

In the Name of God

Journal of

Information Systems & Telecommunication

Vol. 12, No.4, October-December 2024, Serial Number 47

Research Institute for Information and Communication Technology

Iranian Association of Information and Communication Technology

Affiliated to: Academic Center for Education, Culture and Research (ACECR)

Manager-in-Charge: Dr. Habibollah Asghari, ACECR, Iran

Editor-in-Chief: Dr. Masoud Shafiee, Amir Kabir University of Technology, Iran

Editorial Board

Dr. Abdolali Abdipour, Professor, Amirkabir University of Technology, Iran

Dr. Ali Akbar Jalali, Professor, Iran University of Science and Technology, Iran

Dr. Alireza Montazemi, Professor, McMaster University, Canada

Dr. Ali Mohammad-Djafari, Associate Professor, Le Centre National de la Recherche Scientifique (CNRS), France

Dr. Hamid Reza Sadegh Mohammadi, Associate Professor, ACECR, Iran

Dr. Mahmoud Moghavvemi, Professor, University of Malaya (UM), Malaysia

Dr. Mehrnoush Shamsfard, Associate Professor, Shahid Beheshti University, Iran

Dr. Omid Mahdi Ebadati, Associate Professor, Kharazmi University, Iran

Dr. Rahim Saeidi, Assistant Professor, Aalto University, Finland

Dr. Ramezan Ali Sadeghzadeh, Professor, Khajeh Nasireddin Toosi University of Technology, Iran

Dr. Sha'ban Elahi, Associate Professor, Tarbiat Modares University, Iran

Dr. Shohreh Kasaei, Professor, Sharif University of Technology, Iran

Dr. Saeed Ghazi Maghrebi, Assistant Professor, ACECR, Iran

Dr. Zabih Ghasemlooy, Professor, Northumbria University, UK

Executive Editor: Dr. Fatemeh Kheirkhah

Executive Manager: Shirin Gilaki

Executive Assistants: Mahdokht Ghahari, Ali BoozarPoor

Print ISSN: 2322-1437

Online ISSN: 2345-2773

Publication License: 91/13216

Editorial Office Address: No.5, Saeedi Alley, Kalej Intersection., Enghelab Ave., Tehran, Iran,

P.O.Box: 13145-799

Tel: (+9821) 88930150 Fax: (+9821) 88930157

E-mail: info@jist.ir , infojist@gmail.com

URL: www.jist.ir

Indexed by:

- | | |
|---|-------------------------|
| - SCOPUS | www.Scopus.com |
| - Index Copernicus International | www.indexcopernicus.com |
| - Islamic World Science Citation Center (ISC) | www.isc.gov.ir |
| - Directory of open Access Journals | www.Doaj.org |
| - Scientific Information Database (SID) | www.sid.ir |
| - Regional Information Center for Science and Technology (RICEST) | www.ricest.ac.ir |
| - Magiran | www.magiran.com |

Publisher:

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

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

Acknowledgement

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

(A-Z)

- Alaeiyan, Mohammad Hadi, K.N. Toosi University of Technology, Tehran, Iran
- Azarkasb, Seyed Omid, K.N. Toosi University of Technology, Tehran, Iran
- Entezari Maleki, Reza, Iran University of Science and Technology (IUST), Tehran, Iran
- Fadaeieslam, Mohammad Javad, Semnan University, Iran
- Farsi, Hassan, University of Birjand, South Khorasan, Iran
- Farsijani, Hassan, Shahid Beheshti University, Tehran, Iran
- Ghasemzadeh, Ardalán, Urmia University of Technology, West Azerbaijan, Iran
- Ghaemi, Reza, Islamic Azad University, Quchan Branch, Iran
- Hosseini, Monireh, K. N. Toosi University of Technology, Tehran, Iran
- Kasaei, Shohreh, Sharif University, Tehran, Iran
- Khazaei, Mehdi, Kermanshah University of Technology, Kermanshah, Iran
- Kheirkhah, Fatemeh, ACECR, Tehran, Iran
- Khorshidi, Mohammadreza, University of Birjand, South Khorasan, Iran
- Kadhim, Ghadah, University of Baghdad, Baghdad, Iraq
- Marvi, Hossein, Shahrood University of Technology, Semnan Province, Iran
- Mohammadzadeh, Sajjad, University of Birjand, South Khorasan, Iran
- Moayedi, Fatemeh, University of Larestan Higher Education Complex, Fars, Iran
- Montazemi, Alireza, McMaster University, Canada
- Mirroshandel, Seyed Abolghasem, University of Guilan, Rasht, Iran
- Nangir, Mahdi, University of Tabriz, Tabriz, Iran
- Omid Mahdi, Ebadati, Kharazmi University, Tehran, Iran
- Patange, Abhishek, ABB Group, Zurich, Switzerland
- Soleimani Gharehchopogh, Farhad, Islamic Azad University Urmia, Iran
- Sadatfar, Hamid, University of Birjand, Iran
- Tourani, Mahdi, University of Birjand, South Khorasan, Iran
- Tashtarian, Farzad, Islamic Azad Mashad University, Mashad, Iran
- Yari, Alireza, ICT Research Institute, Tehran, Iran
- Zakeri, Bijan, Babol Noshirvani University of Technology, Mazandaran, Iran

Table of Contents

- **Sketch_Based Image Retrieval Using Convolutional Neural Network with Multi_Step Training242**
Azita.Gheitasi, Hassan.Farsi and Sajad.Mohamadzadeh
- **Enhancing Speaker Identification System Based on MFCC Feature Extraction and Gated Recurrent Unit Network.....254**
Mojtaba Sharif Noughabi , Seyyed Mohammad Razavi and Mehran Taghipour-Gorjikotaie
- **Load Balancing Algorithms in Cloud, Fog Computing and Convergence of Fog and Cloud – A Survey 264**
Seyedeh Leili Mirtaheri, Mahya Azari Jafari , Sergio Greco , Ehsan Arianyan and Reza Mansouri
- **An Energy-Aware Approach to Virtual Machine Consolidation Using Classification and the Dragonfly Algorithm in Cloud Data Centers 280**
Nastaran Evaznia , Reza Ebrahimi and Davoud Bahrepour
- **Enhancing the Quality of ICT Regulation in Iran: A Study on the Application of the COBIT IT Governance Framework 291**
Ehsan Baraty , Akbar Nabiollahi and Naser Khani
- **Design and Analysis of a Secure Intelligent Blood Management Information System 300**
Hessah Alhajri, Mostafa Abd El-Barr and Kalim Qureshi

Sketch_Based Image Retrieval Using Convolutional Neural Network with Multi_Step Training

Azita.Gheitasi¹, Hassan.Farsi¹, Sajad.Mohamadzadeh^{1*}

1.Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran

Received: 14 Feb 2023/ Revised: 10 Oct 2023/ Accepted: 27 Nov 2023

Abstract

The expansion of touch-screen devices has provided the possibility of human-machine interactions in the form of free-hand drawings. In sketch-based image retrieval (SBIR) systems, the query image is a simple binary design that represents the mental image of a person with the rough shape of an object. A simple sketch is convenient and efficient for recording ideas visually, and can outdo hundreds of words. The objective is to retrieve a natural image with the same label as the query sketch. This article presents a multi-step training method. Regression functions are used in the deep network structure to improve system performance, and various loss functions are employed for a better convergence of the retrieval system. The convolutional neural network used has two branches, one related to the sketch and the other related to the image, and these two branches can have the same or different architecture. After four training steps, a 56.48% MAP was achieved, indicating the desirable performance of the network.

Keywords: Sketch-Based Image Retrieval (SBIR); Deep Learning; Multi-Step Training; Contrastive loss; Triplet loss.

1- Introduction

The advances in multimedia technologies and the widespread use of the Internet have fundamentally changed human life. Audio, video, and images are known as multimedia data and can be useful in various fields, including military, medical, legal, and commercial [1]. Any system that can analyze and recover these data can be efficient and valuable. Among these, images are the most popular multimedia data [2].

The issue of image retrieval can be done in different ways, for example, content-based image retrieval or (CBIR) can be mentioned, which has been of interest in the past [3]. But here, discussed issue is sketch-based image retrieval. As stated, this issue involves retrieving a natural image with the same label as the query sketch. It mainly focuses on extracting representative and shared features from simple sketches and natural images [4]. Scale-invariant feature transform (SIFT) is one of the most common matching methods previously used in the remote sensing image registration[5]. The challenge in SBIR is that free-hand sketches are inherently abstract and symbolic, which magnifies the cross-domain discrepancy between sketches

and the real image. Deep learning methods are used to alleviate this problem [6].

For a better understanding of the subject, a description could be provided about the differences between sketches and real images. Sketches solely have the holistic shape and salient local shapes (and sometimes symbolic colors), while real images have details on shape, color, and texture. Most sketches contain no background, while real images can have cluttered and complex backgrounds. Even when a sketch and an edge map depict the same object or scene, their abstraction levels are dramatically different. This difference is due to the randomness of the sketch lines, simplification and missing details, disproportion, and unrealistic objects (several parts of objects are drawn unrealistically) in sketches [2]. In general, sketches represent the shape and spatial position, while real images include other useful information, such as color and texture [1]. Sketches are considered a highly scattered signal compared to real images, and their analysis is challenging due to the low input information and the abstractness of sketches. Therefore, comparing low-detail images with pixel-dense real images is difficult [7].

A method of collecting sketch data is edge detection techniques and algorithms, such as the fuzzy-based ACO algorithm [8] or using fuzzy cognitive map [9]. This paper, presents a comprehensive investigation of triplet embedding strategies evaluating on three databases (Quick-Draw, TU-

✉ Sajad Mohamadzadeh
s.mohamadzadeh@birjand.ac.ir

Berlin, and Sketchy). Similar to papers on deep networks for object recognition [10], the present study explores appropriate CNN architectures, weight-sharing schemes, and training methodologies to learn a low-dimensional embedding for the representation of both sketches and photographs in practical terms as a space amenable for a fast approximation of the nearest neighbor (ANN) search (e.g., L2 norm) for SBIR. Also, a novel triplet architecture and training methodology is proposed that is capable of generalizing across hundreds of object categories, and its performance is demonstrated in comparison to existing SBIR methods by a significant margin on leading benchmarks.

We propose a multi-step training methodology and investigate several network designs, comparing the Siamese architecture with the Heterogeneous and Hybrid ones. We aimed to develop a training strategy for partial sharing networks.

2- Related works

Sketch-based image retrieval (SBIR) has been studied since the early 1990s, and content-based retrieval (CBIR) were the subject of discussions from 1990 to 1994 [2]. This field remains attractive to researchers. For example, one researcher on CBIR has presented a method based on the combination of Hadamard matrix, discrete wavelet transform (HDWT2), and discrete cosine transform of DCT [11]. From 1994 onwards, studies on sketch-based image retrieval (SBIR) began [2]. Del Bimbo and et al. [12] introduced a module called the object localization, which separated and selected the main areas of an image with the help of rectangles, normalized these windows to be the same size, and then coded their spatial relationships. In this method, only the main subjects of the image were selected and compared. So far, all the reviewed works have employed pixel-based similarity metrics, but these metrics usually require costly computations and have little flexibility. Later, the feature extraction module was introduced to extract various feature types, which were robust to edge variations. Chans et al. [13] believed that users tended to ignore details when drawing the sketches and proposed a curvelet model to extract and encode the prominent edge segments of images. Rajendra and Cheng [14] used a multi-scale representation of edge maps to indicate changes in the level of detail in human-drawn sketches. They believed that the combination of scales preserved the details of the sketch. In another method, a binary mask was used for objects that spatially matched the real image. Another method is gallery displaying module, which uses K-means tree and best-bin-first strategy in combination. The combination of these two algorithms accelerated the recovery speed by several times [2, 15].

Another pixel-based method is OCM, which seeks the closest edge pixel in the sketch that is related to the image. More recently, with the introduction of deep learning and the

use of deep neural networks, research in the field of SBIR took a new form [16, 17]. Convolution networks are comprehensive and efficient in image processing and alleviate numerous deficiencies and ambiguities of data. Neural network-based methods are generally robust in identifying data patterns, superior in speed, flexible against environmental changes, and provide better performance than classic statistical models [18]. Recently, custom architectures such as Alex-Net, Google-Net [19] combined CNN models, and multi-objective ranking networks [20] have been used to rank and predict features. Sketch-A-Net is a deep networks designed for sketch-based image retrieval problem [6].

It explores recognition (rather than search) using a single-branch network resembling a short-form Alex-Net [10]. Sketch-A-Net is a component of the works of Bhattacharjee et al. [21] and Sain et al. [7]. Sketch-A-Net is also explored in the present study and compared with several other contemporary architectures.

An early work on multi-branch networks for sketch retrieval (of 3D objects) was the contrastive loss network by Wang et al. [22], which independently learned branch weights to bridge the domains of sketch and 2D renderings of silhouette edges. In a recent short paper, Qi et al. [23] propose a two-branch Siamese network with contrastive loss. Their results, although comparable with other methods using shallow features, are still far behind state-of-the-art by a large margin. As we show later, learning a single function to map disparate domains to the search space appears to underperform designs where branch weights are learned independently or semi-independently.

Triplet CNNs employ three branches [24, 25]: (i) an anchor branch, which models the reference object, (ii) one branch representing positive examples (which should be similar to the anchor) and (iii) another modeling negative examples (which should differ from the anchor). The triplet loss function is responsible for guiding the training stage considering the relationship between the three models. Triplet CNNs have recently been explored for face identification [26], tracking [27], photographic visual search [28], and sketched queries to refine search within a single object class (e. g. fine-grain search within a dataset of shoes) [7]. Similarly, a fine-grained approach to SBIR was adopted by the recent Sketchy system of Sangkloy et al. [29] in which careful reproduction of stroke detail is invited for object instance search. Researchers report that using a fully-shared network was better than using two branches without weight sharing. However, the authors in [29] suggest it is more beneficial to avoid sharing any layers in a cross-category retrieval context. Also, a hybrid design was explored by Bui et al. [30] using the same architecture on both branches but sharing certain layers. However, as their model learns a mapping between sketch and edge map (rather than image directly) its performance is limited. Furthermore, it is still unclear whether triplet loss works better than contrastive loss.

This paper uses a generic multi-step training methodology for cross-domain learning that leverages several loss functions in training shared networks as illustrated in Figures 1, 2 and 3. Also an extensive evaluation of ConvNet architectures and weight-sharing strategies is carried out.

3- Proposed Method

The present study proposes a multi-step training method and examines several network architectures. This method, training the network independently at first (without sharing the weights) and then training in shared manner to modify and improve the performance. Lastly other data sets are applied to modify the weights of the training system. Two functions are used in this training process: contrast loss and triple loss.

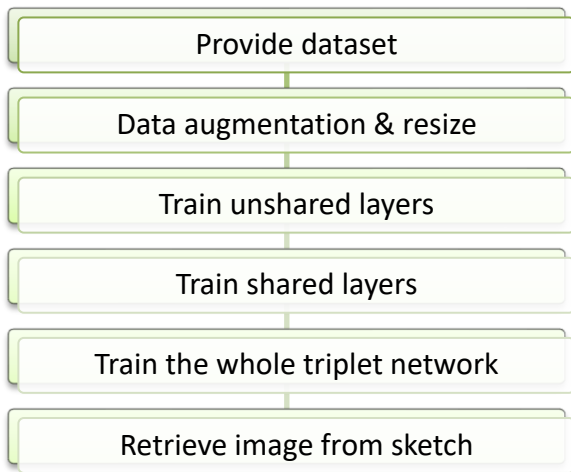


Fig. 1 Block diagram of the multi-step training SBIR system

Figure 1 shows the block diagram of the proposed multi-step training sketch-based image retrieval method. Datasets were collected in the first block. In the upcoming research, three well-known and extensive datasets in sketch-based image retrieval are used: Sketchy, TU-Berlin, and Quick-Draw. In the second block, some pre-processing is done on datasets for equalization before entering into the networks for training. First, all the images were set to 256×256 . Since the datasets contained sketches with different thicknesses, they were all equalized via the thinning method used in [10]. The data augmentation process was done (fully explained in Section 3.3. later). After pre-processing the data, the training phase began. The third block involved the unshared training step, which is the first step of the proposed training. At this step, the training was done independently without sharing the weights of the layers. That is, the sketch branch and the image branch were trained separately using Soft-max loss for a simple classification. The fourth block involved shared training, the second step of the proposed method. In this step,

a two-branch network was formed, and the unshared layers of the previous step were frozen. Soft-max loss and contrast loss (Eq. 3) functions were used to train shared layers in this step. In the next block, the third step of the proposed method, all the layers were defrosted. The training then continued by forming a triplet network and triplet loss and soft-max loss functions. After these steps, the image of the sketch was finally retrieved.

Table 1 shows the summary of the literature.

Table 1. The summary of the literature

<i>step</i>	<i>explanation</i>
1	Collecting datasets: (Sketchy, Tu-Berlin, Quick-Draw)
2	Pre-processing images: (resize all datasets to 256×256 , equalizing, and ...)
3	Unshared training: (for sketch branch training was done independently without sharing the weights of the layers, and for image branch training was done separately using soft-max loss for simple classification.)
4	Shared training: (we have two-branch network. unshared layers of the previous step were frozen. Soft-max and contrast loss functions were used)
5	Training triplet network: (all the layers were defrosted. Training continued by forming a triplet network and triplet and soft-max loss functions.)
6	Retrieval: (the image of the sketch was retrieved)

3-1- Architecture

Investigating a sketch-based image retrieval problem, requires at least one deep convolution bifurcation network. The branch architecture related to sketch and image can be the same or different. This paper, investigated Sketch-A-Net, Alex-Net, VGG-16 and InceptionV1 (Google-Net) for the sketch branch and Alex-Net, VGG-16, and InceptionV1 for the image branch. Low-level features are often learned in the lower layers of the convolutional network, while semantic features are obtained by training the upper layers. Therefore, in this process, the upper layers are trained jointly and the lower layers independently. All possible permutations with the mentioned architectures are explored for the sketch and image branches. When the architectures of the sketch and image branches are completely different, one or more fully connected layers are required to unify the branches.

Here, the loss functions used in the training process are described. Let $X^s = \{x^s\}$ and $X^l = \{x^l\}$ be collections of training sketches and images, respectively. The contrastive loss function accepts a pair of input examples (x^s, x^l) and regresses their embedding closer or pushes them away, depending on whether x^s and x^l are similar [10]. Let Y represents the label of a training pair (x^s, x^l) so that:

$$Y = \begin{cases} 0 & \text{if } (x^s, x^l) \text{ are similar} \\ 1 & \text{if } (x^s, x^l) \text{ are dissimilar} \end{cases} \quad (1)$$

The cross-domain Euclidean distance between the outputs of the two branches is calculated as:

$$D(x^s, x^l) = \|F_{\theta_s, \theta_c}^s(x^s) - F_{\theta_l, \theta_c}^l(x^l)\|_2 \quad (2)$$

Where parameters θ_s and θ_l represent domain-specific layers, θ_c is the shared part, and $F_{\theta_s, \theta_c}^S(x^S)$ and $F_{\theta_l, \theta_c}^l(x^l)$ are the embedding functions for sketch and image domains, respectively.

The contrastive loss is thus defined as:

$$\mathcal{L}_c(Y, X^S, X^l) = \frac{1}{2}(1 - Y)D^2(X^S, X^l) + \frac{1}{2}Y\{m - D^2(X^S, X^l)\}_+ \quad (3)$$

In which $\{\cdot\}_+$ is hinge loss function, and m is a defining margin and acceptable threshold for the dissimilarity of the sketch and image.

Triplet loss [7] maintains a relative distance between the anchoring example and both a similar and a dissimilar example. For the triplet input (X^S, X_+^l, X_-^l) , where X^S is an anchor sketch, and X_+^l is a similar and X_-^l is a dissimilar image, the triplet loss defined as:

$$\mathcal{L}_c(X^S, X_+^l, X_-^l) = \frac{1}{2}\{m + D^2(X^S, X_+^l) - D^2(X^S, X_-^l)\}_+ \quad (4)$$

The CNN network consists of three branches to accommodate the triplet input (X^S, X_+^l, X_-^l) : a sketch branch

(anchor) and two identical image branches (positive and negative). The value of margin m is set as 0.2 in all experiments Suggested in reference [10].

