [42]. Meanwhile, certain technique has been introduced to hide encrypted data using EK-EQR algorithm [43] to improve traceability system [44]. There is also necessity in the further improvement for QR code to protect information with steganography or cryptography in order to guard the knowledge against the attackers [45], which consider the error correction levels up to 30% [46] to the linkage of multimedia data [47], [48]. Interestingly, mobile computing present a great way to easily apply attendance systems [49] to large and complex battlefield interconnection systems [50], allowing QR codes to work and communicate with users anytime, anywhere and on any device.

Table 2. Data retrieving with two webcams, the detected user's face

No	Distnsce (cm)	Test 1	Test 2	Test 3	Test 4	Test 5
1	1	0	0	0	0	0
2	5	0	0	0	0	0
3	10	0	0	1	1	1
4	15	1	1	1	1	1
5	20	1	1	1	1	1
6	25	1	1	1	1	1
7	30	1	1	1	1	1
8	35	1	1	1	1	1
9	40	1	1	1	1	1
10	45	1	1	1	1	1
11	50	1	1	1	1	1
12	55	1	1	1	0	1
13	60	1	1	0	0	0
14	65	0	0	0	0	0
15	70	0	0	0	0	0
16	75	0	0	0	0	0

Table 3. Data retrieving with two webcams, undetected user's face

No	Distnsce (cm)	Test 1	Test 2	Test 3	Test 4	Test 5
1	1	0	0	0	0	0
2	5	0	0	0	0	0
3	10	0	0	0	0	0
4	15	0	0	0	0	0
5	20	0	0	0	0	0
6	25	0	0	0	0	0
7	30	0	0	0	0	0
8	35	0	0	0	0	0
9	40	0	0	0	0	0
10	45	0	0	0	0	0
11	50	0	0	0	0	0
12	55	0	0	0	0	0
13	60	0	0	0	0	0
14	65	0	0	0	0	0
15	70	0	0	0	0	0
16	75	0	0	0	0	0

4. RESULTS AND DISCUSSION

The discussion begins with getting the parameters contained in Tables 1 to 3:

- Column Distance (cm) is the distance between the QR code image and the webcam in centimeters.
- Column 1st test, 2nd test to 5th test are repeated data collection activities.
- Numbers 0 and 1 describe whether or not is a change in information on the monitor screen. Its function is to find out the response from the webcam whether it can still process (indicated by changes in information data on the monitor screen) or cannot process QR code images (indicated by no change in data at all). For number 1, it means that the information on the screen has changed, while the number 0 means that there is no change in the information on the screen.

Then, in Table 1 there are rows containing non similar numbers, such as those in rows 3, 12, 13, while The other rows have similar numbers. So that the non-similar rows will find the approximate value is:

- Row 3 = $(1 + 0 + 1 + 0 + 1) / 5 = 3/5 = 0.6$ and becomes 1.
- Row 12 = $(1 + 1 + 1 + 0 + 1) / 5 = 4/5 = 0.8$ and becomes 1.
- Row 13 = $(0 + 1 + 1 + 0 + 0) / 5 = 2/5 = 0.4$ and becomes 0.

From the calculation of this approach, the condition in Table 1 is:

- The distance at 10 cm has changes in information.
- The distance at 55 cm has changes in information.
- The distance at 60 cm has no change in information.

Then in Table 2 there are rows containing non-similar numbers, such as rows 3, 12, 13., while the other rows have similar numbers. So that the non-similar rows will find the approximate value is:

- Row 3 = $(0 + 0 + 1 + 1 + 1) / 5 = 3/5 = 0.6$ and becomes 1
- Row 12 = $(1 + 1 + 1 + 0 + 1) / 5 = 4/5 = 0.8$ and becomes 1
- Row 13 = $(1 + 1 + 0 + 0 + 0) / 5 = 2/5 = 0.4$ and becomes 0

From the calculation of this approach, the condition in Table 2 is:

- The distance at 10 cm has changes in information.
- The distance at 55 cm has changes in information.
- The distance at 60 cm has no change in information.

Continued in Table 3, all rows contain the number 0 which means there is no change or the system does not respond. Therefore, they can be summarized into the following Tables 4 and 5:

Table 4. The results of the conclusion Table 1

No	Distance (cm)	Status
1	1	Unchange
2	5	Unchange
3	10	Change
4	15	Change
5	20	Change
6	25	Change
7	30	Change
8	35	Change
9	40	Change
10	45	Change
11	50	Change
12	55	Change
13	60	Unchange
14	65	Unchange
15	70	Unchange
16	75	Unchange

Table 5. The results of the conclusion Table 2

No	Distance (cm)	Status
1	1	Unchange
2	5	Unchange
3	10	Change
4	15	Change
5	20	Change
6	25	Change
7	30	Change
8	35	Change
9	40	Change
10	45	Change
11	50	Change
12	55	Change
13	60	Unchange
14	65	Unchange
15	70	Unchange
16	75	Unchange

5. CONCLUSION

In summary, perform face detection using a QR code as a face identifier that accesses the QR code so that it gets a track record. The success of reading the QR code is influenced by 2 factors, namely the first is the distance between the QR code object and the webcam and the second is detecting the user's face. With multicam, these two factors can work simultaneously, because each camera handles each process. Without these two factors, the system will not work optimally. In addition, the QR code reader (QR code reader) and QR code generator (QR code Generator) cannot be used without a VPS. This is because every transaction can be stored/recorded in it, in the form of the user's face and information. Besides, the dynamic type of QR code reader applied in this study requires a QR code generator for synchronizing information, so that the information cannot be directly understood by users when using any QR code reader device on the market. With the ability to detect the user's face and store it, it is hoped that in the next research the system can secure the information contained in the QRCode based on the user's face through face recognition.

REFERENCES

- [1] A. Abas, Y. Yusof, R. Din, F. Azali, and B. Osman, "Increasing data storage of coloured QR code using compress, multiplexing and multilayered technique," *Bull. Electr. Eng. Informatics*, vol. 9, no. 6, pp. 2555–2561, 2020, doi: 10.11591/eei.v9i6.2481.
- [2] P. Satanasawapak, W. Kawsewai, S. Promlee, and A. Vilamat, "Residential access control system using QR code and the IoT," *Int. J. Electr. Comput. Eng.*, vol. 11, no. 4, pp. 3267–3274, 2021, doi: 10.11591/ijece.v11i4.pp3267-3274.
- [3] F. Kumagai, "What Can Be Done Before a Municipality 'Disappears': Making the Best of Negative Municipal Resources," in *Municipal Power and Population Decline in Japan: Goki-Shichido and Regional Variations*, Singapore: Springer Singapore, 2020, pp. 229–264.
- [4] S. C. Cha, H. Wang, Z. Tan, Y. J. Joung, Y. C. Tseng, and K. H. Yeh, "On Privacy Aware Carriers for Value-Possessed e-Invoices Considering Intelligence Mining," *IEEE Trans. Emerg. Top. Comput. Intell.*, vol. 4, no. 5, pp. 641–652, 2020, doi: 10.1109/TETCI.2019.2938547.
- [5] M. Xu *et al.*, "Stylized Aesthetic QR code," *IEEE Trans. Multimed.*, vol. 21, no. 8, pp. 1960–1970, 2019, doi: 10.1109/TMM.2019.2891420.
- [6] S. Li, J. Shang, Z. Duan, and J. Huang, "Fast detection method of quick response code based on run-length coding," *IET Image Process.*, vol. 12, no. 4, pp. 546–551(5), Apr. 2018, [Online]. Available: <https://digital-library.theiet.org/content/journals/10.1049/iet-ipr.2017.0677>.
- [7] U. A. Waqas, M. Khan, and S. I. Batool, "A new watermarking scheme based on Daubechies wavelet and chaotic map for quick response code images," *Multimed. Tools Appl.*, vol. 79, no. 9, pp. 6891–6914, 2020, doi: 10.1007/s11042-019-08570-5.
- [8] E. Ozan, "QR code Based Signage to Support Automated Driving Systems on Rural Area Roads," in *Industrial Engineering and Operations Management II*, 2019, pp. 109–116.
- [9] M. Abdolahi, H. Jiang, and B. Kaminska, "Structural colour {QR} codes for multichannel information storage with enhanced optical security and life expectancy," *Nanotechnology*, vol. 30, no. 40, p. 405301, Jul. 2019, doi: 10.1088/1361-6528/ab2d3b.
- [10] I. Farida, F. R. Agung, R. Aisyah, and D. Nasrudin, "Learning crude oil based on environmental literacy," *J. Phys. Conf. Ser.*, vol. 1563, no. 1, 2020, doi: 10.1088/1742-6596/1563/1/012047.
- [11] R. Focardi, F. L. Luccio, and H. A. M. Wahsheh, "Usable security for QR code," *J. Inf. Secur. Appl.*, vol. 48, p. 102369, 2019, doi: <https://doi.org/10.1016/j.jisa.2019.102369>.
- [12] J. Jabbar, S. I. Malik, G. AlFarsi, and R. M. Tawafak, "The Impact of WhatsApp on Employees in Higher Education," in *Recent Advances in Intelligent Systems and Smart Applications*, M. Al-Emran, K. Shaalan, and A. E. Hassaniien, Eds. Cham: Springer International Publishing, 2021, pp. 639–651.
- [13] W. Liu, B. Wang, Y. Li, and M. Wu, "Screen-camera communication system based on dynamic QR code," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 790, no. 1, 2020, doi: 10.1088/1757-899X/790/1/012012.

- [14] H. Baidong and Z. Yukun, "Research on Quickpass Payment Terminal Application System Based on dynamic QR code," *J. Phys. Conf. Ser.*, vol. 1168, no. 3, 2019, doi: 10.1088/1742-6596/1168/3/032059.
- [15] A. R. Jones, M. D. Aspinall, and M. J. Joyce, "A remotely triggered fast neutron detection instrument based on a plastic organic scintillator," *Rev. Sci. Instrum.*, vol. 89, no. 2, p. 23115, 2018, doi: 10.1063/1.5012121.
- [16] J. Choi and F. J. Contractor, "Improving the Progress of Research & Development (R&D) Projects by Selecting an Optimal Alliance Structure and Partner Type," *Br. J. Manag.*, vol. 30, no. 4, pp. 791–809, 2019, doi: 10.1111/1467-8551.12267.
- [17] K. Lee, Y. Jeong, and B. Yoon, "Developing an research and development (R&D) process improvement system to simulate the performance of R&D activities," *Comput. Ind.*, vol. 92–93, pp. 178–193, 2017, doi: <https://doi.org/10.1016/j.compind.2017.08.001>.
- [18] H. A. Adam, Julham, A. R. Lubis, F. Fachrizal, and Y. Fatmi, "Design information seating chart system in classroom with wireless sensor network," in *Journal of Physics: Conference Series*, 2019, vol. 1361, no. 1, doi: 10.1088/1742-6596/1361/1/012007.
- [19] T. Fadhilah Iskandar, M. Lubis, T. Fabrianti Kusumasari, and A. Ridho Lubis, "Comparison between client-side and server-side rendering in the web development," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 801, no. 1, 2020, doi: 10.1088/1757-899X/801/1/012136.
- [20] I. Kamil, Julham, M. Lubis, and A. R. Lubis, "Management maintenance system for remote control based on microcontroller and virtual private serve," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 16, no. 3, 2019, doi: 10.11591/ijeecs.v16.i3.pp1349-1355.
- [21] A. R. Lubis, F. Fachrizal, and H. Maulana, "Database management optimization using PostgreSQL replication database in database system," *Adv. Sci. Lett.*, vol. 23, no. 5, 2017, doi: 10.1166/asl.2017.8286.
- [22] Y. Y. Lase, Haryadi, and Y. Fatmi, "Analysis of effective storage time to determine the quality of milk using simple additive weighting method," *J. Phys. Conf. Ser.*, vol. 1361, no. 1, 2019, doi: 10.1088/1742-6596/1361/1/012077.
- [23] A. Ramadhan, M. Lubis, W. Puspitasari, and A. R. Lubis, "Development of Web Stock Opname Application With SAP Business One Using Scrum Method," in *2019 International Conference of Computer Science and Information Technology (ICoSNIKOM)*, Nov. 2019, pp. 1–5, doi: 10.1109/ICoSNIKOM48755.2019.9111526.
- [24] A. R. Lubis, M. K. M. Nasution, O. S. Sitompul, and E. M. Zamzami, "A Framework of Utilizing Big Data of Social Media to Find Out the Habits of Users Using Keyword," 2020, pp. 140–144.
- [25] A. R. Lubis *et al.*, "Obtaining Value From The Constraints in Finding User Habitual Words," pp. 8–11, 2020.
- [26] A. R. Lubis, M. K. M. Nasution, O. Salim Sitompul, and E. Muisa Zamzami, "The effect of the TF-IDF algorithm in times series in forecasting word on social media," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 22, no. 2, p. 976, 2021, doi: 10.11591/ijeecs.v22.i2.pp976-984.
- [27] M. Almasi, "An Investigation on Face Detection Applications," *Int. J. Comput. Appl.*, vol. 177, no. 21, pp. 17–23, 2019, doi: 10.5120/ijca2019919664.
- [28] V. G. Patel and A. Suthar, "Human Face Detection and Tracking," *Int. J. Comput. Eng. Technol.*, vol. 9, no. 4, pp. 187–195, 2018, doi: 10.13140/RG.2.2.24467.43042.
- [29] S. Singh, "QR code Analysis," *Int. J. Adv. Res. Comput. Sci. Softw. Eng.*, vol. 6, no. 5, pp. 89–92, 2016, [Online]. Available: www.ijarcsse.com.
- [30] S. Nasir, S. Al-Qaraawi, and M. Croock, "Design and implementation a network mobile application for plants shopping center using QR code," *Int. J. Electr. Comput. Eng.*, vol. 10, no. 6, pp. 5940–5950, 2020, doi: 10.11591/ijece.v10i6.pp5940-5950.
- [31] S. K. Thamer and B. N. Ameen, "A new method for ciphering a message using QR code," *Comput. Sci. Eng.*, vol. 6, no. 2, pp. 19–24, 2016, doi: 10.5923/j.computer.20160602.01.
- [32] X. Cao, L. Feng, P. Cao, and J. Hu, "Secure QR code Scheme Based on Visual Cryptography," vol. 133, pp. 433–436, 2016, doi: 10.2991/aaiie-16.2016.99.
- [33] M. S. Ahamed and H. Asiful Mustafa, "A Secure QR code System for Sharing Personal Confidential Information," *5th Int. Conf. Comput. Commun. Chem. Mater. Electron. Eng. IC4ME2 2019*, pp. 11–12, 2019, doi: 10.1109/IC4ME247184.2019.9036521.
- [34] A. T. Purnomo, Y. S. Gondokaryono, and C. S. Kim, "Mutual authentication in securing mobile payment system using encrypted QR code based on public key infrastructure," *Proc. 2016 6th Int. Conf. Syst. Eng. Technol. ICSET 2016*, pp. 194–198, 2017, doi: 10.1109/FIT.2016.7857564.
- [35] V. Mavroeidis and M. Nicho, "Quick response code secure: A cryptographically secure anti-phishing tool for QR code attacks," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10446 LNCS, no. August 2017, pp. 313–324, 2017, doi: 10.1007/978-3-319-65127-9_25.
- [36] A. R. Lubis, F. Fachrizal, M. Lubis, and H. M. Tahir, "Wireless service at Public University: A survey of users perception on security aspects," in *2018 International Conference on Information and Communications Technology, ICOIACT 2018*, 2018, vol. 2018-Janua, doi: 10.1109/ICOIACT.2018.8350786.
- [37] Julham, F. Fachrizal, H. A. Adam, Y. Fatmi, and A. R. Lubis, "Security of data communications between embedded arduino systems with substitution encryption," in *Proceedings of the 2nd International Conference on Informatics and Computing, ICIC 2017*, 2018, vol. 2018-Janua, doi: 10.1109/IAC.2017.8280578.
- [38] S. Dutta, S. Singh, and P. Ghodke, "Secure Login System Using QR-Image," *Int. J. Comput. Appl. Technol. Res.*, vol. 8, no. 1, pp. 1–3, 2019, doi: 10.7753/ijcatr0801.1001.
- [39] H. Tribak and Y. Zaz, "QR code Recognition based on Principal Components Analysis Method," *Int. J. Adv. Comput. Sci. Appl.*, vol. 8, no. 4, pp. 241–248, 2017, doi: 10.14569/ijacsa.2017.080433.
- [40] F. Masalha and N. Hirzallah, "A Students Attendance System Using QR code," *Int. J. Adv. Comput. Sci. Appl.*, vol. 5, no. 3, pp. 75–79, 2014, [Online]. Available: http://bvs.sld.cu/revistas/mciego/vol14_supl2_08/articulos/a6_v14_supl208.htm.
- [41] S. Goyal, S. Yadav, and M. Mathuria, "Exploring concept of QR code and its benefits in digital education system," *2016 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2016*, pp. 1141–1147, 2016, doi: 10.1109/ICACCI.2016.7732198.
- [42] S. L. Fong, D. Wui Yung Chin, R. A. Abbas, A. Jamal, and F. Y. H. Ahmed, "Smart City Bus Application with QR code: A Review," *2019 IEEE Int. Conf. Autom. Control Intell. Syst. I2CACIS 2019 - Proc.*, no. June, pp. 34–39, 2019, doi: 10.1109/I2CACIS.2019.8825047.
- [43] N. Naik, N. Kadam, and M. Bhalekar, "Technique to Hide Encrypted Data in QR codes using EK-EQR Algorithm," *Int. J. Comput. Appl.*, vol. 161, no. 12, pp. 25–28, 2017, doi: 10.5120/ijca2017913397.
- [44] L. Tarjan, I. Šenk, S. Tegeltija, S. Stankovski, and G. Ostojic, "A readability analysis for QR code application in a traceability system," *Comput. Electron. Agric.*, vol. 109, pp. 1–11, 2014, doi: 10.1016/j.compag.2014.08.015.
- [45] M. Alajmi, I. Elashry, H. S. El-Sayed, and O. S. Farag Allah, "Steganography of Encrypted Messages Inside Valid QR codes," *IEEE Access*, vol. 8, pp. 27861–27873, 2020, doi: 10.1109/ACCESS.2020.2971984.
- [46] A. Mishra and M. Mathuria, "A Review on QR code," *Int. J. Comput. Appl.*, vol. 164, no. 9, pp. 17–19, 2017, doi: 10.5120/ijca2017913739.
- [47] M. Z. Alksasbeh, B. A. Y. Alqaralleh, T. Abukhalil, A. Abukaraki, T. Al Rawashdeh, and M. Al-Jaafreh, "Smart detection of offensive words in social media using the soundex algorithm and permuterm index," *Int. J. Electr. Comput. Eng.*, vol. 11, no. 5, pp.

4431–4438, 2021, doi: 10.11591/ijece.v11i5.pp4431-4438.




[48] C. L. Li, Y. Su, and R. Z. Wang, "Generating photomosaics with QR code capability," *Mathematics*, vol. 8, no. 9, 2020, doi: 10.3390/math8091613.

[49] F. Zailani and A. A. Shamsu Adli, "QR code Attendance System," 2020.




[50] M. Guha and J. C. Calliott, "Interpreting the Signs The Prospects of QR Coding the Battlespace," *Journal of Indo-Pacific Aff.*, pp. 201–235, 2021.

BIOGRAPHIES OF AUTHORS






Julham, ST., M.Kom.    holds a master from Universitas Putra Indonesia Yptk Padang in 2008. He is currently an associate professor at Department of Computer Engineering and Informatics, Politeknik Negeri Medan, Indonesia. His research includes computer networking, embedded system, computer vision, artificial intelligence. He can be contacted at email: julham@polmed.ac.id






Muharman Lubis, Ph.D. IT.    has finished his Doctoral degree recently in Information Technology at 2017 in International Islamic University Malaysia, he also received his Master degree from same university at 2011 and Bachelor degree from University Utara Malaysia at 2008, both in Information Technology. He joined as a Lecturer in the School of Industrial Engineering, Telkom University, in 2017. His research interests include privacy protection, information security awareness, knowledge management and project management. He can be contacted at email: muharmanlubis@telkomuniversity.ac.id






Arif Ridho Lubis, M.Sc. IT.    he got master from University Utara Malaysia in 2012 and graduate from University Utara Malaysia in 2011, both information technology. He is a lecturer in Department of Computer Engineering and Informatics, Politeknik Negeri Medan in 2015. His research interest includes computer science, network, science and project management. He can be contacted at email: arifridho@polmed.ac.id



Al-Khowarizmi, M.Kom.    was born in Medan, Indonesia, in 1992. He is a lecturer in Department of Information Technology-Faculty of Computer Science and Information Technology at Universitas Muhammadiyah Sumatera. He got master from University of Sumatera Utara in 2017 and graduate from Universitas Harapan Medan in 2014, both information system. His main research interest is data science, big data, machine learning, neural network, artificial intelligence and business intelligence. He can be contacted at email: alkhwarizmi@umsu.ac.id



Idham Kamil, S.T., M.T.    he obtained his master's degree from Universitas Sumatera Utara in 2007 and graduated from Universitas Sumatera Utara in 1996, both in Mechanical engineering. He is a lecturer at the Department of Mechanical Engineering at the Medan State Polytechnic. His research interests include management maintenance, renewable energy, and Networking. He can be contacted at email: idhamkamil@polmed.ac.id

Indonesian part of speech tagging using maximum entropy markov model on Indonesian manually tagged corpus

Denis Eka Cahyani¹, Winda Mustikaningtyas²

¹Department of Mathematics, Universitas Negeri Malang, Malang, Indonesia

²Department of Informatics, Universitas Sebelas Maret, Solo, Indonesia

Article Info

Article history:

Received May 30, 2021

Revised Nov 5, 2021

Accepted Dec 27, 2021

Keywords:

Bigram

Maximum entropy markov model

N-gram

Part of speech tagging

Trigram

ABSTRACT

This research discusses the development of a part of speech (POS) tagging system to solve the problem of word ambiguity. This paper presents a new method, namely maximum entropy markov model (MEMM) to solve word ambiguity on the Indonesian dataset. A manually labeled "Indonesian manually tagged corpus" was used as data. Furthermore, the corpus is processed using the entropy formula to obtain the weight of the value of the word being searched for, then calculating it into the MEMM Bigram and MEMM Trigram algorithms with the previously obtained rules to determine the part of speech (POS) tag that has the highest probability. The results obtained show POS tagging using the MEMM method has advantages over the methods used previously which used the same data. This paper improves a performance evaluation of research previously. The resulting average accuracy is 83.04% for the MEMM Bigram algorithm and 86.66% for the MEMM Trigram. The MEMM Trigram algorithm is better than the MEMM Bigram algorithm.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Denis Eka Cahyani

Department of Mathematics, Universitas Negeri Malang

Jl. Semarang No.5, Sumber Sari, Lowokwaru, Malang, Jawa Timur 65145, Indonesia

Email: denis.eka.cahyani.fmipa@um.ac.id

1. INTRODUCTION

Recently, paying attention to writing sentence patterns is very important to do. Writing sentence patterns following the rules of the Indonesian language will make it easier for the person reading the sentence to be accepted so that it will reduce the bad impact of misperceptions between people in understanding the meaning of the sentence. Therefore, the writing that is made is expected to be conveyed informatively and communicatively to the general public. This research will discuss the importance of doing part of speech (POS) tagging in Indonesian. POS tagging is the process of automatically giving word-class labels to a word in a sentence so that it can help in arranging sentences according to good sentence patterns [1], [2].

In building Indonesian POS tagger, there are problems related to word ambiguity [3]. The ambiguity of the word in question is when there is the same word but has a different POS tag depending on the context of the sentence. An example is "Bisa ular kobra bisa mematikan" or "Cobra's venom can be deadly". The three words "bisa" or "venom" or "can" in the sentence are considered homonyms because they have different meanings but the pronunciation and spelling are the same. The first "bisa" or "venom" is a type of noun, while the second "bisa" or "can" is a type of verb. The difference in word type labeling is a problem because it will affect the POS tagging results and word ambiguity [4]. Ambiguous words can make readers confused because this ambiguous word has a double meaning. So it is feared that it could cause

misunderstandings when reading the sentence [5]. Based on this background, it is important to do research related to solve the ambiguity problem of POS tagging in Indonesian language.

Research on POS tagging for Indonesian has been developed previously. Several studies related to POS tagging in Indonesian that has been carried out are using rule-based methods which produce an accuracy rate of 79% [6], conditional random field (CRF) produces an accuracy rate of 83.72% using the 10-fold cross-validation test on the corpus II [7], hidden markov model (HMM) Bigram-viterbi and HMM Trigram-viterbi produce accuracy rates of 77.56% and 61.67% [8]. The previous study has a large dataset of more than 250,000 tokens as in the study [8], but the accuracy performance results have not been optimal. The accuracy performance needs to be improved so that it can solve the ambiguity problem in the POS tagging.

Previous research related to POS tagging was also carried out using HMM which resulted an accuracy rate of 96.50% [9]. Then, bidirectional long short-term memory produces an accuracy rate of 96.92% [10]. Other research related to POS tagging that has been conducted using the deep neural network for Turkish produces an accuracy rate of 88.7% [11], deep learning for Nepali produces an accuracy of 99% [12], maximum entropy for English produces an accuracy of 96.6% [13], HMM for Azerbaijani language yields an accuracy of 90% [14], HMM and morphological rules for Myanmar language get 94% precision [15], and CRF and Bi-LSTM for the arabic tweets get 96.5% accuracy [16]. Previous studies generate optimal accuracy values, but still using small amounts of dataset to conduct research experiments, as in research [12] using 100,720 tokens and 4,325 sentences. In this developed research, a larger number of dataset are used, namely 256,682 tokens and 10,000 sentences. So that this study has the novelty of using a larger amount of data than previous studies.

Based on the problems and several related studies, this research was conducted using the maximum entropy markov model (MEMM) method. The main contribution of this paper is to present a new method to solve the ambiguity problem of POS tagging, namely MEMM using large datasets. This paper improves a performance evaluation of research previously. The MEMM method provides a good level of accuracy in handling POS tagging because it can handle complex problems [17].

MEMM is a graphical model used to combine markov chain and maximum entropy (ME) features. The ME method can overcome the weakness of the HMM method where HMM can only calculate the possible observation words conditioned on the tag. ME can complement the HMM method by estimating the distribution parameters used for the transition probability separately and the POS tagging process can be carried out more efficiently [1]. MEMM can calculate a single probability function in each state and then compare it with the previous word and the word that will be given a word class label [1].

The MEMM method has some advantages over the HMM method. The MEMM method is considered to be able to solve the multi-feature representation problem which is a problem in the HMM method. The MEMM offers increased freedom in selecting features to represent observations [18]. For example, HMM has not paid attention to word spelling in the completion of the POS tagging case, while the MEMM method has paid attention to word spelling in solving the POS tagging case. In contrast to HMM which assumes independence between features [19], MEMM does not assume independence between features. Therefore, MEMM makes it possible to define many correlated yet informative features [1].

This paper discusses POS tagging using MEMM in the Indonesian dataset. The POS tagging uses the Indonesian language dataset from "Indonesian manually tagged corpus" in previous research [20] to label ambiguous words and words as a whole. The dataset is an adaptation of the penn tree bank corpus which is widely used as a reference for English POS tagging research and the POS tag values have been annotated manually. The dataset has large amount of data, consists of 10,000 sentences with a total of 256,682 tokens.

2. RESEARCH METHOD

The development of Indonesian POS tagging using MEMM approach. The stages taken in Indonesian POS tagging research using the MEMM algorithm are preprocessing, implementation of the MEMM algorithm, evaluation and analysis. The explanation of each stage in this research is:

2.1. Preprocessing

The data used in this study were taken from Indonesian manually tagged corpus, where all the data is presented in the form of sentences that have been sorted from manual processes. The dataset used consists of 10,000 sentences with a total of 256,682 tokens and consists of 23 POS tags (such as noun-NN, proper noun-NNP, and VB). The distribution of tags in the dataset is not balanced, where the most tags are NN with 55,575 tokens and the least tag is UH with 29 tokens. The results from the preprocessing stage will be input in the word error detection process. The preprocessing stage that will be carried out in this study [21], [22]: i) filtering, is the process of removing tags and words; ii) tokenizing, is the process of cutting the input string based on each word that composes it; iii) case folding, is the process of converting the entire text in a

document into a standard form, namely in lowercase form so that only letters A to Z are accepted while characters other than letters are eliminated; iv) removing spaces (“ ”) in a phrase that has more than one word. Example: “*Pesta olahragga*” or “Games”, “*Rumah sakit*” or “Hospital”.

2.2. Implementation of the maximum entropy markov model algorithm

After the preprocessing stage, the data on the corpus is processed using the MEMM feature which is used to make contextual predictions. The MEMM is the most common form of classification of maximum entropy [23], [24]. Maximum entropy is defined as the average maximum information value for a set of events X with a uniform probability value distribution [25]. The application of the MEMM algorithm begins by providing text input to the system. The text is preprocessed, then each word in the sentence will look for the probability value of the word class against the word class of the previous word in the corpus. The calculation of probability begins by calculating the probability of the first word by looking at the previous word (start). The probability of the second to the last word will be calculated by looking at the previous word-class using (1) [1].

$$H(x) = - \sum_x P(x) \log_2 P(x) \quad (1)$$

Where, $H(x)$ is entropy's value on variable X , $P(x)$ is the value of $\frac{Y}{x}$, x : the all words that appear in the sentence, $\log_2 P(x)$: formulated with the basic logarithm $\frac{\log x}{\log 2}$.

The difference between the implementation of MEMM Bigram and MEMM Trigram is if the MEMM Bigram is observed for the previous 1 tag, while MEMM Trigram is observed for the previous 2 tags [26]. Markov chain in applying the MEMM method serves to calculate the probability of an observable sequence of events [27]. If the word weight is known, then the MEMM calculation is carried out using (2). The output results obtained in this process are words and word classes by finding the best probability tags using (3) [1]:

$$p(c|x) = \frac{\exp(\sum_{i=0}^N w_{ci} f_i(c, x))}{\sum_{c' \in C} \exp(\sum_{i=0}^N w_{c'i} f_i(c', x))} \quad (2)$$

$$p(c|x) = \operatorname{argmax}_{c' \in C} \frac{\exp(\sum_{i=0}^N w_{ci} f_i(c, x))}{\sum_{c' \in C} \exp(\sum_{i=0}^N w_{c'i} f_i(c', x))} \quad (3)$$

Where:

- e : 2,7
- c : The word class of the designated data
- x : Words from the entire dataset
- c' : Entire class of predefined words
- $f_i(c, x)$: Feature i for a particular class c for a given observation x .
- w : Weighted word value
- i : Observation word index

After knowing the best probability for each word in the sentence, the calculation of perplexity is a measure of the performance of language modeling based on word probability in the corpus. Perplexity is applied in this research as a form of validation against the comparison of the accuracy results obtained from the MEMM Bigram and MEMM Trigram methods. Perplexity generated by normalizing the testing data based on the number of words, meaning that minimizing perplexity is the same as maximizing probability. To calculate the perplexity in each sentence, the calculation for the testing data $W = w_1 w_2 \dots w_N$ can be done using (4) [1]. Bigram perplexity can be calculated using (5). And the Trigram perplexity can be calculated using (6) [1].

$$PP(W) = P(w_1 w_2 \dots w_N)^{-\frac{1}{N}} = \sqrt[N]{\frac{1}{P(w_1 w_2 \dots w_N)}} \quad (4)$$

Where, $PP(W)$ is perplexity to the sentence, P is probability, w are the occurrence of the word on the corpus, N is total words in testing data.

$$PP(W) = \sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i|w_{i-1})}} \quad (5)$$

$$PP(W) = \sqrt[N]{\prod_{i=1}^N \frac{1}{P(w_i|w_{i-2}w_{i-1})}} \quad (6)$$

Where, $PP(W)$ is perplexity to the sentence, P is probability, w are the occurrence of the word on the corpus, N is total words in testing data.

The calculation of the probability of occurring words on the corpus is added with Laplace smoothing to handle the probability value of 0 (zero). It can be written in (7) for Bigram and (8) for Trigram. Meanwhile, the perplexity in the whole sentence can be calculated using (s_1, s_2, \dots, s_m) which is part of the corpus can be seen in (9) [1].

$$P(w_i|w_{i-1}) = \frac{C(w_{i-1}, w_i) + 1}{C(w_{i-1}) + V} \quad (7)$$

$$P(w_i|w_{i-1}w_{i-2}) = \frac{C(w_{i-2}w_{i-1}, w_i) + 1}{C(w_{i-2}w_{i-1}) + V} \quad (8)$$

Where, C is number of words, w_i are the designated word, w_{i-1} are the occurrence of 1 previous word in the corpus, w_{i-2} are the occurrence of 2 previous words in the corpus, V is total words in the training data (the same word counts as 1).

$$PP(C) = \sqrt[N]{\frac{1}{PP(s_1s_2 \dots s_m)}} \quad (9)$$

Where, $PP(C)$ is perplexity corpus, s are the perplexity result of each sentence, N is total words in testing data, m are the whole sentence on the data testing.

2.3. Evaluation and analysis

The evaluation carried out in this study consisted of three scenarios. The first evaluation was carried out to determine the level of accuracy of all words using the 10-fold cross-validation scenario. The second evaluation was carried out to determine the level of accuracy of the whole word using artificial testing data outside the corpus. The third evaluation is carried out to determine the level of accuracy on ambiguous words that are predicted to be correct.

In the first evaluation, the research was carried out applying the 10-fold cross-validation testing technique [28]. The dataset is divided into 10 parts, one part is used for testing and 9 parts are used for modeling (training) [29]. This means that when there is a corpus of 10,000 sentences of data, 1,000 sentences are used as testing data and 9,000 other sentences are used as training data [30]. Then, the second evaluation was done by dividing 1,000 sentences used as testing data (data outside the corpus) and 10,000 sentences used as training data. Artificial data outside the corpus used in the second test was obtained by manually collecting tokens in the Indonesian manually tagged corpus dataset. The third evaluation was done by calculating the number of ambiguous words in the corpus from the calculation results of MEMM Bigram and MEMM Trigram for the number of ambiguous words predicted accurately by the system based on the first evaluation scenario.

The accuracy of each result obtained from the 10-fold cross-validation is calculated by comparing the results with the original data. Accuracy can be obtained using (10) [31], [32]. Then the accuracy between MEMM Bigram and MEMM Trigram is compared so that it is known which method is better.

$$Accuracy = \frac{\text{number of data predicted correctly}}{\text{number of data predicted}} \quad (10)$$

3. RESULTS AND DISCUSSION

The results of applying the MEMM Bigram and MEMM Trigram algorithms are: From the test scenario with 10-fold cross-validation in first evaluation, there are a total of 10 experiments. The results of the calculation of the accuracy and the average accuracy of all the words obtained can be seen in Table 1.

The results of applying the MEMM Bigram and MEMM Trigram algorithms in Table 1 can be concluded that the highest accuracy is obtained in scenario 5 with an accuracy value of 85.18% for the MEMM Bigram algorithm and scenario 6 of 89.13% for the MEMM Trigram algorithm. The resulting average accuracy in first evaluation is 83.04% for the MEMM Bigram algorithm and 86.66% for the MEMM Trigram, respectively. In the second evaluation, data sharing was carried out, namely 10,000 sentences of training data and 1000 artificial sentences outside the corpus. From the second evaluation, the accuracy results for all words were 93.85% using MEMM Bigram and 94.17% using MEMM Trigram.

Table 1. Calculation of accuracy and average accuracy of overall words

Scenario	MEMM Bigram accuracy	MEMM Trigram accuracy
1	81.40%	85.11%
2	84.50%	87.76%
3	84.16%	87.71%
4	84.34%	88.64%
5	85.18%	88.90%
6	85.15%	89.13%
7	81.51%	84.97%
8	80.65%	84.58%
9	82.93%	85.02%
10	80.55%	84.80%
Average of Accuracy	83.04%	86.66%

The results of the application of the MEMM Trigram algorithm show higher accuracy than the MEMM Bigram in both the first and second evaluations. This is proven using perplexity. Perplexity validation was carried out in this study by taking 10 testing data (data outside the corpus). The results of calculating the perplexity Bigram and Trigram for 10 data testing can be seen in Table 2.

The results in Table 2 are then calculated using (9) and the results are 0.194216429 for MEMM Bigram and 0.181184234 for MEMM Trigram. These results indicate that the perplexity on the Trigram is smaller than the perplexity on the Bigram. These result also proven that the accuracy results using the Trigram are better than the Bigram accuracy results in the building POS tagger using the MEMM algorithm.

Table 2. Perplexity results

No	Perplexity Bigram results	Perplexity Trigram results
1	47820.33741	63053.74758
2	72302.91253	136004.8056
3	106849.6051	129885.1291
4	186060.2143	225805.7507
5	134523.1939	134519.3956
6	58395.40241	99034.39428
7	47024.58565	99298.23232
8	30833.01764	78606.7157
9	41918.24686	96455.01441
10	75934.56894	111161.9007
Result	0.181184234	0.194216429

On the third evaluation, the system is successful in labeling all words in the corpus. In addition, ambiguous words in the corpus, which numbered 91,851 were also successfully labeled by the system. Of all the ambiguous words, there are some ambiguous words in the corpus which can be properly labeled. However, not all ambiguous words are labeled correctly, there are also ambiguous words that are incorrectly labeled. Table 3 is an example of ambiguous words contained in the corpus. Table 4 is the result of ambiguous words that were predicted correctly.

Table 4 is the result of the test for handling the problem of word ambiguity. The number of ambiguous words predicted accurately using the MEMM Bigram algorithm is 87,099 from 91,851 words, and using MEMM Trigram algorithm is 89,650 from 91,851 words. From the number of ambiguous words

predicted correctly, the accuracy results in third evaluation obtained are 94.83% using MEMM Bigram and 97.60% using MEMM Trigram.

Table 3. Examples of ambiguous words

Word	Tag	Total number
“Akan” or “will” or “for”	IN	1
	MD	1806
“Hingga” or “until” or “to”	IN	348
	SC	39
“Untuk” or “to” or “for”	IN	396
	NN	1
	NNP	14
	SC	1839

Table 4. Results of ambiguous words predict exactly

Scenario	Ambiguous words predicted by Bigram	Ambiguous words predicted by Trigram
1	8911	9189
2	9455	9725
3	9612	9983
4	9519	9828
5	8936	9313
6	8228	8464
7	7711	7826
8	8691	8854
9	8264	8476
10	7772	7992
Total	87099	89650

Based on the results in Table 4, of all the ambiguous words contained in the corpus, not all are labeled correctly. There are also ambiguous words that are incorrectly labeled. Ambiguous word labeling is of two types that is ambiguous word labeling with exact Bigram results and incorrect Trigrams and ambiguous word labeling with incorrect Bigram results and exact Trigrams. The example of sentences from ambiguous word labeling of the first type in the corpus contained in the 6905th sentence, namely there is an ambiguous word “*agama*” or “religion”. This word can be categorized as an noun (NN tag) or an proper noun (NNP tag). In the context of the sentence in the example of the 6905th sentence, the word “*agama*” has an NN tag. The ambiguous word “*agama*” can be predicted correctly because the highest probability is 0.14348388496879 for the NN tag based on the calculation using MEMM Bigram. Meanwhile, the ambiguous word “*agama*” is predicted to be incorrect because the highest probability is 0.16282123413471 which is owned by the NNP tag based on the results of calculations using the MEMM Trigram.

The example of sentences from ambiguous word labeling of the second type in the corpus contained in the 9325th sentence, which contains the ambiguous word “*informasi*” or “information”, which can be categorized as an NN tag or an NNP tag. In the context of the sentence in the example of 9325th sentence, the word “*informasi*” has an NN tag. The ambiguous word “*informasi*” cannot be predicted with accuracy because the highest probability of 0.18027414056131 is owned by the NNP tag based on the results of calculations using MEMM Bigram. Meanwhile, the ambiguous word “*informasi*” is predicted precisely because the highest probability is 0.2361516773282 owned by the NN tag based on the results of calculations using the MEMM Trigram.