An intermediate, fully-connected (FC) layer is added without post-activation to learn the dimensionality reduction during the training steps. An embedding layer lower-dim is added between layer FC7 (D= 4096) and the output layer FC8 (D = 250) without activation ReLU (fig.1). The connection from FC7 to FC8 is linear. The presence of the domain reduction layer does not affect the performance of the classification layer.

3-2- Training

The proposed multi-step training has four steps:

- Step 1

In this step, the unshared layers learn the features distinctive to their domain without being mixed with other domains (figure 2).

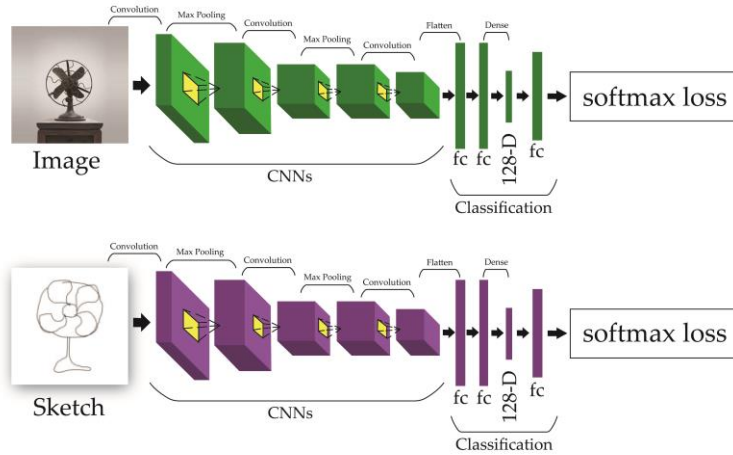


Fig. 2 Training the unshared layers

\mathcal{L}_E and \mathcal{L}_R denote the cross entropy and regularization losses:

$$\mathcal{L}_E(Z) = -\log\left(\frac{e^{zy}}{\sum_i e^{zi}}\right) \quad (5)$$

$$\mathcal{L}_R(\theta) = \frac{1}{2}\sum_i \theta_i^2 \quad (6)$$

So, in step 1, equations 7 and 8 show the representative model for each domain:

$$\arg \min_{\theta_s, \theta_c} \sum_l \mathcal{L}_E(F^S(X_i^S)) + \lambda \mathcal{L}_R(\theta_s, \theta_c) \quad (7)$$

$$\arg \min_{\theta_l, \theta_c} \sum_i \mathcal{L}_E(F^l(X_i^l)) + \lambda \mathcal{L}_R(\theta_l, \theta_c) \quad (8)$$

Where λ is the weight decay term, and θ_c was learned independently.

- Step 2

In this step, the shared layers learn the high-level common features between the two domains by comparing and contrasting the low-level features from both domains (figure 3).

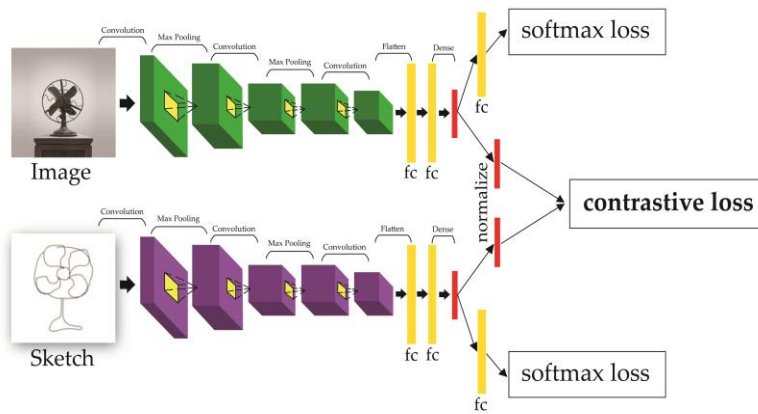


Fig. 3 Training the shared layers

Equation 9 shows the model for the two domains together:

$$\arg \min_{\theta_c} \sum_i \mathcal{L}_E(F^S(X_i^S)) + \sum_i \mathcal{L}_E(F^I(X_i^I)) + \alpha \sum_i \mathcal{L}_C(Y_i, X_i^S, X_i^I) + \lambda \mathcal{L}_R(\theta_C)$$
 (9)
 In which α is the weight of the regression term. As [10] suggests, $\alpha = 2.0$ in all experiments.

$$\arg \min_{\theta_S, \theta_I, \theta_C} \sum_i \mathcal{L}_E(F^S(X_i^S)) + \sum_i \mathcal{L}_E(F^I(X_{i+}^I)) + \sum_i \mathcal{L}_E(F^I(X_{i-}^I)) + \alpha \sum_i \mathcal{L}_T(X_i^S, X_{i+}^I, X_{i-}^I) + \lambda \mathcal{L}_R(\theta_S, \theta_I, \theta_C)$$
 (10)
 - Step 4

- Step 3

In this step, at the beginning of training, two loss functions are applied equally, and then the weight of the triple loss is increased ($\alpha = 2.0$). Figure 3 and Equation 10 display the learning regression in this step.

In this step, the model is modified further by repeating Step 3 on another dataset (figure 4). This training method allows shared and unshared layers to be trained independently in separate steps. In this method, the possibility of partial sharing across the branches is provided, which further reduces overfitting due to the significant reduction of training parameters. At the same time, learning flexibility is maintained for each domain.

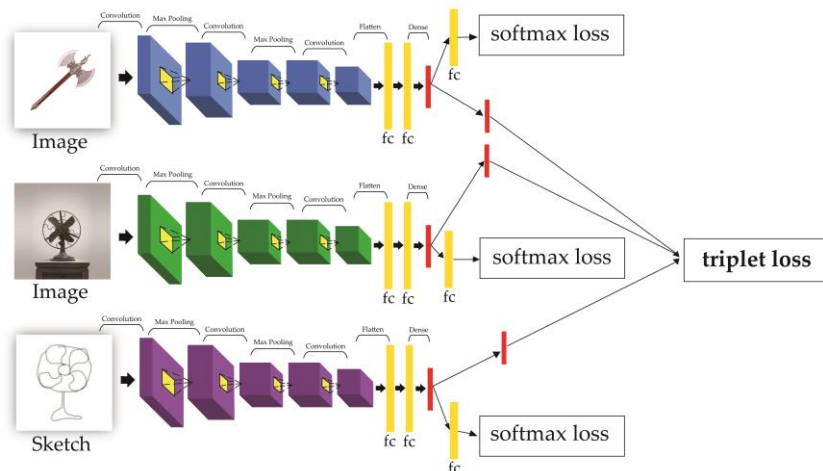


Fig. 4 Training the whole triplet network

It appears that triple and contrast loss functions are important in the training process, but they are not enough to adjust the

training. Therefore, the soft-max loss function was also used in all training steps. Past research has also shown that the

soft-max loss function plays an important role in the convergence of training [10].

3-3- Data Augmentation

Data augmentation is essential in preventing overfitting, especially when the training data is limited. In the proposed method, the following procedures were used to increase the data.

1. A random cut with a dimension of 225×225 as input for Sketch-A-Net network, 227×227 for Alex-net network, and 224×224 for VGG and Inception networks.
2. A random rotation in the range of [-5,5] degrees;
3. A random scaling in the range of [0.9 – 1.1];
4. A random horizontal rotation;
5. The method used only for sketches is called line ranking [10].

This method, is applicable for sketches with at least ten lines. The lines of the sketch are divided into four equal groups based on their importance so that the lines of the first group are the primary lines (the most important lines that related to the more coarse structure of the object) this group of lines is always kept, and the lines of the following groups decrease in importance each time. When one of the groups (except group one) is removed, a new sketch image is obtained every time [10].

4- Exprimental Results

The proposed multi-step training process was tested on several architectures of convolutional networks with sketch and image input. The impact of data augmentation operations on the training process was also evaluated.

4-1- Evaluation Ceriteria

4.1.1 Precision

Precision is one of the most common evaluation criteria used in classification problems. It is based on the ratio of the correctly classified samples to the total number of identified samples (samples that are incorrectly and correctly classified) [31]. The formula for calculating precision is as follows.

$$Precision = \frac{TP}{TP+FP} \times 100 \quad (11)$$

Where TP shows the correctly identified samples and FP shows the misidentified samples.

4.1.2 Recall

The recall is a measure obtained from the ratio of correctly classified samples to the sum of samples that are correctly identified and samples that are incorrectly rejected [32]. It is expressed as the below formula.

$$Recall = \frac{TP}{TP+FN} \quad (12)$$

4.1.3 Mean Average Precision

Average precision is calculated as the weighted mean of precisions at each threshold. The weight is the increase in recall from the prior threshold. The mean average precision is the average of AP of each class [33].

$$AP = \frac{1}{N} \sum_r P_{interp}(r) \quad (13)$$

Where $P_{interp}(r)$ is precision in point recall (r), and MAP is the average AP in each dataset class.

4.1.4 Kendall's Correlation Coefficient (τ_b)

Let $(x_1, y_1), \dots, (x_n, y_n)$ be a set of observations of the joint random variables X and Y, such that all the values of (x_i) and (y_i) are unique (ties are neglected for simplicity). Any pair of observations (x_i, y_i) and (x_j, y_j) , where $i < j$, are considered concordant if the sorting order of (x_i, x_j) and (y_i, y_j) agrees. That is if either both $x_i > x_j$ and $y_i > y_j$ or both $x_i < x_j$ and $y_i < y_j$ are true. Otherwise, they are discordant [10].

The Kendall τ coefficient is defined as:

$$\tau = \frac{(\text{number of concordant pairs}) - (\text{number of discordant pairs})}{\text{number of pairs}} = \frac{1 - \frac{2(\text{number of discordant pairs})}{\binom{n}{2}}}{\binom{n}{2}} \quad (14)$$

In which $\binom{n}{2} = \frac{n(n-1)}{2}$ is the binomial coefficient for the number of ways to choose two from n items.

4-2- Datasets

The proposed networks were evaluated using three datasets.

1) Tu-Berlin:

It is one of the most famous datasets in sketch-based image retrieval and includes 250 classes with 80 images in each, providing a total of 20,000 PNG images of hand-drawn sketches with a size of 128×128 (Figure 6(a)) [34, 35]. This dataset was used for training and testing the first three training steps.

2) Quick-Draw: This dataset has highly simple sketches. It contains 330,000 sketches and 204,000 images with a size of 256×256, divided into 110 classes (Figure 6(b)). It was used to adjust and modify the training model in the fourth step.

3) Sketchy: It is a large dataset of sketches and original images. It contains 75471 hand-drawn images with 125 classes. Of these, 100 classes are shared with the Tu-Berlin dataset, and 25 classes are new

(Figure 6(c)) [36]. This dataset was used to evaluate the proposed model.

Since the Tu-Berlin dataset includes only sketches, Internet databases such as Creative Commons [10] and older datasets such as Flickr-15 [37] or Google search engine were also searched to obtain the original images.

4-3- Training and Testing

A total of 25% of the Tu-Berlin images were selected randomly as the training set, and the remaining 75% were

used as the test set. For simplicity, Sketch-A-Net architecture was used for the sketch branch, and Alex-Net architecture for the image branch. Slight changes were made in the Sketch-A-Net architecture to share the weights between the two networks in such a way that layers 6-7 were taken from the Alex-net network, and layers 4-5 were modified as a combination of the two networks. The sketch branch was trained from the beginning, while the image branch was trained using the pre-trained weights from ImageNet. Figure 5 shows the results of these steps for the proposed multi-step training process.

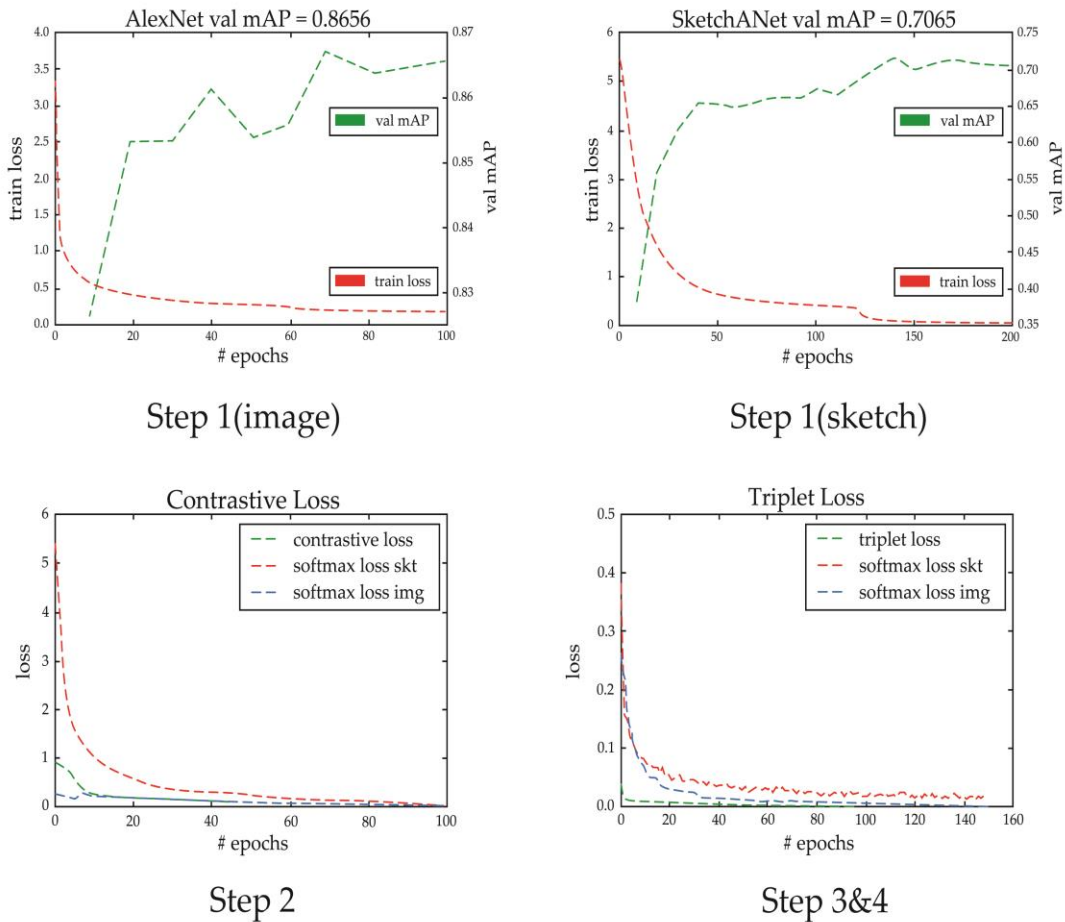


Fig. 5 4 step training of the Sketch-A-Net –and Alex-Net model










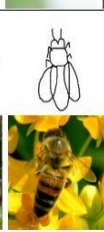


lable dataset	Lion	bee	airplane	sheep
Tu-Berlin				
Quick-Draw				
Sketchy				

Figure 6. Examples of the three datasets used in the multi-steps training SBIR

4-4- Testing Different Architectures

Four different examples of convolution-based network architectures were tested for the sketch and image branches. Different sharing layers were applied for each possible combination according to their architecture and network structure.

The investigations, showed that partial sharing always worked better than full sharing or no sharing at all. However, the layer of each network with the best performance in sharing could only be determined by testing. For example, for Alex-Net - Alex-Net mode, the best performance was achieved when Conv 5 layer was shared. In AlexNet-VGG16, the best performance sharing belonged to sharing the layer FC 7, and in Sketch-A-Net – Alex-Net, sharing layer FC-6 sharing achieved the best performance. In VGG 16-VGG 16, sharing block 5 performed better, and in Inception V1-Inception V1, sharing incept.4e achieved better performance. Subsequently, all possible permutations and sharing were tested to determine the optimal performance of the reviewed architectures. Figure7 shows the results of this review. As the Sketch-A-Net architecture can only be applied to the sketch-edge map mode and does not work on natural images. Therefore, this architecture was not used for the image branches except for one where the images were turned into edge maps.

As the diagram results show, the sketch branch architecture should not be more complicated than the image branch architecture. As can be seen, the designs of VGG16-AlexNet, Inception V1-AlexNet, and Inception V1-VGG16 are better

than their counterparts. Also, if Inception V1 architecture is selected for the image branch, Sketch-A-Net would be more suitable for the sketch branch than Alex-Net or VGG-16, even though it has a simpler architecture.

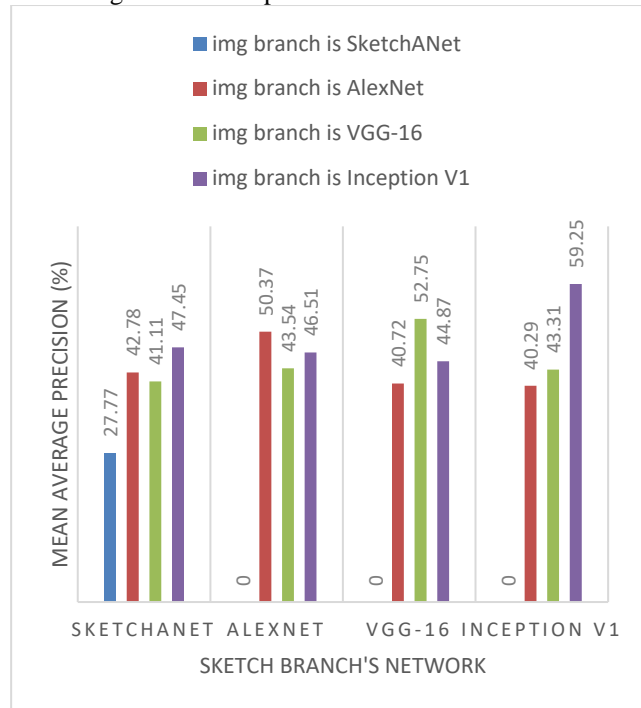


Figure 7. The best performance of different combinations of networks on Sketchy dataset

It can also be seen that using the same architecture for sketch and image branches leads to better performance. Subsequently, the best performance belongs to the design of Inception V1-Inception V1. This architecture was applied to the Sketchy dataset, and the increased output in the range of 64 to 1024 was examined. It was observed that as the dimensions increased, the MAP improved continuously. However, this also led to an increase in the retrieval speed. Therefore, the MAP evaluation criteria and retrieval speed were balanced by selecting a dimension of 256 with a 56.32 map and a recovery time of 6.2 ms for the final model.

4-5- Evaluation the Final Model

The proposed model, using Inception V1-Inception V1, Inception e4 block sharing, and the output dimension of 256 on the Sketchy dataset, was compared with other works. Table 2 shows the comparison of the proposed multistep method with several other research based on the MAP criteria.

Table 2. SBIR comparison based on MAP criteria

<i>method</i>	<i>Dim.</i>	<i>mAP (%)</i>
Siamese with contrastive loss[23]	64	19.54
Rst-SP-SHELO[31]	3060	20.05
Triplet sketch-edgemap[30]	100	24.45
Query-adaptive re-ranking CNN[21]	5120	32.30
Sketchy triplet[29]	1024	35.91
proposed Step 2	256	42.12
sketret [38]	256	43.70
proposed Step 3	256	48.53
cross modal (binary) [39]	64	50.60
cross modal [39]	64	52.30
SBTKNet[40]	512	55.30
hybrid cnn (without shape feature)[41]	64	55.30
proposed Step 4	256	56.48

The Siamese method (Table 2) uses contrastive loss, and introduces. A novel convolutional neural network for SBIR based on the Siamese network [23]. This method primarily draws output feature vectors for input sketch-image pairs with similar labels closer and pushes irrelevant pairs away. This is achieved by jointly tuning two convolutional neural networks which linked by one loss function. As can be seen, the results of this method are lower than all the presented research.