For example namely the labeling of the ambiguous word Bigram which is predicted to be correct, while the Trigram is predicted to be incorrect and vice versa is one of the drawbacks of this study, there is no maximum entropy feature that is used specifically to distinguish NN and NNP tags so that the system is difficult to distinguish between the two tags. In addition, this is also because the tags in the dataset are imbalanced, such as the “NN” tag has a total of 55,575 words, and the “UH” tag has a small number of words, which is 29 words. This imbalanced dataset affects the inaccuracy problem the probability value obtained by each word based on the calculation of MEMM Bigram and MEMM Trigram.

The difference between NN and NNP tags is NN tag indicate nouns in general and the writing is not capitalized unless it is at the beginning of the sentence. Meanwhile, the NNP tag shows specific nouns and the writing uses capital letters [33]. For example, the application of the maximum entropy feature to differentiate NN and NNP tags is [1].

$$f_1(c, x) \begin{cases} 1 & \text{if } w_i \text{ is lower_case } C = NN \\ 0 & \text{otherwise} \end{cases}$$

$$f_2(c, x) \begin{cases} 1 & \text{if } w_i \text{ is upper_first } C = NNP \\ 0 & \text{otherwise} \end{cases}$$

Another problem that has not been fully resolved is phrases that have two words (multi-words), for the example: “*pesta olahraga*” or “Games”. This study still uses the usual tokenization process, namely separating words in each sentence into separate tokens. Then the tokenization is continued by removing the spaces for each token to make multi-words into a single word. To solve this multi words problem, it should be necessary to add a multiword expressions (MWE) tokenizer. MWE tokenizer itself will separate data on documents that are predicted to be MWE into separate tokens, so there is no need to remove spaces in each token.

Evaluation of system performance in this study improves the result compared to previous studies. This study also has advantages in shaping the model. The modeling in this study is quite easy through the stochastic tagger method. The model of the stochastic tagger method can be obtained by conducting training data on training data. The model that is formed can be used immediately to test the testing data. Another advantage of doing POS tagging using the MEMM algorithm is that it offers a variety of multi-feature representations so that it can form various maximum entropy functions to get the best accuracy results. The results of accuracy using MEMM in first evaluation reached 83.04% (Bigram) and 86.66% (Trigram) better than previous studies which used the same data [20]. The previous study using HMM Bigram-viterbi and HMM Trigram-viterbi only produce accuracy rates of 77.56% and 61.67% [8] and another previous study using rule-based methods produce the accuracy rate of 79% [6]. It proves that POS tagging using MEMM has advantages over the methods used previously. In addition, previous studies [8] have not used the second and third evaluation scenarios as used in this study. This is the advantage of this research because it evaluates using artificial testing data outside the corpus and calculating the number of ambiguous words in the corpus predicted accurately.

This research also has advantages in using a large number of datasets, which consist of 10,000 sentences and 256,682 tokens. The number of datasets is greater than the dataset used in the study [12] which only uses 100,720 tokens and 4,325 sentences. But the accuracy produced in this study is still below previous research [12] because previous studies used deep learning methods and this research used machine learning methods. The use of deep learning methods will usually improve accuracy results better than machine learning methods. Therefore, the future research is expected to use deep learning methods for the development of POS tagging.

4. CONCLUSION

Based on the research, it can be concluded that the research carried out succeeded in building an Indonesian POS tagger called “Indonesian manually tagged corpus” using the MEMM Bgram algorithm and the MEMM Trigram algorithm. The Indonesian corpus used consists of 10,000 sentences that have been given the POS tag manually, then the corpus is processed using MEMM Bigram and MEMM Trigram to get the entropy value. From the research results obtained, it is generally proven that using the MEMM method has advantages over the methods used previously which used the same data. This paper improves a performance evaluation of research previously. The resulting average accuracy is 83.04% for the MEMM Bigram algorithm and 86.66% for the MEMM Trigram in first evaluation. Meanwhile, in the second evaluation using testing data outside the corpus, the results obtained accuracy of 93.85% using MEMM Bigram and 94.17% using MEMM Trigram. In the third evaluation for ambiguous words, the accuracy results are 94.83% using MEMM Bigram and 97.60% using MEMM Trigram. From the results obtained, it can generally be concluded that the Trigram MEMM algorithm is better than the Bigram MEMM algorithm because the Trigram MEMM perplexity value has a lower value than the perplexity value in the MEMM Bigram. The problem of inaccuracy in POS tagging in this paper is influenced by the probability value obtained by each word based on the calculation of MEMM Bigram and MEMM Trigram. The suggestion for future research is the utilization of deep learning methods for the development of POS tagging and build a maximum entropy function that affects the distribution of the number of tags in the corpus. Example of making the maximum entropy feature as a function of distinguishing NN and NNP tags. In addition, tokenizer development also needs to be done to deal with words with phrases of more than one word or multi-word problems. To solve the multi-word problem, it should be necessary to add a MWE tokenizer, because using the maximum entropy markov model only, the NN and NNP tags are still biased so that they

cannot be distinguished significantly. But using the maximum entropy markov model has solved the problem of HMM where HMM can only calculate the possible observation words conditioned on the tag.




REFERENCES

- [1] D. Jurafsky and J. H. Martin, "Speech and language processing: An introduction to natural language processing," *Speech and Language Processing an Introduction to Natural Language Processing Computational Linguistics and Speech Recognition*, pp. 1–18, 2001.
- [2] X. Xue and J. Zhang, "Part-of-speech tagging of building codes empowered by deep learning and transformational rules," *Advanced Engineering Informatics*, vol. 47, Art. no. 101235, Jan. 2021, doi: 10.1016/j.aei.2020.101235.
- [3] L. Settipalli, R. Vedantham, and A. Chopparapu, "Ambiguity level assessment for large corpus using Embedded POS tagger with DNFC parameter," *International Journal of Pure and Applied Mathematics*, vol. 118, no. 7, pp. 507–511, 2018.
- [4] A. D. W. Khan, J. A. Nasir, T. Amjad, S. Arafat, N. Aljohani, and F. S. Alotaibi, "Urdu part of speech tagging using conditional random fields," *Language Resources and Evaluation*, vol. 53, pp. 331–362, 2019, doi: 10.1007/s10579-018-9439-6.
- [5] E. Alayaboosar, A. Moloodi, and M. Kouhestani, "Word sense disambiguation focusing on POS tag disambiguation in persian: A rule-based approach," *International Journal of Information Science and Management*, vol. 17, no. 2, pp. 119–134, 2019.
- [6] F. Rashel, A. Luthfi, A. Dinakaramani, and R. Manurung, "Building an Indonesian rule-based part-of-speech tagger," in *2014 International Conference on Asian Language Processing (IALP)*, Oct. 2014, pp. 70–73, doi: 10.1109/IALP.2014.6973521.
- [7] F. Pisceldo, M. Adriani, and R. Manurung, "Probabilistic part of speech Tagging for Bahasa Indonesia," in *Proceedings of the 3rd International MALINDO Workshop, collocated event ACL-IJCNLP*, 2009, May.
- [8] D. E. Cahyani and M. J. Vindiyanto, "Indonesian part of speech Tagging using hidden Markov model Ngram & Viterbi," in *2019 4th International Conference on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, Nov. 2019, pp. 353–358, doi: 10.1109/ICITISEE48480.2019.9003989.
- [9] A. F. Wicaksono and A. Purwarianti, "HMM based part-of-speech tagger for Bahasa Indonesia. In fourth international MALINDO workshop, Jakarta," *4th International MALINDO (Malaysian-Indonesian Language) Workshop*, 2010.
- [10] D. Handrata, C. N. Purwanto, F. H. Chandra, J. Santoso, and Gunawan, "Part of speech Tagging for Indonesian language using bidirectional long short-term memory," *2019 1st International Conference on Cybernetics and Intelligent System, ICORIS 2019*, no. August, pp. 85–88, 2019, doi: 10.1109/ICORIS.2019.8874871.
- [11] C. A. Bahecevan, E. Kutlu, and T. Yildiz, "Deep neural network architecture for part-of-speech Tagging for Turkish language," *UBMK 2018-3rd International Conference on Computer Science and Engineering*, pp. 235–238, 2018, doi: 10.1109/UBMK.2018.8566272.
- [12] G. Prabha, P. V. Jyothsna, K. K. Shahina, B. Premjith, and K. P. Soman, "A deep learning approach for part-of-speech Tagging in Nepali language," in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Sep. 2018, pp. 1132–1136, doi: 10.1109/ICACCI.2018.8554812.
- [13] A. Ratnaparkhi, "A maximum entropy model for part-of-speech Tagging," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pp. 133–142, 1996.
- [14] S. Mammadov, S. Rustamov, A. Mustafali, Z. Sadigov, R. Mollayev, and Z. Mammadov, "Part-of-speech Tagging for Azerbaijani language," in *2018 IEEE 12th International Conference on Application of Information and Communication Technologies (AICT)*, Oct. 2018, pp. 1–6, doi: 10.1109/ICAICT.2018.8747154.
- [15] D. L. Cing and K. M. Soe, "Improving accuracy of part-of-speech (POS) tagging using hidden markov model and morphological analysis for Myanmar Language," *International Journal of Electrical and Computer Engineering*, vol. 10, no. 2, 2020, doi: 10.11591/ijece.v10i2.pp2023-2030.
- [16] W. AlKhawter and N. Al-Twairesh, "Part-of-speech tagging for Arabic tweets using CRF and Bi-LSTM," *Computer Speech and Language*, vol. 65, Art. no. 101138, Jan. 2021, doi: 10.1016/j.csl.2020.101138.
- [17] J. Xiao, X. Wang, and B. Liu, "The study of a nonstationary maximum entropy Markov model and its application on the pos-tagging task," *ACM Transactions on Asian Language Information Processing*, vol. 6, no. 2, Art. no. 7, Sep. 2007, doi: 10.1145/1282080.1282082.
- [18] A. Sharma and V. Yadav, "Approaches to part of speech tagging in Hindi language: A review," *International Journal of Advanced Science and Technology*, vol. 29, no. 5, pp. 283–291, 2020.
- [19] M. Bezoui, "Speech recognition of Moroccan dialect using hidden Markov models," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 1, pp. 7–13, Mar. 2019, doi: 10.11591/ijai.v8.i1.pp7-13.
- [20] A. Dinakaramani, F. Rashel, A. Luthfi, and R. Manurung, "Designing an Indonesian part of speech tagset and manually tagged Indonesian corpus," in *2014 International Conference on Asian Language Processing (IALP)*, Oct. 2014, pp. 66–69, doi: 10.1109/IALP.2014.6973519.
- [21] D. D. Palmer, *Handbook of natural language processing*. CRC Press, 2000.
- [22] D. Virmani and S. Taneja, "A text preprocessing approach for efficacious information retrieval," in *Advances in Intelligent Systems and Computing*, pp. 13–22, 2019.
- [23] A. McCallum, D. Freitag, and F. C. N. Pereira, "Maximum entropy Markov models for information extraction and segmentation," in *Proceedings of the Seventeenth International Conference on Machine Learning*, pp. 591–598, 2020.
- [24] C. Lv, D. Pan, Y. Li, J. Li, and Z. Wang, "A novel Chinese entity relationship extraction method based on the bidirectional maximum entropy markov model," *Complexity*, vol. 2021, pp. 1–8, Jan. 2021, doi: 10.1155/2021/6610965.
- [25] D. MacKay, "Information theory, inference, and learning algorithms," *IEEE Transactions on Information Theory*, vol. 50, no. 10, pp. 2544–2545, Oct. 2004, doi: 10.1109/TIT.2004.834752.
- [26] H. Walia, A. Rana, and V. Kansal, "Comparative analysis of different classifiers for case based model in Punjabi word sense disambiguation," *Investigacion Operacional*, vol. 41, no. 2, pp. 273–288, 2020.
- [27] H. Wang, H. Fei, Q. Yu, W. Zhao, J. Yan, and T. Hong, "A motifs-based maximum entropy markov model for realtime reliability prediction in system of systems," *Journal of Systems and Software*, vol. 151, pp. 180–193, May 2019, doi: 10.1016/j.jss.2019.02.023.
- [28] P. Refaeilzadeh, L. Tang, and H. Liu, "Cross-Validation," in *Encyclopedia of Database Systems*, L. LIU and M. T. ÖZSU, Eds. Boston, MA: Springer US, pp. 532–538, 2009.
- [29] H. Zhou, "Cross-validation and ROC," in *Learn Data Mining Through Excel*, Berkeley, CA: Apress, pp. 67–81, 2020.
- [30] K. Srinivasan, A. K. Cherukuri, D. R. Vincent, A. Garg, and B. Y. Chen, "An efficient implementation of artificial neural networks with K-fold cross-validation for process optimization," *Journal of Internet Technology*, vol. 20, no. 4, pp. 1213–1225,




- 2019.
- [31] A. Baratloo, M. Hosseini, A. Negida, and G. El Ashal, "Part 1: Simple definition and calculation of accuracy, sensitivity and specificity," *Emergency (Tehran, Iran)*, vol. 3, no. 2, pp. 48–49, 2015.
- [32] R. Vulanović and T. Mosavi Miangah, "A comparison of the accuracy of parts-of-speech Tagging systems based on a mathematical model," *Journal of Quantitative Linguistics*, vol. 26, no. 3, pp. 256–265, Jul. 2019, doi: 10.1080/09296174.2018.1474517.
- [33] S. Raharjo, R. Wardoyo, and A. E. Putra, "Detecting proper nouns in Indonesian-language translation of the Quran using a guided method," *Journal of King Saud University-computer and Information Sciences*, vol. 32, no. 5, pp. 583–591, Jun. 2020, doi: 10.1016/j.jksuci.2018.06.009.

BIOGRAPHIES OF AUTHORS



Denis Eka Cahyani    holds a Bachelor of Computer Science (S. Kom.) in Computer Science, Master of Computer Science (M.Kom.) in Computer Science, besides several professional certificates and skills. She is currently lecturing with the department of Mathematics at Universitas Negeri Malang, Malang, Indonesia. She is a member of the Engineers and the Institute of Electrical and Electronics Engineers (IEEE) Indonesia Section. Her research areas of interest include Natural Language Processing, Artificial Intelligent and Data Science. She can be contacted at email: denis.eka.cahyani.fmipa@um.ac.id.



Winda Mustikaningtyas    holds a Bachelor of Informatics degree from Sebelas Maret University, Indonesia in 2020. She is a junior analyst at Grup Pengembangan Aplikasi Sistem Informasi OJK, Jakarta, Indonesia. The project currently being developed is the Non-Bank Financial Industry Supervision Information System in supervisory unit. She can be contacted at email: windaamustikaningtyas@gmail.com

Effective predictive modelling for coronary artery diseases using support vector machine

Kuncahyo Setyo Nugroho, Anantha Yullian Sukmadewa, Angga Vidianto, Wayan Firdaus Mahmudy

Department of Informatics Engineering, Faculty of Computer Science, Brawijaya University, Malang, Indonesia

Article Info

Article history:

Received Jun 21, 2021

Revised Dec 24, 2021

Accepted Jan 2, 2022

Keywords:

Coronary artery disease prediction

Effective support vector machines

Good support vector machines parameters

ABSTRACT

Coronary artery disease (CAD) is a category of cardiovascular disease that causes the highest mortality rate in the world. CAD occurs due to plaque build-up on the walls of the arteries that supply blood to the heart and other organs of the body. To control the mortality rate, a practical model that is capable of predicting CAD is needed. Machine learning approaches have been used in solving various problems in various domains, including biomedicine. However, real-world data often has an unbalanced class distribution that can interfere with classifier performance. In addition, data has many features to process. This study focuses on effective modeling capable of predicting CAD using feature selection to handle high dimensional data and feature resampling to handle unbalanced data. Feature selection is very effective by eliminating irrelevant features from the training data. Hyperparameter tuning is also done to find the best combination of parameters in support vector machines (SVM). Our results show that the SVM cross-validated ten times has a more accurate training result. Furthermore, the grid search on SVM cross-validated ten times had more accurate training model results and achieved 88% accuracy on the test data.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Kuncahyo Setyo Nugroho

Department of Informatics Engineering, Faculty of Computer Science, Brawijaya University

Jalan Veteran, No. 8, Lowokwaru, Malang, East Java 65145, Indonesia

Email: ksnugroho26@gmail.com

1. INTRODUCTION

The heart is an essential organ in the cardiovascular system that requires a supply of blood that contains oxygen. The coronary circulation supplies blood to the heart. The aorta divides into two major coronary arteries, each of which branches off into smaller arteries and supplies blood to the entire heart muscle [1]. Cardiovascular disease (CVD) is a group of diseases. Coronary heart disease (CHD), coronary artery disease (CAD), and acute coronary syndrome (ACS) are included in CVD [2]. CAD happens because of plaque build-up in the wall of arteries that supply blood to the heart and other parts of the [3]. The condition known as atherosclerosis occurs when plaque begins to accumulate in these arteries. As the plaque hardens, the coronary arteries narrow, decreasing the blood supply to the heart. A blood clot on the plaque's surface may occur if it ruptures. In the majority of situations, a big blood clot can totally stop the coronary arteries' blood flow. A heart attack, if left untreated, can result in major health consequences and, in the worst-case scenario, death. As a result, cardiovascular disease is the leading cause of death globally [4].

There is an adequate need for the early detection of patients with CAD. A machine learning approach can solve problems in the biomedical domain [5]–[8]. Machine learning gives the computer ability to learn and improve from experience automatically. Machine learning algorithms have several major categories based on their learning approach, input and output data, and problem type: supervised,

unsupervised, and reinforcement learning [9]. Support vector machines (SVMs) used in supervised learning have been shown to be extremely effective at solving classification problems in a variety of biomedical fields [6], [10]–[12].

There are many studies conducted to diagnose CAD with machine learning in recent years. The most widely used dataset in CAD diagnosis is the Z-Alizadehsani dataset [13]. Using this dataset, [14] applied data mining techniques to diagnose CAD based on the symptoms and characteristics of the patient's ECG. Their research used sequential minimal optimization (SMO) and naïve bayes (NB) classifier and a combination of both to diagnose CAD. Testing with 10-fold cross-validation shows that the combination of SMO-naïve bayes is superior by achieving more than 88.52% accuracy than SMO of 86.95% and naïve bayes of 87.22%. In another study, [15] using SMO, naïve bayes, bagging with SMO, and neural network to diagnose the same disease. Information gain is used to determine which features are most effective for diagnosing CAD. As a result, SMO with information gain obtained the best performance with an accuracy of 94.08%.

Alizadehsani *et al.* used the feature selection technique used in NB, C4.5, and SVM to diagnose CAD. Using 10-fold cross-validation, SVM has the highest accuracy of 96.40% [13]. To increase accuracy, [16] used random trees (RT), decision tree (DT), SVM, and chi-squared automatic interaction detection (CHAID) to select features based on predefined criteria for CAD diagnosis. Random trees are the best method by selecting 40 significant features and bringing out an accuracy of 91.47% [17] using hybrid PCA, DT, and firefly optimization techniques to optimize the accuracy of existing models. The PCA algorithm is used to extract features, the firefly optimization technique is used to optimize the feature selection, and DT is used to classify the data. They achieve 93% accuracy with a low classification error rate also low false positive and negative rates.

Other studies have also been conducted on the prediction of disease in the biomedical field. SVM is used to predict diabetes and pre-diabetes [10]. The SVM model is used to identify characteristics that best classify individuals into different diabetes subtypes. Their model got 83.47% for detection of diagnosed diabetes or diagnosed diabetes compared to [18] model that got 82.1%. In this research, they conclude SVM is a promising model for detecting a complex disease using common and simple variables. According to [11], [19] SVM has superior accuracy when predicting heart disease, diabetes, and parkinson's disease. SVM was also reported to obtain better accuracy than random forest in breast cancer prediction [20].

Machine learning assuming a balanced number of instances in each class. When using unbalanced data can lead to inaccurate model prediction results. Synthetic minority oversampling technique (SMOTE) and adaptive synthetic (ADASYN) sampling are alternatives to overcome unbalanced data by creating synthetic data in the minority class [21]. SMOTE, which is integrated with the prediction model is reported to improve the prediction model's performance [22], [23]. In CAD prediction, SMOTE on artificial neural networks, DT, and SVM showed an increase in the accuracy obtained from the original data [24]. Meanwhile, [25] using ADASYN with SVM to diagnose Parkinson's disease effectively. Both studies do not employ feature selection to determine the most essential features for output prediction.

Based on previous studies, a combination of feature selection and feature resampling in CAD prediction has never been done before. Both techniques are reported to improve the performance of the resulting model. The main contribution of this study is that we propose a framework for building an effective model using feature selection and feature resampling in CAD predictions. Feature selection is used to find the most relevant features to CAD predictions. While handling imbalanced data, we reviewed several feature resampling. We use SVM with hyperparameter tuning to find the combination of parameters to make an effective CAD prediction.

2. RESEARCH METHOD

There are four main steps to complete this research, as shown in Figure 1. The first step is data exploration, followed by data preprocessing. Next, we use feature selection to determine which features have the most importance on the target variable. After identifying the relevant features, the dataset is divided into training and testing sets for the purpose of implementing multiple machine learning algorithms. The last step is model evaluation. This section discusses the processes and procedures involved in doing this research.

2.1. Dataset description

We use the Z-Alizadehsani dataset downloaded from the UCI machine learning repository. The dataset contains records of 303 patients who visited the Shaheed Rajaei Cardiovascular, Medical, and Research Center in Iran. Each patient has 54 features to diagnose CAD. These features are grouped into four categories: demographic, symptom and examination, electrocardiogram (ECG), and laboratory and echo features. Patients are categorized as having CAD if they experience stenosis in one of their coronary arteries

more than or equal to 50%. A total of 216 patients in the dataset had this disease, while the rest were normal patients. This shows that the dataset has an unbalanced class distribution. The target feature on the dataset is cath with a CAD value for patients with coronary artery disease and normal for normal patients.

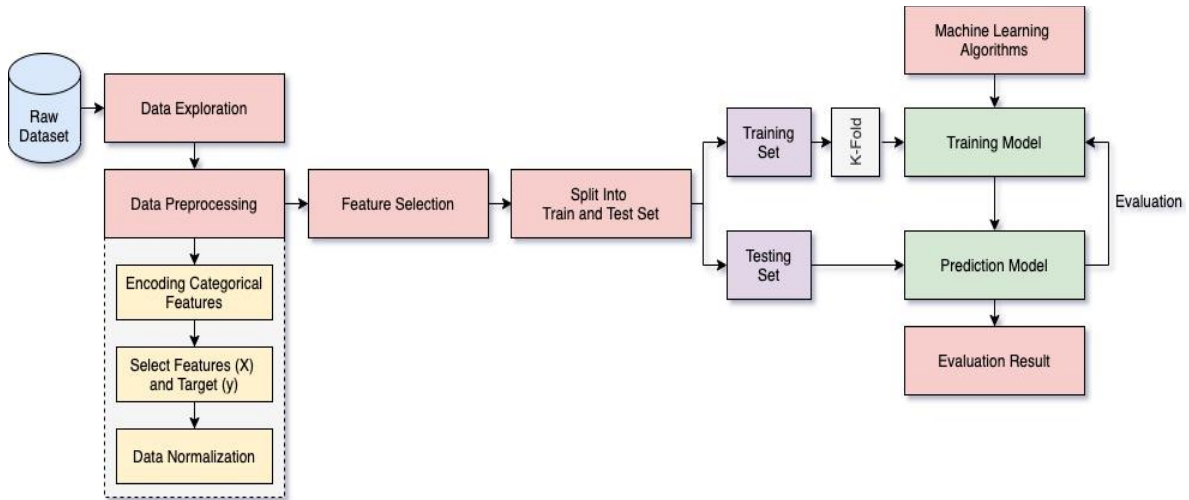


Figure 1. Research method design for CAD prediction

2.2. Data exploration

Our dataset has many diverse features, so this step is taken to explore the dataset to get useful insights through visualization and data analysis. This step also helps us find out the missing values and identify the types of numeric features and categorical features in the dataset.

2.3. Data preprocessing

Real-world datasets have incomplete, inconsistent, and even have missing value on specific features. Data preprocessing used to clean and format the raw data in the dataset so that machine learning algorithms can easily represent the feature set. In this study, we implemented several data preprocessing steps. The first step is to convert categorical features to numeric values because machine learning algorithms can only read and process numeric values. Next, we create a feature matrix that is used as the input variable and the target variable. The input feature is stored into X variable while the target feature is stored into Y variable. The final step is normalizing the data to rescale the numeric features into ranges 0 and 1 used a min-max scaler, as shown in (1).

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

2.4. Feature selection

The features used to train machine learning algorithms have a significant impact on the performance of the final model. Irrelevant features can have a negative impact on the resulting model [26]. To identify features that affect the target variable, feature selection can be used. Feature selection is the process of reducing the number of features in a dataset in order to improve the model's performance [27]. We used feature selection to predict which features were most important in influencing patients with CAD or not. extremely randomized trees classifier is an ensemble learning type used for feature selection. In this method, each decision tree is generated from the training sample. Then, at each test node, the decision tree is given a random sample of k features of all features, where each decision tree must choose the best feature to separate the data based on the Gini Index value. This random feature will provide several uncorrelated decision trees. This value is referred to as the feature's Gini Importance. To make the feature selection process easier, each feature is graded according to its Gini Importance.

2.5. Data separation

Data separation is used to evaluate machine learning algorithms' performance when predicting data that was not used to train the model. Divide the dataset into two subsets using the data separation process.

The first subset is utilized to train machine learning algorithms in order to generate prediction models. The second subset is the test set on which the prediction model is evaluated. We trained on 75% of the dataset and tested the model on the remaining 25%.

2.6. Stratified k-fold

When performing the data separation procedure, the main problem must be enough data to divide the dataset into training data and test data as data representations following the problem domain. Therefore, this procedure is not suitable for evaluating model performance if there are few datasets available. There will not be enough data on the training or testing subset for the model to learn the effective mapping from input to output. Prediction performance can be too optimistic (good prediction) or too pessimistic (bad).

An alternative method that can be used if do not have enough data is the K-Fold procedure by folding K as much data and repeating the process as many as K as well. One type of this procedure is a Stratified K-fold, as shown in Figure 2. Stratified K-Fold is helpful if the available dataset is few and has an unbalanced class distribution. We want to maintain the class imbalance to represent some information about what the model is trying to predict. In this study, we use a combination of the Stratified K-Fold procedure to conduct a final evaluation of the performance of the implemented model. After separated the training and testing set in the previous steps, we further divided the training set into validation set to validate the machine learning algorithms performance during the k-fold iteration process. We also performed feature resampling to balance the distribution of classes in the training set during this process. The generated prediction model is finally tested using the testing set as the final result of the predicted performance.

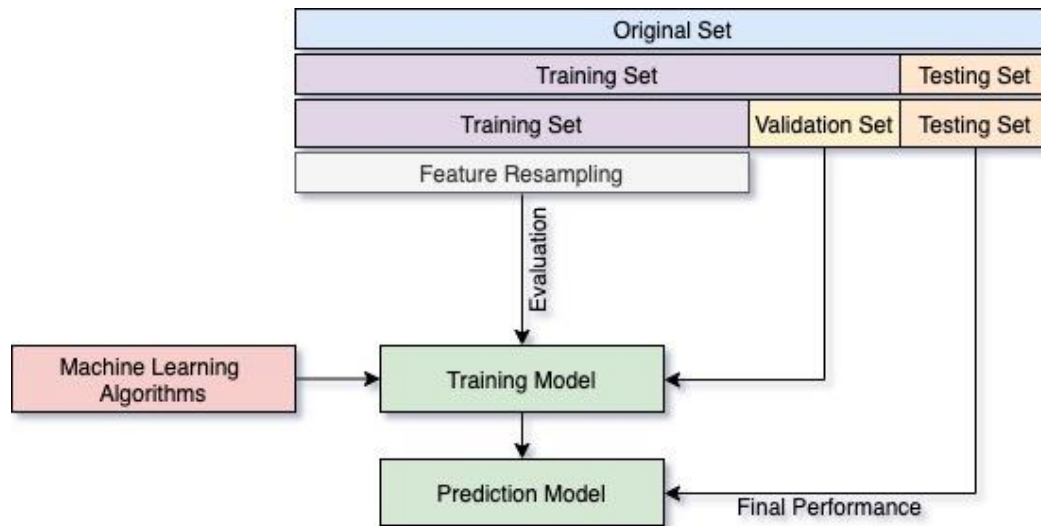


Figure 2. Stratified k-fold schema

2.7. Feature resampling

Imbalance data causes the model to be biased in choosing the majority class. There are many ways to handle a dataset with an unbalanced class distribution, including collecting more data, trying variations in machine learning algorithms, using both oversampling and undersampling techniques. Collecting more data is impossible because it requires more time and costs, while undersampling techniques can cause the loss of important information in the dataset. Therefore, in this study, we focus on oversampling techniques SMOTE and ADASYN so that we hope not to lose any information that might be valuable in the dataset. SMOTE uses a k-nearest neighbors (k-NN)-based distance approach to create synthetic data [28]. First, the data is randomly selected from the minority class, then K is the closest neighbor of the data. Synthetic data is generated between randomly selected and K-nearest data. This step is repeated until the minority class has the same proportion as the majority class. Meanwhile, ADASYN is a variation of SMOTE by creating synthetic data based on data density [29]. The synthetic data generated will be inversely proportional to the density of the minority class. That is, more synthetic data is generated in the feature space where the density of the minority class is low, and less or even less synthetic data is generated in the high density minority class [21].

2.8. Support vector machine

The SVM is a highly effective classification algorithm by finding decision boundaries known as hyperplanes. The optimal hyperplane separates the instances correctly into each class. The margins on the optimal hyperplane and instances in training are maximized to fit the data. The SVM model is not delicate to other information focuses. Its point is to track down the best division line, for example, the ideal hyperplane between the two classes of tests, to have the most significant distance conceivable to every one of the two classes of help vectors. The separator line dictates the indicator include for each prescient class. Figure 3 shows the vector machine in 2-dimensional space [16].

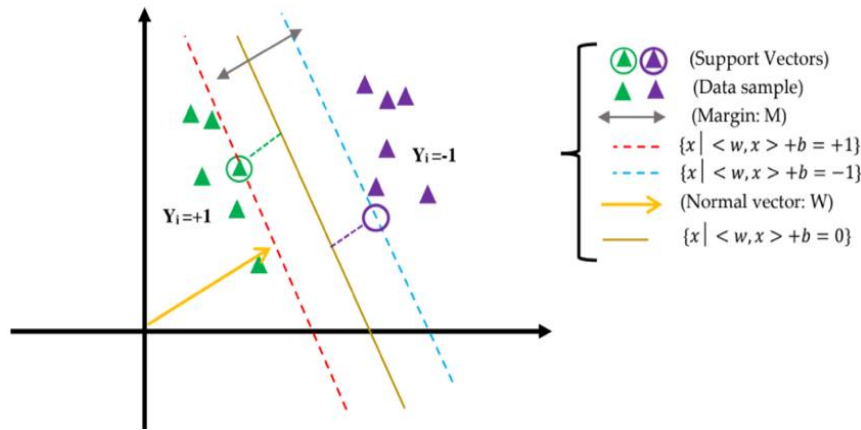


Figure 3. SVM in 2-dimensional space [16]

A hyperplane with a wider margin is projected to be more accurate than one with a smaller margin when classifying future data. Therefore, the hyperplane with the largest margin will be searched for. The function has the following in (2) [30].

$$y(x) = \text{sgn}[\sum_{i=1}^m \alpha_i y_i(x_i, x) + b] \tag{2}$$

However, in (2) can be applied if the sample data used can be separated linearly. Kernel methods enable the transformation of data into huge dimensions for classification challenges. As is the case with data samples that cannot be split linearly, the kernel function converts the data to a higher-dimensional space without actually changing it to that space. In (3) can be applied when the data sample situation cannot be separated linearly.

$$y(x) = \text{sgn}[\sum_{i=1}^m \alpha_i y_i K(x_i, x) + b] \tag{3}$$

The kernel function $K(x_i, x)$ is equals to (x_i, x) and x is the non-linear space from the original space to high dimensional space. Where, r and d are kernel parameters, and the four basic kernels are given as follows in (4)-(7). In this study, we use all kernels to find the best SVM performance.

$$\text{Linear: } K(x_i, x_j) = x_i^T x_j \tag{4}$$

$$\text{Polynomial: } K(x_i, x_j) = (\gamma x_i^T x_j + r)^d, \gamma > 0 \tag{5}$$

$$\text{Radial Bias Function (RBF): } K(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2), \gamma > 0 \tag{6}$$

$$\text{Sigmoid: } K(x_i, x_j) = \tanh(\gamma x_i^T x_j + r) \tag{7}$$

2.9. Evaluation metrics

The model generated during the training phase is used to obtain predictive results from population data. In the classification, the confusion matrix describes the model's performance by calculating which classes are predicted correctly and incorrectly and what types of errors are made. True positive (TP) is

defined as positive instances that are predicted to be true. For example, a patient with CAD is predicted to have true CAD. True negative (TN) is defined as negative instances that are predicted to be true. For example, a patient who does not have CAD is predicted not to have CAD. False positive (FP) is negative instances that are predicted as positive instances. For example, a patient who does not have CAD is predicted to have CAD. False negative (FN) is positive instances that are predicted as negative instances. For example, a patient who has CAD is predicted not to have CAD.

The most frequently used performance metric based on the confusion matrix for classification is accuracy. Accuracy is the ratio of true predictions (TP and TN) with the overall data that describes the level of closeness of the predicted value to the actual value, as shown in (8). In the training phase, the model's accuracy is obtained from the average of each fold in the cross-validation. The standard deviation was also calculated to see the variance. The problem with unbalanced data is negative instances with the majority class and positive instances with fewer classes. To interpreting the model performance with unbalanced data, receiver operating characteristic (ROC) curve are used. The ROC curve is obtained from the true positive rate (TPR) as in (9) and the false positive rate (FPR) as in (10).

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \quad (8)$$

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

$$FPR = \frac{TN}{TN+FP} \quad (10)$$

3. RESULTS AND DISCUSSION

Based on the research framework in Figure 1, the first step we take is preprocessing by changing all categorical features in the dataset to numeric values and normalizing them using the minmax scaler. Next, we use feature selection to determine which features are significant and have an effect on the target variable. We choose features using the additional trees classifier. From the results of feature selection, we found that there are 16 features that are correlated with the target variables shown in Table 1. In each subsequent test, we compare the model's performance using all the features and feature selection results. We wanted to find out if feature selection could improve the performance of a given model.

Table 1. The features used are based on the selection results

Feature name	Feature category
Age	Demographic
Weight	Demographic
BMI	Demographic
HTN	Demographic
BP	Symptom and examination
Typical chest pain	Symptom and examination
Atypical	Symptom and examination
Nonanginal	Symptom and examination
Tinversion	ECG
FBS	Laboratory and echo
TG	Laboratory and echo
ESR	Laboratory and echo
Neut	Laboratory and echo
EF-TTE	Laboratory and echo
Region RMWA	Laboratory and echo

To evaluate the predictive model, we divided the dataset into 75% for the training set and the remainder for the test set. Next, the 10-layer cross-validation procedure we applied to the training set. In this procedure, the training set is randomly divided into ten sections, 9 folds are used to train the classification model and the remaining 1 fold is divided to validate the model. This procedure is performed 10 folds. The dataset character has class imbalance, the value of cross validation is a metric that can be used to evaluate the model because the folds are made with the same number of samples for each class so that the class distribution can be optimally balanced. This procedure can ultimately provide sufficient representation of the minority and majority classes in each group.

We compared the performance of the various number of classifications in CAD predictions on different unbalanced and balanced datasets. We choose SVM as our main model. We comparing the performance of the SVM model with several other methods such as k-NN, naïve bayes, and decision tree without any cross-validation procedures. The next experiment is to implement the resampling feature in the cross-validation procedure by creating synthetic data for the minority class. We used two different oversampling techniques, SMOTE and ADASYN. Unbalanced training data shown in Figure 4, while the balanced training data results are shown in Figures 5(a) and (b).

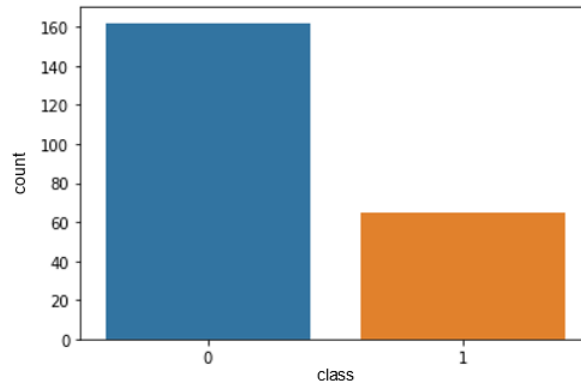


Figure 1. Unbalanced on training data

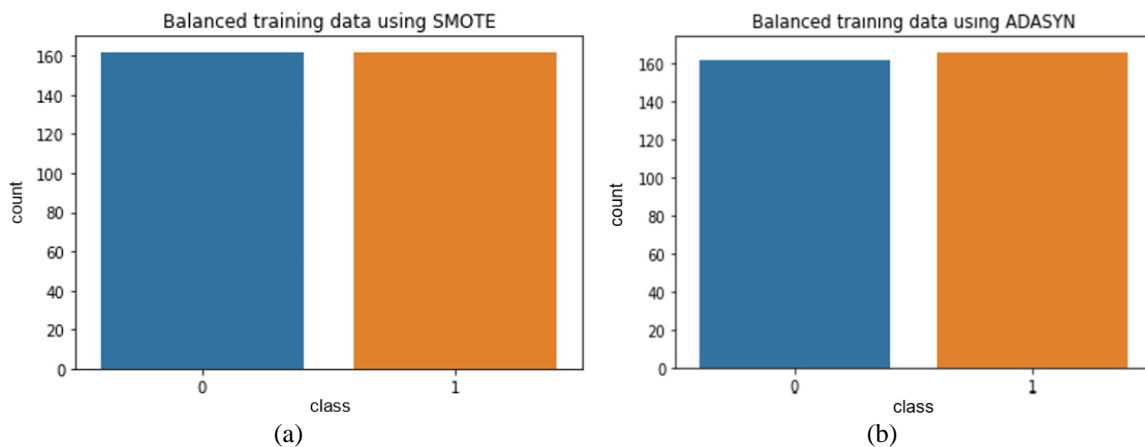


Figure 2. The result of feature resampling (a) using SMOTE and (b) using ADASYN

The majority of machine learning algorithms will not produce optimal results if the parameters are not properly specified. In order to build a good classification model, it is very important to select the parameters in a machine learning algorithm. Effective parameter initialization is situation-dependent, and each situation may require unique parameters. By specifying the appropriate parameters, the model can be guided to its optimal solution [31]. Parameter optimization is time consuming if done manually, especially since it has many parameters. The biggest problem in setting up an SVM model is choosing kernel functions and their parameter values [32], [33]. Incorrect parameter settings lead to poor classification results. Therefore, the last step we took was a grid search for hyper-parameters tuning to find the optimal SVM parameter. Hyper-parameters are defined using the minimum value, maximum value, and the number of steps. The parameters we are looking for are shown in Table 2. Machine learning algorithms are trained on imbalanced data using all features and from the feature selection results, as shown in Table 3.