Another method is the Rst-SP-SHELO (Table 2). This method includes RST-SHELO and improved version of SHELO (Soft Histogram of Edge Local Orientations), which is an advanced, efficient method for describing sketches. In this research, the sketch token approach is used to detect image contours utilizing mid-level features. The square root normalization is used for a better normalization of SHELO and improved performance of the retrieval system. The result of this research is marginally better than the Siamese method with contrastive loss but is yet to be desirable.

In the triplet sketch-edge map method [30], convolutional neural networks and triplet loss are used. The SBIR problem is proposed as a cross-domain modeling problem where a depiction invariant embedding of sketch and photo data is learned by regression over a Siamese CNN architecture with half-shared weights and modified triplet loss function. The results of this method are better than the previous two methods but are still insufficient.

Another method shown in Table 2 is the Query-adaptive re-ranking CNN, which uses the localization technique. It also uses the Sketch-A-Net architecture to locate the candidate object proposals, exploit appearance information to resolve the ambiguities in object proposals and refine the search results. In this research, adaptive search is formulated as a subgraph selection problem and solved by the maximum flow algorithm. The results of this method are better than the previous ones (approx. 32.30).

The Sketchy triplet method is used in [29]. This method trains the Sketchy dataset by cross-domain convolutional networks that embed sketches and photos in a common feature space. The results are similar to that of the GN-Triplet network (Google-Net) with triplet loss.

Another example is the SKETRET, which is a ZS-SBIR retrieval method. In this research, a new framework is introduced, which adapts the bi-level domain of sketch and image features using adversarial learning. This framework alleviates the mentioned problems by providing modality-independent features and a class-discriminative latent space. This research achieves slightly better results than the proposed method in the second step.

Binary and non-binary cross-modal methods [39] also involve a ZS-SBIR problem. The study [39] proposes a novel progressive cross-modal semantic network, which first, explicitly aligns the sketch and image features to semantic features and then projects the aligned features to a common space for subsequent retrieval. Cross-reconstruction loss functions are often used to improve the alignment features, and multi-modal Euclidean loss is used for the similarity between the image-sketch pair retrieval features. The results for the binary and non-binary modes (Table 2) are higher than the proposed method in the third step.

SBTK-Net and hybrid CNN (without shape feature) methods achieve similar results (Table 2). In the SBTK-Net method, a simple and efficient framework is proposed that does not require large computational training resources. In the training and inference steps, only one CNN has been used. A pre-trained Image Net CNN (i.e., Res-Net 50) has been set with three learning objectives: Domain balanced quadruplet loss for learning distinctive features; semantic classification loss to preserve the learned semantic knowledge; semantic knowledge preservation loss to reduce the computational cost and increase the accuracy of the process. In the hybrid CNN method (without the shape feature), sketch recognition supposedly benefits from learning the appearance and shape representation. Therefore, a new architecture called hybrid CNN is proposed, that consists of A-NET and S-NET, describing the appearance and shape information, respectively.

As Table 2 shows, the proposed method of the present study achieves higher results after finishing all four steps than other methods.

Table 3 compares the performance of the proposed multi-step training system with other studies based on the percentage of precision criterion.

Table 3. SBIR comparison based on the precision criterion

<i>method</i>	<i>precision (%)</i>
Sketchy triplet[29]	53.42
cross modal (binary)[39]	61.50
cross modal[39]	61.60
proposed Step 2	63.21
proposed Step 3	69.35
fine-grained sbir[42]	78.02
semi supervised learning[16]	76.22
proposed Step 3	78.36

The results show that the Sketchy triplet and binary and non-binary cross-modal methods have a lower precision than the proposed method. The fine-grained SBIR method in [42] investigates the FG-SBIR problem. The introduced [42], FG-SBIR framework [42] starts retrieving as soon as the user starts drawing. Also, a mutual retrieval framework based on reinforcement learning is developed that directly optimizes the rank of the ground-truth photo over a complete sketch drawing episode. In addition, in the semi-supervised learning method, (FG-SBIR), a novel semi-supervised framework for cross-modal retrieval has been introduced, along with a discriminator-guided mechanism to guide against unfaithful generation and a distillation loss-based regularizer to provide tolerance against noisy training samples. In this research, generation and retrieval are considered two conjugate problems, and a common learning method is devised for each module to benefit mutually. These two methods have acceptable precision, but the proposed method achieves better result. After completing the four steps.

Table 4 shows the performance of the proposed multi-step training system using Kendall's correlation coefficient (τ_b) [10].

Kendall's correlation coefficient is used in limited studies on SBIR, but it is a suitable evaluation criterion. As Table 3 shows, the proposed multi-step training method performs better in terms of Kendall's correlation coefficient criterion than methods such as Triplet sketch-edge map and Sketchy triplet.

Table 4. The comparison based on Kendall's correlation coefficient (τ_b)

<i>method</i>	<i>Dim.</i>	τ_b
Triplet sketch-edgemap[30]	100	0.22
proposed Step 2	256	0.33
proposed Step 3	256	0.36
Sketchy triplet[29]	1024	0.37
proposed Step 4	256	0.48

In this article, we investigated the performance of four CNN network architectures and evaluated all possible permutations for the image branch and sketch in order to find the best combination of the network as well as the appropriate loss function with it, in order to optimize and increase the accuracy of retrieve. Our simultaneous attention to the network architecture, different methods of data augmentation and its impact on the training process and finding the appropriate loss function with the help of training weighting for each network combination has made this research unique. On the other hand, we have tried to use

datasets that includes different image styles due to the breadth and diversity of the subject, so that we could investigate and cover the challenges related to the dataset.

5- Conclusions

This paper proposed a hybrid convolutional neural network that uses dual and triple architectures for sketch-based image learning and retrieval. Various experiments and examinations of different convolutional neural networks (e.g., Sketch-A-Net, Alex-Net, VGG-16 and Inception V1), determined the best network architecture combination model for the proposed retrieval system. Regression functions were used in the deep neural network structure to improve system performance. Different layers were tested for weight sharing, and investigations and methods suggestions were carried out for preprocessing the training data. Various Loss functions were used for better convergence of the retrieval system. Three large, well-known datasets (Sketchy, TU-Berlin and Quick-Draw) were used in the training, testing, and evaluation process. Lastly, the final model was examined based on three evaluation criteria: MAP=56.48%, Precision=78.36%, and $\tau_b=0.48$. The entire training process of the proposed model was carried out on Pytorch platform. Further research on this topic could continue by exploring multi-domain learning, for example sketch-photo 3D models mapping or multi-style artwork retrieval. Recently, deep convolutional generative adversarial networks (DC-GANs) have shown great potential for sketch-based issues and so might offer an interesting alternative to SBIR for sketch-photo matching. Currently DC-GANs suffer limitations in variety of object classes that can be explored when trained.

References

- [1] D. Birari, D. Hiran, and V. Narawade, "Survey on Sketch Based Image and Data Retrieval", in ICCCE 2019 Springer, 2020, pp. 285-290.
- [2] Y. Li, and W. Li, "A survey of sketch-based image retrieval", Machine Vision and Applications, Vol. 29, No. 7, 2018, pp. 1083-1100.
- [3] S. Mohammadzadeh, and H. Farsi, "Image retrieval using color-texture features extracted from Gabor-Walsh wavelet pyramid", Journal of Information Systems and Telecommunication, Vol. 2, No. 1, 2014, pp. 31-40.
- [4] L. Liu, F. Shen, Y. Shen, X. Liu, and L. Shao, "Deep sketch hashing: Fast free-hand sketch-based image retrieval", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 2862-2871.
- [5] Z. Hossein-Nejad, H. Agahi, and A. Mahmoodzadeh, "Remote Sensing Image Registration based on a Geometrical Model Matching", Journal of Information Systems and Telecommunication (JIST), Vol. 5, No. 36, 2021, pp. 41.
- [6] J. Zhang, F. Shen, L. Liu, F. Zhu, M. Yu, L. Shao, H. T. Shen, and L. Van Gool, "Generative domain-migration hashing for sketch-to-image retrieval", in Proceedings of the European conference on computer vision (ECCV), 2018, pp. 297-314.
- [7] A. Sain, A. K. Bhunia, Y. Yang, T. Xiang, and Y. Z. Song, "Style me up: Towards style-agnostic sketch-based image

- retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 8504-8513.
- [8] Z. Dorrani, H. Farsi, and S. Mohamadzadeh, "Image edge detection with fuzzy ant colony optimization algorithm", *International Journal of Engineering*, Vol. 33, No. 12, 2020, pp. 2464-2470.
- [9] E. Askari, and S. Motamed, "Computational Model for Image Processing in the Minds of People with Visual Agnosia using Fuzzy Cognitive Map", *Journal of Information Systems and Telecommunication (JIST)*, Vol. 2, No. 42, 2023, pp. 102.
- [10] T. Bui, L. Ribeiro, M. Ponti and J. Collomosse, "Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression", *Computers & Graphics*, 2018, Vol. 71, pp. 77-87.
- [11] H. Farsi, and S. Mohamadzadeh, "Combining Hadamard matrix, discrete wavelet transform and DCT features based on PCA and KNN for image retrieval", *Majlesi Journal of Electrical Engineering*, Vol. 7, No. 1, 2013, pp. 9-15.
- [12] A. Del Bimbo, and P. Pala, "Visual image retrieval by elastic matching of user sketches", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 19, No. 2, 1997, pp. 121-132.
- [13] Y. Chans, Z. Lie, D. P. Lopresti, and S. Y. Kung, "Feature-based approach for image retrieval by sketch", in *Multimedia Storage and Archiving Systems II*, 1997, Vol. 3229, pp. 220-231.
- [14] R. K. Rajendran, and S. F. Chang, "Image retrieval with sketches and compositions", in *2000 IEEE International Conference on Multimedia and Expo. ICME 2000. Proceedings. Latest Advances in the Fast Changing World of Multimedia (Cat. No. 00TH8532)*, 2000, Vol. 2, pp. 717-720.
- [15] M. Rezaei, and M. Rezaei, "Foreground-Back ground Segmentation using K-Means Clustering Algorithm and Support Vector Machin", *Journal of Information Systems and Telecommunication (JIST)*, Vol. 1, No. 41, 2023, pp. 65.
- [16] A. K. Bhunia, P. N. Chowdhury, A. Sain, Y. Yang, T. Xiang, and Y. Z. Song, "More photos are all you need: Semi-supervised learning for fine-grained sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 4247-4256.
- [17] S. Dey, P. Riba, A. Dutta, J. Lladós, and Y. Z. Song, "Doodle to search: Practical zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 2179-2188.
- [18] A. Gheitasi, H. Farsi, and S. Mohamadzadeh, "Estimation of hand skeletal postures by using deep convolutional neural networks", *International Journal of Engineering*, Vol. 33, No. 4, 2020, pp. 552-559.
- [19] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, "Going deeper with convolutions", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1-9.
- [20] Q. Yu, F. Liu, Y. Z. Song, T. Xiang, T. M. Hospedales, and C. C. Loy, "Sketch me that shoe", in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 799-807.
- [21] S. D. Bhattacharjee, J. Yuan, W. Hong, and X. Ruan, "Query adaptive instance search using object sketches", in Proceedings of the 24th ACM international conference on Multimedia, 2016, pp. 1306-1315.
- [22] F. Wang, L. Kang, and Y. Li, "Sketch-based 3d shape retrieval using convolutional neural networks", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 1875-1883.
- [23] Y. Qi, Y. Z. Song, H. Zhang, and J. Liu, "Sketch-based image retrieval via siamese convolutional neural network", in 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 2460-2464.
- [24] A. Fuentes, and J. M. Saavedra, "Sketch-qnet: A quadruplet convnet for color sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2134-2141.
- [25] K. R. V. Vakili_Zare, and H. Rezaei, "K_Nearest Neighbor Classification Using Data and Deep Neural Networks", in 3rd International Conference on Soft Computing, 2019, pp. 1034-1040.
- [26] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering", in Proceedings of the IEEE conference on computer vision and pattern recognition, 2015, pp. 815-823.
- [27] X. Wang, and A. Gupta, "Unsupervised learning of visual representations using videos", in Proceedings of the IEEE international conference on computer vision, 2015, pp. 2794-2802.
- [28] A. Gordo, J. Almazan, J. Revaud, and D. Larlus, "Deep image retrieval: Learning global representations for image search", in European conference on computer vision, 2016, pp. 241-257.
- [29] P. Sangkloy, N. Burnell, C. Ham and J. Hays, "The sketchy database: learning to retrieve badly drawn bunnies", *ACM Transactions on Graphics (TOG)*, Vol. 35, No. 4, 2016, pp. 1-12.
- [30] T. Bui, L. Ribeiro, M. Ponti, and J. Collomosse, "Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network", *Computer Vision and Image Understanding*, Vol. 164, 2017, pp. 27-37.
- [31] J. M. Saavedra, "Rst-shelo: sketch-based image retrieval using sketch tokens and square root normalization", *Multimedia Tools and Applications*, Vol. 76, No. 1, 2017, pp. 931-951.
- [32] C. Bai, J. Chen, Q. Ma, P. Hao, and S. Chen, "Cross-domain representation learning by domain-migration generative adversarial network for sketch-based image retrieval", *Journal of Visual Communication and Image Representation*, Vol. 71, 2020, pp. 102835.
- [33] a. P. R. G. G. Rajput, "Sketch Based Image Retrieval in Large Databases Using Edge Features", *International Journal of Recent Technology and Engineering (IJRTE)*, Vol. 08, 2020, pp. 2277-3878.
- [34] A. Dutta, and Z. Akata, "Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 5089-5098.
- [35] Q. Liu, L. Xie, H. Wang, and A. L. Yuille, "Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval", in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 3662-3671.
- [36] T. Dutta, A. Singh, and S. Biswas, "Style guide: zero-shot sketch-based image retrieval using style-guided image generation", *IEEE Transactions on Multimedia*, Vol. 23, 2020, pp. 2833-2842.
- [37] P. Torres, and J. M. Saavedra, "Compact and effective representations for sketch-based image retrieval", in

- Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 2115-2123.
- [38] R. Chavhan, "Zero-Shot Sketch Based Image Retrieval", INDIAN INSTITUTE OF TECHNOLOGY BOMBAY, 2021.
- [39] C. Deng, X. Xu, H. Wang, M. Yang, and D. Tao, "Progressive cross-modal semantic network for zero-shot sketch-based image retrieval", *IEEE Transactions on Image Processing*, Vol. 29, 2020, pp. 8892-8902.
- [40] O. Tursun, S. Denman, S. Sridharan, E. Goan, and C. Fookes, "An efficient framework for zero-shot sketch-based image retrieval", *Pattern Recognition*, Vol. 126, 2022, pp. 108528.
- [41] X. Zhang, Y. Huang, Q. Zou, Y. Pei, R. Zhang, and S. Wang, "A hybrid convolutional neural network for sketch recognition", *Pattern Recognition Letters*, Vol. 130, 2020, pp. 73-82.
- [42] A. K. Bhunia, Y. Yang, T. M. Hospedales, T. Xiang, and, Y. Z. Song, "Sketch less for more: On-the-fly fine-grained sketch-based image retrieval", in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 9779-9788.

Enhancing Speaker Identification System Based on MFCC Feature Extraction and Gated Recurrent Unit Network

Mojtaba Sharif Noughabi¹, Seyyed Mohammad Razavi^{1*}, Mehran Taghipour-Gorjikolaie¹

¹. Department of Electrical and Computer Engineering, University of Birjand, Birjand, Iran

Received: 25 Oct 2024/ Revised: 04 Dec 2024/ Accepted: 28 Dec 2024

Abstract

One of the biometric detection methods is to identify people based on speech signals. The implementation of a speaker identification (SI) system can be done in many different ways, and recently, many researchers have been focusing on using deep neural networks. One of the types of deep neural networks is recurrent neural networks, where memory and recurrent parts are handled by layers such as LSTM or Gated Recurrent Unit (GRU). In this paper, we propose a new structure as a classifier in the speaker identification system, which significantly improves the recognition rate by combining a convolutional neural network with two layers of GRU (CNN+ GRU). MFCC coefficients that have been extracted as cell arrays from each period of Pt speech will be used as sequence vectors for the input of proposed classifier. The performance of the SI system has improved in comparison to basic methods according to experiments conducted on two databases, LibriSpeech and VoxCeleb1. When Pt is longer, the system performs better, so that on the LibriSpeech database with 251 speakers, recognition accuracy is equal to 92.94% for Pt=1s, and it rises to 99.92% for Pt=9s. The proposed CNN+GRU classifier has a low sensitivity to specific genders, which can be said to be almost zero.

Keywords: Speaker Identification; Gated Recurrent Unit Network (GRU); Convolutional Neural Network (CNN); MFCC.

1- Introduction

One of the topics of interest in various research from the past is the use of biometric features, such as face image, eyes iris, fingerprints, and voice, to recognize people. Speech biometrics can be given more attention since they don't require special equipment and can be obtained remotely through telephone lines. Voice can also aid in identifying the speaker's emotions, gender, language, and health status, in addition to conveying their identity. Our focus in this article is speaker recognition through speech signals. Speaker recognition is divided into two general subcategories: speaker identification and verification. In the identification phase after receiving the speech signal by the system, his identity is recognized, but in the verification phase, a person claims to be a specific identity using a speech signal, and the system responds to reject or validate their claim.

In these two systems, three basic stages of feature extraction, modeling, and decision-making can be used for both text-independent and text-dependent purposes [1, 2].

Mel Frequency Cepstral Coefficients (MFCC) is commonly used as a practical and important feature in experiments during the feature extraction stage. MFCC is the basis for features like MFCCT [3], SHMFCC [4], which are used in speaker recognition systems and will be explained in more detail in the next section. In addition to speaker recognition, the MFCC feature is also used in other applications such as speech emotion recognition [5]. Other features such as Power Normalized Cepstral Coefficients (PNCC) [6] and Linear Predictive Cepstral Coefficients (LPCC) [7] should also be employed during this phase. In the modeling stage, older methods such as Gaussian mixture model (GMM) and identity vector (i-vector) are used as basic methods, while Models based on deep neural networks (DNN), convolutional neural networks (CNN), and recurrent neural networks (RNN) are also used. Local connectivity and weight sharing in CNN reduce the number of parameters to be learned [8]. In addition to speaker recognition, the use of convolutional networks has been considered in various

✉ Corresponding Author
smrazavi@birjand.ac.ir

speech tasks such as speech recognition and infant cry classification [9] and image processing tasks such as person reidentification [10] and facial expression recognition [11]. Vector quantization, cosine distance, support vector machine (SVM) or neural networks are some of the methods that can be used to perform the decision-making process. It must be pointed out that in some articles, Mel Spectrogram images or the raw speech signal are utilized for convolutional neural network input instead of feature extraction from the speech signal [12-14]. Deep neural networks are employed in three different modes in speaker recognition approaches. In the initial scenario, the network extracts features, while in the second scenario, it classifies them. In the third scenario, both feature extraction and classification are done by the deep network [15]. The second scenario has been used in this article and by employing a recurrent deep neural network, we have observed a significant improvement in system performance compared to other methods.

This paper is broken up into five parts. The subject under study was introduced in the first part, and in the second part, an overview of works relevant to the article will be provided. The method used will be explained in the third part. In the fourth part, the experiments and their results will be reviewed, and in the fifth part, the conclusions and suggestions will be presented.

2- Related Works

In this section, we will briefly review some of the research related to our work.