Based on Table 3, SVM shows that it has the highest accuracy in the training and testing phases, so it is superior to all the models we use. The SVM accuracy obtained when using the full feature is better than using feature selection. This is consistent with research [10] that SVM produces better accuracy when using all features (high dimensional data). We see that our base model does a reasonable job of modeling the data. But when viewed, the accuracy value of training and testing has quite a difference. We want the training value to be the same or close to the test value. Therefore, our next experiment uses stratified k-fold to

validate the model. Based on Table 4, the average accuracy during the training phase decreases for the whole model. This shows that our model can predict the test data more accurately. Now we just focus only on our main model, the SVM. Next, we will perform feature resampling with SMOTE and ADASYN as an effort to balance the training data during the cross-validation procedure. We also perform model-based predictions on the test set. Based on Table 5, the SVM model provides good performance when using SMOTE. When using feature selection, training accuracy drops while testing accuracy goes up. This makes it possible for our model to more accurately predict test data. To improve the performance of the final model, we take hyperparameter-tuning to find the most optimal parameters in the SVM using features from the feature selection results and SMOTE in stratified k-fold. Based on grid-search, the best SVM model is obtained using the parameters C=1000, degree=1, gamma=0.001, kernel: RBF. By using cross-validation, the highest model accuracy obtained in the test set reached 88%, as shown in Table 6. Based on the confusion matrix in Figure 3, the model shows the number of TP=48 and TN=19. While the ROC curve in Figure 4 shows that the model has good performance because the curve is away from the baseline to the TPR axis. This means that the model classifies more data instances correctly.

Table 2. Grid search for hyper-parameters tuning to find optimal SVM parameters

SVM parameter	Value range
C	0.1, 1, 10, 100, 1000
Gamma	1, 0.1, 0.001, 0.001, 0.0001
Degree	1, 2, 3, 4, 5, 6
Kernel	Linear, Polynomial, RBF, Sigmoid

Table 3. Training and testing score several algorithm on the imbalanced dataset

Model	Full Feature		Feature Feature	
	Training	Testing	Training	Testing
SVM	0.942	0.855	0.925	0.815
k-NN	0.885	0.776	0.881	0.868
Naïve Bayes	0.894	0.828	0.867	0.815
Decision Tree	1.000	0.723	1.000	0.776

Table 4. Average cross-validation score trained on the imbalanced dataset

Model	Full feature	Feature feature
	Cross validation score	
SVM	0.872 ± 0.055	0.855 ± 0.072
k-NN	0.828 ± 0.037	0.842 ± 0.074
Naïve Bayes	0.854 ± 0.042	0.863 ± 0.059
Decision Tree	0.833 ± 0.041	0.797 ± 0.048

Table 5. SVM performance with feature resampling trained on the balanced dataset

	Full Feature			Feature Selection		
	Training	Cross Validation	Testing	Training	Cross Validation	Testing
SMOTE	0.942	0.859 ± 0.038	0.855	0.920	0.868 ± 0.091	0.881
ADASYN	0.942	0.849 ± 0.091	0.881	0.876	0.828 ± 0.105	0.815

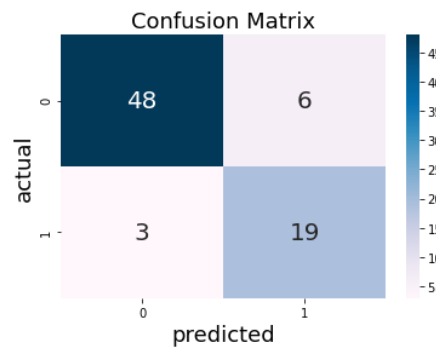


Figure 3. Confusion matrix of SVM balanced dataset with feature selection

Table 6. Hyper-parameter tuning for SVM performance in balanced dataset with feature selection

Training	Cross Validate	Testing
0.925	0.877 ± 0.05	0.881

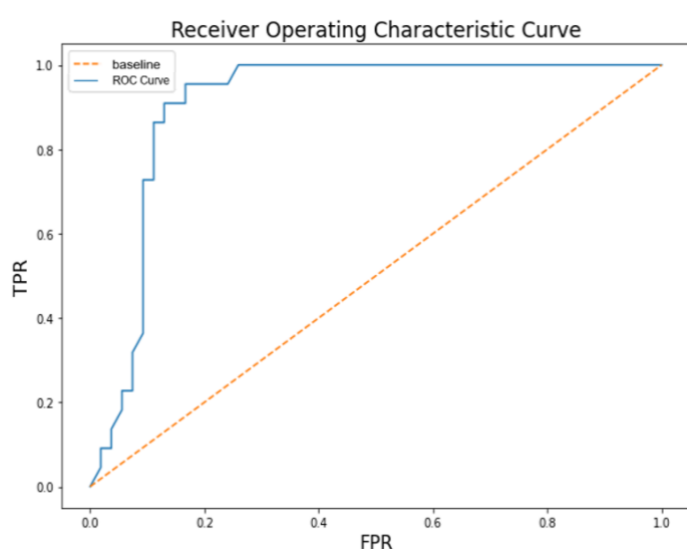


Figure 4. ROC curve of SVM balanced dataset with feature selection

4. CONCLUSION

This study has succeeded in making predictions for CAD using SVM. SVM has better performance than all tested algorithms, such as decision tree, k-NN, and naïve bayes. The main challenge of this research is to deal with unbalanced data and the many features in the dataset. Although the first testers without using feature selection and model resampling features have worked well for modeling the data, there is sufficient distance between the examiners and the training accuracy. Ideally, training and testing accuracy are relatively close. We use extra tree class-based feature selection to generate 16 updated features on the target variable. Our tests using random trees classifier for feature selection and SMOTE for feature resampling show that the best model performance is the testing accuracy of 88%.

REFERENCES





- [1] "Heart disease risk factors," *Texas Heart Institute*. <https://www.texasheart.org/heart-health/heart-information-center/topics/heart-disease-risk-factors> (accessed May 26, 2021).
- [2] F. Sanchis-Gomar, C. Perez-Quilis, R. Leischik, and A. Lucia, "Epidemiology of coronary heart disease and acute coronary syndrome," *Annals of Translational Medicine*, vol. 4, no. 13, pp. 256–256, Jul. 2016, doi: 10.21037/atm.2016.06.33.
- [3] "Coronary artery disease (CAD)," https://www.cdc.gov/heartdisease/coronary_ad.htm (accessed May 26, 2021).
- [4] H. Animesh, K. M. Subrata, G. Amit, M. Arkomita, and A. Mukherje, "Heart disease diagnosis and prediction using machine learning and data mining techniques: a review," *Advances in Computational Sciences and Technology*, vol. 10, no. 7, pp. 2137–2159, 2017.
- [5] R. Maheshwari, K. Moudgil, H. Parekh, and R. Sawant, "A machine learning based medical data analytics and visualization research platform," in *2018 International Conference on Current Trends towards Converging Technologies (ICCTCT)*, Mar. 2018, pp. 1–5, doi: 10.1109/ICCTCT.2018.8550953.
- [6] L. Muflikhah, N. Widodo, W. F. Mahmudy, and Solimun, "Prediction of liver cancer based on DNA sequence using ensemble method," in *2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, Dec. 2020, pp. 37–41, doi: 10.1109/ISRITI51436.2020.9315341.
- [7] A. Ridok, N. Widodo, W. F. Mahmudy, and M. Rifa'i, "A hybrid feature selection on AIRS method for identifying breast cancer diseases," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 11, no. 1, pp. 728–735, Feb. 2021, doi: 10.11591/ijece.v11i1.pp728-735.
- [8] S. Sumiati, H. Saragih, T. Abdul Rahman, and A. Triayudi, "Expert system for heart disease based on electrocardiogram data using certainty factor with multiple rule," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 43–50, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp43-50.
- [9] S. Sah, "Machine learning: a review of learning types," *Preprints*, Jul. 2020, doi: 10.20944/preprints202007.0230.v1.
- [10] W. Yu, T. Liu, R. Valdez, M. Gwinn, and M. J. Khoury, "Application of support vector machine modeling for prediction of common diseases: the case of diabetes and pre-diabetes," *BMC Medical Informatics and Decision Making*, vol. 10, no. 1, Dec. 2010, doi: 10.1186/1472-6947-10-16.
- [11] S. Uddin, A. Khan, M. E. Hossain, and M. A. Moni, "Comparing different supervised machine learning algorithms for disease prediction," *BMC Medical Informatics and Decision Making*, vol. 19, no. 1, Dec. 2019, doi: 10.1186/s12911-019-1004-8.
- [12] A. Ridok, N. Widodo, W. F. Mahmudy, and M. Rifa'i, "FC-SVM: DNA binding Proteins prediction with average blocks (AB)

Effective predictive modelling for coronary artery diseases using support ... (Kuncahyo Setyo Nugroho)




- descriptors using SVM with FC feature selection,” in *2019 International Conference on Sustainable Information Engineering and Technology (SIET)*, Sep. 2019, pp. 22–27, doi: 10.1109/SIET48054.2019.8986070.
- [13] R. Alizadehsani *et al.*, “Non-invasive detection of coronary artery disease in high-risk patients based on the stenosis prediction of separate coronary arteries,” *Computer Methods and Programs in Biomedicine*, vol. 162, pp. 119–127, Aug. 2018, doi: 10.1016/j.cmpb.2018.05.009.
- [14] R. Alizadehsani *et al.*, “Diagnosing coronary artery disease via data mining algorithms by considering laboratory and echocardiography features,” *Research in Cardiovascular Medicine*, vol. 2, no. 3, 2013, doi: 10.5812/cardiovasmed.10888.
- [15] R. Alizadehsani *et al.*, “A data mining approach for diagnosis of coronary artery disease,” *Computer Methods and Programs in Biomedicine*, vol. 111, no. 1, pp. 52–61, Jul. 2013, doi: 10.1016/j.cmpb.2013.03.004.
- [16] J. H. Joloudari *et al.*, “Coronary artery disease diagnosis; ranking the significant features using a random trees model,” *International Journal of Environmental Research and Public Health*, vol. 17, no. 3, Jan. 2020, doi: 10.3390/ijerph17030731.
- [17] Savita, G. Sharma, G. Rani, and V. S. Dhaka, “Efficient predictive modelling for classification of coronary artery diseases using machine learning approach,” *IOP Conference Series: Materials Science and Engineering*, vol. 1099, no. 1, p. 012068, Mar. 2021, doi: 10.1088/1757-899X/1099/1/012068.
- [18] K. E. Heikes, D. M. Eddy, B. Arondekar, and L. Schlessinger, “Diabetes risk calculator: a simple tool for detecting undiagnosed diabetes and pre-diabetes,” *Diabetes Care*, vol. 31, no. 5, pp. 1040–1045, May 2008, doi: 10.2337/dc07-1150.
- [19] K. Shankar, S. K. Lakshmanaprabu, D. Gupta, A. Maselena, and V. H. C. de Albuquerque, “Optimal feature-based multi-kernel SVM approach for thyroid disease classification,” *The Journal of Supercomputing*, vol. 76, no. 2, pp. 1128–1143, Feb. 2020, doi: 10.1007/s11227-018-2469-4.
- [20] C. Aroef, Y. Rivan, and Z. Rustam, “Comparing random forest and support vector machines for breast cancer classification,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 18, no. 2, Apr. 2020, doi: 10.12928/telkomnika.v18i2.14785.
- [21] K. Davagdorj, J. S. Lee, V. H. Pham, and K. H. Ryu, “A comparative analysis of machine learning methods for class imbalance in a smoking cessation intervention,” *Applied Sciences*, vol. 10, no. 9, May 2020, doi: 10.3390/app10093307.
- [22] N. Santoso, W. Wibowo, and H. Hikmawati, “Integration of synthetic minority oversampling technique for imbalanced class,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 13, no. 1, pp. 102–108, Jan. 2019, doi: 10.11591/ijeecs.v13.i1.pp102-108.
- [23] K. Davagdorj, J. S. Lee, K. H. Park, P. V. Huy, and K. H. Ryu, “Synthetic oversampling based decision support framework to solve class imbalance problem in smoking cessation program,” *International Journal of Applied Science and Engineering*, vol. 17, no. 3, pp. 223–235, 2020.
- [24] I. D. Apostolopoulos, “Investigating the synthetic minority class oversampling technique (SMOTE) on an imbalanced cardiovascular disease (CVD) dataset,” *International Journal of Engineering Applied Sciences and Technology*, vol. 4, no. 9, pp. 431–434, Jan. 2020, doi: 10.33564/IJEAST.2020.v04i09.058.
- [25] C. Taleb, M. Khachab, C. Mokbel, and L. Likforman-Sulem, “A reliable method to predict parkinson’s disease stage and progression based on handwriting and re-sampling approaches,” in *2018 IEEE 2nd International Workshop on Arabic and Derived Script Analysis and Recognition (ASAR)*, Mar. 2018, pp. 7–12, doi: 10.1109/ASAR.2018.8480209.
- [26] A. Ridok, W. F. Mahmudy, and M. Rifai, “An improved artificial immune recognition system with fast correlation based filter (FCBF) for feature selection,” in *2017 Fourth International Conference on Image Information Processing (ICIIP)*, Dec. 2017, pp. 1–6, doi: 10.1109/ICIIP.2017.8313761.
- [27] A. M. A. and P. A. Thomas, “Comparative review of feature selection and classification modeling,” in *2019 International Conference on Advances in Computing, Communication and Control (ICAC3)*, Dec. 2019, pp. 1–9, doi: 10.1109/ICAC347590.2019.9036816.
- [28] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: synthetic minority over-sampling technique,” *Journal of Artificial Intelligence Research*, vol. 16, pp. 321–357, Jun. 2002, doi: 10.1613/jair.953.
- [29] H. He, Y. Bai, E. A. Garcia, and S. Li, “ADASYN: adaptive synthetic sampling approach for imbalanced learning,” in *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, Jun. 2008, pp. 1322–1328, doi: 10.1109/IJCNN.2008.4633969.
- [30] I. C. Dipto, T. Islam, H. M. M. Rahman, and M. A. Rahman, “Comparison of different machine learning algorithms for the prediction of coronary artery disease,” *Journal of Data Analysis and Information Processing*, vol. 8, no. 2, pp. 41–68, 2020, doi: 10.4236/jdaip.2020.82003.
- [31] G. A. Fanshuri Alfarisy, W. F. Mahmudy, and M. H. Natsir, “Good parameters for PSO in optimizing laying hen diet,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 4, pp. 2419–2432, Aug. 2018, doi: 10.11591/ijece.v8i4.pp2419-2432.
- [32] I. Syarif, A. Prugel-Bennett, and G. Wills, “SVM parameter optimization using grid search and genetic algorithm to improve classification performance,” *TELKOMNIKA (Telecommunication Computing Electronics and Control)*, vol. 14, no. 4, pp. 1502–1509, Dec. 2016, doi: 10.12928/telkomnika.v14i4.3956.
- [33] L. Muflikhah and D. J. Haryanto, “High performance of polynomial kernel at SVM algorithm for sentiment analysis,” *Journal of Information Technology and Computer Science*, vol. 3, no. 2, pp. 194–201, Nov. 2018, doi: 10.25126/jitecs.20183260.

BIOGRAPHIES OF AUTHORS






Kuncahyo Setyo Nugroho     received a bachelor of computer degree from the Department of Informatics Engineering, Faculty of Engineering, Widayagama University, Indonesia, in 2019. He is currently pursuing a master's degree at the Department of Informatics Engineering, Faculty of Computer Science, Brawijaya University, Indonesia. He is a member of the Intelligent Systems Research Laboratory, with interest in affective computing. He also has research interests in machine learning, deep learning, and natural language processing. He can be contacted at email: ksnugroho26@gmail.com or ksnugroho@student.ub.ac.id.






Anantha Yullian Sukmadewa    obtained a bachelor's degree in Educational Informatics Engineering from The Department of Informatics Engineering, Malang State University in 2019. He is currently continuing his master's studies at The Department of Computer Science, Brawijaya University, Indonesia. He can be contacted at email: ananthayullian@gmail.com or ananthayullian@student.ub.ac.id.



Angga Vidiyanto    completed his bachelor's degree at Department of Informatics Engineering, State Polytechnic of Malang in 2015. He started his career as a web developer and database engineer at a telecommunications company for four years. He is currently continuing his master's studies at Department of Computer Science, University of Brawijaya, Indonesia. He can be contacted at email: angga.vidianto@gmail.com or anggavidianto@student.ub.ac.id.



Wayan Firdaus Mahmudy    obtained a Bachelor of Science degree from the Mathematics Department, Brawijaya University in 1995. His Master in Informatics Engineering degree was obtained from the Sepuluh Nopember Institute of Technology, Surabaya in 1999 while a Ph.D. in Manufacturing Engineering was obtained from the University of South Australia in 2014. He is a Professor at Department of Computer Science, Brawijaya University (UB), Indonesia. His research interests include optimization of combinatorial problems and machine learning. He can be contacted at email: wayanfm@ub.ac.id.

An efficient machine learning-based COVID-19 identification utilizing chest X-ray images

Mahmoud Masadeh¹, Ayah Masadeh¹, Omar Alshorman², Falak H Khasawneh³, Mahmoud Ali Masadeh⁴

¹Computer Engineering Department, Yarmouk University, Irbid, Jordan

²Faculty of Engineering and AlShrouk Trading Company, Najran University, Najran, Saudi Arabia

³Department of Applied Medical Sciences, Zarqa University College, Al-Balqa Applied University, Al-Salt, Jordan

⁴Ministry of Education, Amman-Al Abdli, Amman, Jordan

Article Info

Article history:

Received Jul 10, 2021

Revised Dec 27, 2021

Accepted Jan 5, 2022

Keywords:

Convolutional neural

COVID-19

Deep learning

Diagnosis

Machine learning

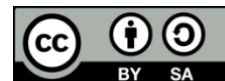
Network

X-ray

ABSTRACT

There is no well-known vaccine for coronavirus disease (COVID-19) with 100% efficiency. COVID-19 patients suffer from a lung infection, where lung-related problems can be effectively diagnosed with image techniques. The golden test for COVID-19 diagnosis is the RT-PCR test, which is costly, time-consuming and unavailable for various countries. Thus, machine learning-based tools are a viable solution. Here, we used a labelled chest X-ray of three categories, then performed data cleaning and augmentation to use the data in deep learning-based convolutional neural network (CNN) models. We compared the performance of different models that we gradually built and analyzed their accuracy. For that, we used 2905 chest X-ray scan samples. We were able to develop a model with the best accuracy of 97.44% for identifying COVID-19 using X-ray images. Thus, in this paper, we attested the feasibility of efficiently applying machine learning (ML) based models for medical image classification.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Mahmoud Masadeh

Computer Engineering Department, Yarmouk University

Irbid 21163, Jordan

Email: mahmoud.s@yu.edu.jo

1. INTRODUCTION

Currently, the entire world is suffering due to the coronavirus disease (COVID-19). Most of the countries all over the world started vaccinating their people for COVID-19 [1]. Thus, vaccines generate protection against the illness, as a consequence of developing an immune response to the SARS-Cov-2 virus. However, COVID-19 is still threatening global health care, since none of the widely used vaccines has approved 100% efficiency without any side effect [2]. Moreover, the ability of the countries to obtain the vaccine is as low as their ability to afford reverse transcription-polymerase chain reaction (RT-PCR) tests for their residents [3]. As of 26 Dec 2021, the total number of cases confirmed to be infected has surpassed 238 million, in more than 220 countries, with more than 4.82 million deaths from COVID-19. Thus, it has caused calamitous consequences on everyday activities, health care, and global economics. Figure 1(a) shows the average number of worldwide new cases per week while Figure 1(b) shows the average number of worldwide deaths per week. As shown in Figure 2 the number of people, as 25 Sept 2021, who have received at least one dose of a vaccine is 44.8% of the world population, while the number of people who fully vaccinated is around 32.9% of the world population. However, more than 20 countries did not reach 1% yet while 40 countries have less than 5% of the population being vaccinated. Such numbers show that reaching to

100% vaccinated world is a long-journey, specially for low-income countries, where the best solution is COVID-19 avoidance and early detection [4].

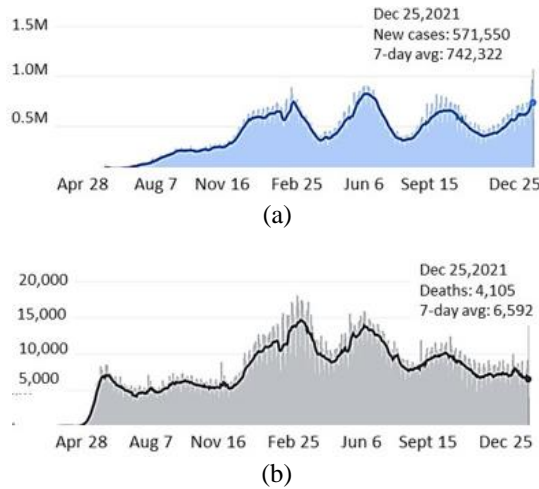


Figure 1. Average number of world wide (a) new cases per week and (b) deaths per week

Since the beginning of the pandemic, different repositories, e.g., Github and Kaggle, presented online chest X-ray images for natural and infected peoples where such images include notable knowledge regarding the COVID-19 virus [5]. The diagnosing process of COVID-19 required the direct contact of the infected patients with the medical staff, which is very risky task [6]. Thus, more precautions are required. Moreover, the treatment procedure should be strict to reduce the risk of infecting the healthy ones [7]. This plague is developing in a sequential process that conveys from one person to another after contacting COVID-19 infected bodies [8]. The diagnosis process of this epidemic includes various personnel, e.g., doctors, nurses, lab technicians and hospital staff [9]. Therefore, to degrade the effect of COVID-19 several approaches have been applied [10]. Moreover, medical imaging is an effective methodology for examining and forecasting the influence of COVID-19 on an individual’s well-being [11].

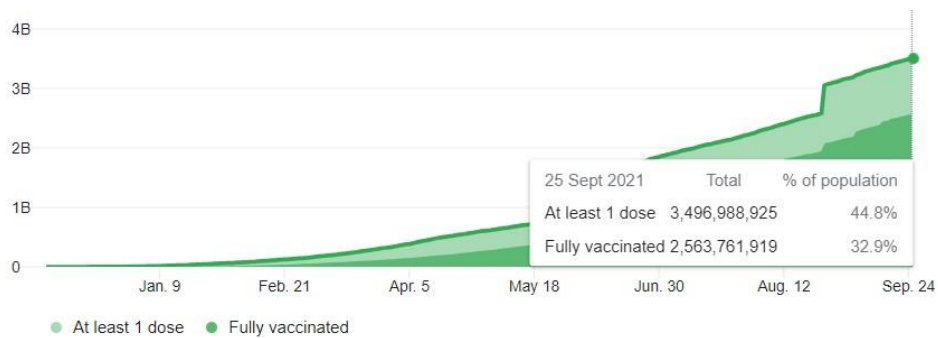


Figure 2. The number of people worldwide who have received a minimum of one dose of a vaccine

Medical systems, including the healthcare givers, are facing tremendous demands on the available resources, e.g., medical tools, RT-PCR test kits, and hospital beds [12]. Thus, the ability to offer a prioritized hospitalization for severe cases is crucial, where the inability to do so exacerbates the plague rate and significantly hinders disease containment [13]. To help the medical systems, especially in the developed countries with limited resources, various prediction models have been introduced for fast, cheap, and reliable diagnosis of COVID-19 [14].

The main symptoms for COVID-19 include respiratory dysfunction with moderate to critical conditions, e.g., sore throat, coughing, and diarrhea, where the virus circulates by air and physical contact [15]. Moreover, elderly people with chronic diseases need special handling since they are vulnerable and

could exhibit serious breakdown if affected with this disease. Thus, the most familiar response to this outbreak is patient isolation and infection-control standards. Whenever the RT-PCR test kits are unavailable due to time and/or cost constraints, machine learning-based approaches could be used, where the smart analysis of X-ray and computerized tomography (CT) images can easily identify COVID-19 cases [16]. However, there is no golden ML-based model where a highly accurate approximation is acceptable in such cases [17], [18].

Various ML-based techniques are used for disease detection using medical images and signals, e.g., electroencephalogram (EEG) [19], electrocardiogram (ECG), and electromyography (EMG) [20]. For example, the authors of [21] introduced a ML-based decision assistance scheme for the prediction of diabetes, while the authors of [22] used various ML-based techniques for cancer detection. The performance of such ML-based approaches is extremely influenced by the quality of the non-optimal and time-consuming hand-engineered features [23]. Therefore, implementing an automatic extraction of optimal features, from the input data, for efficient feature extraction with enhanced classification precision, is required. Thus, various techniques based on deep learning (DL) could be utilized [24], [25].

The authors of [26] suggested using deep CNNs for segmenting lung X-ray images. Thus, they were able to enhance the performance of clinical diagnosis of different diseases of the lungs. In [27], the authors applied CNNs for automatic feature learning and classification. Then, revealed a reliable system to recognize breast cancer using histology images [28]. However, using DL models requires large datasets for training, where such a huge dataset is not always available. Moreover, the majority of medical images datasets are imbalanced with inadequate representation, in addition to privacy and confidentiality concerns of medical data [29]. Therefore, different data augmentation methods are introduced to defeat the shortage of labelled data and model over-fitting. For that, data alteration, e.g., scaling, flipping, rotation, and transformation, is used [30].

According to [31], there exist three main mechanisms to utilize CCNs for the classification of medical images; preparing the CCN from whatever is available (the scratch); using already pre-trained (off-the-shelf) CCNs; unsupervised CCN pre-training with supervised fine-tuning. In this work, we used the first mechanism (trained the CCN from the scratch). In this paper, we introduce an efficient DL-based model that predicts a positive SARS-CoV-2 infection based on X-ray images. We utilized a well-known machine learning method, CNN, which is effective to realize. The proposed model is also able to recognize pneumonia as well as normal cases. We trained the model on publicly available data, i.e, Kaggle repository, which is used by various related works. Thus, our model can be realized globally for efficient monitoring and prioritization of testing for the virus in the worldwide community.

Next, we describe some of the work that is closely related to this paper based on utilizing X-ray and CT images for disease diagnosis. The authors of [32] suggested a Multi-level thresholding and SVM based methodology for COVID-19 detection utilizing 40 images of chest X-ray where the obtained accuracy was 97.48%. Similarly, the authors of [33] introduced a novel framework of DL to diagnose COVID-19, called COVIDX-Net. Based on 50 images of chest X-ray, it supports the practitioners to automatically detect COVID-19. Based on 25 COVID-19 positives and 25 normal images, the accomplished accuracy was 90%.

The scholars of [34] introduced various approaches for COVID-19 discovery with the challenges facing the RT-PCR. Thus, it is highly obliged to realize an automatic identification method to restrict the expanse of the virus through direct communication. While there are various DL approaches for COVID-19 detection, they are able to detect subjects suffering from pneumonia without the ability to accurately specify the exact cause of pneumonia, i.e., whether it is COVID-19, bacterial, or fungal attack. The authors of [35] proposed using AI tools, by radiologists and healthcare professionals, for fast and reliable COVID-19 diagnosis. Due to a shortage of publicly available datasets, the authors built a dataset of 170 X-ray images from multiple sources. Then, build a forecasting technique based on DL and transfer learning algorithms. Experimental results reveal a 94.1% accuracy based on modified CNN.

In [36], the authors designed a DL-based model, i.e., COVID-Net, which delivered a 92.4% success rate using 13975 radiography images. The used images are collected from various open access data. A large dataset of 1427 X-ray images was collected in [37], where the outcomes indicate that DL with X-ray imaging can obtain meaningful biomarkers related to the COVID-19 disease with an accuracy of 96.46%. In their study, the authors of [38] investigated the realization of various classification models for COVID-19 detection where they practiced eleven different CNN models. The best classification model realized accuracy of 95.33% for the identification of COVID-19, where that model was based on a deep feature plus SVM. The study of [39] proposed five pre-trained CNN-based models for COVID-19 detection using chest X-ray images. The models InceptionV3 and ResNet50 achieved an average accuracy of 95.4% and 96.1%, respectively.

The authors of [40] trained a DL-based model on the ResNet-101 CNN architecture using a dataset of 4376 X-ray images. They achieved an accuracy of 71.9%, where such low accuracy is due to the shortage

of the used images in the testing stage. Various work proposed using CT images for COVID-19 diagnosis, e.g., [41]–[44], which achieved an accuracy of 86%, 82.9%, 90.8%, and 86.7%, respectively. It is known that a chest X-ray is a reliable first-look exam with low cost, while a CT scan is suitable for exact analysis and therapy with high cost. Moreover, CT radiation causes attention for subjects who need dynamic monitoring, in addition to investigating children and young patients [45]. Therefore, in this work, we focus on using X-ray images while keeping CT images as future work.

2. CNN-BASED PROPOSED MODEL

2.1. Convolutional neural networks (CNNs)

Artificial intelligence (AI) can contribute to the fight against COVID-19 in various ways such as immediate alarms and warnings, examination and prediction, following and forecast, treatments, and cures, data dashboards, and social control. Machine learning (ML) uses algorithms to analyze big-data and learn from it [46]. Then, it will be able to make a prediction or a decision about new unseen data [47]. A well-known technique to realize machine learning is deep learning (DL), which employs algorithms caused by the composition and functionality of the neurons of the human brain. Artificial neural networks (ANN) are DL models that are based on the structure of the brain's neural network, which is a lightweight design for data classification problems. For a given set of inputs, the activation function of a neuron specifies its result. The activation function is inspired by the activity of our brain, where several neurons fire, i.e., are operated, by several stimuli. ANNs are an expensive solution for image classification problems since 2D images are converted to 1D vectors. Thus, 1D vectors will have a large number of trainable parameters with unusual storage and processing requirement. On the other hand, CNN, i.e., ConvNets, are a specific type of ANN that use convolution operation instead of the common multiplication in one of their layers at least. Thus, in this work, we implement CNN in Python.

The fundamental construction block for CNN are filters, i.e., kernels, which are applied to deduce the related features from the input images utilizing the convolution process. CNNs include two main steps, where the first step is feature learning, which includes: i) convolution for features learning, ii) application of activation function to obtain non-linearity, and iii) pooling to decrease dimensionality and maintain spatial invariance. The second step is the classification, where the output of the feature extraction is fed into fully connected layers. Then, the output is expressed as the chance of an image pertaining to an appropriate class. Generally, the CNNs have various advantages clouding their ability to automatically learn the filters which are used for feature extraction from the inputs. Moreover, CNNs capture the spatial features from images, i.e., the organization of pixels and the correlation between them. Besides, an individual filter is employed across various parts on input images to compose a feature map, i.e., parameter sharing. Section 3.4 explains the architecture of the proposed CNN model, including feature extraction and classification. Moreover, Table 1 explains the parameters of the layers of the proposed/constructed model.

A well-known problem in ML-based models is overfitting and underfitting. Model overfitting is when the model has high prediction accuracy for the training data while it has low prediction accuracy for the testing data. Furthermore, underfitting is when the model is unable to properly predict the class of the data even for the data it was trained on. To solve model underfitting we could increase the complexity of the model or add more features to the input sample. Data augmentation is achieved by making a conscious modification to the training data, e.g., flip, rotate, crop, zoom, and vary the color, in order to produce new data and reduce model overfitting. Thus, we ensured that our model is free from over- & under-fitting. Figure 3 shows the main phases of our CNN-based proposed design. It includes six main steps, which are data preprocessing, importing the necessary modules, data partitioning, building and adding layer(s) to the module, fitting the model, and building a Flask webpage as a graphical user interface (GUI).

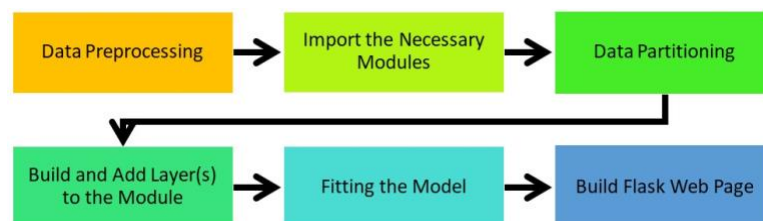


Figure 3. The main phases of the CNN-based proposed design

2.2. Data Set of X-ray images

In this work, we handled chest X-ray images of three classes, i.e., normal, viral pneumonia, and COVID-19 infected where Figure 4 depicts a sample set of the used images. For the collected dataset which covers three classes, we collect 1345 X-ray images of the normal class, 1342 X-ray images of the pneumonia class, and 217 images of the COVID-19 class. Thus, the complete number of images of the dataset is 2905. This data has been obtained from the Kaggle repository [48] and used by previous works, e.g., [32]–[40]. In the exploratory analysis we divide the data into a training set of 2324 images, i.e., 80% of the data, and a validation set of 581 images, i.e., 20% of the data such data partitioning is used by most of ML models and various related works. For parameter tuning, we used 464 images while performed final model testing by the remaining 117 images.

Regarding data preprocessing, we exported the X-ray images, with their APIs, from the Kaggle repository [48]. Then, we cleaned the exported data. It is well-known that DL methods require a large dataset of images to obtain a reliable result, which is not always available due to privacy issues, cost, and data generation time. Thus, we augmented the data through performing data transformation, e.g., rotating and scaling, to extend the size of the training data. Moreover, data enlargement solves the problem of model overfitting and improves the achieved accuracy.

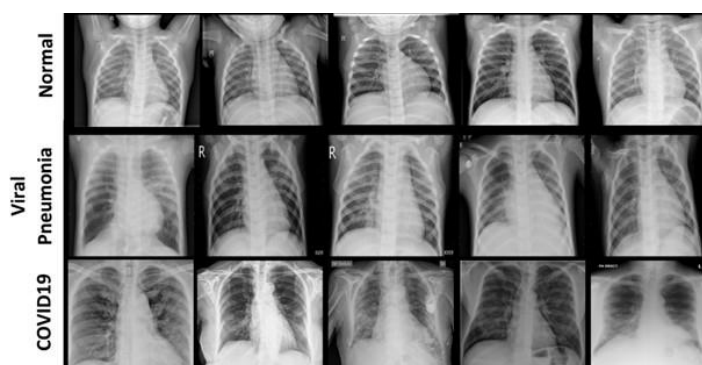


Figure 4. A sample of data set of X-ray images

2.3. Import the required modules

We used Keras, which is a deep learning Python library. It offers simple and compatible application programming interface (API) with easily explained error messages. Thus, the required user actions are minimal. If needed, comprehensive documentation and guidance are also available. Keras enables a fast transition from the idea to the result. We used Adam as an optimization algorithm and Relu as an activation function. Moreover, we used various libraries as explained next.

With images, efficient computing requires performing operations on data as vectors and matrices. Thus, linear algebra is required where the scientific computing library, i.e., NumPy, is utilized. Communicating with the operating system is achieved by various functions implemented by the OS module in Python. For data science and analytics, we use the Pandas module which runs on top of the NumPy library. The sequential module allows creating a deep learning model by adding layers to it. However, some architectures are not linear stacks, e.g., a Siamese NN, which concurrently works on two different input vectors. However, in this work, we use the sequential model. Flatten library converts multi-dimension matrices to vectors.

2.4. Developing deep learning-based model

For the implementation of the target design, we considered various features, i.e., design specifications, that are related to the neural network and the associated web page. The main three features are: i) time complexity where an efficient algorithm will have a short execution time, ii) performance where a high prediction accuracy is obtained by changing the number of layers of the NN, the number of the nodes per layer, and the type of the activation function, and iii) novelty where a new design is implemented without using previous designs.

We obtained the data from the Kaggle repository, cleaned it, and performed data augmentation. Then, this data is applied to train the suggested model. For more useful investigation, we implemented various models with different settings. Then, we examine their performance to determine the accuracy. We plan to reduce the loss function with succeeding epochs, where one Epoch is when a complete dataset is

moved forward and backward within the neural network once. Therefore, we adopted Adam optimizer with a learning rate (LR)=0.001 for model training. We have employed the categorical cross-entropy loss to train our model. In the proposed models, we used the default activation function, i.e., the rectified linear unit (ReLU), which is shown in Figure 5 where $y=Max(0,x)$. Figure 6 shows the structure of the proposed model where more implementation details are given in Table 1.

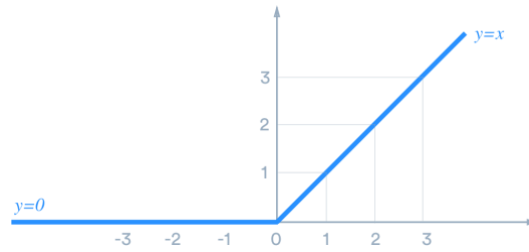


Figure 5. Activation functions: ReLU

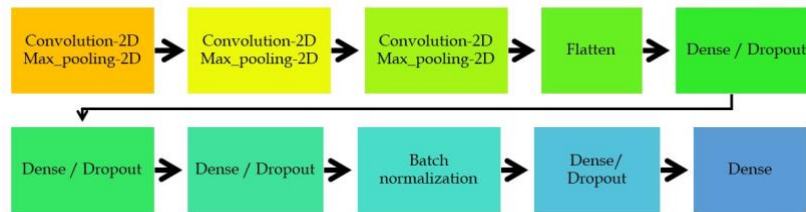


Figure 6. The architecture of the proposed CNN Model

Table 1. Parameters of the layers of the proposed model.

Number of Layers	Layer (Type)	Output Shape	Number of Trainable Parameters
1	Conv-2D	(222, 222, 32)	896
2	Max pool-2D	(111, 111, 32)	0
3	Conv-2D	(109, 109, 64)	18496
4	Max pool-2D	(36, 36, 64)	0
5	Conv-2D	(34, 34, 64)	36928
6	Max pool-2D	(17, 17, 64)	0
7	Flatten	(18496)	0
8	Dense	(512)	9470464
9	Dropout	(512)	0
10	Dense	(256)	131328
11	Dropout	(256)	0
12	Dense	(128)	32896
13	Dropout	(128)	0
14	Batch Normalization	(128)	512
15	Dense	(64)	8256
16	Dropout	(64)	0
17	Dense	(3)	195

Padding is a technique that allows us to keep the size of the filtered image the same as the original image. However, we did not use padding in any convolutional layer (P=0). This is why for an input image of size $n \times n$, we obtain an output image of size $n-2 \times n-2$. Application of the filter to the input image requires moving the filter from left to right and from top to bottom. The amount of this movement is defined as stride. The default value of the stride is (1,1) for the horizontal and vertical movement, where such value works well for most of the cases. Therefore, we applied the default value of the filters in all convolutional layers in this work. When performing a stride convolution for an input image of size $n \times n$, padding is (P), the stride is (S), and the filter size is $F \times F$. Then the size of the output is defined as in (1).

$$Output\ size = \left(\frac{n+2P-F}{S} + 1\right)\left(\frac{n+2P-F}{S} + 1\right) \tag{1}$$

As shown in Figure 6, we used convolution and maximum pooling thrice, where the used X-ray images are of size 224×224 pixels. The size of the filter for the three convolutional layers is 3×3 while $S=1$.

For the first layer, the convolution result is an images of size 222×222 . Then, we applied the max pool function with $S=2$ and $F=2$, utilizing 32 filters. The result of maximum pooling is images of size of 111×111 . Regarding the second layer, the max pool function has $S=3$ and $F=3$, utilizing 64 filters. On the other hand, the third layer has $S=2$ and $F=2$, which is similar to the first layer. The output shape of the 3 layers is shown in Table 1, which are directly obtained according to (1).

The result of applying 2D convolution and max-pooling has the size of $(17, 17, 64)$. Then, Flatten which converts multi-dimension matrices to vector, is applied to convert the data from $17 \times 17 \times 64$ to 18496. The dense, i.e., fully connected layer, which is a linear operation on the layer's input vector, is used 5 times (layer 8, 10, 12, 15 and 17). The input vector for the first dense layer is 512 which is reduced by 2 to reach 64 for the 4th dense layer, while the 5th dense layer has an output vector of 3 which represents one class of the X-ray possible types. Dropout is a well-known regularization procedure, which aims to decrease the complexity of the model with the aim to stop overfitting. Thus, we applied dropout 4 times directly after applying dense operation. Batch normalization technique is designed to standardize the inputs to a layer in CNN. Thus, it significantly expediting the training process with improved performance. In our model, we applied batch normalization once after the third pair of dense-dropout.