As previously mentioned, there are different approaches to implementing speaker recognition systems using deep neural networks, one of them was using the DNN in the classification stage. MFCC features are obtained from speech with specific lengths in [4], and a feature matrix is produced as a result. To increase the dataset, the MFCC feature vectors of every matrix are randomly arranged in terms of their placement in the matrix, without altering the vectors themselves and Form a new feature matrix together again. The name for this new feature is SHMFCC. These feature matrices are fed into a deep neural network that has five layers, consisting of one input layer, three hidden layers, and one output layer. The hidden layer is comprised of 300 neurons, a Batch Normalization layer, and a dropout layer with a probability of 0.35%. Improvement in system performance was observed during tests on two databases, LibriSpeech and VoxCeleb1.

paper [3] takes into account multiple feature vectors after extracting the MFCC feature from speeches with a certain length instead of using these vectors directly as feature vectors. By gathering 12 statistical features from these multiple vectors, a new feature vector called MFCCT was created. The new feature vector is put into a deep neural

network that has 7 layers, one of which is input, five hidden layers with 200 neurons in each layer, and an output layer at the end. The proposed method has achieved relatively good results by running it on the LibriSpeech database.

Reference [16] Focuses on the use of the MFCC feature as well as other features that are commonly derived from MFCC. In the classification phase, SVM was utilized, and in the testing phase, an accuracy rate of about 90% was achieved using the ELSDSR database with only 22 speakers. Ashar et al in [17] Achieved an accuracy of 80% for the data set with 60-speaker by extracting the 39 MFCC feature vectors from speech frames. A deep neural network with one input layer, several hidden layers, and one output layer has been used for classification. In [18], different methods are used to modify the MFCC and PNCC features, and the resulting feature vector is provided to the ELM classifier. The proposed methods for TIMIT and SITW databases achieved a maximum accuracy of 97.52 and 97.66, respectively.

Reasearchers in [19] Has achieved an accuracy of 87.65 with artificial neural networks, 89.96 with recurrent neural networks, and 99.23 with convolutional neural networks. TIMIT data has used to extract the MFCC feature of speech frames for 100 speakers. Speaker data is used to extract 12 MFCC features for each frame in [20] by considering different shapes for framing windows. The database that was utilized has 800 speeches from 16 speakers, which were prepared by the article's authors. For classification, a deep neural network with 6 hidden layers is employed. 94.37% is the average for best performance when using HANNING window.

The implementation of [21] involves the use of an open set speaker recognition system. The extracted feature is the MFCC, and the GMM-UBM model is employed during the classification process. The THYUG-20 SRE databases and speakers from noise-free parts of the LibriSpeech database were used to implement the proposed system, and accuracy levels of between 73 and 86% were achieved. In [22], an attempt has been made to enhance the speaker identification system by using reverberation modeling and techniques to cancelable speakers that can be removed. In this study, features such as wavelet-domain, MFCC and features based on DCT were extracted from the speech signal and a neural network was utilized for classification. For the experiments, the speech data of 15 Arabic speakers, including 10 men and 5 women, was utilized. The proposed method's accuracy in noisy and noiseless conditions ranges between 35 and 100%.

By extracting MFCC and MSE (Multiband Spectral Entropy) features of speech and employing various classifiers, such as KNN and DNN, the highest accuracy for ELSDSR data with 22 speakers was achieved using research [23], which resulted in 93.99% accuracy for 22 speakers. Although tests were performed on 40 speakers from the LibriSpeech database, accuracy was less than

expected. The feature vector is formed when the MFCC feature and its derivatives, along with other features like the formant frequency, are extracted and combined in [24]. This feature vector was employed in the proposed LSTM and BLSTM classifiers, which displayed a 92.75% and 95.52% accuracy rate for the YOHO database with 138 speakers, respectively.

Extracting the MFCC feature from Audio-MNIST data with 60 speakers and 500 speeches per speaker, and then using various classifiers such as SVM, KNN, LR, Nave Bayes, and so on, was done in [25]. The proposed speaker recognition system has achieved the highest accuracy of 97.1% with the SVM classifier. By extracting features from the speech signal, such as MFCC, amplitude, energy, and others, [26] was able to achieve different results, and various classification methods such as MSVM, KNN, DNN, LSTM, and Hybrid LSTM were utilized to achieve them. The Hybrid LSTM classifier achieved a high efficiency of 92.65% for 100 speakers, including 50 women and 50 men from the LibriSpeech database.

By utilizing the MFCC feature and a deep convolutional neural network for classifying, the [27] was able to achieve the highest accuracy of 94% using 251 speakers and 3-seconds long speeches. Paper [11] Has inputted raw audio signals without extracting features, simply by detecting silence in speech and separating speech parts into two different neural networks named sincNET and sincGAN. A good accuracy between 85 and 99.27% was achieved after testing these methods on TIMIT and LibriSpeech data.

The extraction of different speech features such as MFCC, PLP and PLCC and the application of classification methods such as GMM-SVM and Ivectors-PLDA and their fusion using the sparse method have been done in [28]. Experiments on NIST 2004 data show better efficiency of the speaker authentication system using the sparse method. in [29] is designed a speaker recognition system by extracting the MFCC feature from speech frames and forming feature vectors from speech parts with different lengths, such as 1 second and 3 seconds and then applying various classification methods. LibriSpeech data was used to test this system and it achieved the highest accuracy of 99.31% within speeches with a length of 9 seconds. In [30], it is proposed to use Neurogram coefficients to enhance the speaker identification system's robustness. Neurogram is a 2-D time-frequency representation which was constructed by combining the neural responses (i.e., feature) from 25 AN (Auditory Nerve) fibers. The test results on the YOHO database show that the proposed method performs better than basic methods such as MFCC coefficients, especially in noisy conditions. GMM-UBM is the classification method employed.

3- Proposed Method

3-1- Feature Extraction

Our proposed methods in this article are primarily focused on classification, but some suggestions will be made for feature extraction as well. Our first task involves extracting MFCC coefficients from a speech with a specific length. Algorithm 1 is used to obtain the set of features that can be applied to the input of a recurrent deep neural network for classification purposes.

The extraction of features for each of the training, validation, and testing sections, as demonstrated in algorithm 1, results in a set of features that contain a cell array for each speech interval (Pt).

Each of these arrays is considered an input to the classifier. For example, if $P_t = 1$ sec, for each speech of this length, a cell array with dimensions of 13×66 is obtained. The number of MFCC coefficients in each frame is 13, and there are 66 frames in P_t 's length. The number of frames is determined by the length of the frame and the amount of overlap, which can be compared with the basic methods, they are regarded as being equal to 25 and 10 milliseconds, respectively, in this article.

This cell array is inputted as a sequence into the deep neural network, which will be explained in detail later.

Algorithm1. How to Extract Features of Speaker Utterance
Input: path to speaker utterances
Segment utterances random to train, validation and test, 70, 15,15 percentage respectively
Procedure: Get MFCC Features (*path*)
 $M \leftarrow$ total of class (Train or Validation or Test)
 $A2 = \{ \}$ (a cell array for save total features and at end contain features cell for each class)
 $J \leftarrow 1$ (counter for classes)
 While $J \leq M$
 $P_t \leftarrow$ Periods select of Utterance
 $N \leftarrow$ total utterance of class J
 $I \leftarrow 1$ (counter for utterances in each classes)
 $A1 = \{ \}$ (a cell array for save features that is empty each iteration)
 While $I \leq N$
 $A \leftarrow$ 13 MFCC features matrix from frames of utterance with P_t length
 $A1\{I\} = A$
 end
 $A2 = [A2, A1]$
 end

It is clear that as P_t becomes larger, the number of frames and thus the length of the cell array increases. As P_t increases, the feature extraction cycle may not include certain speeches in the database because of their short length to decrease the number of speeches that are deleted,

speeches with a length of more than 0.5Pt and less than 1Pt should be continued with the part of speech that belongs to the same class until they reach the length of Pt. Of course, this part is reversed and then added to the speech. The speech is added to the feature extraction cycle after doing this.

We chose and displayed one of the speeches used in the experiments to enhance our understanding of this proposed method. As depicted in Fig. 1, the speech that was selected has a length of 11 seconds. If Pt=3, we can extract three frames with a full length of 3 seconds from this speech. However, the final frame is 1 second shorter than 3 seconds, which means it will be 2 seconds long. As this frame is longer than 0.5Pt, we'll continue with a portion of the speech that's related to the same speaker, so that its length is as long as Pt and it can be extracted from that feature. This work improves the system's performance by 3% as demonstrated by the test results. This work's results will be displayed in the test results section with the 'AUGMENTED' symbol.

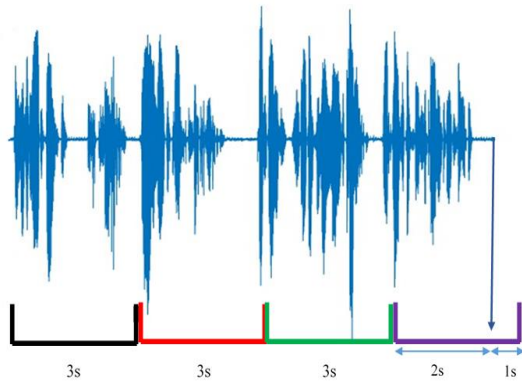


Fig. 1 How to frame speeches in experiments

The diagram in Fig. 2 illustrates the steps required to obtain the MFCC feature. Fig. 2 shows that there are eight steps to extract MFCC features from speech. These steps are explained in more detail below.

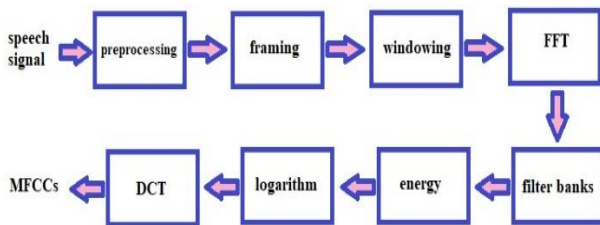


Fig. 2 Steps to calculate MFCCs

The initial step is preprocessing. In this step, a high-pass filter, also known as pre-emphasis, is applied to the speech signal to compensate for the amplitude at higher frequencies. The Eq. (1) represents this filter.

$$P(z) = 1 - az^{-1} \tag{1}$$

In the next step, the signal will be framed. The instability of the speech signal is the main reason for this action, which can be considered almost stationary because of the shortening of the speech signal in the frames. It's obvious that this action is taken to decrease the amount of input data and save time, while also analyzing the signal more closely. The frequency of the speech signal usually determines the number and length of frames. Sometimes, the signals are framed in such a way that the frames overlap with each other, and this overlap can reach up to 50%. To eliminate the discontinuity between the frames' borders, we multiply each frame in a window in the next step. The Hamming window obtained by Eq. (2) is used to calculate these coefficients.

$$w(n) = 0.54 - 0.46 * \cos\left(\frac{2k\pi}{N-1}\right) \quad k = 0,1, \dots, N - 1 \tag{2}$$

In Eq. (2), N is the length of the window, which is equal to the length of the frames. The discrete Fourier transform is performed on the windowed frames in the fourth step.

The human ear's auditory properties are the main inspiration for MFCCs. The function of the human ear is not based on physical understanding, but logarithmically and based on Eq. (3).

$$f_{mel} = 2595 \log\left(1 + \frac{f}{700}\right) \tag{3}$$

The frequency used in Equation 3 is f, while f_mel is the frequency that is converted from the linear domain to the Mel domain. The human ear's accuracy in understanding low frequencies is high, but it is low in understanding high frequencies, as shown in this equation. Mel Frequency Cepstral Coefficients are calculated using a set of filter banks to convert frequencies from Hertz scale to Mel. A triangular filter bank is the usual choice for this step. The bandwidth of triangular filters is greater at higher frequencies than at lower frequencies, which suggests that the human ear is less sensitive to frequency changes at higher frequencies than at lower frequencies. This filter bank is shown in Fig. 3.

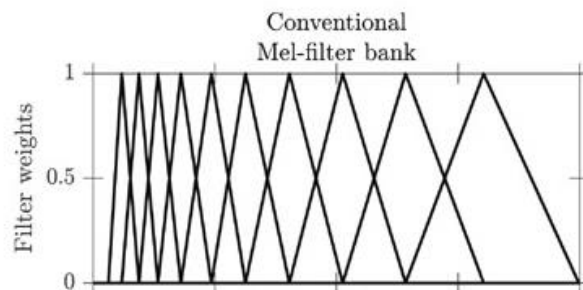


Fig. 3 Triangular filter bank [31]

After that, the energy of each of the filter banks is calculated. To decrease the numbers obtained from energy, the logarithm is employed with Eq. (4).

$$X'(m) = \log(X_1(m)) \tag{4}$$

Finally, we get the cosine transformation for the resulting coefficients by employing Eq. (5).

$$Ceps_{MFCC}(l) = \sum_{m=1}^M X'(m) \cdot \cos(l \frac{\pi}{m} \cdot (m - \frac{1}{2})) \tag{5}$$

The length of each frame is M and the filter bank number is l in this equation. Mel Frequency Cepstral Coefficients are obtained by using Eq. (5) and typically yield 13 or 14 coefficients for each frame. Of course, it should be noted that in [4], in order to achieve the desired results, approximately 60 MFCC coefficients have been extracted from each frame and To improve performance in some parts of the test, non-speech parts have been eliminated before (VAD), While using our method, we have obtained better results with the same 13 MFCC coefficients without performing VAD.

3-2- Classification

In some of the articles reviewed for this step, a deep neural network with multiple hidden layers has been utilized, like in [4], where the structure of Fig. 4 is utilized in the deep neural network.

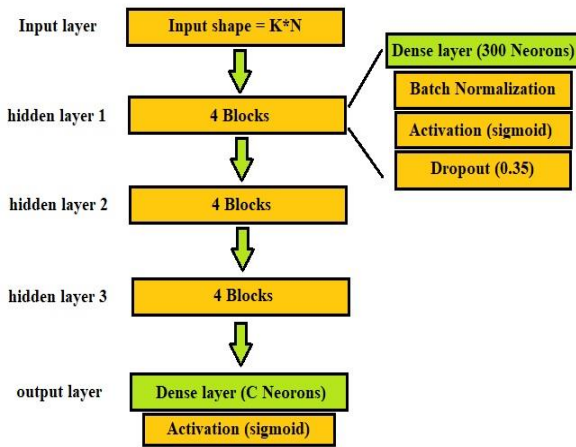


Fig. 4 The deep neural network used in [4]

The GMM-UBM model is one of the common methods used by some researches [21]. In [26], there are various methods for data classification, but the Hybrid LSTM classifier is the most efficient. The architecture of this classifier is depicted in. methods for data classification, but the Hybrid LSTM classifier is the most efficient. The architecture of this classifier is depicted in Fig. 5.

As stated in the related works section, [27] employs a CNN classifier. The proposed classification consists of 13 layers, but we choose not to display their details here. sincNET and

sincGAN are utilized in [14]. These two classifications are composed of several layers, which include convolutional layers, batch normalization, and activators. For more details, refer to the reference mentioned. After performing VAD, raw audio signals are inputted into these two networks. paper [29] has presented a number of approaches for classification, including 1D-CNN, 2D-CNN, LSTM, and CRNN. There are several convolutional layers in its CNN classifier, and at the end, there is a GAP layer that enhances detection accuracy.

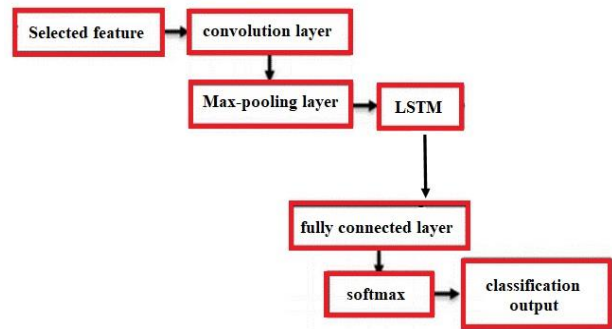


Fig. 5 Hybrid LSTM classifier architecture used in [26]

The objective of this article is to utilize a recurrent neural network that has a structure consisting of layers, as demonstrated in Fig. 5. The GRU architecture (Gated Recurrent Unit) was introduced in 2014 [32]. The purpose of this architecture is to address the shortcomings of traditional recurrent neural networks, such as gradient fading, and also to decrease the overhead of the LSTM architecture.

Deep learning models based on time series, such as Simple RNN, LSTM, and GRU, are appropriate for granting access based on previous access histories [33]. The problem of long dependency on RNN networks can be resolved with the use of GRU, a type of LSTM [34]. The module structure of GRU is repetitive and based on the attention mechanism [35], which is more straightforward than long and short-term memory because each recurrent neural network feature of the module is the same. Furthermore, unlike the LSTM with three gates, GRU has two gates: a reset gate and an update gate. The update gate is used to supervise the extent to which the knowledge of the previously hidden state is extended to the current state. The greater the value of the update gate, the more knowledge of the previous state is introduced. Therefore, if the reset gate is used to adjust the degree of knowledge transfer of the past state, the smaller the value of the reset gate, the more it will be transferred [36]. Due to its simpler structure and fewer parameters than the LSTM, the GRU neural network model can train faster and produce larger networks more easily [37].

Compared to LSTM, GRU has fewer hyperparameters and is less computationally intensive [38]. A GRU layer's internal structure is shown in Fig. 6. In this figure, X_t represents the input vector and h_t represents the state

memory variable at different moments. σ is the sigmoid activation function and \tanh is the tangent function. The structure of the proposed neural network is shown in Fig. 7. This figure displays that the input of the network is sequence-based. The technique for obtaining the feature set was explained in the previous section. In this set, k is the number of MFCC coefficients, which is considered equal to 13, and N is the number of frames in the desired Pt.

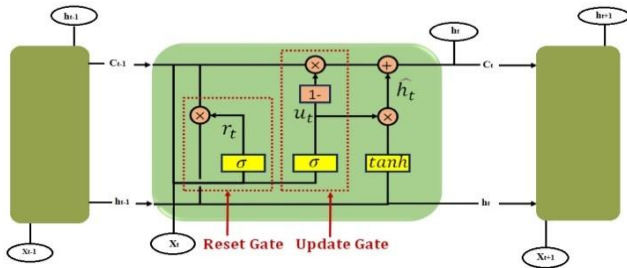


Fig. 6 A GRU layer's internal structure

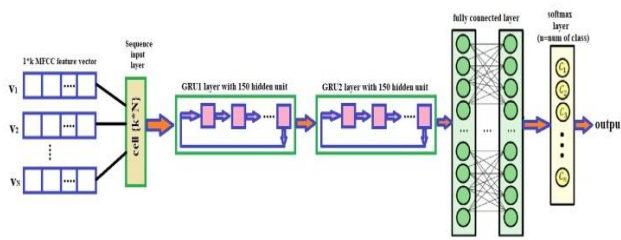


Fig. 7 structure of the proposed neural network (CNN+GRU)

First, the input speech to the system is examined and if the conditions are met, the augmented process is performed. Then the MFCC feature set is extracted from it. This sequential feature set is then fed into a GRU layer. The output of this layer is fed into another GRU layer. Each of these GRU layers has 150 hidden units. Each hidden unit has an internal state that holds information from previous inputs and uses it in the next process. The main task of the hidden unit in a recurrent network is to integrate new input information with the previous internal state. At each time t , the hidden layer receives new input information from the input layer and combines it with the previously maintained internal state.

This combination of information helps the hidden layer to recognize complex temporal patterns in sequential data. Using an LSTM layer instead of second GRU has a significant impact on the performance of the neural network, which is why we chose every double layer of GRU type.

After the GRU layers, a fully connected layer is placed to convert the features extracted from the hidden layers into an output vector. The output dimension of this layer is equal to the number of classes. finally, a Softmax activating layer is put on. The output of this layer is a probability distribution

and performs the final classification task. CNN+ GRU is the name we use for this method.

Initial learning rate is set to 0.01 and the adam optimizer is employed. MATLAB 2023 software and a single GPU platform were utilized for the implementation. The speaker identification system performs better with the proposed classifier, as evidenced by the results. This classifier is not sensitive to gender, and it will be mentioned in the results section.

4- Simulation Results

In this section, we evaluate the performance of the proposed methods by evaluating them on two different databases.

4-1- Database

The experiments employed databases from the relevant articles to ensure that the results were comparable. The LibriSpeech dataset is one of the datasets, taken from the LibriVox audio book collection and has about 1000 hours of speech that are sampled at 16 KHz. The train-clean-100 set is the subset of this database that we used, and it contains speech without any noise. Of the 251 speakers in this subset, 100 speakers, including 50 men and 50 women, were selected as part of the experiment. VoxCeleb1 is another database that has been utilized. The collection contains over 100,000 speeches belonging to 1,251 celebrity speakers, taken from videos posted on YouTube. However, the speeches are not completely clean. 100 speakers from this database with an equal proportion of men and women were chosen for the experiments.

In every experiment, 70% of the data set was utilized for training, 15% for validation, and 15% for testing.

4-2- Evaluation Criteria

Choosing the appropriate evaluation criteria is essential when checking the system's performance. Speaker recognition systems can be evaluated using various criteria. In speaker recognition systems, accuracy of performance (ACC) is one of the most common criteria, and speaker recognition systems that use deep neural networks are typically evaluated with this criterion. Equal error rate (EER), MinDCF, and ROC and DET curves are used in speaker recognition and verification systems to evaluate their performance. To compare the results of the articles that have used this criterion, we use the ACC value as an evaluation criterion for the speaker identification system designed in this paper.

4-3- Results

The databases used in this paper were described in Sections 4-2. To perform the tests, the speaker's speech is segmented according to the selected Pt. The augmentation process is also performed if necessary. We divide the specified segment into 25 ms frames with 10 ms overlap and extract 13 MFCC coefficients from each frame. For each segment of speech, a feature set with dimensions $13*N$ is obtained, where 13 is the number of MFCC coefficients and N is the number of frames of that segment of speech. This feature set is then fed into the proposed CNN+GRU network.

The LibriSpeech database was used for our initial experiment, which involved selecting Pt values of 1, 3, and 5 seconds. Table 1 shows the test data results.

Table 1: Comparing the proposed method's ACC% results and basic methods with the LibriSpeech database

Methods	Pt (s)		
	1	3	5
MFCCT+DNN [3] (VAD , Num. class=100)	52.9	78.4	83.8
MFCC+DNN [4] (NO VAD , Num. class=100)	93.2	94.1	94.7
MFCC+DNN [23] (NO VAD , Num. class=40)	88.78	---	---
MFCC+CNN+GRU (OURS) (NO VAD , Num. class=100)	95.77	99.38	99.76
MFCC+CNN+GRU (OURS) (AUGMENTED ,NO VAD , Num. class=100)	95.92	99.60	99.70

Table 1 shows that the proposed method, regardless of the Pt, provides a superior output compared to the basic methods in all three cases. An improvement of more than 26% has been made when compared to method [3] and more than 4% when compared to method [4]. The results are improved by implementing the AUGMENTED method.

The VoxCeleb1 database was utilized in the next experiment. Although this set is not clean and contains background speech, the system's performance is impacted, but the proposed method still performs better. The evaluation results for the test data are shown in Table 2.

Table 2: Comparing the proposed method's ACC% results and basic methods with the VoxCeleb1 database

Methods	Pt (s)		
	1	3	5
MFCCT+DNN [3] (VAD , Num. class=100)	52.9	78.4	83.8
MFCC+DNN [4] (NO VAD , Num. class=100)	93.2	94.1	94.7
MFCC+DNN [23] (NO VAD , Num. class=40)	88.78	---	---
MFCC+CNN+GRU (OURS) (NO VAD , Num. class=100)	95.77	99.38	99.76
MFCC+CNN+GRU (OURS) (AUGMENTED ,NO VAD , Num. class=100)	95.92	99.60	99.70

The results of Table 2 also show that the proposed method has better performance. For two modes of Pt = 3, 5 s, there was an average improvement of more than 39% was observed in the performance of this method compared to the method [3] and more than 11% when compared to method [4]. A relative improvement in the results has been achieved by using the AUGMENTED method, just like the previous experiment.

The third experiment utilized the total LibriSpeech-clean-100 database, which has 251 speakers, with 126 male and 125 female speakers. There have been no modifications to the features of the proposed CNN+ GRU neural network, and the feature that was extracted is MFCCs. The results for the proposed method and other studied methods are shown in Table 3.

Table 3: Comparing the proposed method's ACC% results and basic methods with the LibriSpeech database

Methods	Pt (s)		
	1	3	8 or 9
Fusion of features + Hybrid LSTM [26]	92.65	---	---
MFCCT+GMM-UBM (VAD) [21]	---	---	86 (8s)
MFCC+CNN (VAD) [27]	---	94	---
RAW signals + sincNET (VAD) [14]	---	98.86	---
RAW signals + sincGAN (VAD) [14]	---	98.94	---
MFCC + 1D-CNN (VAD) [27]	90.21	97.02	99.31 (9s)
MFCC+ A-LSTM (VAD) [27]	88.48	96.98	99.22 (9s)
MFCC+ CRNN (VAD) [27]	91.98	95.94	98.10 (9s)
MFCC+CNN+GRU (NO VAD) (OURS)	92.94	99.02	99.62 (8s) 99.92 (9s)

Table 3 illustrates that the proposed method still performs well despite the increase in speakers from 100 to 251. Regardless of the length of the speech, the proposed method has the best performance among the studied methods in all cases.

The graph in Fig. 8 is drawn to better display and compare the results obtained in Table 3.

The proposed classifier's sensitivity to the specific gender was tested in the fourth test. Identification in previous experiments was performed irrespective of the speaker's gender, which is demonstrated in the results of this section under the title of gender.

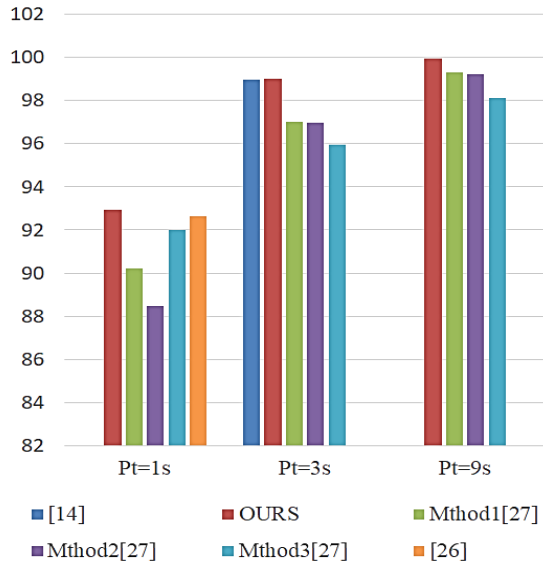


Fig. 8 A chart to compare the results of Table 3

The selected speeches from the LibriSpeech database are divided into male and female speakers in this part of the experiment and after extracting features, we insert them into the proposed classifier for identification. The results are compared in Fig. 9. The AUGMENTED method was not utilized to obtain these results.

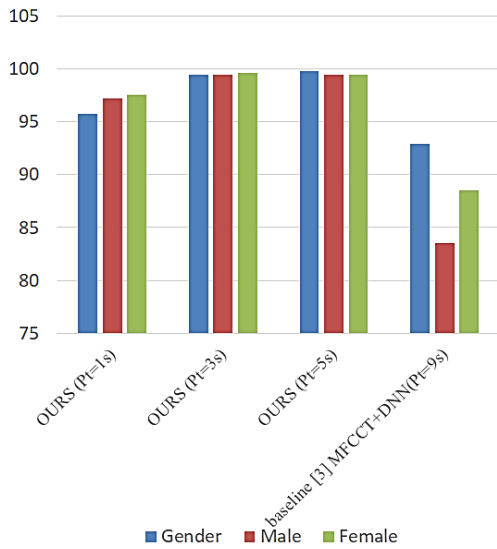


Fig. 9 The ACC% output of proposed classifier in three modes: gender, male, and female with the LibriSpeech database

Fig. 9 illustrates that the proposed classifier does not significantly decrease in performance when compared to specific genders. However, it can detect females and males better than genders in some cases.

The method used for speaker identification in this article, like many existing methods, has limitations, some of which

arise during implementation. For example, training the system requires a large amount of data, and receiving long speech from speakers may not be possible in some situations. Also, in real environments, there is a possibility of noise being added to speech, which reduces the efficiency of the system. Training time also creates limitations if it is long. Of course, in this article, by using GRU layers instead of LSTM in the proposed classifier, the training time has been significantly reduced. The training time of the proposed system with different methods and the accuracy obtained are shown in Table 4. In all methods, MiniBatchSize is 52.

Table 4: Comparison of training time in the proposed system with different databases and methods

Methode: MFCC+CNN+GRU Database: LibriSpeech (100 speaker)			
Pt(s)	epochs	Traning time (min)	Acc of training (%)
1	15	28	95.84
3	28	20	99.40
5	41	21	99.19
Methode: MFCC+CNN+GRU (Augmented) Database: LibriSpeech (100 speaker)			
Pt(s)	epochs	Traning time (min)	Acc of training (%)
1	16	32	95.82
3	27	23	99.63
5	41	29	99.81
Methode: MFCC+CNN+GRU Database: LibriSpeech (251 speaker)			
Pt(s)	epochs	Traning time (min)	Acc of training (%)
1	42	104	92.91
3	60	82	99.00
8	55	39	99.72
9	70	52	99.92
Methode: MFCC+CNN+GRU Database: VoxCeleb1 (100 speakers)			
Pt(s)	epochs	Traning time (min)	Acc of training (%)
1	32	30	71.89
3	70	27	88.00
5	80	20	89.14
Methode: MFCC+CNN+GRU (Augmented) Database: VoxCeleb1 (100 speakers)			
Pt(s)	epochs	Traning time (min)	Acc of training (%)
1	50	54	72.30
3	70	35	89.14

Table 4 shows that as Pt increases, the training time decreases proportionally to the number of epochs and the accuracy of the system on the training data increases. The training time is also not long.

5- Conclusions

To enhance the performance of the speaker identification system, a convolutional neural network utilizing GRU layers was proposed in this article. Since the input to the GRU section is a sequence, the speech in the database is split into equal parts based on the considered Pt. From each part, the feature vector set of MFCCs is extracted in the form of cell arrays and sent to the proposed neural network named CNN+GRU.

The proposed method's efficiency is shown in the implementations on two different databases and with varying numbers of speakers. The system's efficiency increases as the Pt length increases. In one case, with an increase of Pt from 1s to 3s, the recognition rate increases from 71.25% to 88.87%. Increasing the length of the speech through the proposed AUGMENTED method can improve system efficiency to some extent. The proposed method also displayed a low level of sensitivity towards specific gender. It can be inferred that using the GRU layer in CNN instead of LSTM enhances both the SI system's performance and calculation speed.

References

- [1] S. Hourri and J. Kharroubi, "A Novel Scoring Method Based on Distance Calculation for Similarity Measurement in Text-Independent Speaker Verification," *Procedia Computer Science*, vol. 148, pp. 256–265, 2019.
- [2] M. Chaiani, M. Bengherabi, S. A. Selouani and M. Boudraa, "Dysarthric speaker identification with constrained training durations," 2018 International Conference on Signal, Image, Vision and their Applications (SIVA), Guelma, Algeria, 2018, pp. 1-6.
- [3] R. Jahangir et al., "Text-Independent Speaker Identification Through Feature Fusion and Deep Neural Network," in *IEEE Access*, vol. 8, pp. 32187-32202, 2020.
- [4] M. Barhoush, A. Hallawa and A. Schmeink, "Robust Automatic Speaker Identification System Using Shuffled MFCC Features," 2021 IEEE International Conference on Machine Learning and Applied Network Technologies (ICMLANT), Soyapango, El Salvador, 2021, pp. 1-6.
- [5] S. Langari , H. Marvi, and M. Zahedi, "Efficient speech emotion recognition using modified feature extraction," *Informatics in Medicine Unlocked*, vol. 20, p. 100424, Jan. 2020
- [6] X. Liu, M. Sahidullah and T. Kinnunen, "Optimized Power Normalized Cepstral Coefficients Towards Robust Deep Speaker Verification," 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Cartagena, Colombia, 2021, pp. 185-190.
- [7] P. Sandhya, V. Spoorthy, S. G. Koolagudi and N. V. Sobhana, "Spectral Features for Emotional Speaker Recognition," 2020 Third International Conference on Advances in Electronics, Computers and Communications (ICAIECC), Bengaluru, India, 2020, pp. 1-6.
- [8] K. Aghajani and E. P. Afrakoti I., "Speech emotion recognition using Scalogram based deep structure," *International Journal of Engineering. Transactions B: Applications*, vol. 33, no. 2, Feb. 2020.
- [9] A. Abbaskhah, Hamed Sedighi, and Hossein Marvi, "Infant cry classification by MFCC feature extraction with MLP and CNN structures," *Biomedical Signal Processing and Control*, vol. 86, pp. 105261–105261, Sep. 2023.
- [10] A. Sezavar, H. Farsi, and S. Mohamadzadeh, "A New Model for Person Reidentification Using Deep CNN and Autoencoders," *Iranian Journal of Energy and Environment*, vol. 14, no. 4, pp. 314–320, 2023.
- [11] E. Ghasemi, S. M. Razavi, S. Mohamadzadeh, and M. Taghipour-Gorjikolaie, "Facial Expression Recognition through Suboptimal Filter Design Using a Metaheuristic Kidney Algorithm," *Journal of Electrical and Computer Engineering Innovations*, vol. 12, no. 2, pp. 425–438, 2024.
- [12] A. Nagrani , J. S. Chung , and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset". arXiv preprint arXiv:1706.08612. 2017.
- [13] J. W. Jung , H. S. Heo , I. H. Yang , H. J. Shim , and H. J. Yu, "Avoiding speaker overfitting in end-to-end dnns using raw waveform for text-independent speaker verification" . extraction, vol. 8, no. 12, pp. 23-24, 2018.
- [14] G. Wei, Y. Zhang, H. Min, and Y. Xu, "End-to-end speaker identification research based on multi-scale SincNet and CGAN," *Neural Computing and Applications*, vol. 35, no. 30, pp. 22209–22222, Aug. 2023.
- [15] S. S. Tirumala and S. R. Shahamiri, "A review on deep learning approaches in speaker identification". In *Proceedings of the 8th international conference on signal processing systems*, Nov. 2016, pp. 142-147.
- [16] K. A. Abdalmalak and A. Gallardo-Antolín, "Enhancement of a text-independent speaker verification system by using feature combination and parallel structure classifiers," *Neural Computing and Applications*, vol. 29, no. 3, pp. 637–651, Jul. 2016.
- [17] A. Ashar, M. S. Bhatti and U. Mushtaq, "Speaker Identification Using a Hybrid CNN-MFCC Approach," 2020 International Conference on Emerging Trends in Smart Technologies (ICETST), Karachi, Pakistan, 2020, pp. 1-4.
- [18] B. K. P and R. K. M, "ELM speaker identification for limited dataset using multitaper based MFCC and PNCC features with fusion score," *Multimedia Tools and Applications*, vol. 79, no. 39–40, pp. 28859–28883, Aug. 2020.
- [19] M. K. Singh, "A text independent speaker identification system using ANN, RNN, and CNN classification technique," *Multimedia Tools and Applications*, vol. 83, no. 16, pp. 48105–48117, Nov. 2023.
- [20] M. R. Firmansyah, R. Hidayat and A. Bejo, "Comparison of Windowing Function on Feature Extraction Using MFCC for Speaker Identification," 2021 International Conference on Intelligent Cybernetics Technology & Applications (ICICyTA), Bandung, Indonesia, 2021, pp. 1-5.
- [21] S. Chakraborty and R. Parekh, "An improved approach to open set text-independent speaker identification (OSTI-SI)," 2017 Third International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 2017, pp. 51-56.
- [22] E. S. Hassan et al., "Enhancing speaker identification through reverberation modeling and cancelable techniques using ANNs," *PLoS ONE*, vol. 19, no. 2, p. e0294235, Feb. 2024.
- [23] J. I. Ramírez-Hernández, A. Manzo-Martínez, F. Gaxiola, L. C. González-Gurrola, V. C. Álvarez-Oliva, and R. López-

- Santillán, "A comparison between MFCC and MSE features for Text-Independent speaker recognition using machine learning algorithms," in *Studies in computational intelligence*, 2023, pp. 123–140.
- [24] N. M. Almarshady, A. A. Alashban, and Y. A. Alotaibi, "Analysis and investigation of speaker identification problems using deep learning networks and the YOHO English Speech Dataset," *Applied Sciences*, vol. 13, no. 17, p. 9567, Aug. 2023.
- [25] S.Hizlisoy , and , R. S. Arslan , "Text independent speaker recognition based on MFCC and machine learning". *Selcuk University Journal of Engineering Sciences*, vol. 20, no. 3, pp. 73-78, 2021.
- [26] V. S. R. Gade and S. Manickam, "Speaker recognition using Improved Butterfly Optimization Algorithm with hybrid Long Short Term Memory network," *Multimedia Tools and Applications*, vol.13, pp.1-23, Feb. 2024.
- [27] A. Fikri and A. Zahra, "Speaker Identification in Multiple Languages: Regional, Indonesian, and English with Short Utterance," *International Journal of Emerging Technology and Advanced Engineering*, vol. 13, no. 9, pp. 25–35, Oct. 2023.
- [28] M. Hasheminejad , and H. Farsi, (2016). "Instance Based Sparse Classifier Fusion for Speaker Verification". *Journal of Information Systems and Telecommunication (JIST)*, vol. 3, no. 15, pp. 1, 2016.
- [29] R. Li , J. Y. Jiang , J. Liu , C. C. Hsieh , and W. Wang, "Automatic speaker recognition with limited data". In *Proceedings of the 13th International Conference on Web Search and Data Mining*, Jan. 2020, pp. 340-348.
- [30] Md. A. Islam, W. A. Jassim, N. S. Cheok, and M. S. A. Zilany, "A robust speaker identification system using the responses from a model of the auditory periphery," *PLoS ONE*, vol. 11, no. 7, p. e0158520, Jul. 2016.
- [31] S. Nagarajan, S. S. S. Nettimi, L. S. Kumar, M. K. Nath, and A. Kanhe, "Speech emotion recognition using cepstral features extracted with novel triangular filter banks based on bark and ERB frequency scales," *Digital Signal Processing*, vol. 104, p. 102763, Sep. 2020.
- [32] J. Chung, Ç. Gülçehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv (Cornell University)*, Jan. 2014.
- [33] N. Mohammadi, A. Rezakhani, H. H. S. Javadi, and P. Asghari, "FLHB-AC: Federated Learning History-Based Access Control using Deep Neural Networks in healthcare system," *Journal of Information Systems and Telecommunication (JIST)*, vol. 12, no. 46, pp. 90–104, Jun. 2024.
- [34] A. Shewalkar, D. Nyavanandi, and S. A. Ludwig, "Performance Evaluation of Deep Neural Networks Applied to Speech Recognition: RNN, LSTM and GRU," *Journal of Artificial Intelligence and Soft Computing Research*, vol. 9, no. 4, pp. 235–245, Oct. 2019.
- [35] A. Barati, H. Farsi, and S. Mohamadzadeh, "Integration of the latent variable knowledge into deep image captioning with Bayesian modeling," *IET Image Processing*, , vol. 17, no. 7, pp. 2256–2271, 2024.
- [36] H. S. Munir, S. Ren, M. Mustafa, C. N. Siddique, and S. Qayyum, "Attention based GRU-LSTM for software defect prediction," *PLoS ONE*, vol. 16, no. 3, p. e0247444, Mar. 2021.
- [37] C. Yin, D. Tang, F. Zhang, Q. Tang, Y. Feng, and Z. He, "Students learning performance prediction based on feature extraction algorithm and attention-based bidirectional gated recurrent unit network," *PLoS ONE*, vol. 18, no. 10, p. e0286156, Oct. 2023.
- [38] Y. Wang et al., "Prediction of outpatients with conjunctivitis in Xinjiang based on LSTM and GRU models," *PLoS ONE*, vol. 18, no. 9, p. e0290541, Sep. 2023.