3. RESULTS AND DISCUSSION

Figure 7 shows the training loss and validation loss for the 30 epochs of model training. We notice that the training loss starts with the maximum of 1.83 for the first epoch and reaches 0.073 for the last epoch. Regarding the validation loss, it starts by a value of 0.74 and reaches a maximum of 1.46 at epoch 11. Then, the loss starts decreasing until it reaches 0.155 at the last epoch.

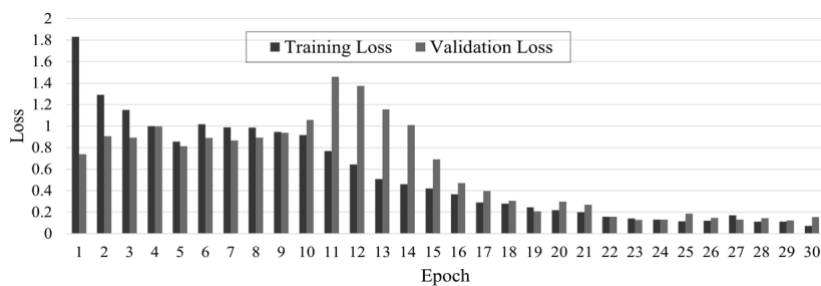


Figure 7. The training and validation loss of the model with successive epochs

Figure 8 shows the training and validation accuracy for the 30 epochs of model training. Regarding the accuracy of the training model, it starts by 32.1% and reaches the maximum of 97.55% at the last epoch. The last step was building a flask web page, where flask is a web framework written in Python. It does not require specific tools or libraries, does not include an abstraction method for the database, and it is a reliable framework for web users. We evaluated the proposed model based on accuracy as given in (2). The obtained confusion matrix, for evaluating the accuracy of the model on the testing data, is shown in Figure 9. The achievable accuracy is 97.44% for testing while the accuracy for the training was 97.55% as shown in Figure 8.

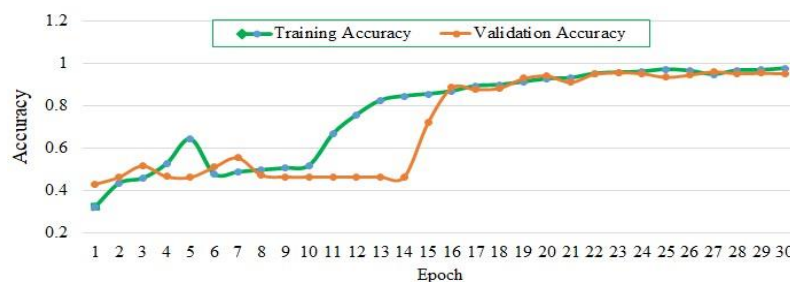


Figure 8. The training and validation accuracy of the model with successive epochs

The model is called through an interface using flask page as shown in Figure 10. Accordingly, Figures 10(a), 10(b), and 10(c) show the results provided by the model for given images, where these images are classified as normal, COVID-19, and viral pneumonia, respectively. In section 2. we explained various works which are closely related to this work. Table 2 shows a tabular comparison of the proposed model with various related work, i.e., based on chest X-ray and CT images. Moreover, the table shows the number of used images, the type of the images, the used method, and the final accuracy of the model.

		Predicted Class		
		COVID-19	Normal	Viral Pneumonia
Actual Class	COVID-19	10	0	0
	Normal	1	51	1
	Viral Pneumonia	1	0	53

Figure 9. The confusion matrix of the testing data

$$Accuracy = \frac{True\ Positive + True\ Negative}{True\ Positive + True\ Negative + False\ Positive + False\ Negative} \quad (2)$$

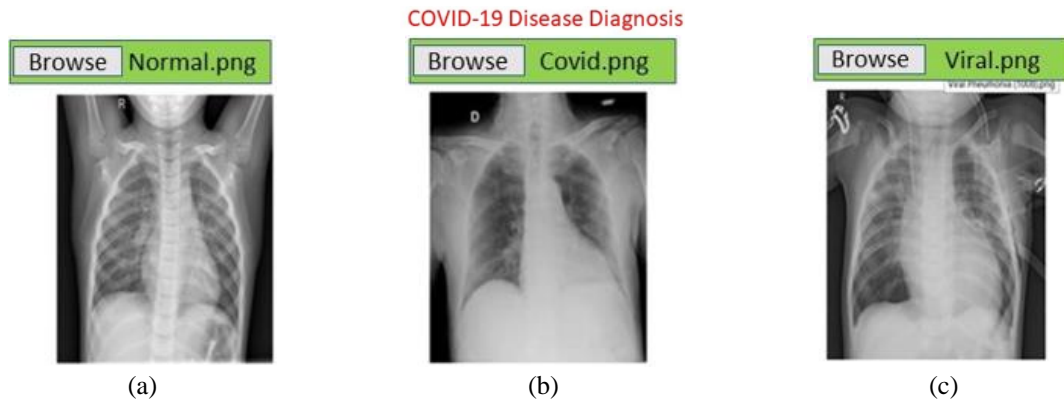


Figure 10. The GUI of the flask page for COVID-19 detection (a) result: normal, (b) result: COVID-19, and (c) result: viral pneumonia

Table 2. Comparison between related DL models for covid-19 identification

Work	Number of Cases	Type of Images	Method	Accuracy (%)
[33]	50	X-ray	COVIDX-Net	90
[35]	170	X-ray	CNN	94.1
[36]	13645	X-ray	COVID-Net	92.4
[37]	1427	X-ray	VGG-19	96.78
[38]	50	X-ray	ResNet50+SVM	95.33
[39]	100	X-ray	InceptionV3	95.4
[39]	100	X-ray	ResNet50	96.1
[40]	4376	X-ray	ResNet-101	71.9
[41]	1485	CT	DRE-Net	86
[42]	453	CT	M-Inception	82.9
[43]	542	CT	UNet+3D Deep Network	90.8
[44]	618	CT	ResNet + Location Attention	86.7
Proposed	2905	X-ray	CNN	97.44

Various authors used the dataset [48] for Covid detection, where they utilized standard CNN architectures, e.g., ResNet. Because of its compelling results, ResNet quickly became one of the most popular architectures in various computer vision tasks. Although ResNet has proven powerful in many applications, one major drawback is that a deeper network usually requires weeks for training, making it practically infeasible in real-world applications. ResNet-50 is a CNN that is 50 layers deep, while ResNet-101 CNN is 101 layers deep. We evaluated the accuracy of our proposed architecture using datasets of

previous related works, i.e., [33]–[40]. We obtained a competitive accuracy. The use of mobile phones to scan X-ray images for the detection of COVID-19 is considered as an extension of this work.

4. CONCLUSION

The latest world pandemic of COVID-19 is progressing daily without a 100% efficient vaccine that has no side effect. Therefore, it is important to quickly identify the infected people to decrease the extent of the COVID-19 by prevention procedures and providing the required health care for the infected ones. Most countries, especially those that have limited resources, are unable to provide the required fast, cheap, and accurate tests to detect the infected cases. Therefore, in this paper, based on a limited set of chest X-ray scans, we investigated various CNN models to detect COVID-19 cases. Our proposed design is completely automated including automatic feature extraction. Moreover, the build models can classify the X-ray into infected, normal, or pneumonia, i.e., three-class classification, with a test accuracy of 97.44% and training accuracy of 97.55%. We tested the proposed model/framework on various data sets which are used by previous works. The testing accuracy was indistinguishable with the accuracy of testing the model based on the training data. Our CNN-based model is usable by radiologists to aid in confirming their initial screening of patients where a negligible accuracy loss is accepted. Thus, this approach is applicable for automatic disease diagnosis e.g., remote places affected by COVID-19. Moreover, we could use it to diagnose other chest-related diseases. To verify the high accuracy obtained, we tested the models based on the newly introduced public data and obtained almost the same accuracy. The main limitation of the work is using a limited amount of labelled X-ray images for COVID-19. Thus, future work will make the model more accurate by using more images. Moreover, the use of magnetic resonance imaging (MRI) and computed tomography (CT) images could be investigated in addition to X-ray images. Finally, COVID-19 belongs to the same family of severe acute respiratory syndrome coronavirus (SARS-CoV) and Middle East Respiratory Syndrome-related Coronavirus (MERS-CoV). Thus, it is possible to detect SARS-CoV and MERS-CoV utilizing chest x-ray and the proposed model/framework.

REFERENCES




- [1] S. Bagchi, "The world's largest COVID-19 vaccination campaign," *The Lancet Infectious Diseases*, vol. 21, no. 3, p. 323, Mar. 2021, doi: 10.1016/S1473-3099(21)00081-5.
- [2] "How effective are the covid-19 vaccines?" 2021, Accessed: Mar. 25, 2021. [Online]. Available: <https://www.statista.com/chart/23510/estimatedeffectiveness-of-covid-19-vaccine-candidates/>.
- [3] S. A. Gómez-Ochoa *et al.*, "COVID-19 in health-care workers: a living systematic review and meta-analysis of prevalence, risk factors, clinical characteristics, and outcomes," *American Journal of Epidemiology*, vol. 190, no. 1, pp. 161–175, Jan. 2021, doi: 10.1093/aje/kwaa191.
- [4] C. Vianello *et al.*, "A perspective on early detection systems models for COVID-19 spreading," *Biochemical and Biophysical Research Communications*, vol. 538, pp. 244–252, Jan. 2021, doi: 10.1016/j.bbrc.2020.12.010.
- [5] B. Aksoy and O. K. M. Salman, "Detection of COVID-19 disease in chest X-Ray Images with capsul networks: application with cloud computing," *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 33, no. 3, pp. 527–541, May 2021, doi: 10.1080/0952813X.2021.1908431.
- [6] C. D. L. Goulart *et al.*, "Lifestyle and rehabilitation during the COVID-19 pandemic: guidance for health professionals and support for exercise and rehabilitation programs," *Expert Review of Anti-infective Therapy*, vol. 19, no. 11, pp. 1385–1396, Nov. 2021, doi: 10.1080/14787210.2021.1917994.
- [7] C. Basile *et al.*, "Recommendations for the prevention, mitigation and containment of the emerging SARS-CoV-2 (COVID-19) pandemic in haemodialysis centres," *Nephrology Dialysis Transplantation*, vol. 35, no. 5, pp. 737–741, May 2020, doi: 10.1093/ndt/gfaa069.
- [8] L. Yan *et al.*, "Prediction of criticality in patients with severe Covid-19 infection using three clinical features: a machine learning-based prognostic model with clinical data in Wuhan," *medRxiv*, 2020, doi: 10.1101/2020.02.27.20028027.
- [9] Y. M. Arabi *et al.*, "How the COVID-19 pandemic will change the future of critical care," *Intensive Care Medicine*, vol. 47, no. 3, pp. 282–291, Mar. 2021, doi: 10.1007/s00134-021-06352-y.
- [10] V. Kumar Singh, M. Abdel-Nasser, N. Pandey, and D. Puig, "LungINFseg: segmenting COVID-19 infected regions in lung CT Images based on a receptive-field-aware deep learning framework," *Diagnostics*, vol. 11, no. 2, p. 158, Jan. 2021, doi: 10.3390/diagnostics11020158.
- [11] S. Bhattacharya *et al.*, "Deep learning and medical image processing for coronavirus (COVID-19) pandemic: A survey," *Sustainable Cities and Society*, vol. 65, p. 102589, Feb. 2021, doi: 10.1016/j.scs.2020.102589.
- [12] T. Hoshina *et al.*, "Intensive diagnostic management of coronavirus disease 2019 (COVID-19) in academic settings in Japan: challenge and future," *Inflammation and Regeneration*, vol. 40, no. 1, pp. 38–47, Dec. 2020, doi: 10.1186/s41232-020-00147-2.
- [13] L. Craxi, A. Casuccio, E. Amodio, and V. Restivo, "Who should get COVID-19 vaccine first? a survey to evaluate hospital workers' opinion," *Vaccines*, vol. 9, no. 3, p. 189, Feb. 2021, doi: 10.3390/vaccines9030189.
- [14] M. Masadeh, O. Hasan, and S. Tahar, "Approximation-conscious IC testin," in *2018 30th International Conference on Microelectronics (ICM)*, Dec. 2018, pp. 56–59, doi: 10.1109/ICM.2018.8704099.
- [15] T. Singhal, "A review of coronavirus disease-2019 (COVID-19)," *The Indian Journal of Pediatrics*, vol. 87, no. 4, pp. 281–286, Apr. 2020, doi: 10.1007/s12098-020-03263-6.
- [16] D. R. Seshadri *et al.*, "Wearable sensors for COVID-19: a call to action to harness our digital infrastructure for remote patient monitoring and virtual assessments," *Frontiers in Digital Health*, vol. 2, pp. 1–8, Jun. 2020, doi: 10.3389/fdgh.2020.00008.

- [17] M. Masadeh, O. Hasan, and S. Tahar, "Machine learning-based self-compensating approximate computing," pp. 1–6, Jan. 2020, [Online]. Available: <http://arxiv.org/abs/2001.03783>.
- [18] M. Masadeh, O. Hasan, and S. Tahar, "Machine-Learning-based self-tunable design of approximate computing," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 29, no. 4, pp. 800–813, Apr. 2021, doi: 10.1109/TVLSI.2021.3056243.
- [19] O. M. AlShorman and A. M. Alshorman, "Frontal lobe and long-term memory retrieval analysis during pre-learning stress using EEG signals," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 1, pp. 141–145, Feb. 2020, doi: 10.11591/eei.v9i1.1335.
- [20] O. AlShorman *et al.*, "Frontal lobe real-time eeg analysis using machine learning techniques for mental stress detection," *Journal of Integrative Neuroscience*, pp. 1–11, 2021.
- [21] A. Yahyaoui, A. Jamil, J. Rasheed, and M. Yesiltepe, "A decision support system for diabetes prediction using machine learning and deep learning techniques," in *2019 1st International Informatics and Software Engineering Conference (UBMYK)*, Nov. 2019, pp. 1–4, doi: 10.1109/UBMYK48245.2019.8965556.
- [22] M. Gurbina, M. Lascu, and D. Lascu, "Tumor detection and classification of MRI Brain image using different wavelet transforms and support vector machines," in *2019 42nd International Conference on Telecommunications and Signal Processing (TSP)*, Jul. 2019, pp. 505–508, doi: 10.1109/TSP.2019.8769040.
- [23] L. Nanni, S. Ghidoni, and S. Brahnham, "Handcrafted vs. non-handcrafted features for computer vision classification," *Pattern Recognition*, vol. 71, pp. 158–172, Nov. 2017, doi: 10.1016/j.patcog.2017.05.025.
- [24] B. Guragai, O. AlShorman, M. Masadeh, and M. B. Bin Heyat, "A survey on deep learning classification algorithms for motor imagery," in *2020 32nd International Conference on Microelectronics (ICM)*, Dec. 2020, pp. 1–4, doi: 10.1109/ICM50269.2020.9331503.
- [25] K. Mahmoud *et al.*, "Prediction of the effects of environmental factors towards COVID-19 outbreak using AI-based models," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, p. 35, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp35-42.
- [26] H. Trung Huynh and V. Nguyen Nhat Anh, "A deep learning method for lung segmentation on large size chest x-ray image," in *2019 IEEE-RIVF International Conference on Computing and Communication Technologies (RIVF)*, Mar. 2019, pp. 1–5, doi: 10.1109/RIVF.2019.8713648.
- [27] D. Bardou, K. Zhang, and S. M. Ahmad, "Classification of breast cancer based on histology images using convolutional neural networks," *IEEE Access*, vol. 6, pp. 24680–24693, 2018, doi: 10.1109/ACCESS.2018.2831280.
- [28] B. Acs, M. Rantalainen, and J. Hartman, "Artificial intelligence as the next step towards precision pathology," *Journal of Internal Medicine*, vol. 288, no. 1, pp. 62–81, Jul. 2020, doi: 10.1111/joim.13030.
- [29] H. Salehinejad, E. Colak, T. Dowdell, J. Barfett, and S. Valaee, "Synthesizing chest X-Ray pathology for training deep convolutional neural networks," *IEEE Transactions on Medical Imaging*, vol. 38, no. 5, pp. 1197–1206, May 2019, doi: 10.1109/TMI.2018.2881415.
- [30] C. Shorten and T. M. Khoshgoftaar, "A survey on image data augmentation for deep learning," *Journal of Big Data*, vol. 6, no. 1, p. 60, Dec. 2019, doi: 10.1186/s40537-019-0197-0.
- [31] H.-C. Shin *et al.*, "Deep convolutional neural networks for computer-aided detection: cnn architectures, dataset characteristics and transfer learning," *IEEE Transactions on Medical Imaging*, vol. 35, no. 5, pp. 1285–1298, May 2016, doi: 10.1109/TMI.2016.2528162.
- [32] L. N. Mahdy, K. A. Ezzat, H. A. E. Haytham H. Elmousalami, and E. Hassanien, "Automatic X-ray COVID-19 lung image classification system based on multi-level thresholding and support vector machine," *medRxiv*, 2020.
- [33] E. E.-D. Hemdan, M. A. Shouman, and M. E. Karar, "COVIDX-Net: A framework of deep learning classifiers to diagnose COVID-19 in X-Ray images," Mar. 2020, [Online]. Available: <http://arxiv.org/abs/2003.11055>.
- [34] M. Ilyas, H. Rehman, and A. Nait-ali, "Detection of Covid-19 from chest X-ray images using artificial intelligence: an early review," Apr. 2020, [Online]. Available: <http://arxiv.org/abs/2004.05436>.
- [35] H. S. Maghddid, A. T. Asaad, K. Z. Ghafoor, A. S. Sadiq, and M. K. Khan, "Diagnosing COVID-19 pneumonia from X-Ray and CT images using deep learning and transfer learning algorithms," Mar. 2020, [Online]. Available: <http://arxiv.org/abs/2004.00038>.
- [36] L. Wang, Z. Q. Lin, and A. Wong, "COVID-Net: a tailored deep convolutional neural network design for detection of COVID-19 cases from chest X-ray images," *Scientific Reports*, vol. 10, no. 1, pp. 11961–11959, Dec. 2020, doi: 10.1038/s41598-020-76550-z.
- [37] I. D. Apostolopoulos and T. A. Mpesiana, "Covid-19: automatic detection from X-ray images utilizing transfer learning with convolutional neural networks," *Physical and Engineering Sciences in Medicine*, vol. 43, no. 2, pp. 635–640, Jun. 2020, doi: 10.1007/s13246-020-00865-4.
- [38] P. K. Sethy and S. K. Behera, "Detection of coronavirus disease (covid-19) based on deep features and support vector machine," *Preprints*, 2020, doi: 10.20944/preprints202003.0300.v1.
- [39] A. Narin, C. Kaya, and Z. Pamuk, "Automatic detection of coronavirus disease (COVID-19) using X-ray images and deep convolutional neural networks," *Pattern Analysis and Applications*, vol. 24, no. 3, pp. 1207–1220, Aug. 2021, doi: 10.1007/s10044-021-00984-y.
- [40] M. Z. Che Azemin, R. Hassan, M. I. Mohd Tamrin, and M. A. Md Ali, "COVID-19 deep learning prediction model using publicly available radiologist-adjudicated chest X-Ray images as training data: preliminary findings," *International Journal of Biomedical Imaging*, vol. 2020, pp. 1–7, Aug. 2020, doi: 10.1155/2020/8828855.
- [41] S. Ying *et al.*, "Deep learning enables accurate diagnosis of novel coronavirus (covid-19) with ct images," *medRxiv*, 2020, doi: 10.1101/2020.02.23.20026930.
- [42] S. Wang *et al.*, "A deep learning algorithm using CT images to screen for corona virus disease (COVID-19)," *European Radiology*, vol. 31, no. 8, pp. 6096–6104, Aug. 2021, doi: 10.1007/s00330-021-07715-1.
- [43] C. Zheng *et al.*, "Deep learning-based detection for covid-19 from chest ct using weak label," *medRxiv*, 2020, doi: 10.1101/2020.03.12.20027185.
- [44] X. Xu *et al.*, "A deep learning system to screen novel coronavirus disease 2019 pneumonia," *Engineering*, vol. 6, no. 10, pp. 1122–1129, Oct. 2020, doi: 10.1016/j.eng.2020.04.010.
- [45] Y. A. Vasilev *et al.*, "Chest MRI of patients with COVID-19," *Magnetic Resonance Imaging*, vol. 79, pp. 13–19, Jun. 2021, doi: 10.1016/j.mri.2021.03.005.
- [46] S. R. Salkuti, "A survey of big data and machine learning," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 1, pp. 575–580, Feb. 2020, doi: 10.11591/ijece.v10i1.pp575-580.
- [47] F. T. Fernandes, T. A. de Oliveira, C. E. Teixeira, A. F. de M. Batista, G. Dalla Costa, and A. D. P. Chiavegatto Filho, "A multipurpose machine learning approach to predict COVID-19 negative prognosis in São Paulo, Brazil," *Scientific Reports*, vol.




11, no. 1, pp. 3343–3350, Dec. 2021, doi: 10.1038/s41598-021-82885-y.
 [48] “Novel Corona Virus 2019 Dataset.” 2021, Accessed: Mar. 25, 2021. [Online]. Available: <https://www.kaggle.com/sudalairajkumar/novel-corona-virus2019-dataset>.

BIOGRAPHIES OF AUTHORS






Mahmoud Masadeh    received a B.Sc. degree in Computer Engineering, in 2003, Jordan. He holds an M.Sc degree in Computer Engineering from the Delft University of Technology, The Netherlands, as well as an M.Sc. in Management Information Systems from Jordan. He served as a full-time Teaching Assistant (TA) from 2005 to 2011, and as a full-time Instructor from 2013 to 2016, both at the Computer Engineering Department, Yarmouk University. In August 2020, he obtained his Ph.D. from Concordia University. Currently, he is an Assistant Professor at the Computer Engineering Department, Yarmouk University, Jordan. His research focuses on artificial intelligence and its application, smart health care, circuit design, approximate computing, and energy-efficient VLSI circuit design. Mahmoud is a member of IEEE, ACM and member of the Jordanian Engineering Association.






Ayah Masadeh    received a B.Sc. degree in Computer Engineering, in 2021, from Yarmouk University, Jordan. Currently, she is working as an independent researchers. Her research focuses on machine learning, artificial intelligence, and hardware design.






Omar Alshorman    was born in Irbid, Jordan in 1986. He received a B.S. degree in computer engineering from Al-Hussien Bin Talal University - Jordan in 2009, and M.S. from Yarmouk University- Jordan in 2012. In 2014, he joined an electrical engineering department, faculty of engineering, Najran University, Najran, Saudi Arabia as a lecturer. Currently, he is working as a project manager at AlShrouk trading company, Najran University, Najran, Saudi Arabia. His research interests are signal and image processing, healthcare informatics, the internet of things, artificial intelligence, and condition monitoring.



Falak H Khasawneh    received a B.S. degree in medical laboratory sciences from Jordan University of Science and Technology, Jordan in 2014, and M.Sc. in clinical biochemistry/ Medical laboratory Sciences from Jordan University of Science and Technology, Jordan 2017. She served part time lecturer from 2017-2021 at medical laboratory sciences department at Jordan University of Science and Technology, and as full-time lecturer at Zarqa university college, Balqa applied university from 2021-now. Her research fields include many areas of laboratory sciences such as: male fertility, drug effects, infectious diseases, and healthcare.



Mahmud Ali Masadeh    received a B.S. degree in computer science from Irbid National University, Jordan in 2009, and M.Sc in Information and Communication Technology in Education from Al Albayt University, Jordan in 2018. Currently, he is working at Ministry of Education. His research interests are artificial intelligence and its application, E-education, and E-health.

An efficient resource utilization technique for scheduling scientific workload in cloud computing environment

Nagendra Prasad Sodinapalli¹, Subhash Kulakrni¹, Nawaz ahmed Sharief², Prasanth Venkatareddy³

¹Department of Electronics and Communications Engineering, PES Institute of Technology, Affiliated to Visvesvaraya Technological University, Bangalore, India

²Samsung R&D Institute, Bangalore, India

³Department of Electrical Engineering, Nitte Meenakshi Institute of Technology, Bangalore, India

Article Info

Article history:

Received Apr 1, 2021

Revised Dec 19, 2021

Accepted Dec 31, 2021

Keywords:

Data-intensive applications

Energy aware scheduling

Heterogeneous cloud framework

Multi-objective optimization

Problem quality of service

ABSTRACT

Recently, number of data intensive workflow have been generated with growth of internet of things (IoT's) technologies. Heterogeneous cloud framework has been emphasized by existing methodologies for executing these data-intensive workflows. Efficient resource scheduling plays a very important role provisioning workload execution on Heterogeneous cloud framework. Building tradeoff model in meeting energy constraint and workload task deadline requirement is challenging. Recently, number of multi-objective-based workload scheduling aimed at minimizing power budget and meeting task deadline constraint. However, these models induce significant overhead when demand and number of processing core increases. For addressing research problem here, the workload is modelled by considering each sub-task require dynamic memory, cache, accessible slots, execution time, and I/O access requirement. Thus, for utilizing resource more efficiently better cache resource management is needed. Here efficient resource utilization (ERU) model is presented. The ERU model is designed to utilize cache resource more efficiently and reduce last level cache failure and meeting workload task deadline prerequisite. The ERU model is very efficient when compared with standard resource management methodology in terms of reducing execution time, power consumption, and energy consumption for execution scientific workloads on heterogeneous cloud platform.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Nagendra Prasad Sodinapalli

Department of Electronics and Communication Engineering, PES Institute of Technology, Bangalore South Campus

1 km before Electronic city, Hosur Road, Bangalore-560100

Email: nagendraps09@rediffmail.com

1. INTRODUCTION

Unanimous growth and advancement in fabrication industry, number of transistors can be easily embedded into one single chip, also recent development in processor design has caused for adding several central processing unit (CPU) core and large cache in single chip to enhance the performance. However, this has also caused several power mechanism issue which affects the device reliability, device performance and battery life of device. Moreover, power consumption is considered as one of the major reason that lead to the architecture shift towards the multicore chips, this tends to manage the demand in increase in frequency. However, in order to keep up the performance flawless many number of cores needed to be added to the given processor, this makes the power one of the important factor again; the main challenge is managing

huge number of cores for delivering high performance with low power consumption. The performance can be improvised through two scenarios i.e. either through increasing the core and communication width or through increasing frequency. These two scenarios can result in power consumption; further energy is considered as the byproduct of performance and power. Hence it is another motivation for researching the relationship among them.

Power consumption issue has been tackled through the various existing technique such as dynamic power management (DPM) and dynamic voltage frequency scaling (DVFS) [1], in here DVFS controller detects the computational patterns in execution process and further determines the voltage scaling and frequency scaling of CPU core to reduce the energy consumption. Moreover, several processor designs have adopted DVFS per core. For instance, Intel's boost technology and LITTLE cores, where each core holds the capability for scaling the level of voltage and frequency; further in multi-core model, extended control protocol is implied for controlling the core to reduce power consumption [2]. Further other studies have focused energy efficiency indirectly through performance improvisation as in [3], [4], machine learning based approach is used for the complex problem such of minimizing the multicore power consumption [5]. These mechanisms can be applied efficiently to the core; however, such strategy cannot be scaled to the given chip level where management is required to be applied in given coordinated type. Moreover, the overhead of given technique are unacceptable as the number of core increases [6]. Further, reinforcement learning is employed to manage the power consumption due to prior knowledge and also it is adaptive [7]. However, all these DVFS suffers heavily from the low supply voltage constraint and failed to consider the cache reliability and also none of them considered the power allocation among chip clusters dynamically in order to improve the performance or to save power, simultaneously maintaining the scalability of model.

CPU utilization control using feedback information [8] has shown remarkable performance to provide the real time guarantees through workload variations adaption based on given dynamic feedback. Moreover, the main aim of utilization control is enforcing the proper utilization of schedulable bounds in real time scenario on entire processors. This needs to be achieved despite of uncertainty in workload. Hence it is observed that utilization control is capable of meeting all the deadlines of real-time without proper knowledge of workload such as execution times of task. Further power aware utilization is focused by few researchers to achieve the reduction in power consumption and real time guarantees [9]–[11], however the existing work on this mainly depends on the DVFS through assumption that execution time of task can be easily adapted along with CPU frequency. More over the assumption is very much valid considering the real time scenario for task that are memory intensive and computation intensive and have 75% of instructions as store or load. Furthermore, when particular processor is in process CPU frequency and memory intensive tasks are set to highest level, further utilization can also exceed the schedulable bound this results in missing undesired deadlines. Moreover, cache size is divided to core and can be maximized to minimize the cache access latency and cache miss rate which is occurred due to the less memory access delay. Hence CPU utilization is lowered for efficient real time scenario, further if utilization is comparatively lower than given bound while the frequency is at the lowest level, size of active cache can be minimized further and other cache units which are rarely used can be used through putting low power mode to reduce the cache leakage power [12].

In this paper, two-phase utilization was proposed for energy efficiency in the real time scenario for heterogeneous multi-core processing environment, at core level this mechanism utilizes dynamic L2 cache partition and per-core DVFS for addressing the objectives such as reducing the core energy consumption and controlling the CPU utilization for each core. Moreover, utilization occurred due to the periodic real time can be determined through frequency independent and frequency dependent execution time. Further cache partitioning and per-core DVFS is used for adopting the dependent and independent portions of frequency, respectively [13], [14]. However, a main challenge here is traditional control theory optimization model [15] does not handle the optimization objectives. Further, the decision making process to optimize voltage-frequency scaling and last level cache controller are made separately, leading to increased quality of service (QoS) violation when last level cache controller aims to reduce global last level caches. Existing cache optimized based workload scheduling achieves very poor energy efficiency as they don't consider optimizing voltage-frequency scaling in dynamic manner. For example, using less cache resource for execution of workload can lead to increased scaling of voltage-frequency; thus higher energy for task execution is incurred due to quadratic effects. Hence in this paper an efficient resource management technique is developed which was based on the multi-objective control theory [16], [17] to optimize the above two objectives (i.e., utilize cache resource more efficiently with minimal execution time for scheduling scientific workload under heterogeneous cloud computational framework). The efficient resource utilization (ERU) methodology are modelled by employing efficient cache resource usage by optimizing voltage-frequency in dynamic manner and last level cache partitioning in heterogeneous cloud framework. The ERU

model can coordinate the cache size from the given core and further the dynamic cache is resized to reduce the leakage in power consumption of last level caches (LLCs).

The significance of efficient resource utilization model:

- Here efficient resource utilization for execution of data-intensive application for heterogeneous cloud framework is presented.
- ERU model reduces LLC failures with better cache and V/F scaling optimization in dynamic manner; thus utilizes resource more efficiently.
- The ERU methodology achieves very good outcomes when compared with standard resource utilization methodology considering performance metrics such as power consumption, energy efficiency, and execution time.

The paper organization is as follows: in section 2, survey for underlying benefit and limitation of using state-of-art workload scheduling is described. In section 3, the proposed efficient resource utilization working model is described. In section 4, experiment outcome obtained by ERU over various existing workload scheduling model is detailed. Lastly, the research is concluded and future direction work are discussed.

2. LITERATURE SURVEY

This section presents analysis of various existing workload scheduling algorithm under cloud computing environment. In [18]–[20] observed that energy is one of the major concern while designing the multicore chips, here performance and power are two primary energy components which are inversely related to each other, in here multicore chips optimization which process on the parallel load using either performance optimization or power optimization. Hence to achieve that machine learning model was developed based on the dynamic and global controller of power management, moreover controller is used for reducing the power consumption and increases the performance in given power budget. Further it is observed that controller is scalable and does not possess much overhead as there is increase in demand. In [21] observed that existing methods tries to switch off several processors through combining the task on fewer processors for reducing the energy consumption for deadline constrained. However, it is observed that turning of processor might not be necessary to reduce the energy consumption, hence they proposed energy-aware processor merging (EPM) mechanism to choose the particular processor to switch off for energy consumption and quick-EPM was developed to minimize the computational overheads. In [22] proposes cost and energy aware scheduling (CEAS) technique for the cloud scheduler to reduce the workflow execution cost and minimize the energy consumption which meets the deadline prerequisite. In general, CEAS comprises five algorithms, at first virtual machine (VM) selection approach is used that applies the cost utility concept to map the task to their optimal VM-types through the make span constraint. Later two tasks aware technique were employed to minimize the energy consumption and execution cost, further to reuse the VM Instance, VM reuse policy is developed and at last slack time reclamation gets utilized to reduce the energy of these VM Instances.

In [23] observed that any increase in chip temperature possesses various circuit errors, also there is large increase in leakage power consumption. Hence it applied task migration or traditional digital terrain model (DTM) technique for reducing the core temperature as these cores possess high temperature, also to compensate with high demand in data, last level cache (LLC) is attached which helps in reducing on leakage of power by occupying the chip area. Further to reduce the power consumption cache size are made to shrink dynamically, shrinking of cache size not only helps in leakage of power consumption but also helps in creating on chip thermal buffers to further minimize the temperature of chip though exploiting heat transfer. Moreover, resizing of cache are carried out based on cache hotspot generated while execution. In [24] proposed lead i.e. learning enabled-EAD (energy aware dynamic voltage) scaling for given multicore architecture using reinforcement learning and supervised learning. Further lead groups the link and its router into the same voltage-frequency domain and further implements the management strategies of proactive DVFS which mainly rely on the machine learning based offline model to provide the voltage-frequency selection among the pairs of voltage/frequency. Further three supervised learning model were developed based on energy/throughput change, buffer utilization change and buffer utilization, these allows the proactive mode selection mode on the basis of absolute prediction. Further reinforcement learning models were developed which optimizes selection of DVFS mode directly and also removes the requirement for threshold and label engineering.

In [25] showed heterogeneous multi-core processing is adopted mainly in embedded system, as it provides the energy consumption minimization through applying the popular technique like DPM and DVFS. Moreover, effective management of energy based technique exploits the software and hardware level energy minimization technique. energy efficiency partitioning (EEP) is a software level technique where task

allocation to the given heterogeneous clusters impacts the whole energy model. Hence a technique was developed which couples the energy efficient partition problem along with task scheduling as task differ in terms of system on a chip (SoC) circuitry, active processing, execution path, I/O access, memory, cache and instruction mix, these affects the demand in power. Moreover, hardware frequency scaling is used for scaling to minimize the model energy. In [17], [26] proposed two resourceful workflow scheduling technique which considers the monetary cost as well as make span. Hence at first single objective workflow is developed named deadline-constrained cost optimization for hybrid clouds (DCOH) which was mainly deadline constrained cost optimized to minimize the scheduling cost under the given deadline. Furthermore, considering DCOH, multi-objective optimization is proposed for hybrid cloud named MOH to optimize the monetary cost and execution time of workflows.

In [18] observed that single algorithm does not possess optimal solution under different power settings, dynamic slack and various workloads, further the device configuration variation affects the DVFS algorithm. Hence considering the adaptability this paper focused on developing the reinforcement learning that takes execution technique set which are specialized to handle the various conditions and switches to the best technique considering the situations. In [19] developed load balancing approach for allocating the non-real time on the given heterogeneous nodes, further they introduced the processing node frequency of the whole cycle for all the jobs that are assigned. In [20] developed an integer linear programming (ILP) based thread which takes the input through hardware performance which determines the characteristics of thread. Here they used the last level cache and instructions per second as measure of memory bandwidth and CPU load. Moreover, they used performance metric for optimizing the global thread to the core assignment. However, it is not suitable for the real time scenario and it requires to solve the problem of ILP periodically and hence this incurs the marginally scheduling overhead.

From extensive survey it is seen the existing workload scheduling adopting multi-objective optimization induce computation complexity because of NP-hard problem. Further, DVFS based methodologies are highly influenced by fluctuation of clock frequency and when supply voltage is low these model induce significant degradation in performance. Further, focusing only on workload execution makespan metric will result in improper calculation of energy dissipation. This is because different task will have diverse execution paths, intrinsic cache usage, I/O access pattern. Further, it is noticed that power will not always be same even if workload is executed on same kind of processing element. This is because the jobs with higher cache and memory accessibility would incur higher energy. Thus, for overcoming research problem this work presents an efficient resource utilization model for executing scientific workload on heterogeneous computational environment in next section.

3. AN EFFICIENT RESOURCE UTILIZATION TECHNIQUE FOR SCHEDULING SCIENTIFIC WORKLOAD IN CLOUD COMPUTING ENVIRONMENT

This section present efficient resource utilization (ERU) technique for scheduling scientific workload on heterogeneous cloud framework. Here the cache resource is utilized more efficiently in order to improve cloud resource utilization outcomes. The ERU is designed in such a way that it minimizes LLC failures under shared caching environment. Here VM migration is done for meeting cache constraint in reducing LLC failures. The ERU methodology for executing workload in heterogeneous cloud framework is done through two phased as shown in Algorithm 1. First, for enhancing the bandwidth usage and system capacity, the virtual computing node shares its cache memory. Second, reconfiguration of virtual computing machines is done whenever last level cache failures occurs.

Algorithm 1: Efficient resource utilization (ERU) technique for executing data-intensive workflow in heterogeneous computing framework.

```

Step 1. Start
Step 2. Compute and establish  $N_L$ 
Step 3. Compute and establish  $W_L$ 
Phase 1-
Step 4. For each processing node  $j$  from 1 to  $y$  do
Step 5.  $nx_j$ ←collects ( $j$ )
Step 6.  $W_L$ ←sort ( $nx_j$  )
Step 7. Obtain ( $W_L$  )
Step 8. End for
Phase 2-Obtain processing node with maximal and minimal last level cache failures
Step 9. Maximal Node←find Maximal Node ( $N_L$  )
Step 10. Minimal Node←find Minimal Node ( $N_L$  )
Step 11. Maximal VCM←find maximal VCM (Maximal Node)
Step 12. Minimal VCM←find minimal VCM (Minimal Node)
Step 13. if  $T < \text{maximal Node}_{LLC} - \text{minimal Node}_{LLC}$  then

```

Step 14. Interchange (maximal VCM, minimal VCM)
 Step 15. Stop.

The ERU methodology is designed to utilize cache resource more efficiently meeting energy constraint of heterogeneous platform [21]. Thus, the energy incurred $P(t)$ in heterogeneous platform for respective (i.e., t^{th}) interval of time is computed as follows

$$P(t) = e(t)M_a \quad (1)$$

where, $e(t)$ depicts the overall power induced considering frequency level of processing core B_k with L2 cache size which is constant in nature with respect to respective time sessions M_a for executing workload. Here M_a represents the operating period for discharging different information of entire workload-task in process of t^{th} session. Here the cache partitioning size with energy constraint are configured and frequency prerequisite for executing under such constraint is obtained as follows

$$a_k(t) | 1 \leq k \leq i, f_k(t) | 1 \leq k \leq i \quad \min_{\sum_{k=1}^i [V_k - v_k(t)]^2} \quad (2)$$

$$a_k(t) | 1 \leq k \leq i, f_k(t) | 1 \leq k \leq i \quad \min P(t) \quad (3)$$

where $v_k(t)$ represents core utilization B_k in t^{th} session instance, V_k depicts resource utilization sets $V = [V_1, \dots, V_i]^T$ for respective frequency range of $[R_{\uparrow,k}, R_{\downarrow,k}]$ for each processing core B_k , $\{a_k(t) | 1 \leq k \leq i\}$ depicts cache memory partition size and $\{f_k(t) | 1 \leq k \leq i\}$ depicts processing core operating frequency at t^{th} session instance for decreasing variance amid core utilization $v_k(t)$ and utilization sets (V_k).

The processing element of cloud computing framework is composed of two cache elements namely L1 cache and L2 cache. These caches are shared among different core in multi-core shared computational framework. Here each processing element has DVFS capability. Thus, aid in saving significant amount of energy resource. The cache memory is portioned for carrying out various task. Therefore, the L2 cache partition size is represented by $a_k(t)$ considering processing core size B_k . The peak frequency size under certain B_k is represented by $f_{k_1}(t)$. The (2) and (3) must satisfy following constraint

$$R_{\downarrow,k} \leq f_{k_1}(t) \leq R_{\uparrow,k} \quad \text{where, } (1 \leq k \leq i) \quad (4)$$

$$\sum_{k=1}^i a_k(t) \leq A \quad (5)$$

where A depicts total L2 cache size available in heterogeneous cloud computing environment.