Load Balancing Algorithms in Cloud, Fog Computing and Convergence of Fog and Cloud – A Survey

Seyedeh Leili Mirtaheri^{1*}, Mahya Azari Jafari², Sergio Greco¹, Ehsan Arianyan³, Reza Mansouri⁴

¹. Department of Computer Engineering, Modeling, Electronics and Systems, University of Calabria, Italy

². Faculty of Mathematics and Computer Science, Kharazmi University, Tehran, Iran

³. Department of Information Technology, ICT Research Institute (ITRC), Tehran, Iran

⁴. Computer Science Department, Georgia State University, Atlanta, GA, USA

Received: 04 Aug 2022/ Revised: 04 Oct 2023/ Accepted: 18 Nov 2023

Abstract

Cloud computing and fog computing are deployed as computing storage and services for the end-users. Fog computing promotes task performance through storage, computing, and networking services. Instead of taking place in centralized cloud computing data centers, these services can be provided via near-edge devices. Efficient load balancing in distributed computing systems has been the main challenge. The load balancing algorithm has an important role in enhancing the Quality of Service (QoS), throughput, and resource utilization and diminishing the potential cost and its strategy and architecture completely depend on the centralized or distributed architecture of the system and the type of requests. Cloud computing and fog computing use centralized and distributed architectures, respectively. The load balancing algorithm in these two environments cannot be the same. Meanwhile, the demand for near real-time processing requests is drastically increasing; load balancing should be able to handle real-time requests. This paper reviews and investigates the modern and diverse load balancing aspects of fog and cloud computing systems. We also categorize the load balancing algorithms in cloud and fog computing: meta-heuristic algorithms, heuristic algorithms, learning algorithms, and customized algorithms. We propose different research classes about the algorithm's type, objectives, simulation tools, and so forth. This review demonstrates that the most prevalent categories of methods used in load balancing in fog and cloud computing are custom approaches and meta-heuristic algorithms, respectively. While the most renowned load balancing algorithms have not yet succeeded in fog environments, meta-heuristic algorithms have shown their competence in cloud environments impeccably.

Keywords: Fog Computing; Cloud Computing; Convergence of Fog and Cloud; Load Balancing.

1- Introduction

Today, the dramatic development of IoT and mobile internet has caused both objects and people to connect to the internet anytime, anywhere. The substantial number of devices connected to the internet has led to tremendous data. Due to this vast amount of data, current processing and storage equipment cannot meet people's demands, making it difficult to manage them with current technology, including distributed systems and cloud computing.

Cloud computing is a suitable option for data processing because of its high storage and processing potential. Nonetheless, this processing pattern is centralized, and all processing of tasks must be performed literally in a cloud. It means that all requests are sent to a centralized cloud. The centralized point is a challenging issue in cloud computing

because processing resources are not proportional to the network bandwidth [1].

In some applications of IoT, intelligent traffic control systems, smart homes, health-related systems, smart networks, and many other delayed-sensitive systems, we require low latency and mobility. Therefore, the delay is not acceptable to the system caused by exchanging the data with a centralized cloud [2].

Some cloud decisions can be calculated and implemented locally without being transmitted to the cloud, and the near real-time decision-making process cannot tolerate delay. Thus, Fog computing is a promising solution to support: 1) computational demand in real-time and sensitive applications, 2) delays in IoT and geographically distributed devices, 3) high-density network challenges, 4) long service delays, and 5) reduced quality of service [1].

Fog computing is a distributed computational model. This computational model places many heterogeneous network-

connected devices at the network's edges to provide services such as processing, network communication, and storage in a comprehensive manner. Thus, fog computing improves the system's overall performance. Fog computing responds effectively to near real-time applications and improves latency and bandwidth.

To assist future load balancing researchers in fog computing, we surveyed convergence fog, cloud computing, the various infrastructures, mechanisms, and existing algorithms in load balancing. This paper provides a new classification of load balancing algorithms in cloud and fog environments.

In sections 2 and 3 we go through fog and cloud computing definitions with various infrastructures, platforms, and technical aspects. In section 4, diverse load balancing techniques, their advantages, and load balancing metrics are presented. Section 5 is devoted to various classifications of load balancing algorithms. Section 6 gives different analyses of the research done, based on different categories, and finally, section 7 is the conclusion.

2- Cloud Computing

This section studies computational infrastructure and platforming aspects of cloud computing.

2-1- Definition

Cloud computing, as described by the National Institute of Standards and Technology (NIST), is a technology model which facilitates "convenient, resource pooling, ubiquitous, on-demand access which can be easily delivered with different types of service provider interaction."

2-2- Cloud Computing Infrastructure

Public cloud: Public cloud gives open and unrestricted access to infrastructure to the public [2],[3]. Private cloud: When computing takes place inside the data center, it is a private cloud. Community cloud: This model allows the cloud resources to be shared and utilized by more than one organization simultaneously. Virtual private cloud: It is a semi-private cloud deployment model with less infrastructure. Hybrid cloud: It is a combination of two or more clouds (public, private, or community) [3].

2-3- Service Models in Cloud Computing

IaaS (Infrastructure as a Service): [4]

PaaS (Platform as a Services): [4]

SaaS (Software as a Service): [4]

CaaS (Computing as a Service): [5]

SECaaS (Security as a Service): [5]

3- Fog Computing

This section studies computational infrastructure and platforms, features, and architecture of fog computing.

3-1- Definition

Fog computing is a model with constraints on storage, computing, and distributed network services between different devices and classic cloud computing [6]. The OpenFog Consortium elucidates fog computing as a system-level horizontal architecture, distributing storage, computing, control, and networking resources and services along the spectrum from cloud to things.

3-2- Fog Computing Infrastructure

According to the definition in [7], fog infrastructure has four types: private fog, public fog, community fog, and hybrid fog.

Private fog: Created and owned by an organization, a third party, or both, a private fog is deployable off or on-premises. While the fog is managed and operated by its owner, a single organization offers the resources exclusively (e.g., business units).

- Public fog: Created and owned by a government organization, company, academic institute, or a mixture, a public fog is deployed on the properties of the providers. While the fog is managed and operated by its owner, the general public offers the resources for open use.

- Community fog: Created by one or many organizations in a community, a third party, or an amalgamation of them, a community fog may be deployed off or on-premises. While the fog is maintained and operated by its creator(s), the resources are offered exclusively to consumers of a particular community of organizations with shared incentives.

- Hybrid fog: A type of fog computing that integrates a private/public cloud (i.e., a hybrid cloud) with a private/public/community fog, that can be proper due to the physical resource restraints. Consequently, this platform is extended in a scalable architecture as a hybrid cloud, that is elastic, scalable, and with available on-demand resources [7].

3-3- Service Models

Depending on who provides infrastructure, platform, or software, fog computing platforms can be classified as high/low-level virtualized resources in three different categories [7]. We classify fog computing workloads into static and dynamic ones—the last of which contain some metrics like the user, location, and time. In Figure 1, fog computing's service model is presented.

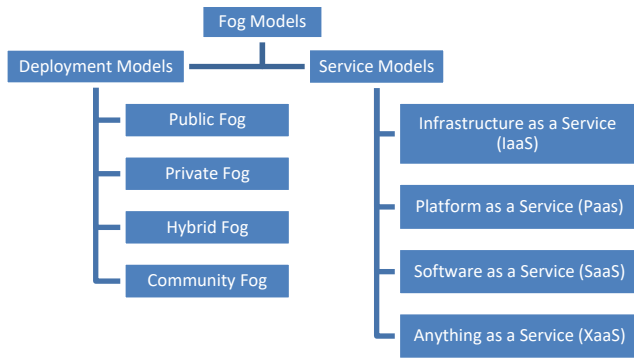


Fig. 1 Service model in fog

3-4- Characteristics of Fog Computing

According to the definitions of fog-based computing, a prominent aspect of this computational model is the proximity of resources to end devices (sensors/Internet of Things devices), which is one of the highlights compared to other computational models. Other fog computing aspects include real-time interactions, low latency, mobility, interoperability, scalability, geographical distribution, heterogeneity, security, low bandwidth consumption, and low energy consumption [6].

Fog-based processing is a new paradigm that tries to expand cloud computing capabilities at the network's edges. Performing a task via cloud computing may take a long time, especially when the network delay is high or the client's load is exceeded.

This case is more sensible in mobile devices because the wireless network delay is higher due to the relatively lower bandwidth. Therefore, researchers advanced the fog computing pattern to solve the problems regarding mobile devices. This computational model can improve performance while reducing energy consumption in environments where mobile devices are available [1]. There are some fog-based hierarchical architectures that add a layer of fog in the middle of cloud centers and end devices. Figure 2 shows the fog-based hierarchical architecture.

4- Load Balancing

Currently, load balance is a significant challenge in cloud computing. There are many requests from thousands of users and customers that need a lot of hardware and bandwidth. A load balancer helps to allocate the workload between different nodes and guarantee that no nodes are overloaded. A load-balancing algorithm's aim is improving the response time by using available resources. Other goals of load balancing algorithms are reducing computational time, increasing throughput, reducing error tolerance and execution time, and so on.

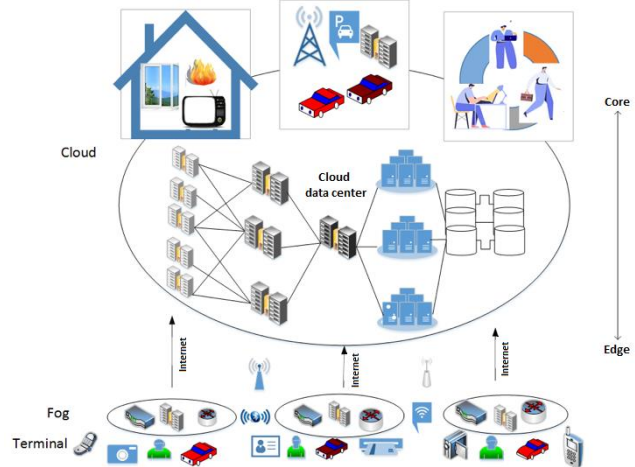


Fig. 2 Fog-based hierarchical architecture

4-1- Advantages of Load Balancing

Load balancing the system's workload improves all computational nodes' efficiency, thus improving its overall efficiency. Some significant advantages of load balancing are as follows [7], [8]:

- The task of waiting time is reduced.
- The task of response time is minimized.
- The exploitability of system resources is maximized.
- The system throughput is maximized.
- The readability and stability of the system are improved.
- It accommodates future modifications.
- Prolonged starvation is avoided for small jobs.
- In load balancing, overall system performance is enhanced by improving the performance of each node.

4-2- Load Balancing Metrics

Some important load balancing metrics are throughput, response time, scalability, resource utilization, fault tolerance, migration time, performance, overload, and energy consumption.

4-3- Types of Load Balancing Algorithms

Contingent upon the initiation of the process, load balancing algorithms are categorizable as follows:

- Sender-initiated: In this type, the sender initiates the process. The sender sends request messages until it finds a receiver accepting the load [9].
- Receiver-initiated: In this type of description algorithm, the process is initiated by the receiver, where it sends request messages until a sender able to get the load is found (a node that is under-loaded) [9].
- Symmetric: This type is an amalgamation of sender-initiated and receiver-initiated algorithms [9].

Subject to the system's current state, load balance algorithms may be categorized into dynamic and static as

well [10]. Figure 3 depicts the taxonomy of load balancing algorithms.

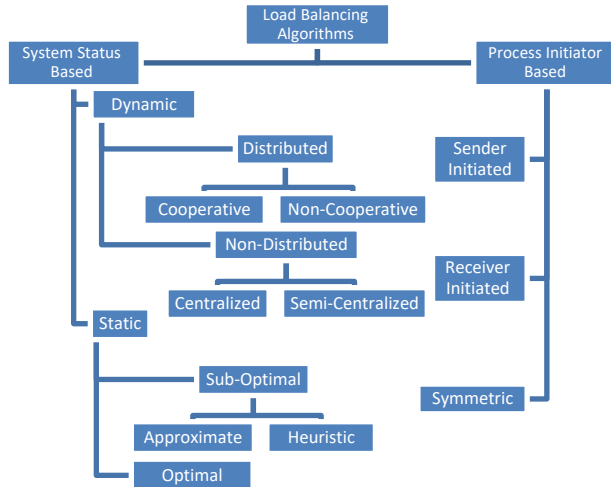


Fig. 3 Taxonomy of load balancing algorithms

4-4- Load balancing in cloud computing

Load balancers are advantageous to cloud environments in which massive workloads overloading a single server is highly likely; hence, many high-level services will be unavailable, and thus, adversely affecting both response time, service reliability, and Service-Level Agreement (SLA)—all of which are critical to business processes.

4-5- Load Balancing in Fog Computing

In fog computing, data sent by IoT devices/sensors is transferred to the fog nodes. Due to the high rate of data generation, some fog nodes get overloaded; hence, a load balancer should be used to offload the tasks to the nodes that are less overloaded [11].

5- Classification of load Balancing Algorithms in Fog and Cloud Computing

Load Balancing (LB) is NP-Pharisees., and finding real solutions for NP-hard algorithms is too costly. Due to the Non-deterministic of this problem, various methods have been used to balance the load among cloud and fog nodes. In this section, as shown in Figure 4, we classify load balancing algorithms into 4 groups: Meta-heuristic algorithms, Heuristic algorithms, Algorithms employing machine learning, and Custom algorithms.

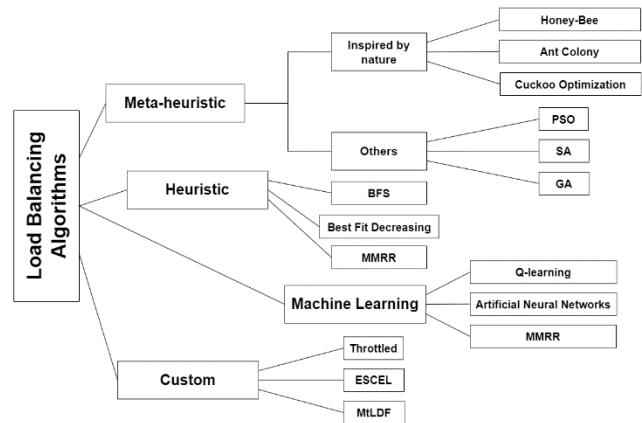


Fig. 4 Classification of load balancing algorithms and some of their examples

5-1- Meta-Heuristic LB Algorithms

In this section, we review several different load balancing strategies. Meta-heuristic methods—taking inspiration from nature or biological behaviors—consider some of the optimization hypotheses meta-heuristic methods, rather than heuristic algorithms, require more time to obtain the final solution. Amongst the meta-heuristic algorithms are the Hill-climbing algorithm, Honey-Bee algorithm, Particle Swarm Optimization (PSO) algorithm, Simulated Annealing (SA) algorithm, Genetic Algorithm (GA), and Ant Colony (ACO) algorithm.

5-2- Classification based on Meta-Heuristic Algorithms in Fog Computing

PSO algorithm, which is a meta-heuristic algorithm, has been used in papers [12], [13], [14]. The authors in [12] applied a new architecture based on SDN networks in the fog/cloud environment called SDCFN to obtain the desired load balance and reduce the distance between intelligent vehicles by virtue of PSO. [13] provides a model based on load balance and energy-aware planning in the fog environment to solve energy consumption issues in smart factories. The PSO algorithm is used to prioritize the workload. [14] presents a fog/cloud-based approach to reduce processing time and response time. It uses PSO-SA algorithms to properly allocate requests (virtual machines) and balance loads between virtual machines. [15] uses four load balancing algorithms, Throttled, PSO, RR, and Active VM Load Balancing (MLB), and four-layered architecture, to manage users' requests for electricity and reduce energy consumption.

[16] proposes a new algorithm for task scheduling with the tasks being modeled with a directed acyclic graph $G(V, E)$, where V are the tasks with their respective weights indicating their execution times, and E are the prerequisite relations between the tasks, with their respective weights

indicating the communication cost of sending a message between two tasks. The authors use the Genetic algorithm to obtain the initial population as answers to the task scheduling problem, and further, use the PSO algorithm to find the optimal solution. They have used a novel cost function based on these two algorithms to measure task execution time on available resources and their method shortens the length of the critical path and reduces the communication costs among the processors.

The Honey-Bee algorithm is another meta-heuristic algorithm used in papers [17] and [18]. [17] advances a new architecture to balance the traffic load. This load balancing method is decentralized, which helps achieve a load balance between virtual machines in the fog environment. By applying the honey bee algorithm and proposing a new architecture, it allocates resources optimally. [18] have used the Honeybee algorithm to prioritize user requests, minimize energy consumption, and reduce execution time applications.

The Hill-Climbing algorithm, which is in the category of meta-heuristic algorithms, is used in paper [19] to balance the fog computing load by managing the request load from consumers to the appropriate virtual machines. Optimal load balancing is a significant matter in fog computing using tabu search fog computing for load balancing [20]. The paper [21] uses ACO and PSO algorithms to efficaciously distribute load balance among the fog nodes.

5-3- Classification based on Meta-Heuristic Algorithms in Cloud Computing

The Ant Colony algorithm, categorized as a meta-heuristic algorithm, is used in papers [22], [23], [24], [25] to balance the cloud computing load. The paper [22] analyzes the performance of four load balancing algorithms which were inspired by nature, to find data center processing time (DCPT) and total response time (TRT) in the cloud environment. The paper [23] proposed a meta-heuristic approach to the ant colony optimization algorithm. This algorithm solves the task scheduling problem by focusing on minimizing the makespan/computation time only on two objectives. The paper [24] has proposed a modern distributed VM migration strategy named ACO-VMM with high scalability and reliability. Moreover, to find the near-optimal mapping between virtual/physical machines, they propose two approaches inspired by two traversing strategies for ants. The paper [25] advanced a new algorithm based on improved ant colony optimization to ameliorate the process of allocating resources and guarantee the quality of service. The Honey-Bee and Cuckoo Optimization Algorithm (COA) algorithms are the meta-heuristic algorithms used in these papers [26] and [27].

[28] proposes a load balancing algorithm focused on saving energy by mimicking the life of a bird family called

cuckoos (COA). Cuckoos raise their young by laying eggs in the roosts of other birds with similar eggs. Using COA, over-utilized hosts are detected, and afterward, some VMs are chosen for migration from these hosts to others. The paper [26] advanced a new efficient load balancing algorithm based on bee colonies, in which the tasks removed from overloaded VMs and under-loaded VMs are regarded as honey bees and food sources, respectively. The effort of this technique is to reduce response time and the number of task migrations.

The paper [27] proposed a novel modified Artificial Bee Colony (ABC) method named Mutation Based ABC (MABC). This algorithm highlights the procedure of detecting under-utilized available servers in the provided data centers. In line with that, the paper [29] introduces the integration of the swarm intelligence algorithm in an artificial bee colony with a heuristic scheduling algorithm named Heuristic task scheduling with Artificial Bee Colony (HABC).

[30] proposes a new version of the meta-heuristic Grey Wolf Optimization (GWO) algorithm, which mimics the hunting behavior of grey wolves, with alpha wolves as leaders, and beta, delta, and omega wolves in the next ranks, forming a hierarchy. The authors model the nodes (VMs) in a cloud infrastructure as preys for a pack of wolves. Using a load threshold and based on estimated loads, they try to find under loaded nodes and recommend them to the server. Their method outperforms PSO, ABC, and GA in terms of makespan, cost, response time, and resource utilization.

The Water Wave Algorithm (WWA) algorithm is another meta-heuristic algorithm used in the paper [31]. The paper aims for resource scheduling in the cloud environment.

The PSO algorithm is another meta-heuristic algorithm used in papers [32] and [33]. The algorithm is based on the heuristic optimization technique and used for analyzing the optimal path of solution space; while putting upload on a specific VM for processing of resources, it moves along all the VM and determines the optimal machine to put the load. The paper [32] introduced a load balancing strategy by using revised PSO task scheduling (LBMPs). The paper [33] proposed a new multi-criteria optimization technique for the weighted task scheduling that is called PSO based α PSO-TBLB (Task Based Load Balancing) load balancing method.