The (2) depict the minimum energy dissipation for executing certain task under heterogeneous cloud computational framework under certain power generation $e(t)$ for t^{th} session instance. The (3) depicts the processing machine frequency leis within in range of each processing core using ERU model. The frequency range variation depends on kind of processing element being used. The (5) depicts summation of every partitioned cache memory which is almost equal to total memory available.

For each processing core, the variation among resource (i.e., core) utilization $v_k(t)$ and utilization sets V_k is reduced utilizing cache aware resource utilization method by modifying the cache partition size and its core frequencies. However, optimizing frequencies based on different cache partition size in static manner induces overhead and affect the processing time of heterogeneous computational environment. Thus, for improving processing time a dynamic optimization model is presented. The model maintain ideal relationship among balancing $v_k(t)$, core frequency $f_k(t)$, and optimizing feature $a_k(t)$ in t^{th} session instance. First, for respective core B_k , the dynamic optimization model gives an ideal relationship among $b_{kp}(t)$, job operational time M_{kp} and optimizing feature $f_k(t)$ in t^{th} session instance and $a_k(t)$. Then, the relationship parameter $b_{kp}(t)$ can be optimized in different manner such as frequency independent or frequency dependent as described in the (6)

$$b_{kp}(t) = s_{kp}(t) + i_{kp} \cdot (f_k(t))^{-1} \quad (6)$$

where $s_{kp}(t)$ depicts frequency, independent segment considering respective operational session instance M_{kp} as processing time of I/O devices does not rely upon frequency of individual core and $i_{kp} \cdot (f_k(t))^{-1}$

depicts frequency dependent segment because it depends on frequency of operating cores. The reserved cache memory for respective job operational instance M_{kp} considering certain I/O device does not take part for executing jobs can be depicted as $s_{kp}(t)$. The parameter $s_{kp}(t)$ plays an ideal relationship among cache failure and cache memory size. The ideal relationship among $s_{kp}(t)$, $a_{kp}(t)$, and allocated caches for heterogeneous computational framework B_k can be estimated using following (7):

$$s_{kp}(t) = \begin{cases} D_{kp}a_{kp}(t) + H_{kp} & 0 \leq a_{kp}(t) \leq X_{kp} \\ Constant & a_{kp}(t) \geq X_{kp} \end{cases} \quad (7)$$

where D_{kp} , H_{kp} are quantified jobs features, and X_{kp} depicts the operational set size within job operational session instance M_{kp} . The (7) shows that whenever operation set size X_{kp} is higher than $a_{kp}(t)$, the cache memory size improves and aiding in minimizing operation session instance. In similar manner, if operation set size X_{kp} is lower than $a_{kp}(t)$, then cache failure will be higher and can't be addressed by allocating additional cache memory. Thus, for managing job execution of real-time scientific application, the relationship among total independent frequency and operation session instance of each job in heterogeneous computing processing element B_k and total cache size $a_k(t)$ given to processing element B_k is established using following (8):

$$s_k(t) = \begin{cases} \sum_p D_{kp}' a_k(t) + \sum_p H_{kp} & 0 \leq a_k(t) \leq X_k \\ Constant & a_k(t) \geq X_k \end{cases} \quad (8)$$

where $D_{kp}' = \frac{D_{kp}a_{kp}(t)}{a_k(t)}$ and $X_k = \sum_p X_{kp}$. The (8) depicts cumulating of (7) for every job on heterogeneous computational processing element B_k . Then, the proposed ERU model aids in minimizing interference among different processing element shared caches can be described using following (9):

$$h_k(t) = \sum_p i_{kp} q_{kp} \cdot (\mathbb{f}_k(t))^{-1} + \sum_p D_{kp}' q_{kp} a_k(t) + \sum_p H_{kp} q_{kp} \quad (9)$$

where $h_k(t)$ depicts the estimated processing element resource utilization and q_{kp} depicts job rate within operational session instance M_{kp} for heterogeneous computing environment B_k . Using (9), it can be shown that $h_k(t)$ is proportionally inverse with respect to processing element frequency $\mathbb{f}_k(t)$. The estimated variation in resource utilization $\Delta h_k(t)$ for heterogeneous computing framework B_k is described using following (10)

$$\Delta h_k(t) = l_k(t) \sum_p i_{kp} q_{kp} + \Delta a_k(t) \sum_p D_{kp}' q_{kp} \quad (10)$$

where $\Delta h_k(t)$ is a linear function with respect to $l_k(t)$ and $\Delta a_k(t)$, $l_k(t) = \left(\frac{1}{\mathbb{f}_k(t)}\right) - \left(\frac{1}{\mathbb{f}_k(t-1)}\right)$ and $\Delta a_k(t) = a_k(t) - a_k(t-1)$. The (10) substitute direct frequency utilization of processing element $\mathbb{f}_k(t)$ to $l_k(t)$. The (10) verifies that $\Delta h_k(t)$ proportional with respect to i_{kp} and D_{kp}' . Therefore, the cost function of heterogeneous computational environment can be minimized using regulator for heterogeneous processing element B_k using following (11) to (13):

$$Z_k(t) = \sum_{c=1}^E \|v_k(t+c-1|t) - \beta \mathbb{f}_k(t+c-1|k)\|^2 + \|u_k(t|t) - u_k(t-1|t)\|^2 \quad (11)$$

$$R_{\downarrow,k} \leq \mathbb{f}_k(t) \leq R_{\uparrow,k} \quad (12)$$

$$a_k(t) \leq a_{quota,k} \quad (13)$$

where $\beta \mathbb{f}_k(t+1|t)$ depicts the pattern considering resource utilization influence/feature $v_k(t+c-1|t)$ must change its present utilization influence $v_k(t)$ to V_k , $u_k(t) = \begin{bmatrix} l_k(t) \\ \Delta a_k(t) \end{bmatrix}$ and E depicts the computed range for estimating the pattern of the device in E operational session instances. The cache size $a_k(t)$ for heterogeneous computational framework B_k is bounded by $a_{quota,k}$ for satisfying (5). Thus, using dynamic model, the least square problem can be minimized, and cache memory can be optimized in efficient manner. The power consumption optimization can be described using efficient resource utilization model can be described using following (14) to (16):

$$e_k(t) = S_k f_k(t)^3 + Y_k a_k(t) + C_k \quad (14)$$

$$R_{\downarrow,k} \leq f_k(t) \leq R_{\uparrow,k} \quad (15)$$

$$a_k(t) \leq a_{quota,k} \quad (16)$$

where S_k, Y_k , and C_k depicts the power factors of the heterogeneous computation framework processing element of virtual computing nodes. The power consumption of heterogeneous computational framework processing element can be described as cumulative of power consumed by different shared caches and processing element. The total power consumption are dependent on leakage power C_k and dynamic power component $S_k f_k(t)^3$. Thus, the cache memory power consumption can be optimized using the ERU model. The benefit of reducing cache resource usage cost plays very important part in workload scheduling. Traditionally, the caching cost is measured in terms size of data used (i.e., high usage means higher cost). However, considering workload deadline prerequisite the novelty of this work is the cache benefits is measured in terms of response time. Therefore the caching benefits \mathcal{D}_g is measure as shown in (17):

$$\mathcal{D}_g = \begin{cases} 0 & Q_g = 0 \\ Q_g * \left(u_{seek} + \frac{T_g}{BW_{cache}} \right) & Q_g \neq 0 \end{cases} \quad (17)$$

where BW_{cache} represent caching I/O bandwidth, T_g represents the task data size, u_{seek} signify time desired for enlisting data in cache partition, and Q_g represents data re-accessibility from the cache. Using [21] the caching cost benefits are Max-Min for addressing data comparability issues

$$\mathcal{D}_{Ben} = \frac{(\mathcal{D}_g - \mathcal{D}_\downarrow)}{(\mathcal{D}_\uparrow - \mathcal{D}_\downarrow)} \quad (18)$$

where \mathcal{D}_\downarrow describes minimal result of cache cost benefit and \mathcal{D}_\uparrow represents maximal result of caching cost benefit. Further, replacing data from cache may induce certain cost. For calculating replacing cost this work takes the number of unused (i.e., garbage) partitions of a data as the measurement. If the particular data blocks are in active/hot mode, these data have less probabilities of being replaced. Thus, they exhibit less replacing cost. The access probabilities of each data block considering window sampling x can be defined using following (19)

$$Q_\ell = \frac{access_\ell}{access} \quad (19)$$

where Q_ℓ defines access probabilities of data blocks ℓ , $access_\ell$ depicts number of time the data blocks ℓ being accessed, $access$ depicts total number times access within time period x . Let us partition the session window into smaller sessions $x_1, x_2, \dots, x_\sigma$. Then, data blocks ℓ access probabilities in different session segments are established:

$$\begin{cases} Q_{\ell_1} = access_{\ell_1} / access_1 \\ Q_{\ell_2} = access_{\ell_2} / access_2 \\ \dots \\ Q_{\ell_\sigma} = access_{\ell_\sigma} / access_\sigma \end{cases} \quad (20)$$

In which, Q_{ℓ_j} depicts the probabilities of data blocks in session x_j , $access_{\ell_j}$ depicts number of times the data blocks ℓ is being accessed in session x_j , $access_j$ depicts total number of times all the data blocks is being accessed in session x_j . The active mode data blocks ℓ can be established using following (12):

$$active_\ell = \frac{Q_{\ell_2} * Q_{\ell_3} * Q_{\ell_4} * Q_{\ell_5} \dots * Q_{\ell_\sigma}}{Q_{\ell_1} * Q_{\ell_2} * Q_{\ell_3} * Q_{\ell_4} \dots * Q_{\ell_{\sigma-1}}} = \frac{Q_{\ell_\sigma}}{Q_{\ell_1}} \quad (21)$$

Thus, replacing cost of data can be described using following (22)

$$\mathcal{R}_g = \sum_{\ell=1}^{\sigma} \frac{Q_{\ell}}{\text{active}_{\ell} * \mathcal{T}_{\ell}} \quad (22)$$

where σ depicts partition size of data blocks for particular task, \mathcal{T}_{ℓ} standard data block size, active_{ℓ} depicts active mode of data blocks ℓ , Q_{ℓ} data block ℓ access probabilities in session x . Similar to cache cost we apply max-min for computing replacing cost where \mathcal{R}_{\downarrow} describes minimal result of caching resource interchanging cost benefits and \mathcal{R}_{\uparrow} describes maximal result of caching resource interchanging cost benefits. In next section, experiment is conducted for validating cache aware resource utilization methodology over standard resource utilization methodology using data-intensive workload with different job-size. The ERU aid in achieving between energy efficiency and meet task deadline prerequisite with minimal execution time.

4. RESULTS AND ANALYSIS

Here, the performance of the system is tested on scientific workflow small ribonucleic acid (sRNA) identification protocol using high-throughput technology (SIPHT) using proposed efficient resource utilization model to verify high efficiency and lower energy consumption of proposed cache aware resource utilization model in heterogeneous computational framework. In this modern era, heterogeneous multi-core architectures have impressed all over across the globe in different areas such as industries, trading departments, and medical applications. Thus, due to extensive demand of multi-core architectures, cloud computing has also stated to add multi-core architecture support. Moreover, graphics processing unit (GPU) instances are favored in contrast to traditional CPU-based resources to improve speed and efficiency of the system. However, improper resource scheduling and enormous amount of energy consumption can reduce the performance of the model in an extensive manner. Therefore, a cache aware efficient resource utilization scientific workload scheduling method is introduced to ensure low energy consumption, high performance of the model and proper resource scheduling using heterogeneous multi-core architectures. This technique helps to speed up the process and performance of the model.

Here, we have conducted various experiments using the proposed ERU model to find energy consumption, power sum, simulation time and average power results which are demonstrated in Table 1 with the help of SIPHT scientific dataset for various jobs 30, 60, 100 and 1000. Our proposed technique ensures very less energy consumption for running SIPHT scientific dataset for SIPHT 30 is 2812.991014 watts, SIPHT 60 is 3158.219947 watts, SIPHT 100 is 3174.261302 watts and SIPHT 1000 is 11211.22691 watts demonstrated in Table 1 which is highly reduced compared with other state-of-art techniques using similar statistics. Table 2 also demonstrates Execution time to finish the task using the proposed ERU technique for various jobs as 30, 50, 100 and 1000 with the help of SIPHT benchmark. The average power outcomes for SIPHT 30 is 21.99945901 W, SIPHT 60 is 21.9994593 W, SIPHT 100 is 21.9994591 W and SIPHT 1000 is 21.99946127 W. Here, Table 2 represents average simulation time comparison of proposed ERU method with other state-of-art-techniques using scientific model SIPHT. Further, this section provides graphical representation of our simulated experiments for various jobs using SIPHT scientific dataset and compared outcome achieved with recent standard resource utilization methodology considering different performance metrics such as average power, energy consumption, simulation time and power sum.

Table 1. Energy efficiency and execution time performance evaluation of proposed ERU model over existing DVFS based workload scheduling algorithm

Parameters	DVFS [10]				ERU			
	<i>Sipht</i> 30	<i>Sipht</i> 60	<i>Sipht</i> 100	<i>Sipht</i> 1000	<i>Sipht</i> 30	<i>Sipht</i> 60	<i>Sipht</i> 100	<i>Sipht</i> 1000
Power Sum (W)	10734800.6	22566020.6	33620561.4	335969269.8	9783513.62	10291173.3	9934905.13	16022788.43
Average Power (W)	28.6557284	28.6557203	28.65572104	28.65572239	21.99945901	21.9994593	21.9994591	21.99946127
Power Consumption (Wh)	4367.658563	11228.74085	20813.04776	1070996.931	2812.991014	3158.219947	3174.261302	11211.22691
Simulation Time (sec)	3746.13	7874.87	11732.58	117243.34	4447.16	4677.92	4515.98	7283.26

Figure 1 shows power sum results in contrast to DVFS-based resource utilization methodology using proposed ERU methodology for data-intensive workload dataset SIPHT for different job size as 30, 60, 100 and 1000. The Figure 2 shows average power results in contrast to DVFS-based methodologies using proposed ERU methodology for data-intensive workload dataset SIPHT for different job size as 30, 60, 100

and 1000. The Figure 3 shows energy consumption results in contrast to DVFS-based methodologies using proposed ERU methodology for data-intensive workload dataset SIPHT for different job size as 30, 60, 100 and 1000.

Table 1. Computation efficiency performance evaluation of proposed ERU model over existing multi objective-based and DVFS-based workload scheduling algorithm

DAGs	Number of nodes	Average Simulation time (s)		
		DCOH [17]	DVFS [10]	ERU
Sipht 30	30	178.92	124.871	148.2386667
Sipht 60	60	194.48	131.2478333	77.96533333
Sipht 100	100	175.55	117.3258	45.1598
Sipht 1000	1000	179.05	117.24334	7.28326

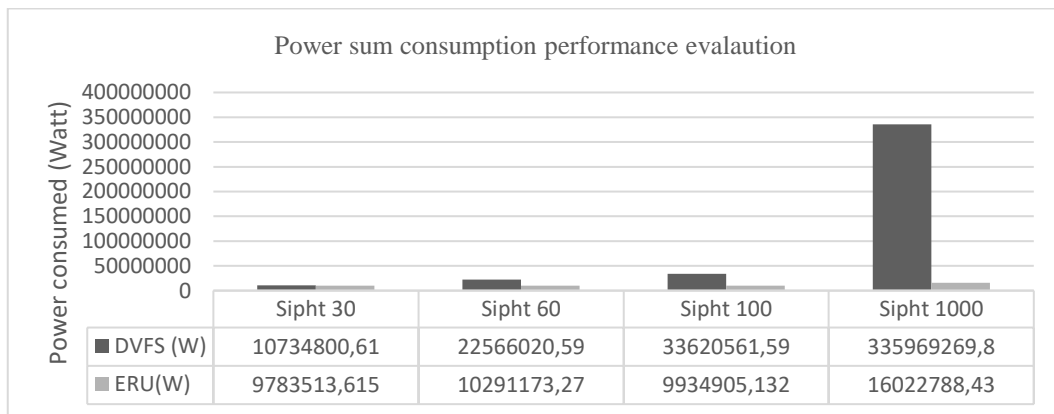


Figure 1. Power sum comparison using proposed ERU model with DVFS-based workload scheduling algorithm

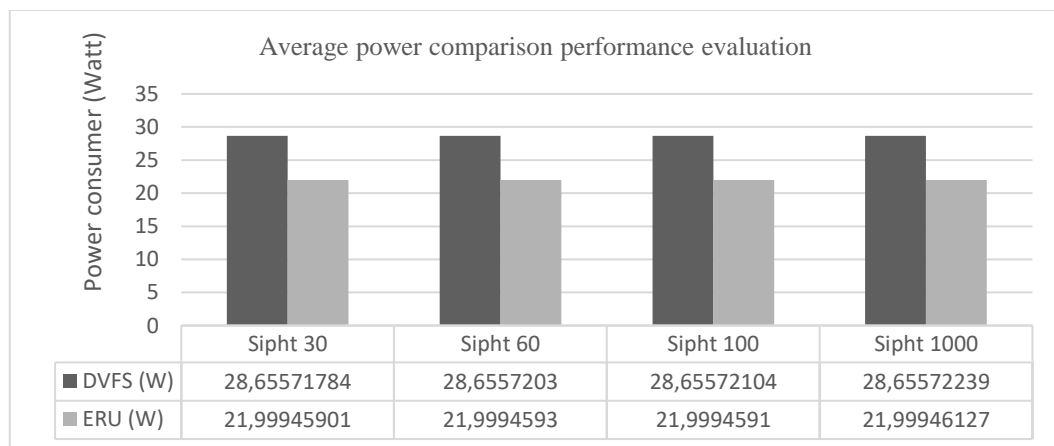


Figure 2. Average power consumption comparison using proposed ERU model with DVFS-based workload scheduling algorithm

The Figure 4 shows execution time results in contrast to DVFS-based methodology using proposed ERU methodology for data-intensive workload dataset SIPHT for different job size as 30, 60, 100 and 1000. The outcomes achieved concludes the supremacy of ERU methodology in terms of average power, power consumption and power sum using SIPHT scientific dataset. Likewise, Figure 5 shows average execution time assessment with DVFS-based and DCOH methodology using modelled ERU methodology for data-intensive workload benchmark SIPHT for different job size as 30, 50, 100 and 1000.

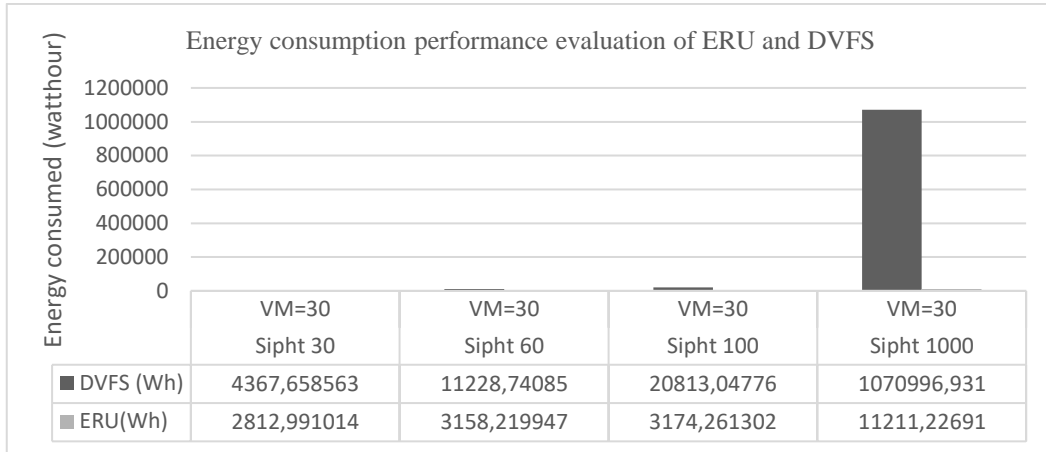


Figure 3. Energy consumption comparison using proposed ERU model with DVFS-based workload scheduling algorithm

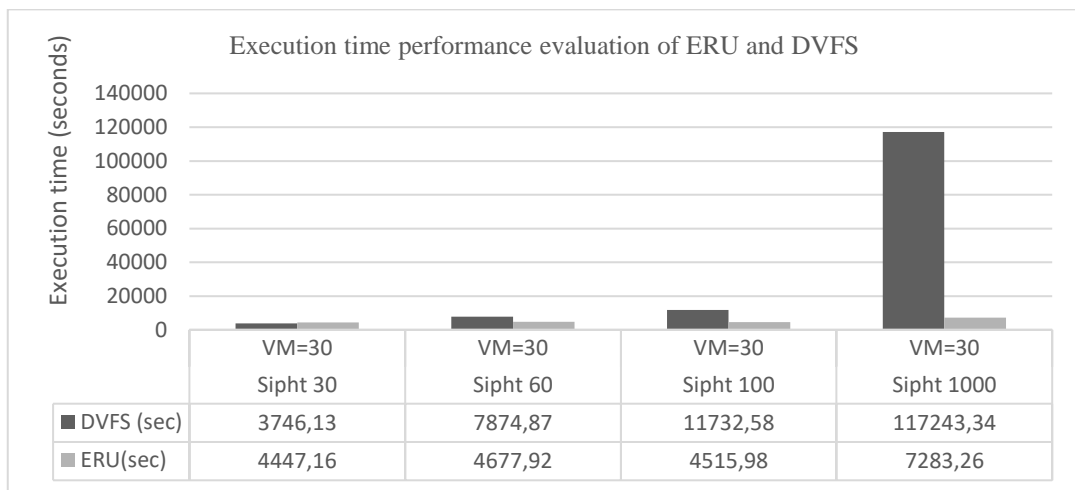


Figure 4. Execution time comparison using proposed ERU model with DVFS-based workload scheduling algorithm



Figure 5. Average execution time comparison using proposed ERU model with multi-objective based DVFS-based workload scheduling algorithm

5. CONCLUSION

Workload scheduling considering dynamic cache memory optimization under heterogeneous multicore environment is a challenging task. Recently, number of methodologies have aimed at bringing good tradeoffs among reducing energy and improving workload execution performance. An effective way of reducing energy dissipation is to employ DVFS technique; and for utilizing resource more efficiently and meet task deadline requires effective cache optimization technique. Thus, this paper presented a two phase cache resource optimization technique enabling V/F scaling in dynamic manner. From experiment it can be seen ERU improves energy efficiency by 44.29% over existing DVFS workload scheduling technique. Further, reduces execution time 43.215% and 61.72% over existing DVFS and DCOH workload scheduling technique, respectively. The proposed ERU workload scheduling model brings good tradeoffs in meeting task deadline with minimal execution time and energy consumption. Future work would consider evaluating performance of ERU considering diverse data intensive workload; and also consider employing evolutionary or deep learning technique to monitor and optimize QoS for executing workload.

ACKNOWLEDGEMENTS

The authors would like to express their cordial thanks to Visvesvaraya Technological University for the much-valued support and advice.




REFERENCES

- [1] S. Mittal, "A survey of techniques for improving energy efficiency in embedded computing systems," *International Journal of Computer Aided Engineering and Technology*, vol. 6, no. 4, pp. 440–459, 2014, doi: 10.1504/IJCAET.2014.065419.
- [2] S. S. Jha, W. Heirman, A. Falcón, J. Tubella, A. González, and L. Eeckhout, "Shared resource aware scheduling on power-constrained tiled many-core processors," *Journal of Parallel and Distributed Computing*, vol. 100, pp. 30–41, Feb. 2017, doi: 10.1016/j.jpdc.2016.10.001.
- [3] A. Das, R. A. Shafik, G. V Merrett, B. M. Al-Hashimi, A. Kumar, and B. Veeravalli, "Reinforcement learning-based inter- and intra-application thermal optimization for lifetime improvement of multicore systems," in *Proceedings of the The 51st Annual Design Automation Conference on Design Automation Conference-DAC '14*, 2014, pp. 1–6, doi: 10.1145/2593069.2593199.
- [4] M. Otoom, P. Trancoso, H. Almasaeid, and M. Alzubaidi, "Scalable and dynamic global power management for multicore chips," in *Proceedings of the 6th Workshop on Parallel Programming and Run-Time Management Techniques for Many-core Architectures-PARMA-DITAM '15*, 2015, pp. 25–30, doi: 10.1145/2701310.2701312.
- [5] U. A. Khan and B. Rinner, "Online learning of timeout policies for dynamic power management," *ACM Transactions on Embedded Computing Systems*, vol. 13, no. 4, pp. 1–25, Dec. 2014, doi: 10.1145/2529992.
- [6] H. Jung and M. Pedram, "Supervised learning based power management for multicore processors," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 29, no. 9, pp. 1395–1408, Sep. 2010, doi: 10.1109/TCAD.2010.2059270.
- [7] Z. Chen and D. Marculescu, "Distributed reinforcement learning for power limited many-core system performance optimization," in *Design, Automation and Test in Europe Conference and Exhibition (DATE), 2015*, 2015, pp. 1521–1526, doi: 10.7873/DATE.2015.0992.
- [8] M. Otoom, P. Trancoso, M. A. Alzubaidi, and H. Almasaeid, "Machine learning-based energy optimization for parallel program execution on multicore chips," *Arabian Journal for Science and Engineering*, vol. 43, no. 12, pp. 7343–7358, Dec. 2018, doi: 10.1007/s13369-018-3079-4.
- [9] G. Xie, G. Zeng, R. Li, and K. Li, "Energy-aware processor merging algorithms for deadline constrained parallel applications in heterogeneous cloud computing," *IEEE Transactions on Sustainable Computing*, vol. 2, no. 2, pp. 62–75, Apr. 2017, doi: 10.1109/TSUSC.2017.2705183.
- [10] Z. Li, J. Ge, H. Hu, W. Song, H. Hu, and B. Luo, "Cost and energy aware scheduling algorithm for scientific workflows with deadline constraint in clouds," *IEEE Transactions on Services Computing*, vol. 11, no. 4, pp. 713–726, Jul. 2018, doi: 10.1109/TSC.2015.2466545.
- [11] K. Li, "Power and performance management for parallel computations in clouds and data centers," *Journal of Computer and System Sciences*, vol. 82, no. 2, pp. 174–190, Mar. 2016, doi: 10.1016/j.jcss.2015.07.001.
- [12] Y.-H. Chen, Y.-L. Tang, Y.-Y. Liu, A. C.-H. Wu, and T. Hwang, "A novel cache-utilization-based dynamic voltage-frequency scaling mechanism for reliability enhancements," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 3, pp. 820–832, Mar. 2017, doi: 10.1109/TVLSI.2016.2614993.
- [13] S. Chakraborty and H. K. Kapoor, "Analysing the role of last level caches in controlling chip temperature," *IEEE Transactions on Sustainable Computing*, vol. 3, no. 4, pp. 289–305, Oct. 2018, doi: 10.1109/TSUSC.2018.2823542.
- [14] Q. Fettes, M. Clark, R. Bunescu, A. Karanth, and A. Louri, "Dynamic voltage and frequency scaling in NoCs with supervised and reinforcement learning techniques," *IEEE Transactions on Computers*, vol. 68, no. 3, pp. 375–389, Mar. 2019, doi: 10.1109/TC.2018.2875476.
- [15] A. Suyagh and Z. Zilic, "Energy and task-aware partitioning on single-ISA clustered heterogeneous processors," *IEEE Transactions on Parallel and Distributed Systems*, vol. 31, no. 2, pp. 306–317, Feb. 2020, doi: 10.1109/TPDS.2019.2937029.
- [16] Z. Zhu, G. Zhang, M. Li, and X. Liu, "Evolutionary multi-objective workflow scheduling in cloud," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 5, pp. 1344–1357, May 2016, doi: 10.1109/TPDS.2015.2446459.
- [17] J. Zhou, T. Wang, P. Cong, P. Lu, T. Wei, and M. Chen, "Cost and makespan-aware workflow scheduling in hybrid clouds," *Journal of Systems Architecture*, vol. 100, p. 101631, Nov. 2019, doi: 10.1016/j.sysarc.2019.08.004.
- [18] F. M. ul Islam and M. Lin, "Hybrid DVFS scheduling for real-time systems based on reinforcement learning," *IEEE Systems Journal*, vol. 11, no. 2, pp. 931–940, Jun. 2017, doi: 10.1109/JSYST.2015.2446205.
- [19] M. U. Karim Khan, M. Shafique, A. Gupta, T. Schumann, and J. Henkel, "Power-efficient load-balancing on heterogeneous computing platforms," in *Proceedings of the 2016 Design, Automation and Test in Europe Conference and Exhibition (DATE)*,





- 2016, pp. 1469–1472, doi: 10.3850/9783981537079_0898.
- [20] V. Petrucci, O. Loques, D. Mossé, R. Melhem, N. A. Gazala, and S. Gobriel, “Energy-efficient thread assignment optimization for heterogeneous multicore systems,” *ACM Transactions on Embedded Computing Systems*, vol. 14, no. 1, pp. 1–26, Jan. 2015, doi: 10.1145/2566618.
- [21] S. Nagendra Prasad, S. Kulkarni, and P. Venkatareddy, “Cache aware task scheduling algorithm for heterogeneous cloud computing environment,” in *2020 Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN)*, Nov. 2020, pp. 154–158, doi: 10.1109/ICRCICN50933.2020.9296177.
- [22] K. Sumalatha and M. S. Anbarasi, “A review on various optimization techniques of resource provisioning in cloud computing,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 629–634, Feb. 2019, doi: 10.11591/ijece.v9i1.pp629-634.
- [23] M. Tarahomi and M. Izadi, “A hybrid algorithm to reduce energy consumption management in cloud data centers,” *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 9, no. 1, pp. 554–561, Feb. 2019, doi: 10.11591/ijece.v9i1.pp554-561.
- [24] S. Chen, Z. Li, B. Yang, and G. Rudolph, “Quantum-inspired hyper-heuristics for energy-aware scheduling on heterogeneous computing systems,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 6, pp. 1796–1810, Jun. 2016, doi: 10.1109/TPDS.2015.2462835.
- [25] K. Khorramnejad, L. Ferdouse, L. Guan, and A. Anpalagan, “Performance of integrated workload scheduling and pre-fetching in multimedia mobile cloud computing,” *Journal of Cloud Computing*, vol. 7, no. 1, p. 13, Dec. 2018, doi: 10.1186/s13677-018-0115-6.
- [26] L. Chunlin, T. Jianhang, and L. Youlong, “Hybrid cloud adaptive scheduling strategy for heterogeneous workloads,” *Journal of Grid Computing*, vol. 17, no. 3, pp. 419–446, Sep. 2019, doi: 10.1007/s10723-019-09481-3.

BIOGRAPHIES OF AUTHORS







Mr. Nagendra Prasad Sodinapalli     received his Bachelor of Engineering in Electrical and Electronics Engineering and M.Tech in VLSI Design and Embedded Systems, in 2008, from Visvesvaraya Technological University, Belgaum, India. Since 2011, he has been working as Assistant Professor in the Department of Electronics and Communication Engineering, PES Institute of Technology South Campus, Bangalore. His experience includes 12 Years of Academic Teaching and 2 years in Industry. His current research is towards DVFS and Design of Scheduling Algorithms for High Performance Computing Applications. He can be contacted at email: nagendraprasad09@rediffmail.com.







Dr. Subhash Kulkarni     received the M.Tech in Electronic Design and Technology from CEDT IISc Bangalore, in 1995 and the Ph.D. from dept. of E&ECE, IIT Kharagpur in 2002, with more than 31 years of academic teaching experience including 20 years of research experience. He conducted tutorial on Level Sets for Image Analysis at 1st international Conference on Signal and Image Processing between 2006 and 2008. From 2011 to 2019, he has been working as a Professor and Head in the Department of Electronics and communication Engineering. Currently he is Principal, PES Institute of Technology, Bangalore South Campus, Bangalore. His research interests mainly include Math Models in Image Processing, High Speed Computational Vedic Architectures, and Control Systems and Deformable models for Image Analysis H-Infinity Control. He has published 72 Papers in International Journals, 35 Papers in National and International Conferences. Under his guidance 9 Scholars were awarded PhD Degree. He can be contacted at email: sskul@pes.edu.



Mr. Nawaz Ahmed Sharief     received the M.Tech in VLSI and Embedded systems from EPCET Bangalore in 2008. With more than 11 years of industry experience in memory layout design. He also has experience as an application engineer in field of VLSI. Since 2011, he has been working in different nodes of Samsung foundry such as 4 nm, 5 nm, 7 nm. He has also worked on Intel foundry's 32 nm, 22 nm nodes and TSMC's 7 nm, 10 nm, 16 nm, 28 nm nodes. Currently he is working as senior staff engineer in Samsung R&D institute. He can be contacted at email: Nawaz.sharief262@gmail.com.



Mr. Prasanth Venkatareddy     received his Bachelor of Engineering in Electrical and Electronics Engineering and Master of Engineering in Power and Energy Systems, in 2005, from Bangalore University, Bangalore, India and the Ph.D. from department of Electrical Engineering VTU Belgaum. His research interests mainly include Robust control systems. His experience includes 17 Years of Academic Teaching and 2 years in Industry. His current research is towards the design of robust controllers to Electrical Motors in industry. He can be contacted at email: prasanth.v@nmit.ac.in.

AraBERT transformer model for Arabic comments and reviews analysis

Hicham El Moubtahij¹, Hajar Abdelali², El Bachir Tazi³

¹Systems and Technologies of Information Team, High School of Technology, University of Ibn Zohr, Agadir, Morocco

²LISAC Laboratory, Faculty of Sciences Dhar Mahraz, University of Sidi Mohamed Ben Abdellah, Fez, Morocco

³Computer Science department, Polydisciplinary Faculty, University of Sidi Mohamed Ben Abdellah, Taza, Morocco

Article Info

Article history:

Received Sep 27, 2021

Revised Dec 24, 2021

Accepted Jan 4, 2022

Keywords:

AraBERT

Arabic language understanding

Farasa segmentation

Natural language processing

ABSTRACT

Arabic language is rich and complex in terms of word morphology compared to other Latin languages. Recently, natural language processing (NLP) field emerges with many researches targeting Arabic language understanding (ALU). In this context, this work presents our developed approach based on the Arabic bidirectional encoder representations from transformers (AraBERT) model where the main required steps are presented in detail. We started by the input text pre-processing, which is, then, segmented using the Farasa segmentation technique. In the next step, the AraBERT model is implemented with the pertinent parameters. The performance of our approach has been evaluated using the ARev dataset which contains more than 40,000 comments-remarks records relate to the tourism sector such as hotel reviews, restaurant reviews and others. Moreover, the obtained results are deeply compared with other relevant states of the art methods, and it shows the competitiveness of our approach that gives important results that can serve as a guide for further improvements in this field.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Hicham El Moubtahij

Systems and Technologies of Information Team, High School of Technology, University of Ibn Zohr
Agadir, Morocco

Email: h.elmoubtahij@uiz.ac.ma

1. INTRODUCTION

Arabic is an international language, spoken by more than 500 million speakers. It is considered as one of the important Semitic languages family. From the Arabian gulf to the atlantic ocean, Arabic language is administrative and official language of more the 21 countries [1]. Arabic is a rich and complex language in terms of word morphology compared to English, the presence of various dialects is some of the distinguishing prominent factors in the language. Moreover, the large differences between the modern standard Arabic (MSA) and the dialectal Arabic (DA) increase this complexity. It should be noted that MSA is employed for formal (administrative) writing and DA is employed for informal daily communication on social media for example [2]. From the work of Guellil *et al.* [3] published in 2021, the DA is divided into six collections: i) Maghrebi (MAGH), ii) Egyptian (EGY), iii) Iraqi (IRQ), iv) Levantine (LEV), v) Gulf (GLF), and vi) others remaining dialect. On the other hand, the Arabic language used on short messaging system (SMS), chat forums and on social media generally is called "Arabizi" [4]. Its written text is a mixture of Latin characters, numerals and some punctuation. For example, the sentence: "يا لاه نساferou", that is translated into English as "let's travel", is written in Arabizi form as "yallah nsaferou" [5].

Despite its spread usage, there is little research in the field of modern computational linguistics interested in the Arabic language compared to other language. However, in the last years, several research

efforts has been made and many paper appear in various language processing tasks. Practically, the named entity recognition (NER) and the sentiment analysis (SA) are the most difficult tasks of Arabic natural language processing (ANLP) [6].

In order to obtain satisfactory results with tolerable performance for ANLP tasks, research works of the last years have focused on the application of transfer learning by the fine-tuning of large pre-trained language models with a relatively small number of samples. It should be mentioned that this approach is based on a self-supervised pre-trained language models. They allow us to represent the set of words as dense vectors in a vector space of minimum dimension and construct continuous distributed representations for texts. Despite the effectiveness of word embedding, it is unable to take into account the relationship between several words and the meaning of complete sentences in the text. Seeing the next two sentences, " نفسها المرأة " " هذه " .On the one hand, their word embedding representations are identical, and on the other hand, their meanings are entirely different. However, the high computational cost is a disadvantage in the training phase of the models (more than 500 TPU working for weeks). Moreover, a huge corpus is needed for the pre-training phase [7], [8].

In this work, we define and describe the important process and steps of our approach base on Arabic bidirectional encoder representations from transformers (AraBERT) transformer model for the Arabic language understanding (ALU). We can effectively classify the comments and the reviews into positive and negative categories. Hence, we evaluated our model on ARev dataset which contains more than 40,000 comments, hotel, restaurant, product, attraction and movie reviews written on a mixture of standard Arabic and Algerian dialect. The experiments show that our approach achieves very good results.

This reminder of this paper is structured as: in section 2, we present the most important techniques and approaches used in the natural language processing (NLP) field to deal with the ALU problem. Then, in section 3 we describe and clarify our model's architecture where BERT represents its basic core. In section 4, we describe the ARev dataset on which we perform our experiments, then we compare our results with those of relevant methods. Finally, section 5 concludes the paper and outlines the main points of our future works.

2. RELATED WORKS

There are various techniques and approaches used in NLP to solve the problem of ALU. In this section, we briefly present some work in this field. The first work on the meaning of words began in 2013 with the word2vec model developed by Mikolov *et al.* [9], then researchers are oriented towards variants of word2vec like GloVe by Pennington *et al.* [10] in 2014 and fast-text by Mikolov *et al.* [11] in 2017. By the introduction of the concept of "contextual information" in 2018, the results were improved noticeably on different tasks [12], increasingly the structures became larger which had superior representations of words and sentences. From this date, the famous models of language comprehension have been developed, for example: i) bidirectional encoder representations from transformers (BERT) [13], ii) universal language model fine-tuning (ULMFiT) [14], iii) text-to-text transfer transformer (T5) [15], iv) A Lite BERT (ALBERT) [16]. These offered improved performance by exploring different pre-training methods, modified model architectures and larger learning corpora.

Concerning the AraBERT model, we note that there is little work done in relation to other languages. In the following we quote some in chronological order. In 2020, Nada *et al.* [17] proposed a new approach for Arabic text summarizer founded on a general-purpose architecture for natural language understanding (NLU), and natural language generation (NLG): generation and understanding of natural language to summarize the Arabic text by extracting and evaluating the most important sentences at this text.

Alami, a member of the LISAC FSDM-USMBA team at SemEval-2020 [18], proposed an effective method for dealing with the offensive Arabic language in Twitter by using AraBERT embeddings. In the First, they started with pre-processing tweets by handling emojis (containing their Arabic meanings), in the next, they substituted each detected emojis by the special token (MASK) into both fine-tuning and inference phases. Then, by applying the AraBERT model they represent tweets tokens. Finally, to decide whether a tweet is offensive or not, they feed the tweet representation into a sigmoid function. There proposed method achieved the best results, a score equal to 90.17% on OffensEval 2020.

In the next year, Faraj and Abdullah [19] published the best solution for the shared task on sentiment and sarcasm detection in the Arabic language. The objective global of the task is to identify whether a tweet is sarcastic or not. The proposed solution is based on the ensemble technique with AraBERT pre-trained model. In their paper, they started by defining the architecture of the model in the shared task. In the next, the hyperparameter and the experiment tuning that lead to this result are presented in detail. Their model is ranked 5th out of 27 teams with an F1 score of 0.5985.

In the recent work of 2021, Hussein *et al.* [20] worked on an effective approach for fighting Tweets COVID-19 Infodemic by using the AraBERT model. The organisation of their approach is: in the first step,

the goal is to transform Twitter jargon, including emoticons and emojis, into plain text by involving a sequence of pre-processing procedures, and they exploited a version of AraBERT in the second step, which was pre-trained on plain text, to fine-tune and classify the tweets concerning their Label. Their approach can be predict 7 binary properties of an Arabic tweet about COVID-19. By using the dataset provided by NLP4IF 2021, they ranked 5th in the Fighting the COVID-19 Infodemic task results with an F1 of 0.664.

3. METHODOLOGY

The objective of this section is to describe and clarify the architecture of our model based on the AraBERT model, where BERT represents the basic core. Subsection 3.1 show BERT model. Our model based on AraBERT see in subsection 3.2.

3.1. BERT model

BERT stands for bidirectional encoder representations from transformers, it came out of Google AI labs in late 2018. We mention that it is: i) more powerful than its predecessors in terms of results; ii) more powerful than its predecessors in terms of learning speed; iii) once pre-trained, in an unsupervised way, it has its own linguistic “representation”. It can be trained in incremental mode (in a supervised way this time) to specialize the model quickly and with little data; and iv) finally, it can work in a multi-model way, taking as input data of different types such as images and/or text, with some manipulations. It has the advantage over its competitors OpenAI’s generative pre-trained transformer (GPT) and embeddings from language models (ELMo) [12] of being bi-directional, it does not have to look only backwards like OpenAI GPT or concatenate the “back” view and the “front” view driven independently like for ELMo, as shown in Figure 1.

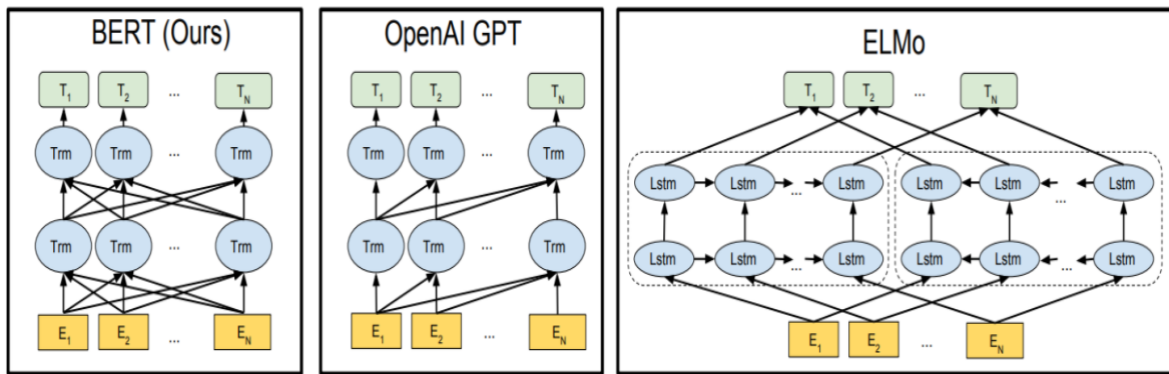


Figure 1. Differences in pre-training model architectures

Examples of what it can do: i) BERT can do the translation. He can even once pre-trained to translate [French/English-English/French] and then [English/German-German/English], translate from French to German without training; ii) BERT can compare the meaning of two sentences to see if they are equivalent; iii) BERT can generate text; iv) BERT can describe and categorize an image; and v) BERT can do logical sentence analysis, i.e. determine if a given element is a subject, a verb, and a direct object complement.

3.1.1. Bidirectional encoder representations from transformers (BERT) architecture

BERT reuses the architecture of transformers (hence the “T” in BERT). Indeed, BERT is nothing more than a superposition of encoders that all have the same structure but do not share the same weights. The “Base” version of BERT consists of 12 encoders. There is another larger version called “Large” which has 24 encoders. Certainly, the large version is more powerful but more demanding on machine resources. The above model has 512 entries, each corresponding to a token. The first entry corresponds to a special token the “[CLS]” for “classification” which allows BERT to be used for a text classification task. It also has 512 outputs of size 768 each (1024 for the base version). The first vector is the classification vector. The output of each of the 12 encoders can be considered as a vector representation of the input sequence. The relevance of this representation is ensured by the attention mechanism implemented by the encoders.

3.1.2. Training procedure

BERT differs from its predecessors (pre-trained NLP models) in the way it is pre-trained on a large dataset consisting of texts from English Wikipedia pages (2,500 million words) as well as a set of books (800 million words). This pre-training is done on two tasks. First, a masked language modelling (MLM) task. Second, a next sentence prediction (NSP) task.

a. Task 1: masked language modelling (MLM)

The objective of this task is to predict the hidden word. Therefore, because of the ability of the transformer architecture to simultaneously take into account the right and left contexts of the target word, this task allows the model to learn even more contextualised representations than one-way models such as ELMo [12]. In practice, target words are sometimes replaced with a special symbol [MASK], or replaced with another random word, or kept as they are as shown in Figure 2.

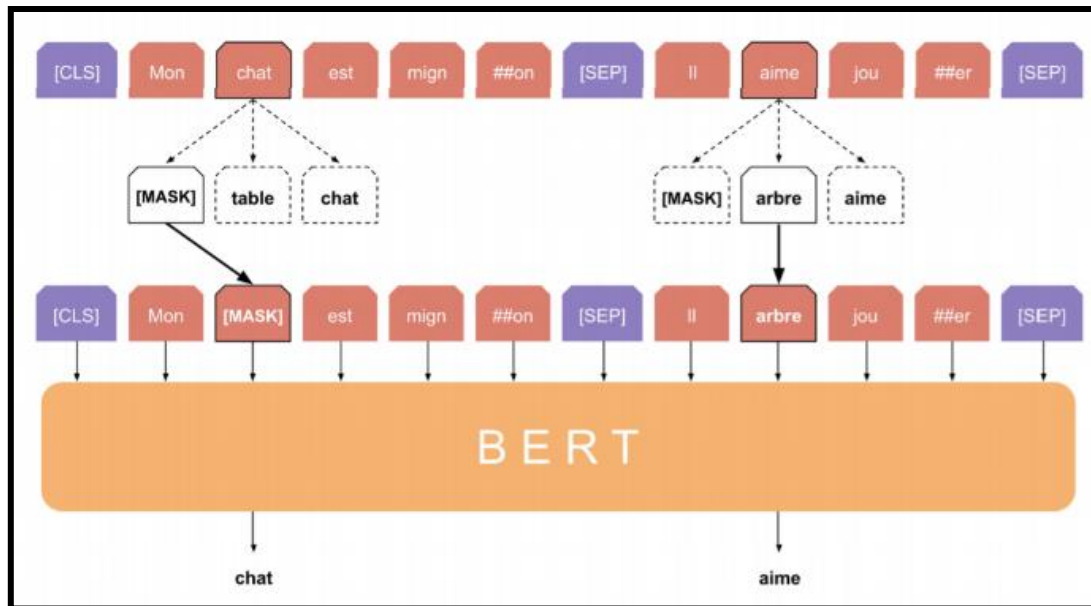


Figure 2. MLM

b. Task 2: next sentence prediction (NSP)

BERT is also trained on a next-sentence prediction task in which it must decide whether two input sentences are consecutive. The rationale for this task is to improve the performance of the model on tasks where the objective is to qualify the relationship between a pair of sentences. In practice, the special symbol representation [CLS] is used to classify each pair of input sentences as well as for any other classification task once the model has been trained.

3.1.3. BERT: fine-tuning

Fine-tuning consists of using a pre-trained version of BERT in a model architecture for a specific NLP task. Adding a basic neural network layer is enough to get very good results. For a text classification task, for example, and more precisely for the analysis of the sentiment of moviegoers' reviews, the architecture of the fitted model may look like this as shown in Figure 3. It is sufficient to add, downstream of BERT, a feed-forward followed by a softmax.

3.2. Our model based on AraBERT

In our approach, we used AraBERT based on the BERT model. It is a widely used model in various NLP tasks for several languages. AraBERT is a pre-trained model for the Arabic language, based on the Google BERT architecture [6] there are six versions of the model: AraBERTv0.1-base, AraBERTv0.2-base, AraBERTv0.2-large, AraBERTv1-base, AraBERTv2-base and AraBERTv2-large. In Table 1 we describe in detail the important information for each version in relation to the pre-training process. The overall view of our model is shown in Figure 4. We have been working on the customer/user review database for the sentiment analysis area, our dataset is titled AREv.

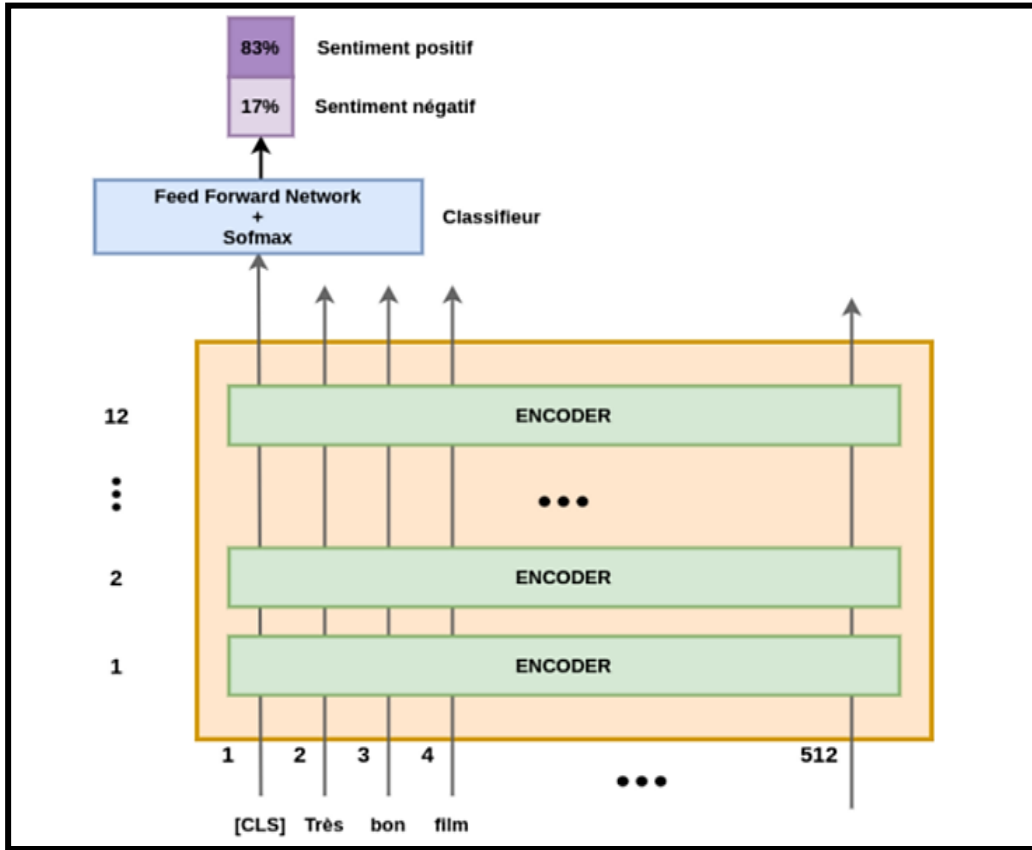


Figure 3. Architecture of the fine-tuning

Table 1. Model pre-training parameters

Model	Size		Pre-segmentation	Sentences	Dataset	
	MB	Param.			Size	Words
AraBERTv0.2-base	543 M	136M	No	200 M	77 GB	8.6 B
AraBERTv0.2-large	1.38 G	371M	No	200 M	77 GB	8.6 B
AraBERTv2-base	543 MB	136M	Yes	200 M	77 GB	8.6 B
AraBERTv2-large	1.38 G	371M	Yes	200 M	77 GB	8.6 B
AraBERTv0.1-base	543 MB	136M	No	77 M	23 GB	2.7 B
AraBERTv1-base	543 MB	136M	Yes	77 M	23 GB	2.7 B

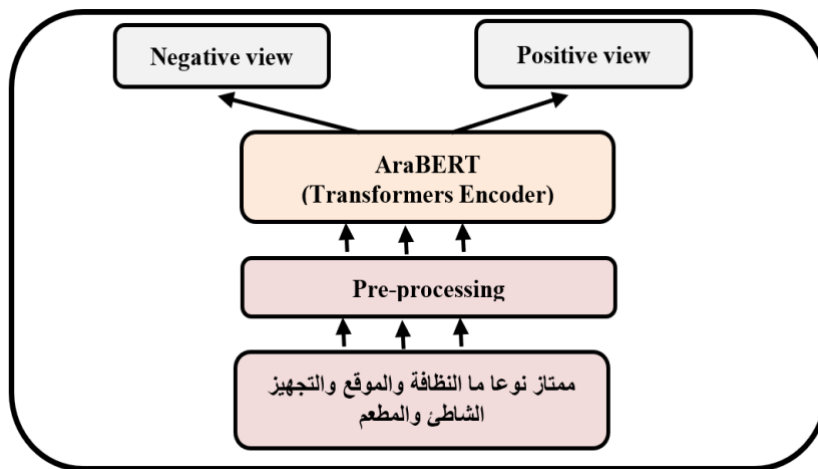


Figure 4. AraBERT architecture overview

At the input of our system, we go through the pre-processing stage where we clean the text of any unsentimental content, such as usernames, hashtags and URLs, and then proceed to segment the text by using the Farasa segmentation [21]. First, we segment the words into stems, prefixes and suffixes. Look for sentence, “ الكتاب – Alkittab “ becomes “ ال + كتا + ب - " Al+kitta+b". Then, in unigram mode, we trained a Sentence Piece [22] on the segmented pre-training dataset to produce a subword vocabulary of more than 59K tokens. It must be noted that before the application of Farasa segmentation, the dataset that is used for pre-training has a size more of 70 GB, more than 8.5 billion words and more than 200 million sentences. To create a well pre-training dataset, we used several websites such as: i) OSIAN Corpus. ii) Arabic Wikipedia dump, iii) Assafir news articles, iv) 1.5 billion word Arabic Corpus, and v) OSCAR unfiltered and sorted In our model based on AraBERT, we successively used two special tokens: Tok1: segment separation (“SEP”) and Tok2: classification (“CLS”). For any classifier, we used it as the first input token which we help us to derive an output vector. Then, in order to obtain the probability distribution on the predicted output classes, we add a simple layer composed of feed-forward and Softmax see (1):

$$P = \text{softmax}(CW) \tag{1}$$

where P is probability of each category, W is matrix of the classification layer, and C is output of the transformers.

4. EXPERIMENT AND RESULTS

4.1. ARev dataset

We evaluated our model on the sentiment analysis task. For this reason, we used the Arabic reviews (Arev) dataset [23]. Using the Facebook API, the ARev dataset is built by more than 100 K comments of the most popular Algerian Facebook pages. We needed tree input for our ARev dataset which are: the Facebook page identifier, the identifier of the Facebook page post and the access token as shown in Figure 5. To enrich our ARev dataset, three open-source datasets of modern standard Arabic and Algerian Arabic comments are used see Table 2. Finally, after pre-processing and deleting the duplicate elements, the dataset is saved in CSV format. The statistics of our dataset are presented in Table 3.

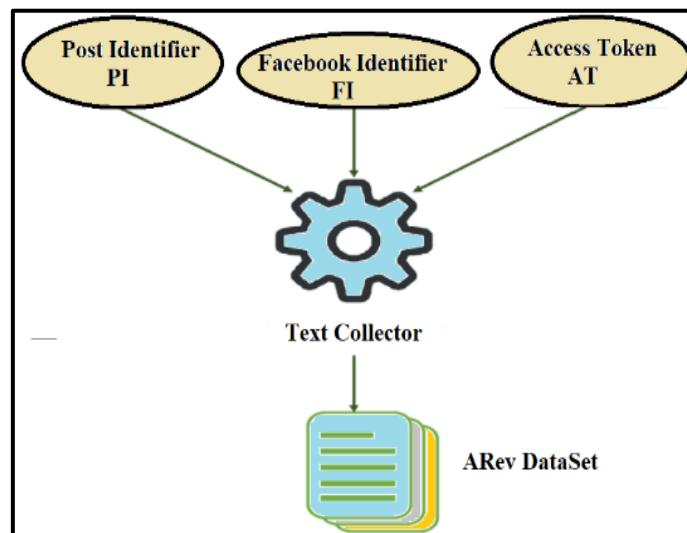


Figure 5. Inputs of dataset collection from Facebook

Table 2. Various datasets used

Datasets	Type of language	Description
LABR [24]		Book reviews
The dataset of Elshahar and El-Beltagy [25]	Standard Arabic	Hotel reviews, restaurant reviews, product reviews, attraction reviews, movie reviews.
The dataset of Mataoui <i>et al.</i> [26]	Algerian Dialect	Comments

Table 3. Statistics on the ARev dataset

	Positive	Negative
Total comments	24932	24932
Total words	1180663	1345029
Avg. words in each comment	47.36	53.95
Avg. characters in each comment	253.15	294.47

4.2. Experimental setup

We used the Google Colab tool to run our experiments where we can take good advantage of TensorFlow's performance. Note that we worked with a masking probability of 15%, a random seed of 34, and a duplication factor was set to 10. In our approach, we worked through the version of AraBERTv1 implemented in the work of [6] where our model was pre-trained on a TPuv2-8 pod. Table 4 resume the parameters used for fine-tuning in our models.

Table 4. Parameter values

Parameter	Value
Learning Rate	1e-4
Epsilon (Adam optimizer)	1e-8
Maximum Sequence Length	256
Epochs	27

4.3. Results and discussion

To show the importance of our module, we compared the result obtained by our approach with those existing in the state of the art for the domain of sentiment analysis. For this reason, we used the accuracy metric, as shown in Table 5. The previous results show that our approach gives an important result that is comparative to those of the state of the art. We obtained an accuracy value of 92.5% for a database containing more than 40,000 comments written by a mixture of standard Arabic and Algerian dialect. However, the approach of Alomari *et al.* [27] gives an accuracy value better than ours by +1.3%, which is a slight difference due to the two reasons following: Firstly, the number of tweets in [27] does not exceed 1800 tweets, secondly, the language mix used in our approach generates more linguistic specifications than the Jordanian dialect. The AraBERT v1 with the best parameters chosen for fine-tuning gives our approach this competitiveness over other models.

Table 5. Performance of our model implemented on AraBERTv1 compared by the previous state of the art systems

Dataset	Descriptions	Language	Accuracy
ASTD [28]	The dataset contains 10,000 tweets.	Egyptian dialect	92.6
Arsen TD lev [29]	The dataset contains 4,000 tweets.	Levantine dialect	59.4
AJGT [27]	The Arabic Jordanian General Tweets dataset contains more than 1,800 tweets.	Jordanian dialect	93.8
ArSarcasm-v2 [30]	Collection of 15,548 sarcasm and sentiment tweets.	Standard Arabic and dialectal Arabic	67.7
ARev Our dataset	The Dataset of a mixture of comments and Hotel reviews, restaurant reviews, product reviews, attraction reviews, movie reviews.	Standard Arabic and Algerian dialect	92.5




5. CONCLUSION AND FUTURE WORK

The automatic understanding of Arabic scripts is still a challenging process and an open issue for researchers in the NLP field. In this work, we have presented our approach based on the AraBERT language model. Also, we have described and detailed the main steps of the proposed architecture using diagrams and examples. The process starts with the input of our model into a pre-processed text from the ARev database, then version 1 of the AraBERT model was implemented by using Farasa segmentation. Moreover, our evaluation is based on the ARev dataset, which contains more than 40,000 comments and reviews. With well-tuned parameters of the AraBERT model, we obtained an accuracy value of 92.5%, which represents a very competitive result. In future work, we aim to address the problem of Arabic text segmentation, try to improve the farasa segmentation version.




REFERENCES

- [1] N. Boudad, R. Faizi, R. Oulad Haj Thami, and R. Chiheb, "Sentiment analysis in Arabic: A review of the literature," *Ain Shams Engineering Journal*, vol. 9, no. 4, pp. 2479–2490, Dec. 2018, doi: 10.1016/j.asej.2017.04.007.
- [2] A. Wadhawan, "Dialect identification in nuanced arabic tweets using farasa segmentation and AraBERT," *arXiv:2102.09749*, Feb. 2021.
- [3] I. Guellil, H. Saâdane, F. Azouaou, B. Gueni, and D. Nouvel, "Arabic natural language processing: An overview," *Journal of King Saud University-Computer and Information Sciences*, vol. 33, no. 5, pp. 497–507, Jun. 2021, doi: 10.1016/j.jksuci.2019.02.006.
- [4] T. Tobaili, "Sentiment analysis for the low-resourced latinised Arabic 'Arabizi'," The Open University, 2020.
- [5] I. Guellil, F. Azouaou, F. Benali, A. E. Hachani, and M. Mendoza, "The role of transliteration in the process of Arabizi translation/sentiment analysis," in *Studies in Computational Intelligence*, Springer International Publishing, 2020, pp. 101–128.
- [6] W. Antoun, F. Baly, and H. Hajj, "AraBERT: transformer-based model for Arabic language understanding," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Jul. 2020, pp. 8440–8451.
- [7] A. Conneau *et al.*, "Unsupervised cross-lingual representation learning at scale," in *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Nov. 2020, pp. 8440–8451.
- [8] D. Adiwardana *et al.*, "Towards a human-like open-domain chatbot," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Jan. 2018, pp. 52–55.
- [9] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in *Advances in neural information processing systems*, 2013, pp. 3111–3119.
- [10] J. Pennington, R. Socher, and C. Manning, "Glove: global vectors for word representation," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2014, pp. 1532–1543, doi: 10.3115/v1/D14-1162.
- [11] T. Mikolov, E. Grave, P. Bojanowski, C. Puhrsch, and A. Joulin, "Advances in pre-training distributed word representations," Dec. 2017, [Online]. Available: <http://arxiv.org/abs/1712.09405>.
- [12] M. Peters *et al.*, "Deep contextualized word representations," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, 2018, pp. 2227–2237, doi: 10.18653/v1/N18-1202.
- [13] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: pre-training of deep bidirectional transformers for language understanding," *arXiv:1810.04805*, Oct. 2018.
- [14] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, Jan. 2018, pp. 328–339, doi: 10.18653/v1/P18-1031.
- [15] C. Raffel *et al.*, "Exploring the limits of transfer learning with a unified text-to-text transformer," *arXiv:1910.10683*, Oct. 2019.
- [16] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "ALBERT: a lite BERT for self-supervised learning of language representations," *arXiv:1909.11942*, Sep. 2019.
- [17] A. M. A. Nada, E. Alajrami, A. A. Al-Saqqa, and S. S. Abu-Naser, "Arabic text summarization using AraBERT model using extractive text summarization approach," *International Journal of Academic Information Systems Research (IAISR)*, vol. 4, no. 8, pp. 6–9, 2020.
- [18] H. Alami, S. Ouatik El Alaoui, A. Benlabib, and N. En-nahnahi, "LISAC FSDM-USMBA Team at SemEval-2020 Task 12: Overcoming AraBERT's pretrain-finetune discrepancy for Arabic offensive language identification," in *Proceedings of the Fourteenth Workshop on Semantic Evaluation*, 2020, pp. 2080–2085, doi: 10.18653/v1/2020.semeval-1.275.
- [19] D. Faraj and M. Abdullah, "SarcasmDet at SemEval-2021 Task 7: detect humor and offensive based on demographic factors using RoBERTa pre-trained model," in *Proceedings of the 15th International Workshop on Semantic Evaluation (SemEval-2021)*, 2021, pp. 527–533, doi: 10.18653/v1/2021.semeval-1.64.
- [20] A. Hussein, N. Ghneim, and A. Joukhadar, "DamascusTeam at NLP4IF2021: fighting the Arabic COVID-19 infodemic on Twitter using AraBERT," in *Proceedings of the Fourth Workshop on NLP for Internet Freedom: Censorship, Disinformation, and Propaganda*, 2021, pp. 93–98, doi: 10.18653/v1/2021.nlp4if-1.13.
- [21] A. Abdelali, K. Darwish, N. Durrani, and H. Mubarak, "Farasa: a fast and furious segmenter for Arabic," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, 2016, pp. 11–16, doi: 10.18653/v1/N16-3003.
- [22] T. Kudo, "Subword regularization: improving neural network translation models with multiple subword candidates," in *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2018, pp. 66–75, doi: 10.18653/v1/P18-1007.
- [23] A. Abdelli, F. Guerrouf, O. Tibermacine, and B. Abdelli, "Sentiment analysis of Arabic Algerian dialect using a supervised method," in *2019 International Conference on Intelligent Systems and Advanced Computing Sciences (ISACS)*, Dec. 2019, pp. 1–6, doi: 10.1109/ISACS48493.2019.9068897.
- [24] M. Aly and A. Atiya, "Labr: A large scale Arabic book reviews dataset," 2013.
- [25] H. ElSahar and S. R. El-Beltagy, "Building large Arabic multi-domain resources for sentiment analysis," in *Computational Linguistics and Intelligent Text Processing*, Springer International Publishing, 2015, pp. 23–34.
- [26] M. Mataoui, O. Zelmati, and M. Boumechache, "A proposed lexicon-based sentiment analysis approach for the vernacular Algerian Arabic," *Research in Computing Science*, vol. 110, no. 1, pp. 55–70, Dec. 2016, doi: 10.13053/rcs-110-1-5.
- [27] K. M. Alomari, H. M. ElSherif, and K. Shaalan, "Arabic tweets sentimental analysis using machine learning," in *Advances in Artificial Intelligence: From Theory to Practice*, Springer International Publishing, 2017, pp. 602–610.
- [28] M. Nabil, M. Aly, and A. Atiya, "ASTD: Arabic sentiment tweets dataset," in *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, 2015, pp. 2515–2519, doi: 10.18653/v1/D15-1299.
- [29] R. Baly, A. Khaddaj, H. Hajj, W. El-Hajj, and K. B. Shaban, "ArSentD-LEV: a multi-topic corpus for target-based sentiment analysis in Arabic levantine tweets," *The 3rd Workshop on Open-Source Arabic Corpora and Processing Tools*, 2018.
- [30] I. A. Farha and W. Magdy, "Benchmarking transformer-based language models for Arabic sentiment and sarcasm detection," in *Proceedings of the Sixth Arabic Natural Language Processing Workshop*, 2021, pp. 21–31.




BIOGRAPHIES OF AUTHORS

Prof. Hicham El Moubtahij    is currently a Professor of Computer Science at the University of Ibn Zohr, Agadir, Morocco. He received his Ph.D. in Computer Science from the University of Sidi Mohamed Ben Abdellah, Fez, Morocco, in 2017. He is now a member of the Systems and Technologies of Information Team at the High School of Technology at the University of Ibn Zohr, Agadir. His current research interests include machine learning, deep learning, Arabic handwriting recognition, Text Mining, and medical imagery. Dr. El Moubtahij Hicham has published articles in indexed international journals and conferences, has been a reviewer for scientific journals, and has served on the program committee of several conferences. He can be contacted at email: h.elmoubtahij@uiz.ac.ma.



Dr. Hajar Abdelali    holder of a bachelor's degree in experimental sciences, a bachelor's degree in mathematics and computer science, a master's degree in information sciences, networks and multimedia from Sidi Mohammed Ben Abdellah, University of Fez, Morocco in 2013. She joined the laboratory XLIM of the University of Poitiers in France in collaboration with the scientific laboratory LIMS of the Faculty of Sciences Dhar Mahraz of Sidi Mohammed Ben Abdellah, University of Fez, Morocco where he obtained his Ph.D. degree in computer science in 2019. She can be contacted at email: abdelali.hajar@usmba.ac.ma.



Prof. El Bachir Tazi    graduated in Electronic Engineering from ENSET Mohammedia Morocco in 1992. He obtained his DEA and DES in Automation and Signal Processing and his PhD in Computer Science from Sidi Mohammed Ben Abdellah University, Faculty of Sciences in Fez, Morocco respectively in 1995, 1999 and 2012. He is now a member of the engineering sciences laboratory and associate professor at Sidi Mohammed Ben Abdellah University, Polydisciplinary Faculty of Taza, Morocco. His areas of interest generally include all areas of automatic recognition based on artificial intelligence methods and applications related to automatic speaker. He can be contacted at email: elbachirtazi@yahoo.fr.

Implementation of FaceNet and support vector machine in a real-time web-based timekeeping application

Ly Quang Vu¹, Phan Thanh Trieu², Hoang-Sy Nguyen³

¹Faculty of Computer Science, University of Information Technology, Ho Chi Minh City, Vietnam

²Faculty of Information Technology, Ton Duc Thang University, Ho Chi Minh City, Vietnam

³Institute of Engineering and Technology, Thu Dau Mot University, Binh Duong Province, Vietnam

Article Info

Article history:

Received Sep 20, 2021

Revised Dec 20, 2021

Accepted Dec 31, 2021

Keywords:

Face detection

Face recognition

Real-time web-based

Multi-task cascaded neural network

Support vector machine

ABSTRACT

This paper presents in detail how to build up and implement a real-time web-based face recognition application. The system works so that images of people are recorded and compared with the references on the database. If they match, the information about their presence will be recorded. As for the system architecture, the multi-task cascaded neural network was deployed for face detection. Followingly, for the recognizing tasks, we conducted a study to compare the accuracy level of three different face recognizing methods on three different public datasets by means of both the literature review and our simulation. From the comparison, it can be drawn that the FaceNet algorithm in-used with the support vector machine (SVM) classifier performs the best among others and is the most suitable candidate for the practical deployment. Eventually, the proposed system can deliver a highly satisfactory result, proving its potentials not only for the research but also the commercial purposes.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Hoang-Sy Nguyen

Institute of Engineering and Technology, Thu Dau Mot University

Binh Duong Province, Vietnam

Email: nguyenhoangsy@tdmu.edu.vn

1. INTRODUCTION

Face recognition technology has been replacing the human's role in recognizing faces. The face-recognizing equipment receives face images or videos containing human faces as input, of which biometric facial data is subsequently extracted and processed to conduct the recognizing task [1]. Because the facial features of an individual are unique [2], they have been acknowledged as an effective means for security purposes, e.g., alternating the use of passwords and identity cards, and allowing authorized access. Among popular face recognition models which have been developed by universities and companies, there are VGGFace [3], DeepFace [4], [5], OpenFace [6], and FaceNet [7]. In [8], a face recognition model based on the histogram of oriented gradients (HOG) and support vector machine (SVM) classifier was investigated. Besides, in [9], a method based on the AdaBoost algorithm was used to train cascade classifiers with feature types such as the HOG and the Haar-like. Although a better performance was achieved, it is computationally demanding as it includes a number of weak classifiers.

On the other hand, direct training operations on faces can be challenging owing to the face occlusion, which is common in practice. To overcome this issue, Zhang *et al.* [10] have based on the Bayesian framework to propose an algorithm that locates the head using the Omega-liked shape formed by the head-shoulder part of a person. This technique has been applied widely in automatic teller machines (ATM). Additionally, in [11], the face-recognizing task was carried with deformable part models (DPM) yielding remarkable results, though it requires heavy computational resources. The DPM-based system was

deployed as well in [12], which offers a reduction in error rate and false-negative face detection. Nonetheless, this technique is limited by the usage of front-view facial images, thus, is not universal.

Recent years have witnessed the rise of convolutional neural network (CNN) application in face detection. Deep CNN (DCNN) [13], region based convolutional neural networks (R-CNN) [14], and another one-or two-stage deep CNN-based systems such as VGGNet [15] and ResNet [16] have showcased their outstanding performance in comparison with their conventional counterparts. However, as there are more convolutional layers added to the CNN, the detecting speed is reduced considerably. To overcome this issue, a number of multi-stage face-detecting algorithms have been investigated, for example, the funnel-structured cascade (FuSt) [17], the pyramid-based cascade model that distills knowledge online and mines hard sample offline [18], which deliver outstanding true positive rate and performance in real-time.

CNNs are driven with data as they are trained with the extracted features and face classification. Additionally, CNNs which are trained with 2D facial data could further be tuned with 3D one for potentially better recognition accuracy. Tornincasa [19] showcased how the pertinent discriminating features from the query faces can be extracted by the use of differential geometry. Dagnes *et al.* [20] have investigated an algorithm that can compute the optimized marker layout to capture the face movement. To deal with the different facial expressions and illumination, radon and wavelet transforms were combined in [21] for the nonlinear feature extraction. Notably, a so-called DeepID model, which is constructed of a large number of CNNs, and its extension were proposed in [22]–[24] with a better feature extracting capability. This is realized thanks to the fact that they can process a variety of face positions and facial patches.

In this paper, we designed a face recognition system based on the FaceNet model with SVM classifier. Then, we compare the accuracy of our proposed method with two other face recognition methods operating on three public datasets to increase the generalization of the study. Finally, the paper showcases how to integrate the system into a web-based timekeeping application. The obtained results regarding the system performance and its implementation are highly applicable for both the research purpose and the practical usage.

2. MATERIALS AND METHODS

2.1. Multi-task cascaded convolutional neural networks (MTCNN)

MTCNN framework detects and aligns faces with unified cascaded CNNs. MTCNN is tasked with three outputs. Firstly, it has to classify whether an input is a face or non-face. Then it has to perform the bounding box regression, and finally localizes the facial landmark. Each layer uses the intakes the output from its preceding layer and in the end, the overall learning target is summed up. Corresponding to these tasks, the MTCNN is constructed of three layers which are in order so-called the proposal network (P-Net), the refine network (R-Net), and the output network (O-Net). The architectures of MTCNN are shown in Figure 1.

Layer 1(P-Net) is a fully convolutional network (FCN), which is used to generate the candidate windows and their corresponding bounding box regression vectors. P-Net combines the overlapping areas of the bounding box vectors to reduce the candidate volume. Layer 2 (R-Net) is a CNN which is differentiated from FCN as its last stage is denser. R-Net intakes the output of P-Net, screens out the false candidates, calibrates with bounding box regression, and merges overlapping candidates using non-maximum suppression (NMS). Layer 3 (O-Net) functions in a similar manner to the R-Net. However, it describes in more detail the faces and delivers five positions of the facial landmarks being the left/right eyes, nose, and left/right mouth corners.

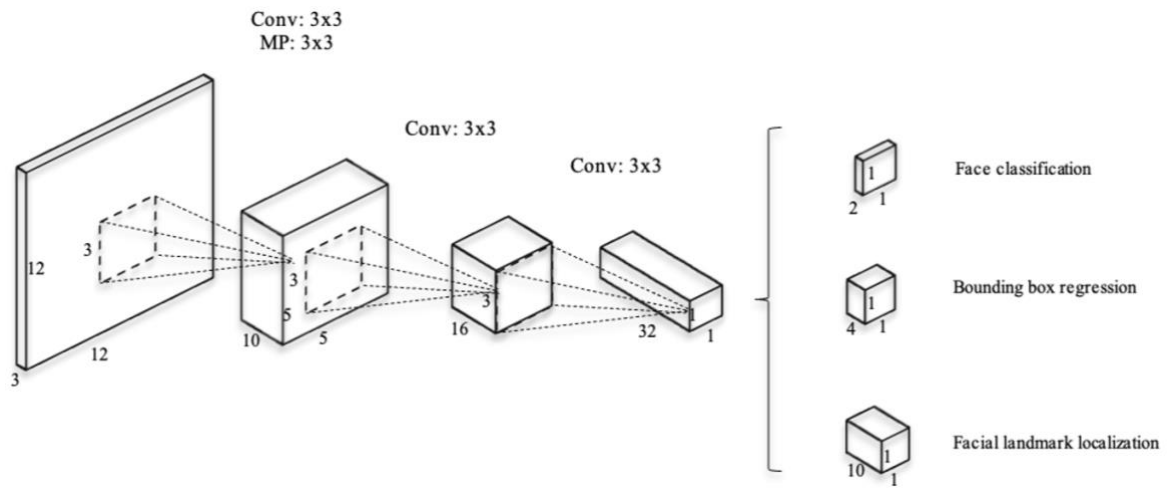
2.2. FaceNet model

Facenet takes an image of the person's face as input and output it into the 128-dimensional euclidean space. The distances of a person's face images would be comparatively closer than that of other random ones. In general, there are two different CNN-based basic architectures in Facenet. The first category adds $1 \times 1 \times d$ convolutional layers between the standard convolutional layers of the Chen *et al.* [25] architecture, then gets a model 22 layers NN1 model. The second category consists of Inception models based on GoogLeNet [26]. The inception module contains 4 branches from the left to right. It employs convolution with 1×1 filters as well as 3×3 and 5×5 filters and a 3×3 max-pooling layer. Each branch uses a 1×1 convolution to achieve time complexity reduction. FaceNet model is a DCNN trained via a triplet loss technique that allows vectors for the same identity to become more similar, while vectors for different identities should become less similar.

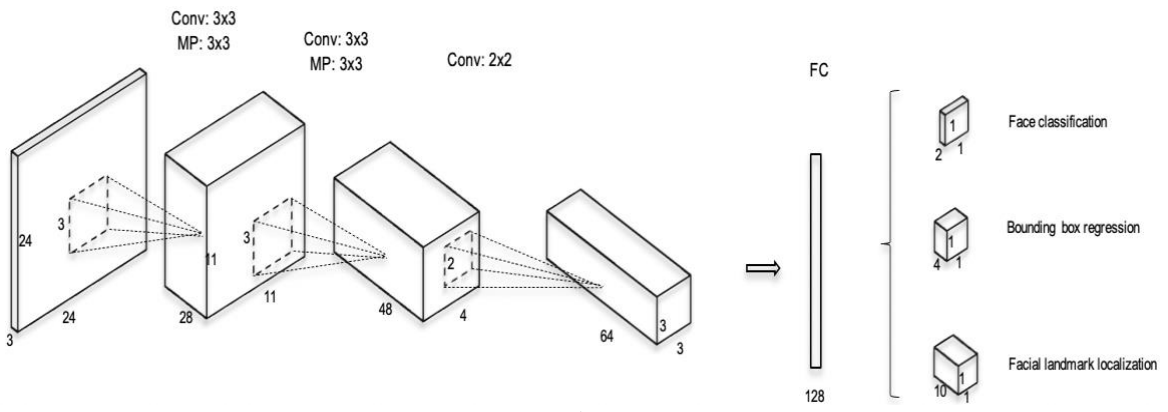
2.3. Triplet loss

The face recognition model is trained with batches of data, each has three images being the anchor, the positive, and the negative images. Specifically, Figure 2 illustrates how the triplet loss operates by maximizing the anchor-negative image distance and minimize the anchor-positive one. Notably, an image is

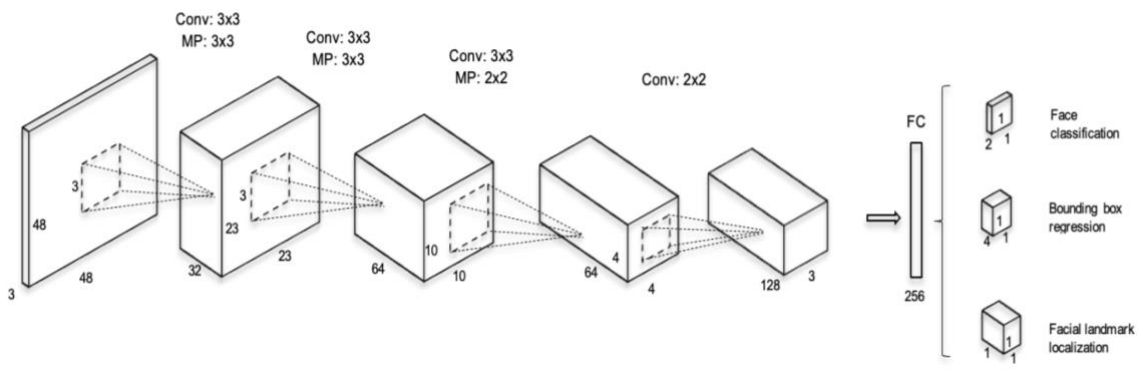
considered positive if it has the same identity as the anchor and vice versa for the negative image. Thanks to this mechanism Triplet loss has been considered one of the best effective ways for learning face image 128-D encodings. Notably, an image is considered positive if it has the same identity as the anchor and vice versa for the negative image. Thanks to this mechanism Triplet loss has been considered one of the best effective ways for learning face image 128-D encodings.