The genetic algorithm, which is a meta-heuristic algorithm, has been used in the paper [34]. The idea behind considering the priority is real-world virtualization. The authors advanced a policy for cloud task scheduling based on the load balancing Enhanced Genetic Algorithm (EGA). This algorithm schedules VMs in a way that load balancing is achieved, and the need for VM migrations is reduced due to its smart way of allocating VMs to physical machines using the fitness function. Table 1 mentions the prominent meta-heuristic solutions to the load balancing problem.

Table 1: Meta-heuristic load balancing algorithms

Author(s)	Objective	Technique	Testbed/Sim.	Target Service
Sefati and Mousavinasab 2022 [30]	Improve makespan utilization and reduce response time	GWO algorithm	Cloudsim	Cloud
Hussein et al. 2020 [21]	Improve response time	ACO and PSO algorithms	MATLAB	Fog
Bukhsh et al. 2018 [14]	Improve response, processing, and execution time	PSO and Simulated Annealing algorithms	Cloud Analyst	Fog
Zahid et al. 2018 [19]	Improve response time, processing time, and delay	Hill-Climbing algorithm	Cloud Analyst & Java	Fog
Abbasi et al. 2018 [15]	Improve response time and delay	Round Robin, PSO, Throttled, and Active VM load balancing algorithms	Cloud Analyst	Fog
Arulkumar and N. Bhalaji 2020 [22]	Improve TRT and DCPT	ACO, PSO, GA, and WWA algorithms	Cloud Analyst	Cloud
Kruekaew and Kimpan 2020 [29]	Maximize productivity and minimize total makespan	ABC and heuristic scheduling algorithms	Cloudsim	Cloud
Sharma and Saini 2019 [18]	Minimize energy consumption and execution time	Honey-bee algorithm	MATLAB	Fog
Alguliyev et al. 2019 [33]	Minimize the task execution and transfer time	PSO algorithm	Cloudsim & JSwarm	Cloud
Tellez et al. 2018 [20]	Minimize memory consumption and computational costs	Tabu-search algorithm	Cloudsim	Fog

5-4- Classification based on Heuristic Algorithms

Heuristic methods are a collection of constraints aimed at finding a suitable solution to a specific problem. Heuristic algorithms offer an approximate solution to the best solution.

5-5- Classification based on Heuristic Algorithms in Fog Computing

The Breadth-First Search (BFS) and Best Fit Decreasing (BFD), which are heuristic algorithms, are used in the papers [35] and [36], respectively. [35] proposed a secure method for load balancing and assigning tasks in edge data centers (EDC). Edge data centers are placed midst the cloud data centers and reduce network congestion, and delay by processing user requests and data in a near real-time—breadth-first search algorithm deployed to balance the workload. In this paper, the major objective is to load balancing between different types of computational nodes. First, [36] proposed a model for load balance in the fog/cloud setting. They considered a heuristic method for proper planning and location of virtual machines with virtual machine migration.

The Min-Min and the Max-Min algorithms are static load balancing algorithms classified as Heuristic algorithms. The paper [37] uses this algorithm. The authors proposed a central load balancing policy in the fog computing setting.

In this paper, a Min-Min algorithm, a simple and easy algorithm, is used to balance the load of requests. Resources are classified as reliable and unreliable in the fog layer. The paper [38] proposes three heuristic algorithms that carry out load balancing among Micro Data Centers (MDCs): minimum load, minimum distance, and Minimum Hop Distance and Load (MHDL).

5-6- Classification based on Heuristic Algorithms in Cloud computing

As mentioned, Min-Min and Max-Min are static load balancing algorithms which belong to the class of heuristic algorithms. The papers [39], [40], [41] use these algorithms.

In the article [39], Min-Min and Max-Min load balancing algorithms were analyzed. The Min-Min algorithm prioritizes tasks with smaller resource demands and minimum completion times when allocating the resources.

The paper [40] proposed a new load balancing algorithm, which combines Max-Min and Round-Robin algorithm (MMRR) to assign virtual machines to different userbase requests.

The authors in [42] propose a solution to load balancing in big data applications performed on clouds. They provide two mathematical optimization models, one to find a host machine with the maximum number of available resources, and another, for task scheduling. With the aim of reducing execution response time, their load balancer, based on the Hill-climbing algorithm, carries out resource allocation and

task scheduling. The key point of their solution is considering a deadline in model optimization for task scheduling and execution that distinguishes the proposed algorithm from existing ones. Their solution transcends FIFO, Round-Robin, MET, Min-Min, Max-Min, Genetic, ESCE, and Throttled algorithms in response time and turnaround time.

The paper [41] proposed a Max-Min scheduling algorithm. The proposed MMSIA algorithm uses the "learned learning" machine learning to improve requests' completion time by clustering requests' sizes and the utilization percent of VMs. Table 2 mentions important heuristic load balancing approaches.

Table 2 : Heuristic load balancing algorithms

Author(s)	Objective	Technique	Testbed/Sim.	Target Service
Aghdashi and Mirtaheeri 2021 [42]	Reduce execution response time	Hill-climbing algorithm	Cloudsim	Cloud
Moses et al. 2020 [40]	Improve response time and cost-effectiveness	Max-Min algorithm	Cloudsim	Cloud
Singh and Auluck 2019 [38]	Improving response time	MHDL algorithm	iFogSim	Fog
Hung et al. 2019 [41]	Improve completion time	Max-Min algorithm	Cloudsim	Cloud
Xu et al. 2018 [36]	Improve load balance among computational nodes	BFD algorithm	Cloudsim	Fog
Manju and Sumathi 2018 [37]	Improve response time	Min-Min algorithm	Cloud Analyst	Fog
Puthal et al. 2018 [35]	Improve delay and response time	BFS algorithm	MATLAB	Fog
Gopinath and Vasudevan 2015 [39]	Improve makespan	Min-Min & Max-Min algorithms	Cloudsim	Cloud

5-7- Classification based on Machine Learning Algorithms

Using machine/deep learning (neural networks) techniques, we can obtain accurate predictions with data trained in different situations and virtual machines in cloud and fog environments. It is also possible to host a virtual machine in a much shorter time. Among the methods used in this type of technique, we can mention KNN, Q-learning, ANN, and so on.

5-8- Classification based on Machine Learning Algorithms in Fog Computing

Q-learning algorithm, which is one of the machine learning techniques, is used in the paper [43] to improve response time, delay, and energy consumption. An algorithm is needed to balance the load due to the uncertainty related to user requests and different computing capacities—I have used an algorithm based on reinforcement learning. As mentioned, the technique of artificial neural networks, which is one of the methods based on machine learning, has been used in [44]. The authors in have used a four-layer architecture to minimize delays and energy consumption,

load balance, and optimally assign and schedule the task in the fog environment.

5-9- Classification based on Machine Learning Algorithms in Cloud Computing

Clustering or cluster analysis, which is one of the machine learning techniques, is used in the papers [45], [46], [47], [48], [49]. In the paper [45], a new heuristic method named LB-BC (Load Balancing based on Bayes and Clustering) is proposed. The LB-BC method uses the Bayes theorem to acquire the posterior probabilities of every candidate physical host.

The article [46] advanced an algorithm for cluster-based load balancing that performs adequately in heterogeneous node environments. This algorithm takes into consideration the tasks' resource-specific requirements and reduces the overhead cost of scanning by dividing the machines into clusters.

The article [47] introduces an algorithm able to provide more fine-tuned analytical data using machine learning methods, which can form the load scheduling mechanism. The algorithm is based on dynamic load balancing.

The paper [48] presents a method for accelerating the training of a distributed machine learning model based on a cloud service. The authors proposed a load balancing

method called fast adaptive reassignment (AdaptFR). The paper [49] proposed a strategy based on a machine-learning algorithm for intelligent VM scheduling that tries to attain

load balancing of the cloud data center. Table 3 shows the load balancing approaches which employ machine learning.

Table 3 : Machine learning load balancing algorithms

Author(s)	Objective	Technique	Testbed/Sim.	Target Service
Baek et al. 2019 [43]	Minimize latency, response time, and overload (extra cost)	Q-learning algorithm, Using three load transfer models for testing	-	Fog
Sharma and Saini 2019 [44]	Improve response time, latency, energy consumption, load balance rate	Artificial Neural Networks (ANN)	iFogSim	Fog
Parida and Panchal 2018 [47]	More efficient load balancing	Dynamic load balancing	AWS	Cloud
Zhao et al. 2016 [45]	Reduce failed number of task deployment events, improve throughput and performance of external cloud services	Naïve Bayes classification and Clustering	Cloudsim	Cloud
Kapoor and Debas 2015 [46]	Improve waiting time, execution time, turnaround time, and throughput	k-Means clustering algorithm	Java	Cloud

5-10- Classification based on Custom Algorithms

Custom algorithms are the proposed algorithms by authors based on innovative models. By studying the load balancing algorithms in fog and cloud, we have faced proposed algorithms that are not based on the known models, and the model is innovated. To continue, we will mention these researches in fog and cloud computing.

5-11- Classification based on Custom Algorithms in Fog Computing

Categories in fog computing are based on customized algorithms, algorithms, or strategies written by the paper authors to improve standards such as improved latency, response time, power consumption, energy consumption, and the like. This category includes techniques such as First-In-First-Out (FIFO), Throttled, Equally Spread Current Execution Load (ESCEL), Min-Min, and Max-Min.

The authors in [50] advance the MOABCQ method, which is a multi-objective task scheduling approach using hybrid artificial bee colony algorithm along with Q-learning. Their method calculates the fitness of the VMs, based on which, considers the selection of them. The MOABCQ method improves throughput, cost reduction, makespan reduction, and resource utilization.

[51] aims to process and prioritize input requests using the queue model under the SLA law. To establish a strategy for allocating resources, [36] introduces a dynamic resource allocation method named DRAM in the fog network. The

introduced technique consists of four main parts for load balancing among nodes in the cloud and fog platforms. DRAM's implementation is such that it allocates the resources statically and schedules them in a dynamic manner in fog services through identifying the spare spaces, global resource allocation based on load balance, partitioning the fog service, and static resource allocation for the subsets of the fog service.

The authors in [52] aim to reduce energy consumption, cost, and time by making appropriate decisions and scheduling load transfer among fog nodes. To optimize and distribute the load in fog settings by taking into account specific multi-tenancy demands (priority and delay), the authors in [53] proposed the Multi-tenant Load Distribution algorithm for Fog Environments (MtLDF).

The authors in [54] proposed an algorithm for load balancing in fog computing focused on graph partitioning. In their paper, the physical node graph model is viewed as a VM graph model. Afterward, depending on the resource distance and task load balancing, using a graph partition and clustering, the VM node provides services to the user.

Fog-Based Radio Access Networks (F-RAN) have an important role in future 5th generation (5G) cellular networks. [55] introduced the concept of virtual FAPs (v-FAPs), set up by several local IoT devices under the control of the FAPs. In this paper, the first authors formulated an optimization problem for optimal task assignment to reduce the maximum resource costs. Then they present a service load balancing algorithm for the v-FAPs to assign appropriate tasks.

Increasing the traffic load in healthcare systems causes all the requests to be sent to the main server to be delayed.

Delays are intolerable in healthcare scenarios. To alleviate this issue, the authors of [56] aim to provide efficient resource utilization by conjugating fog computing support so that the requests are dealt with by foglets, and only crucial requests are sent to the cloud to be processed.

The authors in [57] proposed an adaptive load balancing algorithm called LBA-le (Load Balancing Algorithm for IoT communications within e-health environment). The proposed load-balancing algorithm is based on integrating IoT communication parameters in the flow control process supported by the TCP protocol to consider the network fluctuations and apply them in the e-health domain as well. As the demand for numerous IoT applications increases, fog nodes tend to overload, even close to the sensors; hence the response time of IoT applications and latency increases. As a result, [58] proposed an algorithmic model that takes into consideration the dynamics and heterogeneity of computational nodes in fog computing. These models utilize the predefined policies by the network administrator to assign tasks to the most fitting nodes.

The authors in [59] used comprehensive dynamic resource allocation for load balancing. The method used in this paper includes the following four phases:

1. Service partitioning.
2. Gathering spare space details.
3. Primary static resource allocation.
4. Dynamic resource allocation that secures global load-balancing in the fog environment.

In cloud registering, Load-adjusting is one of the testing undertakings. Various load balancing strategies are proposed for load adjusting. The authors in [60] proposed a heap adjusting calculation. Load adjusting—a dynamic strategy—is the system to adjust the heap to the cloud hubs so that Computing Communication and Signal processing in every hub viably uses the assets and limits the reaction time.

The authors in [61] introduced two new load sharing mechanisms, such as adaptive forwarding, and sequential forwarding, to offload tasks towards the neighboring nodes. The authors in [62] proposed a load balancing technique for IoT-Gateways and network links through the use of Software-Defined Networks (SDN). The main goal of this method is performance improvement in IoT scenarios based on fog computing. The authors in [63] proposed LL(F, T) power-of-random choices based on distributed peer-to-peer load balancing algorithm.

5-12- Classification based on Custom Algorithms in Cloud Computing

The paper [64] proposed algorithm-based Self-learning and Adaptive Load balancer (SSAL). The algorithm focuses on data centers' overall throughput optimization in unstable environments. In order to estimate the recent capabilities of the servers and assign workloads commensurate to the

current relative potential of the servers, SSAL logically splits the time into fixed-length feedback intervals.

The paper [65] introduces a hybrid strategy for load balancing and task scheduling called Dems. The strategy embodies three main algorithms: Querying and Migrating tasks (QMT), On-Demand scheduling, and Staged Task Migration (STM).

The paper [66] aims to enhance the performance of the computing clusters by advancing a combination of centralized and decentralized load balancing. In the proposed load balancing algorithm, computing nodes notify other neighboring nodes of their load and resource usage details to determine their relative state. The resource availability information and load status of all the nodes in each cluster, based on which workload distribution and migration come about, are stored in the main node of the cluster.

The paper [67] aims to optimize the load and schedule resources for each cloud user request with the efficient transformation of the data center by proposing the Fuzzy-based Multidimensional Resource Scheduling and Queuing Network (F-MRSQN) method. The method's major intent is to effectively put integrated scheduling and load balancing algorithms into use, depending on minimum processing time and maximum resource utilization in the cloud environment. This method's main objective is to effectively utilize combined scheduling and load balancing algorithms based on maximum resource usage and minimum processing time in the cloud environment.

The paper [68] presented a new hybrid load balancing algorithm, an amalgamation of randomizing and greedy load balancing algorithms. The main goal is to improve the response time for the user (UserBase) and the processing time of the data center.

The paper [69] advanced a novel mechanism for load balancing. This method is used for calculating server processing power. It is also able to load and obtain PS values, thus reducing the chance of a server being incapable of handling excessive computational requirements.

The authors in [70] advance a model in which, without a central node to manage the system load, each individual node is responsible to estimate its status based on its computing power and the intended volume of load, which will classify themselves into nodes with positive load, the nodes with less computing power relative to their considered load, and nodes with negative load, which will undertake extra portions of the positive nodes' load, leading to load balance. They also define a parameter entitled compensating factor, to address communication delay between nodes, which is calculated from each node's perspective, and to compensate the effect of external load by using information from neighbor nodes status. Their simulation results illustrates significant improvement in comparison with common distributed load balancing approaches in managing dynamic requests.

In the paper [71], a load balancing algorithm based on VM availability is proposed. In particular, the Availability Index (AI) is assessed for all VMs over a specific period, based on which the jobs are allotted to the machines. Table 4 mentions some of recent custom load balancing approaches. [72] presents a new mechanism for flow scheduling in cloud data centers. This method makes decisions based on flows' sizes. Small flows (mice) are sent via the EMCP algorithm and big flows (elephant) are scheduled using bidirectional search. Their approach can balance the network more efficiently than traditional Static, ECMP, and DiFS mechanisms.

5-13- Load Balancing Algorithms in Converged Fog and Cloud Computing

The article [36] advanced a Virtual Machine scheduling approach for load balancing in fog/cloud computing. By exploiting the VM live migration method, the authors design a VM scheduling mechanism through dynamic VM scheduling and VM placement.

The paper [74] introduces a new mechanism for scheduling IoT requests using a made-to-order implementation of the genetic algorithm (GA) as a heuristic procedure that mainly aims to improve the overall latency. The authors study the GA and evaluate it on different problems with various sizes to estimate the effects of the model with different parameters, namely the maximum number of iterations or the population size.

The paper [73] advanced an energy-efficient load balancing mechanism for scientific workflows in the fog/cloud computing environment along with a load balancing algorithm for the fog environment. Load balancing at the fog layer facilitates latency reduction, improving the quality of service, and using the resources properly. The mechanism aims to utilize the resources at the fog layer by minimizing the energy consumed by fog resources.

The authors of [75], proposed different algorithms for load balancing, task scheduling, and resource provisioning, and they recognized some of their drawbacks for further development. They surveyed the fog-integrated cloud environment and its 3-layered architecture in their paper.

The paper [76] advanced a Fault-Tolerant Scheduling Method (FTSM) for distributing service requests to ample devices in IoT-based fog/cloud environments. The method mainly aims to increase the capacity along with reliability and reduce the overhead costs and latency of cloud services. The paper [77] has proposed a reliable scheduling approach, named the Load Balanced Service Scheduling Approach (LBSSA), for allocating users' requests to the resources of

cloud-fog environments. LBSSA mainly aims to achieve proper system utilization, high load balancing, and reliable service for requests within the necessary limits of response times.

6- Analysis of Research

In this article, different load balancing algorithms in the cloud, fog computing, and convergence environment are surveyed and compared. According to the review of selected papers, we give different analyses based on different categories, algorithm types, the objectives of load balancing algorithms, task specifications, selecting a suitable location for task execution, and simulation tools.

The first categorization of the research on the load balancing algorithms is based on their type and approach. We categorize the proposed algorithms into metaheuristic, heuristic, machine learning, and customized algorithms. The results show that custom algorithms are proposed by most of the authors for fog computing load balancing algorithms. In contrast, most authors use meta-heuristic algorithms to balance the load in the cloud environment. The number of researches on proposing load balancing algorithms for converged fog and cloud computing environments is not too many yet, and it's a new scope of research for researchers in the load-balancing field.

In the converged fog and cloud computing, we face two different architectures, distributed nature of the fog computing management model and the centralized nature of the cloud computing management model. Since the load balancing algorithm is contingent upon the architecture of the system, it's a very interesting issue that we can propose a load balancing algorithm in the converged system to balance the load in all cloud and fog nodes. Based on the architectures, there are new issues such as designing distributed load balancing algorithms or centralized or the hybrid model.

By categorizing the research based on the type of algorithms, in the second level in fog and cloud, respectively, we can mention meta-heuristic algorithms in fog computing and custom algorithms in cloud computing, and in the third level, heuristic algorithms in fog and machine learning in cloud environment have been more popular. Finally, in the fourth level, the machine learning algorithms in fog and heuristic algorithms in cloud computing have been used. This issue is depicted using Figure 4 and Figure 5.