(a)



(b)



(c)

Figure 1. The architectures of (a) P-Net, (b) R-Net, and (c) O-Net, where “MP” means max pooling and “Conv” means convolution. The step size in convolution and pooling is 1 and 2, respectively



Figure 2. The triplet loss training

2.4. Proposed approach

The pipeline of our face recognition system is illustrated in Figure 3. It can be further elaborated:

- Firstly, the MTCNN is trained with face images of all the staff in an organization.
- Secondly, images and video frames are input into our system and the MTCNN face detector is applied to recognize the face location. These faces are pre-processed and aligned based on the face landmarks computed by MTCNN. There are five features that are included in the face landmarks which are the nose, left/right eye, and left/right mouth. Moreover, the MTCNN is used as well to construct image pyramids corresponding to the face images.
- Thirdly, the FaceNet algorithm is applied to extract the 128D embeddings from the face images.
- Fourthly, we deploy a search algorithm to find in the database an encoding whose distance with the real-time face image encoding satisfies a threshold value. Once it does, the person is recognized as a staff.
- Consequently, the information about the presence of the recognized staff will be updated to the database. Otherwise, the application screens will display a notification announcing that the person’s face can not be recognized or it is not in the staff list.

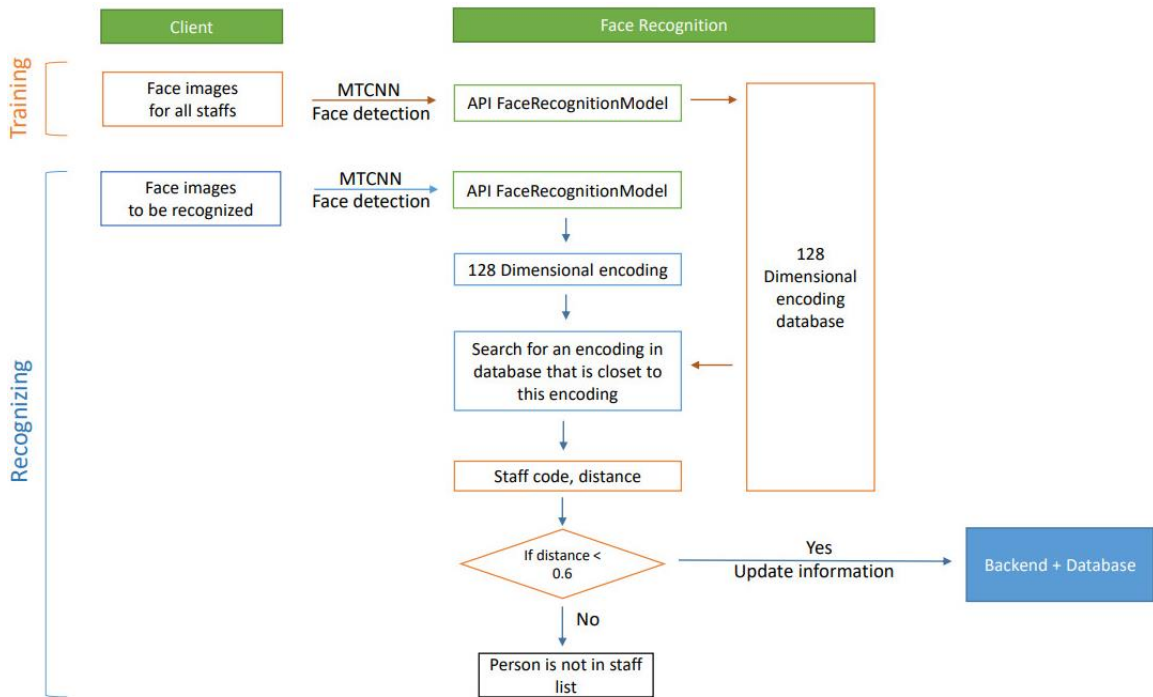


Figure 3. The pipeline of our face recognition system

Remarkably, our system allows working with Euclidean image embeddings, and the network is trained to propose the embedding spaces (squared L2 distances) directly according to the similar faces. As a result, the distance of the same subject images is small and that of different subjects is big. After these embedding spaces have been created, face verification can be performed easily by setting a threshold distance value between two points in the space. Subsequently, the SVM algorithm is applied for the classifying operation.

3. IMPLEMENTATION AND RESULTS

3.1. Datasets used for training and testing

In this paper, three public datasets namely labeled faces in the wild (LFW) [27], our database of faces (ORL) [28], and yale face database [29] were used to assess the accuracy of the in-studied approach. The number of face images and subjects along with their notes are given in Table 1. The three datasets vary largely in the number of images, subjects and configurations. Thus, it is expected that the generality of this study can be ensured.

Table 1. Three in-used public face image datasets

Name	Number of face images and subjects	Challenges	Note
LFW[27]	13,233 images (5,749 subjects)	Face pose, expression, illumination.	Subjects with more than 20 images were selected, resulting in 3,137 images (62 subjects).
ORL[28]	400 images (40 subjects)	Timing, expression (open/close eyes, yes/no smile), illumination, accessories (yes/no glasses).	All subjects were used. Dark background. Upright, frontal face images.
Yale Face Database[29]	165 images (15 subjects)	Expression (happy, neutral, sad, sleepy, surprised, wink), illumination (center/left/right light), accessories (yes/no glasses).	All subjects were used. Grayscale GIF images.

3.2. Experiment results

Table 2 compares the accuracy of the results which were produced from the three datasets using the FaceNet with support vector machine (SVM) classifier. In particular, every image was processed with a Euclidean space technique and compared with its index label. It can be concluded that the FaceNet with SVM can deliver results with relatively a high level of accuracy. Figure 4 demonstrates how the triplet loss function minimize the distances between positive anchors and maximizes the distances between negative ones after being trained with the subset of the LFW dataset.

Table 2. Accuracy comparison using Facenet with SVM

Dataset	FaceNet with SVM [%]
LFW	99.83
ORL	97.5 [2]
Yale Face Database	98.9 [2]

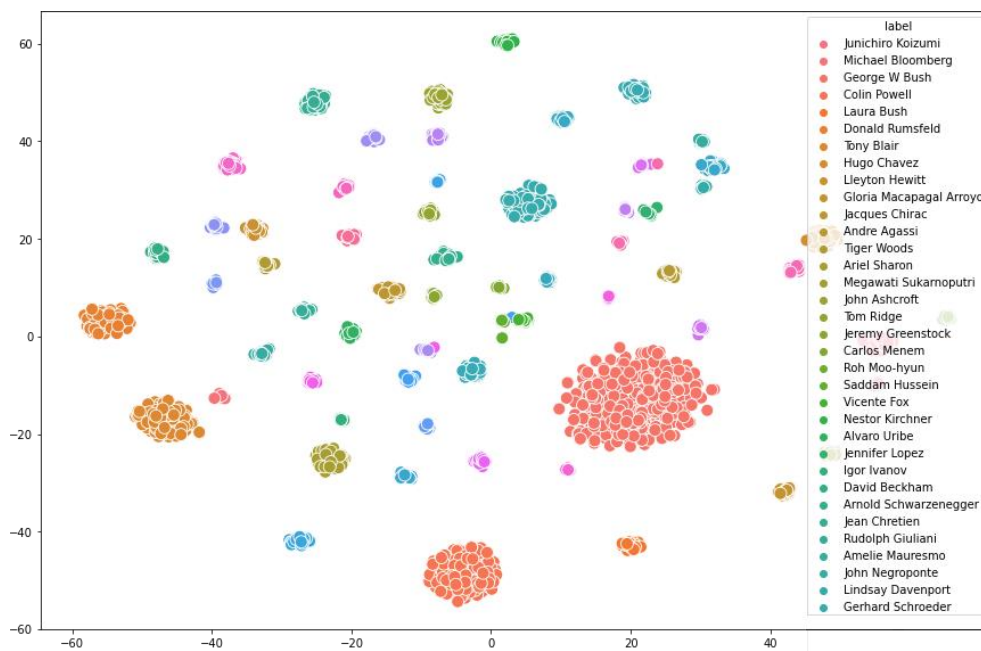


Figure 4. Triplet loss training on the subset of LFW dataset

The results from the FaceNet face recognition is subsequently compared with two other methods namely the principal component analysis (PCA) and SVM classifier [25], and the k-nearest neighbor (K-NN) [26], as can be seen in Table 3, the FaceNet with SourVM can deliver the minimum accuracy level of up to 97.5%, being the highest among others. It should be noted that this model performs well even though there exist challenges being a variety of face poses, expressions, illumination, and the use of accessories.

Table 3. Accuracy comparison using FaceNet with SVM, PCA with SVM, and K-NN

Method Dataset	LFW [%]	ORL [%]	Yale Face Database[%]
FaceNet with SVM	99.83	97.5 [2]	98.9 [2]
PCA with SVM	62.14	95.12	82.35
K-NN	30.24	85.36	52.94

Face recognition can effectively detect human presence in a particular area of interest (AOI) such as office, and educational institution. Herein this paper, the authors succeeded in establishing a web-based timekeeping application. Figure 5 illustrates how the system works. The system consists of a remote server and a database that can be accessed with a web application for monitoring and administrating purposes. An IP camera is set at the entrance to a company to streamline video frames in real-time to the Face recognition API. If a face is detected, an image in that time frame is preprocessed and passed on to the deep CNN to generate 128-byte embedding.

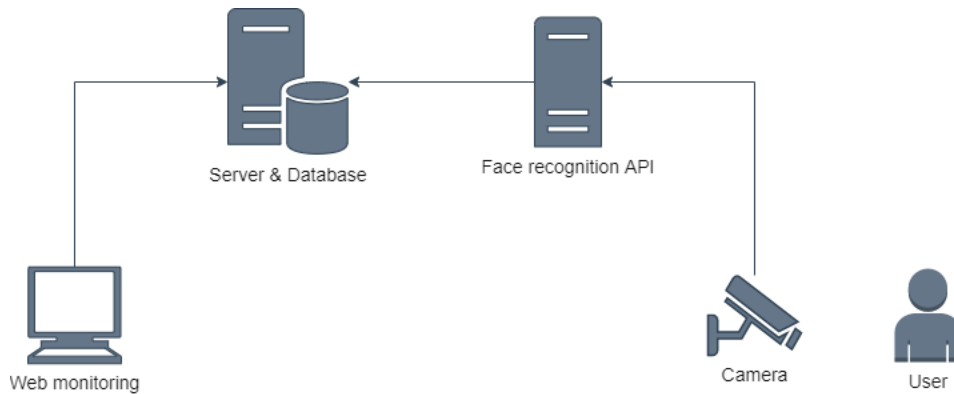


Figure 5. Web-based timekeeping application

Subsequently, the staff’s identities can be determined with the SVM classifier and the data related to the staff’s presence such as the identities, the accuracy percentage, the time, and the date of entry are recorded in the database. Figure 6 illustrates what information a user can see on the web application as a staff is recognized by the system. Specifically, there is a frame identifying the detected face with the recognized name and the accuracy at its bottom. The right side shows a list of recognized staffs along with their ID numbers, full names, ID cards, and the entry time. In case the system cannot recognize a person’s presence due to the missing of data, for example, an entry of new staff or a visitor, the face image of the person will display as shown in Figure 7.

The detected face is framed with red color and labeled with “Unknown”. It should be noted that the time and date of the unrecognized entry are recorded to assist the administrator in preparing corresponding solutions such as adding the information of the new employee, and re-training the model. All the information about the entries of the people as in Figure 6 can be exported to *.xls file as shown in Figure 8.

Besides, the web application has an interface for adding new facial data. Users can open the IP camera from the application to capture new face images in real-time. These images can then be saved in the database and assigned with a unique user ID. Consequently, the face from the image is extracted and labeled with what the administrator may find suitable, for example, the person’s name.

The system was tested with a group of 32 staffs showing the face recognition accuracy of 96%. Nevertheless, the system is sensitive to the lighting changes and angle between the faces and the IP camera, which considerably downgrade the system’s accuracy. Thus, in case the system fails to recognize staff, the staff needs to inform the person in charge of timekeeping for a manual marking.

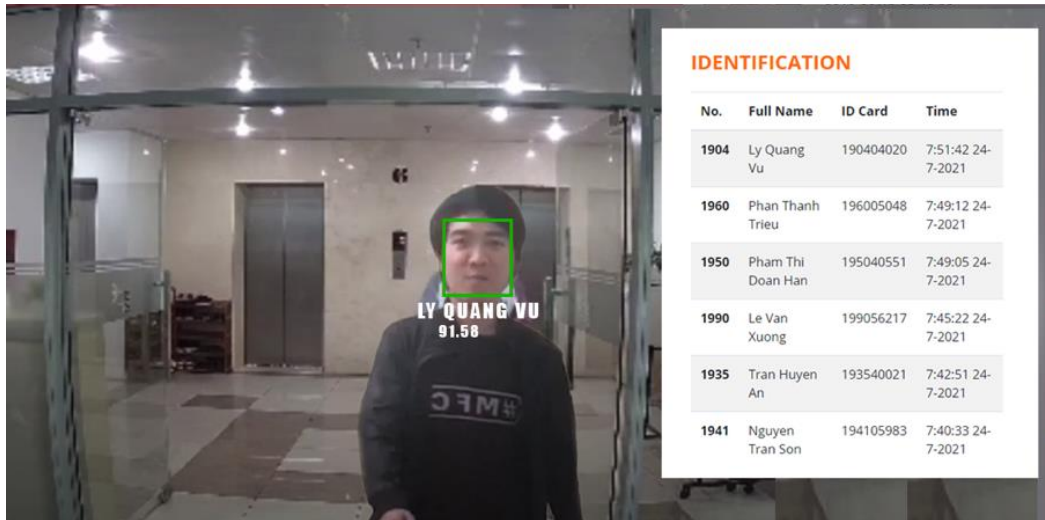


Figure 6. A recognized staff by the system



Figure 7. An unrecognized person

	A	B	C	D
1	id	Full name	ID Card	Time
2	1904	Ly Quang Vu	190404020	7:51:42 24-7-2021
3	1960	Phan Thanh Trieu	196005048	7:49:12 24-7-2021
4	1950	Pham Thi Doan Han	195040551	7:49:05 24-7-2021
5	1990	Le Van Xuong	199056217	7:45:22 24-7-2021
6	1935	Tran Huyen An	193540021	7:42:51 24-7-2021
7	1941	Nguyen Tran Son	194105983	7:40:33 24-7-2021

Figure 8. Data exported to *.xls file.

4. CONCLUSION




To conclude, in our system, the MTCNN algorithm is deployed to detect the faces, generates the embeddings using the pre-trained FaceNet with SVM classifier, then recognizes images that are taken through the system. The system is able to deliver in practice the recognition accuracy of 96%, given that the

images are collected under consistent conditions in terms of lighting and face-camera angle. The comparison study can serve as a foundation for the researchers seeking for optimized face-recognizing algorithms. Additionally, the paper also presents an established web-based application with some key concepts that can potentially be upgraded to a commercial timekeeping product. Application of such products into practice has proven its abilities to save companies and organizations a considerable amount and time and efforts in timekeeping tasks. As more and more powerful algorithms are introduced and implemented into face recognition systems, it is promising that end users will get more benefits from them. For future studies, the system can be more fine-tuned and more training data with noises can be collected to further improve the capability of our proposal.




REFERENCES

- [1] Y. Zhang, S. Wang, H. Xia, and J. Ge, "A novel SVPWM modulation scheme," in *2009 Twenty-Fourth Annual IEEE Applied Power Electronics Conference and Exposition*, Feb. 2009, pp. 128–131, doi: 10.1109/APEC.2009.4802644.
- [2] L. Li, X. Mu, S. Li, and H. Peng, "A review of face recognition technology," *IEEE Access*, vol. 8, pp. 139110–139120, 2020, doi: 10.1109/ACCESS.2020.3011028.
- [3] I. William, D. R. Ignatius Moses Setiadi, E. H. Rachmawanto, H. A. Santoso, and C. A. Sari, "Face recognition using facenet (survey, performance test, and comparison)," in *2019 Fourth International Conference on Informatics and Computing (ICIC)*, Oct. 2019, pp. 1–6, doi: 10.1109/ICIC47613.2019.8985786.
- [4] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *Proceedings of the British Machine Vision Conference 2015*, 2015, pp. 41.1-41.12, doi: 10.5244/C.29.41.
- [5] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf, "DeepFace: closing the gap to human-level performance in face verification," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1701–1708, doi: 10.1109/CVPR.2014.220.
- [6] T. Baltrusaitis, P. Robinson, and L.-P. Morency, "OpenFace: An open source facial behavior analysis toolkit," in *2016 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Mar. 2016, pp. 1–10, doi: 10.1109/WACV.2016.7477553.
- [7] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.
- [8] M. Drożdż and T. Kryjak, "FPGA implementation of multi-scale face detection using HOG features and SVM classifier," *Image Processing & Communications*, vol. 21, no. 3, pp. 27–44, Sep. 2016, doi: 10.1515/ipc-2016-0014.
- [9] C. Ma, N. Trung, H. Uchiyama, H. Nagahara, A. Shimada, and R. Taniguchi, "Adapting local features for face detection in thermal image," *Sensors*, vol. 17, no. 12, Art. no. 2741, Nov. 2017, doi: 10.3390/s17122741.
- [10] T. Zhang, J. Li, W. Jia, J. Sun, and H. Yang, "Fast and robust occluded face detection in ATM surveillance," *Pattern Recognition Letters*, vol. 107, pp. 33–40, May 2018, doi: 10.1016/j.patrec.2017.09.011.
- [11] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," in *Computer Vision [textdash] [ECCV] 2014*, Springer International Publishing, 2014, pp. 720–735.
- [12] D. Marcetic and S. Ribaric, "Deformable part-based robust face detection under occlusion by using face decomposition into face components," in *2016 39th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*, May 2016, pp. 1365–1370, doi: 10.1109/MIPRO.2016.7522352.
- [13] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Processing Letters*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016, doi: 10.1109/LSP.2016.2603342.
- [14] S. Wan, Z. Chen, T. Zhang, B. Zhang, and K. Wong, "Bootstrapping face detection with hard negative examples," Aug. 2016.
- [15] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," Sep. 2014, Available: <http://arxiv.org/abs/1409.1556>.
- [16] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778, doi: 10.1109/CVPR.2016.90.
- [17] S. Wu, M. Kan, Z. He, S. Shan, and X. Chen, "Funnel-structured cascade for multi-view face detection with alignment-awareness," *Neurocomputing*, vol. 221, pp. 138–145, Jan. 2017, doi: 10.1016/j.neucom.2016.09.072.
- [18] S. S. Farfade, M. J. Saberian, and L.-J. Li, "Multi-view face detection using deep convolutional neural networks," in *Proceedings of the 5th ACM on International Conference on Multimedia Retrieval*, Jun. 2015, pp. 643–650, doi: 10.1145/2671188.2749408.
- [19] S. Tornincasa *et al.*, "3D facial action units and expression recognition using a crisp logic," *Computer-Aided Design and Applications*, vol. 16, no. 2, pp. 256–268, Aug. 2018, doi: 10.14733/cadaps.2019.256-268.
- [20] N. Dagnes *et al.*, "Optimal marker set assessment for motion capture of 3D mimic facial movements," *Journal of Biomechanics*, vol. 93, pp. 86–93, Aug. 2019, doi: 10.1016/j.jbiomech.2019.06.012.
- [21] H. D. Vankayalapati and K. Kyamakya, "Nonlinear feature extraction approaches with application to face recognition over large databases," in *2009 2nd International Workshop on Nonlinear Dynamics and Synchronization*, Jul. 2009, pp. 44–48, doi: 10.1109/INDS.2009.5227967.
- [22] Y. Sun, D. Liang, X. Wang, and X. Tang, "DeepID3: face recognition with very deep neural networks," Feb. 2015, [Online]. Available: <http://arxiv.org/abs/1502.00873>.
- [23] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation by joint identification-verification," Jun. 2014, [Online]. Available: <http://arxiv.org/abs/1406.4773>.
- [24] Y. Sun, X. Wang, and X. Tang, "Deep learning face representation from predicting 10,000 classes," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, Jun. 2014, pp. 1891–1898, doi: 10.1109/CVPR.2014.244.
- [25] X. Chen, L. Song, and C. Qiu, "Face recognition by feature extraction and classification," in *2018 12th IEEE International Conference on Anti-counterfeiting, Security, and Identification (ASID)*, Nov. 2018, pp. 43–46, doi: 10.1109/ICASID.2018.8693198.
- [26] H. Zhang and G. Chen, "The research of face recognition based on PCA and k-nearest neighbor," in *2012 Symposium on Photonics and Optoelectronics*, May 2012, pp. 1–4, doi: 10.1109/SOPO.2012.6270975.
- [27] B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database for studying face recognition in unconstrained environments," 2008.
- [28] "ORL (our database of faces)." <https://paperswithcode.com/dataset/orl>.
- [29] "Yale face database." <http://vision.ucsd.edu/content/yale-face-database>.




BIOGRAPHIES OF AUTHORS

Ly Quang Vu    is a student Master of Science (M.Sc.) in the Faculty of Computer Science of Ho Chi Minh City University of Information Technology (VNUHCM-UIT). He is working at Bdoop Services and Trading Joint Stock Company-Ho Chi Minh City, Vietnam. His research interests are in fields of Machine Learning Applications and Image Processing, Data Mining, and Network Security. Email: quangvu.ly@gmail.com.



Phan Thanh Trieu    is a student Master of Science (M.Sc.) in the Faculty of Information Technology (IT) of Ton Duc Thang University, Vietnam (TDTU). He is working at Vietnam Posts and Telecommunications Group (VNPT)-An Giang Province, Vietnam. His research interests are in fields of Machine Learning Applications and Image Processing, Network Communications, and Network Security. Email: phanthanhtrieuag@gmail.com.



Hoang-Sy Nguyen    was born in Binh Duong province, Vietnam. He received the B.S. and MS.c degree from the Department of Computer Science from Ho Chi Minh City University of Information Technology (UIT-HCMC), Vietnam in 2007, 2013, respectively. He received his Ph.D. degree in communication technology, dissertation thesis “Energy harvesting enable relaying networks: Design and performance analysis” from the VSB-Technical University of Ostrava-Czech Republic, in 2019. His research interests include Energy efficient wireless communications, 5G wireless communication networks, Network security, Artificial Intelligence, Cloud Networks, and Big Data. Email: ng.hoangsy@gmail.com.

Identify tooth cone beam computed tomography based on contourlet particle swarm optimization

Hiba Adreese Younis, Dhafar Sami Hammadi, Ansam Nazar Younis

Computer Sciences Department, College of Computer Sciences and Mathematics, Mosul University, Mosul, Iraq

Article Info

Article history:

Received Mar 16, 2021

Revised Dec 16, 2021

Accepted Dec 28, 2021

Keywords:

CBCT

Contourlet

CPSO

Directional filter

Laplace's pyramid

Preprocessing

PSO

ABSTRACT

In this paper certain type of biometric measurements has been used to identify the cone beam computed tomography (CBCT) radiograph of the subject in a fast and reliable way. Where the CBCT radiograph of a person is used as a data and stored in database for later use in a person's recognition process. The aim of this research is to use various stages of the preprocessing operations of the CBCT radiograph to obtain the clearest possible image that will help us in the identification process more easily and precisely. The contourlet transformation was used for feature extraction of each particular CBCT image and the results were processed by a new hybrid particle swarm optimization (PSO) named "contourlet PSO" algorithm (CPSO), which is faster and produce more precise (due to apply contourlet algorithm) than traditional PSO. The proposed algorithm (CPSO) gave a detection ratio of 98% after its application on 100 CBCT radiographs.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Hiba Adreese Younis

Computer Sciences Department, College of Computer Sciences and Mathematics, Mosul University

Mosul, Iraq

Email: hibaadreese@uomosul.edu.iq

1. INTRODUCTION

The personal secret identification number can be lost, forgotten or simply difficult to memorize. Also it may be stolen or hacked on in many occasions. Therefore, biometrics can be used for identity access as an alternative [1], [2]. The main advantage of biometric features is that these are not prone to theft and loss, and do not rely on the memory of their users. Moreover, biometrics, such as finger prints, iris print, hand geometry, ear shape, face and teeth scans don't change significantly over time and it is a difficult for a person to alter his own physiological biometrics or imitate other individuals. So, the biometric record took a lot of interest in the last decades as a safer method for personal identification. Each one of these biometric records has its advantages and disadvantages [1]. Dental biometrics uses dental structure information for the automatic identification of human remains [3]. So the aim of this article is to present a medical system for personal identification depending on forensic dentistry where an artificial intelligent system have been made using the cone beam computed tomography (CBCT) for personal identification, and despite the fact that teeth shape can be subjected to change over time, but the shape, size of jaw and facial bones usually are still clear and constant without any change, so these features can be efficiently used for personal identification and in artificial intelligence algorithms. for this reason we chose CBCT images radiographs as a database for the proposed system. The rest of this paper is outlined as shown: section 2 deals with related works, section 3 deals with CBCT, section 4 deals with contourlet algorithm, section 5 deals with the proposed algorithm, results and analysis are shown in section 6, section 7 deals with conclusion and future works, finally, the acknowledgement.

Dighe and Shriram introduced a method for identifying the individual identity from dental information. The proposed method included three main steps: feature extraction using morphological operations

(mathematical morphology), dental code generation, and dental code matching. The method was implemented on 30 X-ray database [4]. Karunya *et al.* [5] proposed a new system which consists of two stages: first stage includes extraction of features and the second stage is matching. In the first stage the contour model was used for contour extraction. The second stage includes two sub stages evaluation the distances of the images and identification of the subject. The system was applied on ten normal images and fifty-five orthopantomogram (OPG) images. Rehman *et al.* [6] presented an efficient method for human authentication correctly which consists of five main processing stages: preprocessing, segmentation, processing steps for segmentation, feature extraction and biometric analysis. The method was tested on colored teeth images for 14 persons and dental radiographs images for 45 persons.

Oktay *et al.* [7] presented a method for distinguishing humans by comparing two-dimension panoramic dental X-ray images. First each tooth is detected and labeled using support vector machine and graphical probabilistic models. The matching ratings between images were calculated based on an appearance of the tooth and the geometric similarities. Shaker *et al* [8] introduced a method for identity identification based on X-ray image. The method included three stages: preprocessing, feature extraction, and matching. The preprocessing was mean, distance and standard derivation (STD), feature extraction was variance and principal component analysis (PCA). The best results showed an identification rate of 89%. Khudhur *et al.* [9] introduced a system to construct a database containing dental ante mortem radiographic features, later used for post mortem dental matching. The algorithm which applied on X-Ray image included three stages: segmentation of images, classification and extracting features. These features were STD, euler number and area taken from bite-wing X-ray image.

In this research a hybrid method was proposed for discrimination CBCT radiation images. This method is a combination of discriminative features of contourlet coefficients and intelligent features of particle swarm optimization (PSO) algorithm. Firstly, The CBCT radiographic images was preprocessed through different steps to obtain clearest possible image That make the identification process simpler and more reliable. Then contourlet transformation was used for feature extraction of each particular CBCT image. Finally, the PSO algorithm was implemented on the extracted features for identification process. The new hybrid PSO method was faster and yielded more accurate result, and the use of CBCT radiographs images which gives information not found with the traditional two-dimension imaging added a strength point to the research. Also, the process of hybridizing the pso algorithm with a contourlet transformation added strength to the algorithm which improved the efficiency of the algorithm. A comparison with related previous studies were described in Table 1.

Table 1. Comparisons with related studies

No	Research name	Algorithms/methods	No. of images	Recognition rate
1	Dental biometrics for human identification based on dental work and image properties in Periapical radiographs/ 2012	mathematical morphological operations	30 X-ray images	90%
2	Human identification using dental biometrics/2014	Shape registration method, euclidian distance	ten normal images and fifty five OPG images	72%
3	Human identification using dental biometric analysis/2015	Different methods for segmentation and feature extraction	colored teeth images for 14 persons and dental radiographs images for 45 persons	Equal error rate (EER) 88.8% for colored images 85.7% for dental radiographs.
4	Dental X-Ray based human identification system for forensic/ 2017	Standard deviation (STD), Euler number & area taken from bite-wing X-ray image.	80 I 80 X-Ray images	70%
5	Human identification with dental panoramic radiographic images /2018	support vector machine & graphical probabilistic models	206 X-Ray images of 170 various subjects	rank-1 precision of 81% rank-2 accuracy of 89%
6	Identification based dental image/ 2018	Mean, distance, standard derivation (STD), variance & principal component analysis (PCA)	75 75 x-ray images belonging to 115 ppersons (five for every person	89%
7	The use of contourlet transformations in hybridization an development of an intelligent PSO algorithm to distinguish cbct radiation image/ 2020	Particle swarm optimization & contourlet transformation	100 CBCT radiographs images	98%

2. THE CONE BEAM COMPUTED TOMOGRAPHY

Arai *et al.* [10] and Mozzo *et al.* [11] working separately, presented the CBCT for the oral and maxillofacial applications and like computed tomography (CT), offered 3D investigation and increasingly precise imaging contrasted with 2D imaging. The financially savvy innovation of CBCT, prompted quick entrance into the field of dentistry with interest for responsibility of dental experts and dental instructors to investigate the uses of CBCT innovation. Radiographic assessment is necessary in diagnosis treatment planning in dentistry. Aside from packing three-dimensional life structures of the zone being radiographed into a two-dimensional picture, 2D imaging has many important drawbacks (including magnification, distortion, and superimposition), together prompting distortion of structures [12]. The applications of CBCT in dentistry include as: i) implantology: missing teeth substitution by dental implants requests precise visualization of the surgical site for the successful implant installation and to keep away from damage to adjacent important structures; ii) oral and maxillofacial surgery, orthodontics, end odontics, periodontics; iii) applications in temporom and ibular joint disorders; iv) applications in forensic dentistry: one of the parts of forensic dentistry is age estimation. Enamel is usually resistant to changes beyond ordinary wear and tear; on the other hand, the pulpodentinal complex displays physiologic and pathological alterations with aging. usually, to measure these changes, extraction and segmenting of teeth is vital, which isn't constantly a practicable decision. CBCT, conversely, provides a non-invasive substitute; and v) virtual treatment planning and simulations [13].

3. CONTOURLET ALGORITHM

It is a true way to represent two-dimensional images, and is a new way to effectively represent the contour and texture of images [14]. The transformation consists of two-layer filters, where the laplace pyramid transformation is used to achieve multi-domain analysis and obtain discontinuous points. After this, the multidirectional analysis is carried out by the directional filter bank in order to connect the non-continuous points in the form of a linear structure [15], [16]. By incorporating the laplace pyramid and the directional filter bank it produces a multi-directional filter [17].

3.1. Laplace's pyramid

Offers the means to achieve multiscale decomposition. In each step of decomposition it produces a lowpass downsampled version of the original image and a bandpass image. A coarse image with low frequencies and a more accurate image with additional high frequencies including point discontinuities are produced. This pattern can be repeated continuously in the lowpass image and is restricted only from the size of the original image Because of the downsampling [14].

3.2. Directional filter bank

The directional filter bank (DFB) is designed to obtain high frequency contents such as smooth circumference and directional edges [18]. The DFB analyzes each detailed sub-range from laplace's pyramid (LP) to a number of directional sub-ranges. The package passing images from the LP are fed into DFB so that directional information can be obtained. The scheme of the multilayer decomposition (Contourlet). The merging between LP and DFB forms a dual filter bank which is called a pyramidal directional filter bank that analyzes the image into directional subdomains with multiple scales [14], [19]. Figure 1 shows the contourlet transform diagram.

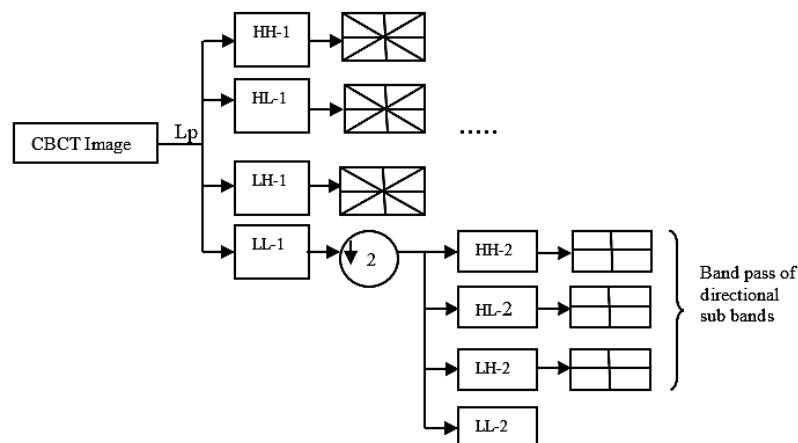


Figure 1. Contourlet transform diagram

4. PROPOSED ALGORITHM

4.1. Preprocessing

This is very important step which includes the basic steps performed on CBCT images. These steps improve the image quality and facilitate to use in the next stages. The steps of primary processing include the following Figure 2:

- Taking CBCT images for a group of persons and incorporating them into the suggested database system as shown Figure 2(a) Then all the CBCT images were read from the available system database so the preprocessing starts for every image according to the following steps.
- The CBCT images are converted from (RGB) to gray scale as shown Figure 2(b).
- Improving the gray scale image color distribution by changing color pixels values through the use of contrast limited at adaptive histogram equalization (CLAHS), which is one of the morphological algorithms where the result used as a mask for the next [20]. This step improves the image to a great extent so the teeth, facial bones and the bones surrounding the teeth appear in dark colors with a higher contrast as shown in Figure 2(c).
- Using another morphological algorithm known as (Erode) and applying it to the previous mask in order to restore the possible lost parts of the image in the previous stages through the use of SE=1, and “disk” function where they are used in rapidly restoring the lost parts without affecting the basic features of the image [21]. so, the teeth, jaws and facial bones become more clear. This step will result to a dark image that focuses on the cavities and bones as shown in Figure 2(d).
- Applying one of the morphological methods named (morphological reconstruction) between the mask and image resulted from the previous stage where the contacted points in the images are extracted and rearranged in a new image [22]. This method utilizes the gray colors in a hybrid way and rebuilds them in a better condition. The results of this stage can be noticed in Figure 2(e).
- The result from this stage gave a clear, less noise image which concentrating on the teeth and bones. the morphological methods and filters are considered effective in the image improvement process and color redistribution [23].
- Implementing sharpening for image improvement in order to make the borders be marked clearly including the edges of the teeth, bones and cavities. Also, dental fillings and the fixed dentures are seen in a clear white color as shown in Figure 2(f).
- The image is converted to binary as shown in the Figure 2(g).
- The black rows and colors surrounding the teeths and bones are deleted from the CBCT images as shown in Figure 2(h).

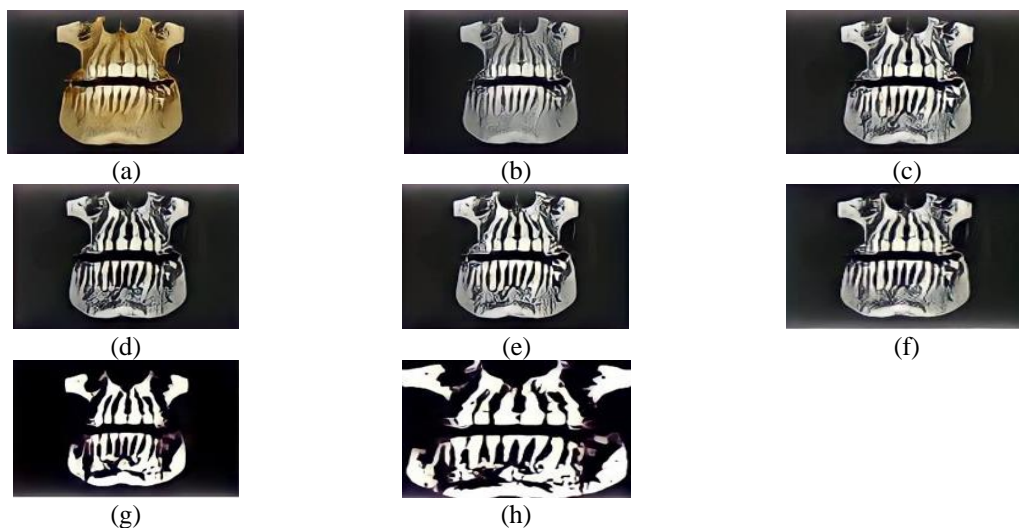


Figure 2. Preprocessing steps results for CBCT image, where (a) the source image of CBCT, (b) the CBCT image after converted from (RGB) to gray scale, (c) the CBCT image after using morphological algorithms (CLAHS), (d) the CBCT image after applying morphological algorithm (Erode), (e) the CBCT image after applying morphological reconstruction, (f) implementing sharpening for the CBCT image, (g) the previous CBCT image converted to binary, and (h) after deleting the black rows and colors surrounding the teeths and bones from output images

4.2. Feature extraction

The enhanced images resulted from previous stage are fed into 2-level contourlet transformation for feature extraction. The coefficients were unique for each individual image which gave a distinct feature for the particular image and thereby helped in the next step for recognition.

4.3. Apply contourlet particle swarm optimization algorithm

The main idea is to imitate the behavior of animals looking for food, such as fish, birds, or bees [24]. A variety of simple variations have been developed in order to increase the speed of convergence and the consistency of the solution found by the PSO [25]. PSO is developed for distinguishing images entered easily using extracted contourlet coefficients as the first generation in PSO algorithm.

The use of contourlet coefficients resulting from contourlet transformations as input to the PSO algorithm made the algorithm faster to reach the solution and gave more accurate results, because the inputs to the algorithm are coefficients of a size less than the size of the original images, and therefore the number of iteration needed to reach the required results became less by 1/10, Where the optimal solution is continuously searched for in these transactions until it reaches the ideal solution depending on search and repetition. The PSO algorithm considers each solution as a particle and has two significant characteristics: position n and velocity v_i . The two characteristics are related to each particle. So that $n = (n_{j1}, n_{j2}, \dots, n_{jN})$ and $v_i = (v_{i1}, v_{i2}, \dots, v_{iN})$, where N reflects the dimensions of the problem and at each stage, the particles in the swarm are given a fitness function at each stage of the search for a solution. The speed and location values are modified in accordance with the (1) and (2):

$$v = w * v + c_1 m_1(n \text{ Best} - 1) + c_2 m_2(p \text{ Best} - 1) \quad (1)$$

$$n = n + v \Delta t \quad (2)$$

where W : represents weight of inertia responsible for regulating the impact of particles on past velocities; c_1 , c_2 : positive constants that are referred to as parameters of acceleration; m_1 , m_2 : random values in each appearance take on new values; Δt : represents the time steps; Best : is the best current position that has entered the particle or passed it until the present moment; $p\text{Best}$: is the best current location reached or moved by a particle of neighboring particles until the present moment. The contourlet PSO (CPSO) algorithm pseudo code will be:

```
Input: Create community member sites by initialized position randomly of the practices:  $n_j$ 
(0) and velocity  $v_j$  (0).
Output: - best position of the global optima  $n^*$ .
 $F(n_j)$ =fitness, Which is calculated from equation [26]:
SSIM (a, b) =
```

$$\frac{(2\mu_a \mu_b + X_1)(2\sigma_{ab} + X_2)}{(\mu_a^2 + \mu_b^2 + X_1)(\mu_a^2 + \mu_b^2 + X_2)} \quad (3)$$

```
Begin
Repeat while max number of iteration is not reached do
Begin
For  $j=1$  to number of particles
IF  $F(n_j) < F(n_{newj})$  then
 $n_j = n_{newj}$ ;
Update  $v_j$ : using (1);
Update  $n_j$ : using (2);
 $j++$ ;
End
End
```

While fitness function for each member of the primary community is done using the similarity scale function (SSIM), where: μ_a : average of a , μ_b : average of b , μ_a^2 : variance of a , μ_b^2 : variance of b , σ_{ab} : Covariance of a and b , X_1, X_2 , are Two variables to allow consistency in the partition process with a non-strong denominator. The following Figure 3 summaries the whole work.

5. RESULTS AND ANALYSIS

The results obtained after processing 30 different CBCT images of different people using the proposed algorithm (CPSO), as shown in Table 2 and Figure 4. It showed 100% detection rate in the training stage, it is the optimum value obtained from any recognition system. In the testing stage, another 57 different radiographs have been processed and the results were 98%. when minor changes to the radiographs were made on 13 images

belonging to the same persons these changes include (a change in the teeth such as extraction, the placement of fillings, or a change in the dimensions of the radiographs of the same person) The result was 100%.

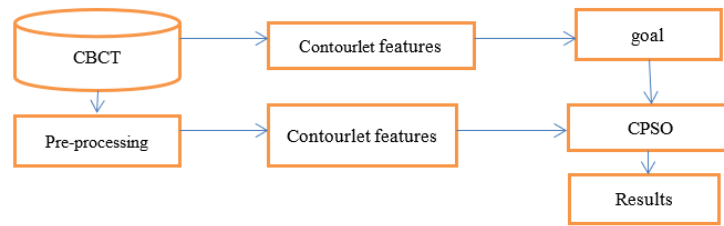


Figure 3. The work stages

Table 2. Comparisons with related studies

No. of processing	Type of processing	Type of dataset	No. of images	DR	ER	WR
1	Training	familiar	30	100	0	0
2	Testing	familiar	57	98	1.8	5
3	Testing	Unfamiliar (after changes)	13	100	0	0

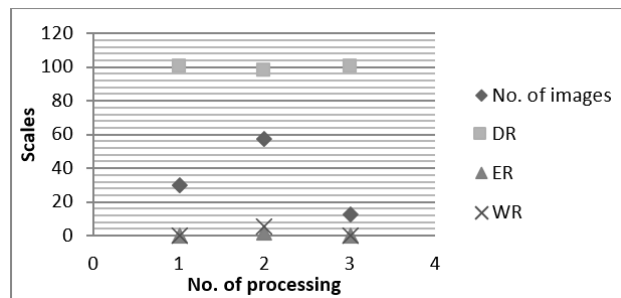


Figure 4. Results of proposed algorithm (CPSO)

These results showed the strength of the system in identifying dental rays through the use of primary treatment, which showed the jaws very clearly, with the cancellation of all unwanted parts. Also, these high percentages show how important it is to use the contourlet method in the process of extracting features and identifying important points from the x-ray images and using them as inputs to the proposed algorithm and obtain matching results in properties by means of the SSIM equation, which was used in developing PSO algorithm and reaching the goals directly with the least mistakes.

The scale used to calculate the overall degrees of discrimination in the system is detection rate (4). Where this measure represents the ability of the system to identify the person to whom the inserted dental rays belong. Therefore, it is calculated from the ratio of the number of correctly identified dental x-rays to the total number of x-ray images available in the system dataset. Therefore, the higher percentage of this scale is considered as a greater force for the system to discover the identity of the people to whom these rays belong. Thus, the value of detection rate is calculated as follows [26].

$$\text{Detection Rate (DR)} = (\text{no. of correctly detected sampels}) / (\text{total no. of samples}) * 100 \quad (4)$$

The extent of the system's errors in identifying people who have dental radiation inserted can be calculated by the scale Error Rate, and that was 1.8%, which is a small percentage that was calculated by the (5). This scale the lower its value, the higher the system has the ability to correctly achieve goals, due to the strength of the algorithm to find the right targets and avoid alien targets, This percentage can be calculated by the ratio of the number of images that were not found by the system from among the images entered to the number of images in the system dataset [26].

$$\text{Error Rate} = (\text{no. of false detected samples}) / (\text{total no. of samples}) * 100 \quad (5)$$

Also, the wrong acceptance rate (6) in this system was 5%, meaning that any radiograph that does not exist in the radiograph dataset of the system is not accepted. And this was done according to the following formula [27], [28]:

$$\text{Wrong acceptance rate} = (\text{no. of samples accepted error}) / (\text{total no. of images}) * 100 \quad (6)$$

6. CONCLUSION

The initial treatment stages that were performed on the images contributed to the clear visibility of the radiograph and in turn helped to obtain high results in the identification process. Using contourlet with the intelligent algorithm increases the ability of the algorithm to find the optimal solution effectively because it gives distinctive values for each image, which represent the characteristics and properties of that particular image. The challenges faced were the limited number of images available in the database which were used in the training and testing stages and difficulty distinguishing in the case of teeth falling out due to accidents or in the case of dental implant. The process of distinguishing dental rays is a good way to distinguish the identity of people, especially after physical changes have occurred to the external features of the human body over time. The radiograph identification rate with this system was high comparing with previous related studies, and thus the hybridization of the PSO algorithm with the Contourlet is considered a successful hybridization.

7. FUTURE WORKS

The possibility of making changes to a number of parameters of the proposed algorithm. This includes increasing the number of images acquired by the CBCT rays, also increasing the number of levels in the contourlet transformation, which leads to an increase in the number of features (coefficients) in the images and observe the magnitude of these effects on the results. Comparison of results using the proposed algorithm on both conventional and CBCT radiographs. Doing some noise on the images acquired in the database, applying to them the steps of the proposed algorithm, and observe the effect of this on the results. Use dental images to determine the identity of the deceased person by matching dental images of the deceased person provided by his relatives. The use of other methods in the discrimination process, such as machine learning.

ACKNOWLEDGEMENTS

Great thanks to the University of Mosul for its support and assistance during making this research.




REFERENCES

- [1] A. Jain, R. Bolle, and S. Pankanti, *Biometrics: personal identification in networked society*. USA: Kluwer Academic Publishers, 2006.
- [2] A. Morales, M. A. Ferrer, and A. Kumar, "Improved palmprint authentication using contactless imaging," Sep. 2010, doi: 10.1109/BTAS.2010.5634472.
- [3] S. Jeyanthi and N. Uma Maheswari and R. Venkatesh, "Human identification using dental biometrics," *International Journal of Advanced Engineering and Global Technology*, vol. 1, no. 5, 2013.
- [4] S. C. Dighe and R. Shriram, "Dental biometrics for human identification based on dental work and image properties in Periapical radiographs," Nov. 2012, doi: 10.1109/TENCON.2012.6412216.
- [5] R. Karunya, A. Askarunisa, and A. Athiraja, "Human identification using dental biometrics," *International Journal of Applied Engineering Research*, vol. 9, no. 1, pp. 4428–4433, 2014.
- [6] F. Rehman, M. U. Akram, K. Faraz, and N. Riaz, "Human identification using dental biometric analysis," in *2015 Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, Apr. 2015, pp. 96–100, doi: 10.1109/DICTAP.2015.7113178.
- [7] A. B. Oktay, "Human identification with dental panoramic radiographic images," *IET Biometrics*, vol. 7, no. 4, pp. 349–355, Nov. 2018, doi: 10.1049/iet-bmt.2017.0078.
- [8] S. H. Shaker and H. Najah, "Identification based dental image," *Journal of Al-Qadisiyah for computer science and mathematics*, vol. 10, no. 3, 2018, doi: 10.29304/jqcm.2018.10.3.438.
- [9] S. Dh Khudhur and M. S. Croock, "Dental X-Ray based human identification system for forensic," *Engineering and Technology Journal*, vol. 35, no. 1, pp. 49–60, 2017.
- [10] Y. Arai, E. Tammissalo, K. Iwai, K. Hashimoto, and K. Shinoda, "Development of a compact computed tomographic apparatus for dental use," *Dentomaxillofacial Radiology*, vol. 28, no. 4, pp. 245–248, 1999, doi: 10.1038/sj.dmfr.4600448.
- [11] P. Mozzo, C. Procacci, A. Tacconi, P. Tinazzi Martini, and I. A. Bergamo Andreis, "A new volumetric CT machine for dental imaging based on the cone-beam technique: Preliminary results," *European Radiology*, vol. 8, no. 9, pp. 1558–1564, Nov. 1998, doi: 10.1007/s003300050586.
- [12] W. C. Scarfe and A. G. Farman, "What is cone-beam CT and how does it work?," *Dental Clinics of North America*, vol. 52, no. 4, pp. 707–730, Oct. 2008, doi: 10.1016/j.cden.2008.05.005.
- [13] F. Yang, R. Jacobs, and G. Willems, "Dental age estimation through volume matching of teeth imaged by cone-beam CT," *Forensic Science International*, vol. 159, no. 1, pp. S78–S83, May 2006, doi: 10.1016/j.forsciint.2006.02.031.
- [14] S. Katsigiannis, E. G. Keramidias, and D. Maroulis, "A contourlet transform feature extraction scheme for ultrasound thyroid texture classification," *Final Draft-Published in Engineering Intelligent Systems, Special issue: Artificial Intelligence Applications and*




- Innovations*, vol. 18, no. 4, 2010.
- [15] K. I. AlSaif and M. M. Salih, "Text embedding based on contourlet transformation coefficients," *International Journal of Information Technology and Business Management*, vol. 12, no. 1, pp. 60–67, 2013.
- [16] D. Guo and J. Chen, "The application of contourlet transform to image denoising," *Procedia Engineering*, vol. 15, pp. 2333–2338, 2011, doi: 10.1016/j.proeng.2011.08.437.
- [17] D. B. Mali and J. B. Jadhav, "Performance comparison of contourlet and wavelet transform in denoising of ultrasound image," *International Journal of Development Research*, vol. 8, no. 10, pp. 23405–23409, 2018.
- [18] A. A. Dawood, "A Contourlet-Based Image Denoising Technique With Coefficient Threshold Level Estimation.," *Tikrit Journal Of Engineering Sciences*, vol. 20, no.4, pp. 11-22, March. 2013.
- [19] A. H. H. AlAsadi, "Contourlet transform based method for medical image denoising," *International Journal of Image Processing (IJIP)*, vol. 9, no. 1, pp. 22–31, 2015.
- [20] N. M. Sasi and V. K. Jayasree, "Contrast limited adaptive histogram equalization for qualitative enhancement of myocardial perfusion images," *Engineering*, vol. 05, no. 10, pp. 326–331, 2013, doi: 10.4236/eng.2013.510B066.
- [21] B. Gatos, S. J. Perantonis, N. Papamarkos, and I. Andreadis, "Fast implementation of morphological operations using binary image block decomposition," *International Journal of Image and Graphics*, vol. 4, no. 2, pp. 183–202, Apr. 2004, doi: 10.1142/S0219467804001361.
- [22] L. Vincent, "Morphological grayscale reconstruction: Definition, efficient algorithm and applications in image analysis," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 1992, vol. 1992-June, pp. 633–635, doi: 10.1109/CVPR.1992.223122.
- [23] I. Jivet, A. Brindusescu, and I. Bogdanov, "Image contrast enhancement using morphological decomposition by reconstruction," *WSEAS Transactions on Circuits and Systems*, vol. 7, no. 8, pp. 822–831, Aug. 2008.
- [24] W. R. Abdul-Adheem, "An enhanced particle swarm optimization algorithm," *International Journal of Electrical and Computer Engineering*, vol. 9, no. 6, pp. 4904–4907, Dec. 2019, doi: 10.11591/ijece.v9i6.pp4904-4907.
- [25] D. Palupi Rini, S. Mariyam Shamsuddin, and S. Sophiyati Yuhaniz, "Particle swarm optimization: technique, system and challenges," *International Journal of Computer Applications*, vol. 14, no. 1, pp. 19–27, Jan. 2011, doi: 10.5120/1810-2331.
- [26] D. S. Hammadi, A. N. YOUNIS, and F. M. RAMO, "Hybridization and modification of the pso algorithm and its use in personal recognition by opg x-ray," *Journal of Engineering Science and Technology*, vol. 16, no. 1, pp. 325–338, 2021.
- [27] M. J. Al-Shamdeen, A. N. Younis, and H. A. Younis, "Metaheuristic algorithm for capital letters images recognition," *International Journal of Mathematics and Computer Science*, vol. 16, no. 2, pp. 577–588, 2020.
- [28] A. N. Younis, "Personal identification system based on multi biometric depending on cuckoo search algorithm," *Journal of Physics: Conference Series*, vol. 1879, no. 2, p. 22080, May 2021, doi: 10.1088/1742-6596/1879/2/022080.

BIOGRAPHIES OF AUTHORS






Hiba Adreese Younis    Graduated from the collage of computer sciences and mathematics, university of Mosul, Iraq at 2003, she worked as a programmer at the same college till 2009 when she started studying masters of science at the college of computer sciences and mathematics (university of Mosul), then she finished MSC. Degree at 2011. Now she works as assistant lecturer at the college of computer sciences and mathematics (university of Mosul) specialized in multimedia processing, she has a research gate account under the name hiba adreese. She can be contacted at email: hibaadreese@uomosul.edu.iq.



Dhafar Sami Hammadi    Graduated from the department of computer sciences collage of computer sciences and mathematics, university of Mosul, Iraq. she worked as a programmer at the same college till 2013 when she started studying masters of science at the college of computer sciences and mathematics (university of Mosul), then she finished MSC. Degree at 2016. Now She work as assistant lecturer at the college of computer sciences and mathematics (university of Mosul) specialized in image processing and artificial intelligence, She can be contacted at email: dhafar_un@uomosul.edu.iq.



Ansam Nazar Younis    She has been an assistant literature at department of computer sciences, college of computer sciences and mathematics, the University of Mosul, Iraq since 2018, Graduated from the Computer Science and Mathematics Collage at the University of Mosul, Iraq in 2005, and worked as a programmer in the same collage until 2013 when she also started studying Masters of Science in the same collage, then she finished MSC. Degree at 2018. General expertise is computer science, and specialty is in the area of artificial intelligence and image processing. She. She has a research gate account under the name Ansam Nazar Younis. She can be contacted at email: anyma8@uomosul.edu.iq.

Paper's title should be the fewest possible words that accurately describe the content of the paper (Center, Bold, 16pt)

Abdel-Rahman Hedar^{1,2}, Patricia Melin³, Kennedy Okokpujie⁴ (10 pt)

¹Department of Computer Science, Faculty of Computers & Information, Assiut University, Assiut, Egypt (8 pt)

²Department of Computer Science in Jamoum, Umm Al-Qura University, Makkah, Saudi Arabia

³Division of Graduate Studies, Tijuana Institute of Technology, Tijuana, Mexico

⁴Department of Electrical and Information Engineering, College of Engineering, Covenant University, Ogun State, Nigeria

Article Info

Article history:

Received month dd, yyyy

Revised month dd, yyyy

Accepted month dd, yyyy

Keywords:

First keyword

Second keyword

Third keyword

Fourth keyword

Fifth keyword

ABSTRACT (10 PT)

An abstract is often presented separate from the article, so it must be able to stand alone. A well-prepared abstract enables the reader to identify the basic content of a document quickly and accurately, to determine its relevance to their interests, and thus to decide whether to read the document in its entirety. The abstract should be informative and completely self-explanatory, provide a clear statement of the problem, the proposed approach or solution, and point out major findings and conclusions. **The Abstract should be 100 to 200 words in length.** References should be avoided, but if essential, then cite the author(s) and year(s). Standard nomenclature should be used, and non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself. No literature should be cited. The keyword list provides the opportunity to add 5 to 7 keywords, used by the indexing and abstracting services, in addition to those already present in the title (9 pt).

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Kennedy Okokpujie

Department of Electrical and Information Engineering, College of Engineering, Covenant University

Km. 10 Idiroko Road, Canaan Land, Ota, Ogun State, Nigeria

Email: kennedy.okokpujie@covenantuniversity.edu.nga

1. INTRODUCTION (10 PT)

The main text format consists of a flat left-right columns on A4 paper (quarto). The margin text from the left and top are 2.5 cm, right and bottom are 2 cm. The manuscript is written in Microsoft Word, single space, Time New Roman 10 pt, and maximum 12 pages for original research article, or maximum 16 pages for review/survey paper, which can be downloaded at the website: <http://ijai.iaescore.com>.

A title of article should be the fewest possible words that accurately describe the content of the paper. The title should be succinct and informative and no more than about 12 words in length. Do not use acronyms or abbreviations in your title and do not mention the method you used, unless your paper reports on the development of a new method. Titles are often used in information-retrieval systems. Avoid writing long formulas with subscripts in the title. Omit all waste words such as "A study of ...", "Investigations of ...", "Implementation of ...", "Observations on ...", "Effect of....", "Analysis of ...", "Design of..." etc.

A concise and factual abstract is required. The abstract should state briefly the purpose of the research, the principal results and major conclusions. An abstract is often presented separately from the article, so it must be able to stand alone. For this reason, References should be avoided, but if essential, then cite the author(s) and year(s). Also, non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself. Immediately after the abstract, provide a maximum of 7 keywords, using American spelling and avoiding general and plural terms and

multiple concepts (avoid, for example, 'and', 'of'). Be sparing with abbreviations: only abbreviations firmly established in the field may be eligible. These keywords will be used for indexing purposes.

Indexing and abstracting services depend on the accuracy of the title, extracting from it keywords useful in cross-referencing and computer searching. An improperly titled paper may never reach the audience for which it was intended, so be specific.

The Introduction section should provide: i) a clear background, ii) a clear statement of the problem, iii) the relevant literature on the subject, iv) the proposed approach or solution, and v) the new value of research which it is innovation (within 3-6 paragraphs). It should be understandable to colleagues from a broad range of scientific disciplines. Organization and citation of the bibliography are made in Institute of Electrical and Electronics Engineers (IEEE) style in sign [1], [2] and so on. The terms in foreign languages are written italic (*italic*). The text should be divided into sections, each with a separate heading and numbered consecutively [3]. The section or subsection headings should be typed on a separate line, e.g., 1. INTRODUCTION. A full article usually follows a standard structure: **1. Introduction, 2. The Comprehensive Theoretical Basis and/or the Proposed Method/Algorithm (optional), 3. Method, 4. Results and Discussion, and 5. Conclusion.** The structure is well-known as **IMRaD** style.

Literature review that has been done author used in the section "INTRODUCTION" to explain the difference of the manuscript with other papers, that it is innovative, it are used in the section "METHOD" to describe the step of research and used in the section "RESULTS AND DISCUSSION" to support the analysis of the results [2]. If the manuscript was written really have high originality, which proposed a new method or algorithm, the additional section after the "INTRODUCTION" section and before the "METHOD" section can be added to explain briefly the theory and/or the proposed method/algorithm [4].

2. METHOD (10 PT)

Explaining research chronological, including research design, research procedure (in the form of algorithms, Pseudocode or other), how to test and data acquisition [5]–[7]. The description of the course of research should be supported references, so the explanation can be accepted scientifically [2], [4]. Figures 1-2 and Table 1 are presented center, as shown below and cited in the manuscript [5], [8]–[13]. The settlement curves produced at SG1 has been illustrated in Figure 2(a) and SG2 has been illustrated Figure 2(b).

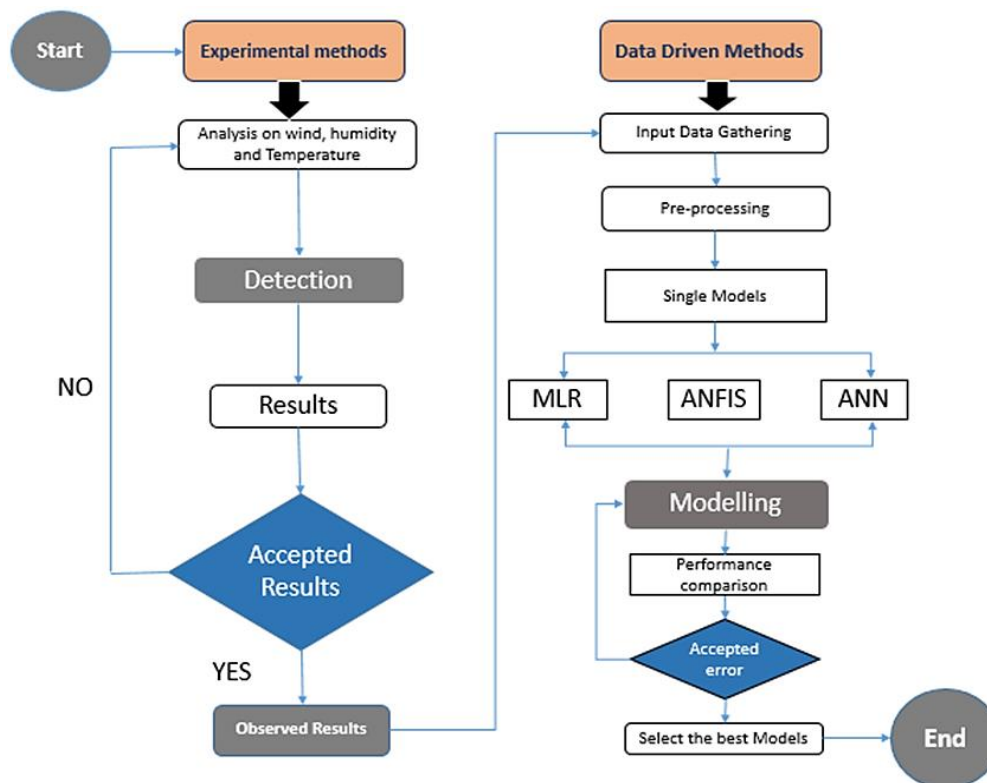


Figure 1. Shows the flowchart of the AI-based models and experimental methods applied

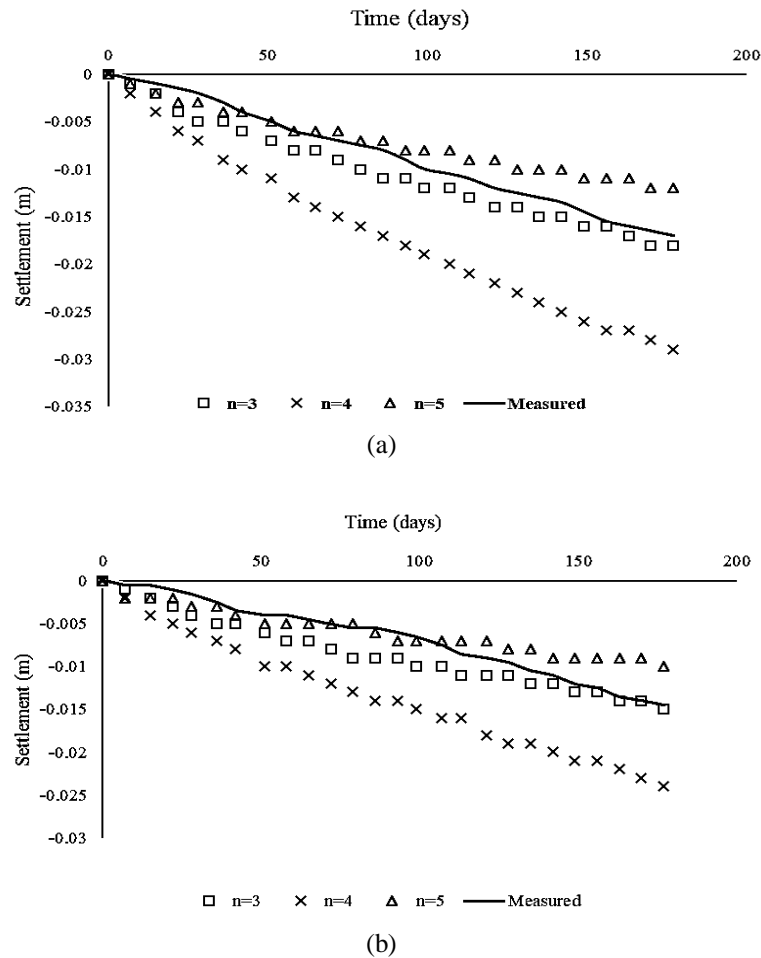


Figure 2. The relationship of soil settlement and time, (a) SG1 and (b) SG2

Table 1. The performance of ...

Variable	Speed (rpm)	Power (kW)
x	10	8.6
y	15	12.4
z	20	15.3

3. RESULTS AND DISCUSSION (10 PT)

In this section, it is explained the results of research and at the same time is given the comprehensive discussion. Results can be presented in figures, graphs, tables and others that make the reader understand easily [14], [15]. The discussion can be made in several sub-sections.

3.1. Sub section 1

Equations should be placed at the center of the line and provided consecutively with equation numbers in parentheses flushed to the right margin, as in (1). The use of Microsoft Equation Editor or MathType is preferred.

$$E_v - E = \frac{h}{2.m} (k_x^2 + k_y^2) \quad (1)$$

All symbols that have been used in the equations should be defined in the following text.

3.2. Sub section 2

Proper citation of other works should be made to avoid plagiarism. When referring to a reference item, please use the reference number as in [16] or [17] for multiple references. The use of "Ref [18]..."

should be employed for any reference citation at the beginning of sentence. For any reference with more than 3 or more authors, only the first author is to be written followed by *et al.* (e.g. in [19]). Examples of reference items of different categories shown in the References section. Each item in the references section should be typed using 9 pt font size [20]–[25].

3.2.1. Subsub section 1

yy

3.2.2. Subsub section 2

zz

4. CONCLUSION (10 PT)

Provide a statement that what is expected, as stated in the "INTRODUCTION" section can ultimately result in "RESULTS AND DISCUSSION" section, so there is compatibility. Moreover, it can also be added the prospect of the development of research results and application prospects of further studies into the next (based on result and discussion).

ACKNOWLEDGEMENTS (10 PT)

Author thanks In most cases, sponsor and financial support acknowledgments.

REFERENCES (10 PT)

The main references are international journals and proceedings. All references should be to the most pertinent, up-to-date sources **and the minimum of references are 25 entries** (for original research paper) and **50 entries** (for review/survey paper). References are written in **IEEE style**. For more complete guide can be accessed at (<http://ipmuonline.com/guide/refstyle.pdf>). Use of a tool such as **EndNote**, **Mendeley**, or **Zotero** for reference management and formatting, and choose **IEEE style**. Please use a consistent format for references-see examples (8 pt):

[1] Journal/Periodicals

Basic Format:

J. K. Author, "Title of paper," *Abbrev. Title of Journal/Periodical*, vol. x, no. x, pp. xxx-xxx, Abbrev. Month, year, doi: xxx.

Examples:

- M. M. Chiampi and L. L. Zilberti, "Induction of electric field in human bodies moving near MRI: An efficient BEM computational procedure," *IEEE Trans. Biomed. Eng.*, vol. 58, pp. 2787–2793, Oct. 2011, doi: 10.1109/TBME.2011.2158315.
- R. Fardel, M. Nagel, F. Nuesch, T. Lippert, and A. Wokaun, "Fabrication of organic light emitting diode pixels by laser-assisted forward transfer," *Appl. Phys. Lett.*, vol. 91, no. 6, Aug. 2007, Art. no. 061103, doi: 10.1063/1.2759475.

[2] Conference Proceedings

Basic Format:

J. K. Author, "Title of paper," in *Abbreviated Name of Conf.*, (location of conference is optional), year, pp. xxx-xxx, doi: xxx.

Examples:

- G. Veruggio, "The EURON roboethics roadmap," in *Proc. Humanoids '06: 6th IEEE-RAS Int. Conf. Humanoid Robots*, 2006, pp. 612–617, doi: 10.1109/ICHR.2006.321337.
- J. Zhao, G. Sun, G. H. Loh, and Y. Xie, "Energy-efficient GPU design with reconfigurable in-package graphics memory," in *Proc. ACM/IEEE Int. Symp. Low Power Electron. Design (ISLPED)*, Jul. 2012, pp. 403–408, doi: 10.1145/2333660.2333752.

[3] Book

Basic Format:

J. K. Author, "Title of chapter in the book," in *Title of His Published Book*, X. Editor, Ed., xth ed. City of Publisher, State (only U.S.), Country: Abbrev. of Publisher, year, ch. x, sec. x, pp. xxx-xxx.

Examples:

- A. Taflove, *Computational Electrodynamics: The Finite-Difference Time-Domain Method* in *Computational Electrodynamics II*, vol. 3, 2nd ed. Norwood, MA, USA: Artech House, 1996.
- R. L. Myer, "Parametric oscillators and nonlinear materials," in *Nonlinear Optics*, vol. 4, P. G. Harper and B. S. Wherret, Eds., San Francisco, CA, USA: Academic, 1977, pp. 47–160.

[4] M. Theses (B.S., M.S.) and Dissertations (Ph.D.)

Basic Format:

J. K. Author, "Title of thesis," M.S. thesis, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

J. K. Author, "Title of dissertation," Ph.D. dissertation, Abbrev. Dept., Abbrev. Univ., City of Univ., Abbrev. State, year.

Examples:

- J. O. Williams, "Narrow-band analyzer," Ph.D. dissertation, Dept. Elect. Eng., Harvard Univ., Cambridge, MA, USA, 1993.
- N. Kawasaki, "Parametric study of thermal and chemical nonequilibrium nozzle flow," M.S. thesis, Dept. Electron. Eng., Osaka Univ., Osaka, Japan, 1993.

*In the reference list, however, list all the authors for up to six authors. Use *et al.* only if: 1) The names are not given and 2) List of authors more than 6. *Example:* J. D. Bellamy *et al.*, Computer Telephony Integration, New York: Wiley, 2010.

See the examples:

REFERENCES

- [1] T. S. Ustun, C. Ozansoy, and A. Zayegh, "Recent developments in microgrids and example cases around the world—A review," *Renew. Sustain. Energy Rev.*, vol. 15, no. 8, pp. 4030–4041, Oct. 2011, doi: 10.1016/j.rser.2011.07.033.
- [2] D. Salomonsson, L. Soder, and A. Sannino, "Protection of Low-Voltage DC Microgrids," *IEEE Trans. Power Deliv.*, vol. 24, no. 3, pp. 1045–1053, Jul. 2009, doi: 10.1109/TPWRD.2009.2016622.
- [3] S. Chakraborty and M. G. Simoes, "Experimental Evaluation of Active Filtering in a Single-Phase High-Frequency AC Microgrid," *IEEE Trans. Energy Convers.*, vol. 24, no. 3, pp. 673–682, Sep. 2009, doi: 10.1109/TEC.2009.2015998.
- [4] S. A. Hosseini, H. A. Abyaneh, S. H. H. Sadeghi, F. Razavi, and A. Nasiri, "An overview of microgrid protection methods and the factors involved," *Renew. Sustain. Energy Rev.*, vol. 64, pp. 174–186, Oct. 2016, doi: 10.1016/j.rser.2016.05.089.
- [5] S. Chen, N. Tai, C. Fan, J. Liu, and S. Hong, "Sequence-component-based current differential protection for transmission lines connected with IIGs," *IET Gener. Transm. Distrib.*, vol. 12, no. 12, pp. 3086–3096, Jul. 2018, doi: 10.1049/iet-gtd.2017.1507.
- [6] S. Parhizi, H. Lotfi, A. Khodaei, and S. Bahramirad, "State of the Art in Research on Microgrids: A Review," *IEEE Access*, vol. 3, pp. 890–925, 2015, doi: 10.1109/ACCESS.2015.2443119.
- [7] S. Chowdhury, S. P. Chowdhury, and P. Crossley, *Microgrids and Active Distribution Networks*. Institution of Engineering and Technology, 2009.
- [8] R. Ndou, J. I. Fadiran, S. Chowdhury, and S. P. Chowdhury, "Performance comparison of voltage and frequency based loss of grid protection schemes for microgrids," in *2013 IEEE Power & Energy Society General Meeting*, 2013, pp. 1–5, doi: 10.1109/PESMG.2013.6672788.
- [9] S. Liu, T. Bi, A. Xue, and Q. Yang, "Fault analysis of different kinds of distributed generators," in *2011 IEEE Power and Energy Society General Meeting*, Jul. 2011, pp. 1–6, doi: 10.1109/PES.2011.6039596.
- [10] K. Jennett, F. Coffele, and C. Booth, "Comprehensive and quantitative analysis of protection problems associated with increasing penetration of inverter-interfaced DG," in *11th IET International Conference on Developments in Power Systems Protection (DPSP 2012)*, 2012, pp. P31–P31, doi: 10.1049/cp.2012.0091.
- [11] P. T. Manditereza and R. Bansal, "Renewable distributed generation: The hidden challenges – A review from the protection perspective," *Renew. Sustain. Energy Rev.*, vol. 58, pp. 1457–1465, May 2016, doi: 10.1016/j.rser.2015.12.276.
- [12] D. M. Bui, S.-L. Chen, K.-Y. Lien, Y.-R. Chang, Y.-D. Lee, and J.-L. Jiang, "Investigation on transient behaviours of a unigrounded low-voltage AC microgrid and evaluation on its available fault protection methods: Review and proposals," *Renew. Sustain. Energy Rev.*, vol. 75, pp. 1417–1452, Aug. 2017, doi: 10.1016/j.rser.2016.11.134.
- [13] T. N. Boutsika and S. A. Papathanassiou, "Short-circuit calculations in networks with distributed generation," *Electr. Power Syst. Res.*, vol. 78, no. 7, pp. 1181–1191, Jul. 2008, doi: 10.1016/j.epsr.2007.10.003.
- [14] H. Margossian, G. Deconinck, and J. Sachau, "Distribution network protection considering grid code requirements for distributed generation," *IET Gener. Transm. Distrib.*, vol. 9, no. 12, pp. 1377–1381, Sep. 2015, doi: 10.1049/iet-gtd.2014.0987.
- [15] O. Núñez-Mata, R. Palma-Behnke, F. Valencia, A. Urrutia-Molina, P. Mendoza-Araya, and G. Jiménez-Estévez, "Coupling an adaptive protection system with an energy management system for microgrids," *Electr. J.*, vol. 32, no. 10, p. 106675, Dec. 2019, doi: 10.1016/j.tej.2019.106675.
- [16] M. Brucoli and T. C. Green, "Fault behaviour in islanded microgrids," in *Proceedings of the 19th international conference on electricity distribution, CIRED*, 2007, pp. 0548-(1-4).
- [17] I. K. Tarsi, A. Sheikholeslami, T. Barforoushi, and S. M. B. Sadati, "Investigating impacts of distributed generation on distribution networks reliability: A mathematical model," in *Proceedings of the 2010 Electric Power Quality and Supply Reliability Conference*, Jun. 2010, pp. 117–124, doi: 10.1109/PQ.2010.5550010.
- [18] L. K. Kumpulainen and K. T. Kauhaniemi, "Analysis of the impact of distributed generation on automatic reclosing," in *IEEE PES Power Systems Conference and Exposition, 2004.*, pp. 1152–1157, doi: 10.1109/PSCE.2004.1397623.
- [19] A. A. Memon and K. Kauhaniemi, "A critical review of AC Microgrid protection issues and available solutions," *Electr. Power Syst. Res.*, vol. 129, pp. 23–31, Dec. 2015, doi: 10.1016/j.epsr.2015.07.006.
- [20] H. A. Abdel-Ghany, A. M. Azmy, N. I. Elkalashy, and E. M. Rashad, "Optimizing DG penetration in distribution networks concerning protection schemes and technical impact," *Electr. Power Syst. Res.*, vol. 128, pp. 113–122, Nov. 2015, doi: 10.1016/j.epsr.2015.07.005.
- [21] S. Chaitusaney and A. Yokoyama, "An Appropriate Distributed Generation Sizing Considering Recloser-Fuse Coordination," in *2005 IEEE/PES Transmission & Distribution Conference & Exposition: Asia and Pacific*, pp. 1–6, doi: 10.1109/TDC.2005.1546838.
- [22] H. H. Zeineldin, Y. A.-R. I. Mohamed, V. Khadkikar, and V. R. Pandi, "A Protection Coordination Index for Evaluating Distributed Generation Impacts on Protection for Meshed Distribution Systems," *IEEE Trans. Smart Grid*, vol. 4, no. 3, pp. 1523–1532, Sep. 2013, doi: 10.1109/TSG.2013.2263745.
- [23] D. Eltigani and S. Masri, "Challenges of integrating renewable energy sources to smart grids: A review," *Renew. Sustain. Energy Rev.*, vol. 52, pp. 770–780, Dec. 2015, doi: 10.1016/j.rser.2015.07.140.
- [24] M. M. Eissa (SIEEE), "Protection techniques with renewable resources and smart grids—A survey," *Renew. Sustain. Energy Rev.*, vol. 52, pp. 1645–1667, Dec. 2015, doi: 10.1016/j.rser.2015.08.031.
- [25] A. Oudalov *et al.*, "Novel Protection Systems for Microgrids," 2009. [Online]. Available: <http://www.microgrids.eu/documents/688.pdf>.

BIOGRAPHIES OF AUTHORS (10 PT)

The recommended number of authors is at least 2. One of them as a corresponding author.

Please attach clear photo (3x4 cm) and vita. Example of biographies of authors:

	<p>Abdel-Rahman Hedar    holds a Doctor of Informatics degree from Kyoto University, Japan in 2004. He also received his B.Sc. and M.Sc. (Mathematics) from Assiut University, Egypt in 1993 and 1997, respectively. He is currently an associate professor at Computer Science Department in Jamoum, Umm Al-Qura University, Makkah, Saudi Arabia. He is also an associate professor of artificial intelligence in Assiut University since January 2012. His research includes meta-heuristics, global optimization, machine learning, data mining, bioinformatics, graph theory and parallel programming. He has published over 70 papers in international journals and conferences. From July 2005 to July 2007, he was a JSPS research fellow in Kyoto University, Japan. He can be contacted at email: ahahmed@uqu.edu.sa or hedar@aun.edu.eg.</p>
	<p>Patricia Melin    received the D.Sc. degree (Doctor Habilitatus D.Sc.) in computer science from the Polish Academy of Sciences, Warsaw, Poland, with the Dissertation “Hybrid Intelligent Systems for Pattern Recognition using Soft Computing”. She is a Professor of Computer Science in the Graduate Division, Tijuana Institute of Technology, Tijuana, Mexico since 1998. In addition, she is serving as Director of Graduate Studies in computer science and Head of the research group on Computational Intelligence (2000–present). Her research interests are in Type-2 Fuzzy Logic, Modular Neural Networks, Pattern Recognition, Neuro-Fuzzy and Genetic-Fuzzy hybrid approaches., She is currently the President of Hispanic American Fuzzy Systems Association (HAFSA) and is the founding Chair of the Mexican Chapter of the IEEE Computational Intelligence Society. She can be contacted at email: pmelin@tectijuana.mx.</p>
	<p>Dr. Kennedy Okokpujie    holds a Bachelor of Engineering (B.Eng.) in Electrical and Electronics Engineering, Master of Science (M.Sc.) in Electrical and Electronics Engineering, Master of Engineering (M.Eng.) in Electronics and Telecommunication Engineering and Master of Business Administration (MBA), Ph.D in Information and Communication Engineering, besides several professional certificates and skills. He is currently lecturing with the department of Electrical and Information Engineering at Covenant University, Ota, Ogun State, Nigeria. He is a member of the Nigeria Society of Engineers and the Institute of Electrical and Electronics Engineers (IEEE). His research areas of interest include Biometrics, Artificial Intelligent, and Digital signal Processing. He can be contacted at email: kennedy.okokpujie@covenantuniversity.edu.ng.</p>