Table 4 : Custom load balancing algorithms

Author(s)	Objective	Technique	Testbed/Sim.	Target Service
Kruekaew and Kimpan 2022 [50]	Optimize task scheduling, maximize VM throughput, create load balance based on makespan	ABC and Q-learning algorithms	Cloudsim	Cloud
Kaur and Aron 2021 [73]	Reduce energy consumption, optimize resource utilization, improve QoS	Energy-aware load balancing framework	iFogSim	Cloud & Fog
Beraldi et al. 2020 [61]	Improve response time	Sequential forwarding and adaptive forwarding algorithms	MATLAB & Omnet++	Fog
Khattak et al. 2019 [56]	Improve latency and reduce traffic overhead by improving QoS	An algorithm to measure patients' heart condition, along with an algorithm for using fog servers	iFogSim & C	Fog
Mirtaheri and Grandinetti 2016 [70]	Reducing the search space in the load balancing problem, via a novel decentralized approach	A new decentralized model for estimating each node's status and load assignment accordingly	MATLAB	Cloud

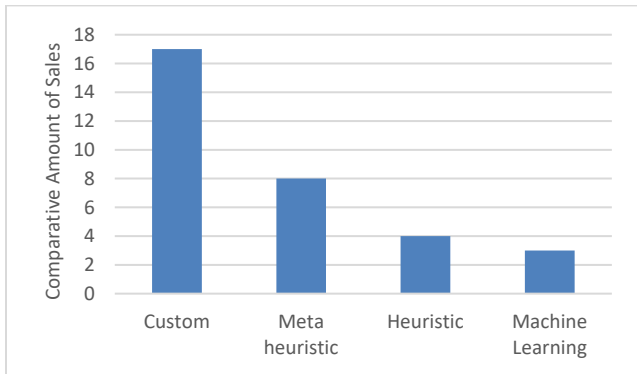


Fig. 5 Load balancing algorithms in fog computing

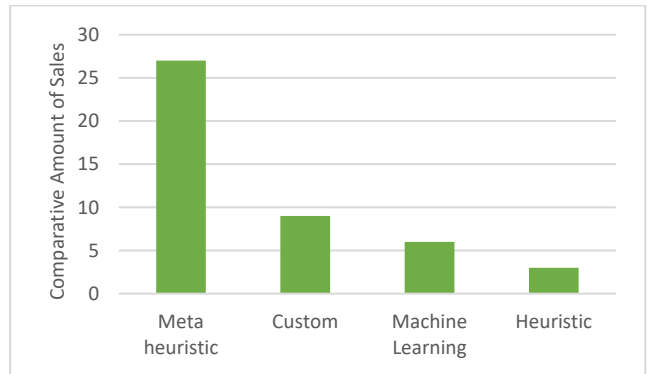


Fig. 6 Load balancing algorithms in cloud computing

In accordance with this categorization, it can be concluded that the famous algorithms have not succeeded yet in solving the load balancing issue in fog computing environments. But on the other hand, metaheuristic algorithms are the most widely used in providing load balancing algorithms in cloud computing environments. For proposing a suitable algorithm to work efficiently in converged fog and cloud computing environments, this categorization can help.

The second categorization of research is done based on the objective of the proposed load balancing algorithms in these researches. In fog computing, response time, execution time, processing time, delay and latency, throughput, computational cost, energy consumption, overload, resource utilization, power consumption, and failure rate are the most popular objectives in balancing the load of the system. Through these objectives, response time is in the first rank, and minimizing the delay and latency in the second rank is the most popular research objective. The statistic chart is shown in Figure 6.

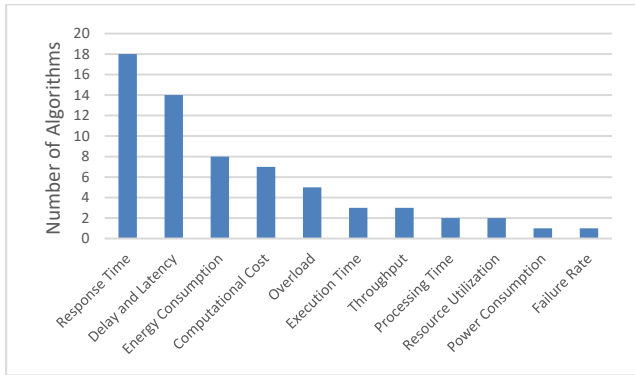


Fig. 7 Objective of load balancing algorithms in fog computing

In cloud computing, Response Time, Execution Time, Processing Time, Waiting Time, Throughput, Computational Cost, Energy Consumption, Completion Time, Makespan, Migration, Accuracy, Resource Utilization, Power Consumption, and Failure rate are the most popular objectives in designing load balancing platform. As mentioned in Figure 7, resource utilization is in the first rank in objectives of research and makespan state in the second rank. It should be noted that the proposed approaches to bring solutions to the load balancing problem commonly use random, Google Cloud Jobs (GoCJ), and synthetic workloads as their datasets for evaluating the above criteria.

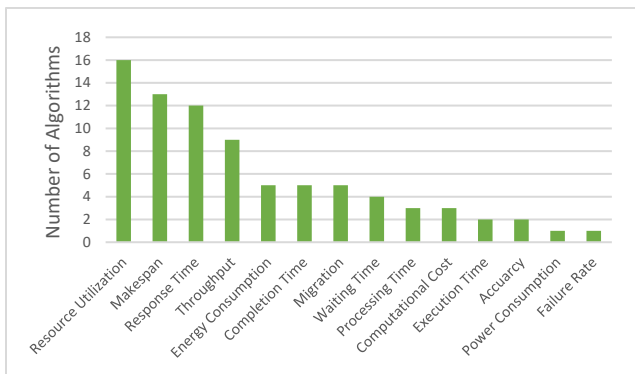


Fig. 8 Objective of load balancing algorithms in cloud computing

Based on these statistics, the objective of balancing the load in the converged fog and cloud computing settings will be a critical challenge. It's possible to select a single objective for the whole in the system or have a separate objective for the fog and cloud environment. For example, the load balancing strategy in distributing the fog-based task can be based on minimizing the response time, and the strategy of load balancing for the cloud-based tasks can improve resource utilization.

The third analysis is based on the task specification. The fog-based tasks have special specifications, and also, the cloud-based tasks have their specifications. The load

balancer should prioritize the tasks based on their specifications to reach the optimum output. Real-time nature of tasks, the priority of the tasks regarding the user's request, size of tasks, runtime duration of the task, computing-intensive tasks, data-intensive tasks, the location of requested data by tasks, the I/O requested tasks, etc. These are the challenges that the load balancer should be able to consider to manage the tasks. Therefore, if we have an environment with fog and cloud services, we should be able to consider the specification of tasks in deciding to run the tasks.

The fourth categorization of issues is about selecting a suitable location for task execution. The load balancer should be informed about the nodes' statuses in the system to find the location. There are different strategies for obtaining this information. However, we can categorize them into two strategies, pooling and interrupting. Pooling means that the load balancer in different periods asks the nodes to send the status of CPU and memory or any other needed information from the node to estimate the load of the system and making decisions about migrating the tasks to other nodes or not. Interrupting strategy means that the nodes send the overloading alarm to the load balancer when the resource utilization rate reaches the specified and predefined rate. By receiving this alarm, the load balancer makes a new decision to migrate the tasks to another suitable node to run.

Statistically, Figure 8 and Figure 9 show the percentage of evaluation tools used to review the literature here in this paper. The CloudAnalyst, iFogSim, and Cloudsim have 4.11% each, MATLAB has 7.20% of usage, C#, Java platform, JMeter, Mininet, Python 3.7, Simply package come next evaluation tools for these literature reviews in fog computing. But, Cloudsim has 67% of usage in cloud computing. Then CloudAnalyst has 14% usage, Java platform, AWS, MATLAB, and Rock Cluster come next evaluation tools for these literature reviews in cloud computing. Some aspects to consider while choosing these tools include the ability to create quantities with different random distributions, providing reports on the infrastructure's performance in the form of figures and curves, and user support, i.e., timely updates, and informative documentation [78].

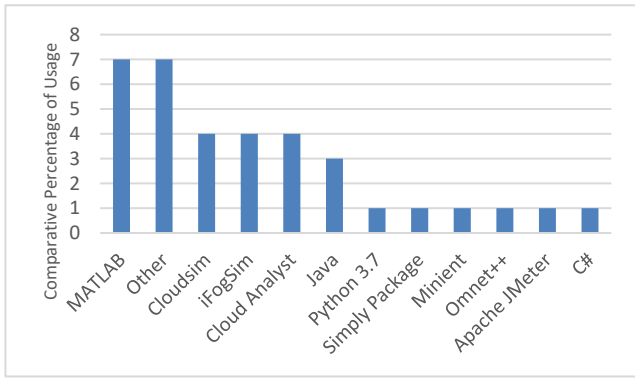


Fig. 9 Evaluation tools for load balancing mechanisms in fog computing

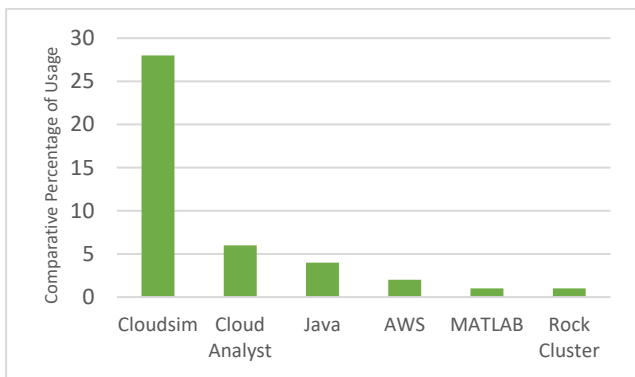


Fig. 10 Evaluation tools for load balancing mechanisms in cloud computing

7- Conclusion and Future Work

Cloud is an enormous system with various aspects, which involves cloud service providers, myriad end-users, service brokers, physical hardware machines, storage capabilities, bandwidth, internet latency, storage capabilities, scheduling algorithms, etc. Fog-based processing is a processing pattern as a result of the rapid advancement of the Internet of Things (IoT), central processing systems, and mobile internet. This processing model responds efficiently to the needs of real-time and delay-sensitive applications. The load balancer is one of the most important issues in the fog and cloud calculation model because overloading the system will reduce efficiency. Therefore, efficient algorithms are needed to load balance, optimal resource allocation, reduce response time, and increase system efficiency.

This review paper discusses new load-balancing algorithms in the fog and cloud and their converged environment. According to the classification, the most common categories of papers written in load balancing in fog and cloud computing, respectively, are custom and meta-heuristic algorithms. For designing suitable computing algorithms in a converged fog and cloud environment, the

presented categories will be useful and give a better understanding of the solutions ahead.

Further research may include considering more aspects of QoS such as security, delay for different routing policies, fault tolerance and etc. With the ongoing advancements in the discipline of artificial intelligence, utilizing optimization techniques along with other machine learning algorithms should also be considered. Taking inspiration from nature has been in the spotlight to bring solutions to the problem of load balance, nevertheless, more nature-inspired algorithms can be developed. Moreover, taking into account green computing, where its usage saves energy and reduces the trade-off between SLA requirements and the energy consumed, is suggested. Furthermore, with 5G networks becoming more and more prevalent, considering their capabilities and flaws can significantly affect future research conducted to solve this problem. Finally, as the progress of research in this area suggest, in the future, the load balancing will be carried out dynamically, and with special focus on the dependent tasks.

References

- [1] P. Hu, S. Dhelim, H. Ning, and T. Qiu, "Survey on fog computing: architecture, key technologies, applications and open issues," *Journal of Network and Computer Applications*, vol. 98, pp. 27–42, Nov. 2017, doi: 10.1016/j.jnca.2017.09.002.
- [2] M. Verma, N. Bhardwaj, and A. K. Yadav, "Real Time Efficient Scheduling Algorithm for Load Balancing in Fog Computing Environment," *International Journal of Information Technology and Computer Science*, vol. 8, no. 4, pp. 1–10, Apr. 2016, doi: 10.5815/ijitcs.2016.04.01.
- [3] S. Verma, A. K. Yadav, D. Motwani, R. S. Raw and H. K. Singh, "An efficient data replication and load balancing technique for fog computing environment," 2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom), 2016, pp. 2888-2895.
- [4] A. Alharthi, F. Yahya, R. J. Walters, and G. B. Wills, "An Overview of Cloud Services Adoption Challenges in Higher Education Institutions," *Proceedings of the 2nd International Workshop on Emerging Software as a Service and Analytics*, 2015, doi: 10.5220/0005529701020109.
- [5] S. Mathew, "Implementation of Cloud Computing in Education - A Revolution," *International Journal of Computer Theory and Engineering*, pp. 473–475, 2012, doi: 10.7763/ijcte.2012.v4.511.
- [6] H. Atlam, R. Walters, and G. Wills, "Fog Computing and the Internet of Things: A Review," *Big Data and Cognitive Computing*, vol. 2, no. 2, p. 10, Apr. 2018, doi: 10.3390/bdcc2020010.
- [7] A. Ahmed, et al. "Fog Computing Applications: Taxonomy and Requirements," *arXiv*, 26 July 2019. doi: 10.48550/arXiv.1907.11621.
- [8] E. Jafarnejad Ghomi, A. Masoud Rahmani, and N. Nasih Qader, "Load-balancing algorithms in cloud computing: A

- survey,” *Journal of Network and Computer Applications*, vol. 88, pp. 50–71, Jun. 2017, doi: 10.1016/j.jnca.2017.04.007.
- [9] Z. M. Elngomi and K. Khanfar "A Comparative Study of Load Balancing Algorithms: A Review Paper," *International Journal of Computer Science and Mobile Computing*, vol. 5, no. 6, pp. 448-458, Jun. 2016.
- [10] M. N. Arab, S. L. Mirtaheri, E. M. Khaneghah, M. Sharifi, and M. Mohammadkhani, "Improving Learning-Based Request Forwarding in Resource Discovery through Load-Awareness," *Lecture Notes in Computer Science*, pp. 73–82, 2011, doi: 10.1007/978-3-642-22947-3_7.
- [11] A. Chandak and N. K. Ray, "A Review of Load Balancing in Fog Computing," 2019 International Conference on Information Technology (ICIT), Dec. 2019, doi: 10.1109/icit48102.2019.00087.
- [12] X. He, Z. Ren, C. Shi, and J. Fang, "A novel load balancing strategy of software-defined cloud/fog networking in the Internet of Vehicles," *China Communications*, vol. 13, no. 2, pp. 140–149, 2016, doi: 10.1109/cc.2016.7405730.
- [13] J. Wan, B. Chen, S. Wang, M. Xia, D. Li, and C. Liu, "Fog Computing for Energy-Aware Load Balancing and Scheduling in Smart Factory," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4548–4556, Oct. 2018, doi: 10.1109/TII.2018.2818932.
- [14] R. Bukhsh, N. Javaid, Z. Ali Khan, F. Ishmanov, M. Afzal, and Z. Wadud, "Towards Fast Response, Reduced Processing and Balanced Load in Fog-Based Data-Driven Smart Grid," *Energies*, vol. 11, no. 12, p. 3345, Nov. 2018, doi: 10.3390/en1123345.
- [15] S. H. Abbasi, N. Javaid, M. H. Ashraf, M. Mehmood, M. Naeem, and M. Rehman, "Load Stabilizing in Fog Computing Environment Using Load Balancing Algorithm," *Lecture Notes on Data Engineering and Communications Technologies*, pp. 737–750, Oct. 2018, doi: 10.1007/978-3-030-02613-4_66.
- [16] A. Kamalinia and A. Ghaffari, "Hybrid Task Scheduling Method for Cloud Computing by Genetic and PSO Algorithms," *Journal of Information Systems and Telecommunication (JIST)*, vol 16, no. 4, pp. 1-10, Sep. 2016, doi: 10.7508/jist.2016.04.008.
- [17] M. Verma and N. B. A. K. Yadav, "An architecture for load balancing techniques for fog computing environment," *International Journal of Computer Science and Communication*, vol. 8, no. 2, pp. 43–49, 2015.
- [18] S. Sharma and H. Saini, "Efficient Solution for Load Balancing in Fog Computing Utilizing Artificial Bee Colony," *International Journal of Ambient Computing and Intelligence*, vol. 10, no. 4, pp. 60–77, Oct. 2019, doi: 10.4018/ijaci.2019100104.
- [19] M. Zahid, N. Javaid, K. Ansar, K. Hassan, M. KaleemUllah Khan, and M. Waqas, "Hill Climbing Load Balancing Algorithm on Fog Computing," *Advances on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 238–251, Oct. 2018, doi: 10.1007/978-3-030-02607-3_22.
- [20] N. Tellez, M. Jimeno, A. Salazar, and E. D. Nino-Ruiz, "A Tabu Search Method for Load Balancing in Fog Computing," *International journal of artificial intelligence*, vol. 16, pp. 106–135, 2018.
- [21] M. K. Hussein and M. H. Mousa, "Efficient Task Offloading for IoT-Based Applications in Fog Computing Using Ant Colony Optimization," *IEEE Access*, vol. 8, pp. 37191–37201, 2020, doi: 10.1109/access.2020.2975741.
- [22] V. Arulkumar and N. Bhalaji, "Performance analysis of nature inspired load balancing algorithm in cloud environment," *Journal of Ambient Intelligence and Humanized Computing*, Jan. 2020, doi: 10.1007/s12652-019-01655-x.
- [23] A. Gupta and R. Garg, "Load Balancing Based Task Scheduling with ACO in Cloud Computing," 2017 International Conference on Computer and Applications (ICCA), Sep. 2017, doi: 10.1109/comapp.2017.8079781.
- [24] W. T. Wen, C. D. Wang, D. S. Wu, and Y. Y. Xie, "An ACO-based Scheduling Strategy on Load Balancing in Cloud Computing Environment," 2015 Ninth International Conference on Frontier of Computer Science and Technology, Aug. 2015, doi: 10.1109/fcst.2015.41.
- [25] P. Verma, S. Shrivastava, and R. K. Pateriya, "Enhancing load balancing in cloud computing by ant colony optimization method," *International Journal of Computer Engineering in Research Trends*, vol. 4, no. 6, pp. 277–284, 2017.
- [26] K. R. Remesh Babu and P. Samuel, "Enhanced Bee Colony Algorithm for Efficient Load Balancing and Scheduling in Cloud," *Advances in Intelligent Systems and Computing*, pp. 67–78, Dec. 2015, doi: 10.1007/978-3-319-28031-8_6.
- [27] S. Singhal, "Load Balancing in Cloud Computing using Mutative Bacterial Foraging Optimization," *Journal of Xidian University*, vol. 14, no. 6, Jun. 2020, doi: 10.37896/jxu14.6/258.
- [28] M. Yakhchi, S. M. Ghafari, S. Yakhchi, M. Fazeli, and A. Patoghi, "Proposing a load balancing method based on Cuckoo Optimization Algorithm for energy management in cloud computing infrastructures," 2015 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO), May 2015, doi: 10.1109/icmsao.2015.7152209..
- [29] B. Kruekaew and W. Kimpan, "Enhancing of Artificial Bee Colony Algorithm for Virtual Machine Scheduling and Load Balancing Problem in Cloud Computing," *International Journal of Computational Intelligence Systems*, vol. 13, no. 1, p. 496, 2020, doi: 10.2991/ijcis.d.200410.002.
- [30] S. Sefati, M. Mousavinasab, and R. Zareh Farkhady, "Load balancing in cloud computing environment using the Grey wolf optimization algorithm based on the reliability: performance evaluation," *The Journal of Supercomputing*, May 2021, doi: 10.1007/s11227-021-03810-8.
- [31] A. V and N. Bhalaji, "Load balancing in cloud computing using water wave algorithm," *Concurrency and Computation: Practice and Experience*, vol. 34, no. 8, Sep. 2019, doi: 10.1002/cpe.5492.
- [32] A. Pradhan and S. K. Bisoy, "A novel load balancing technique for cloud computing platform based on PSO," *Journal of King Saud University - Computer and*

- Information Sciences, Oct. 2020, doi: 10.1016/j.jksuci.2020.10.016.
- [33] R. M. Alguliyev, Y. N. Imamverdiyev, and F. J. Abdullayeva, "PSO-based Load Balancing Method in Cloud Computing," *Automatic Control and Computer Sciences*, vol. 53, no. 1, pp. 45–55, Jan. 2019, doi: 10.3103/s0146411619010024.
- [34] H. Sharma and G. Sekhon, "Load Balancing in Cloud Using Enhanced Genetic Algorithm," *International Journal of Innovations and Advancement in Computer Science*, vol. 6, no. 1, pp. 100–107, 2017.
- [35] D. Puthal, M. S. Obaidat, P. Nanda, M. Prasad, S. P. Mohanty, and A. Y. Zomaya, "Secure and Sustainable Load Balancing of Edge Data Centers in Fog Computing," *IEEE Communications Magazine*, vol. 56, no. 5, pp. 60–65, May 2018, doi: 10.1109/mcom.2018.1700795.
- [36] X. Xu, Q. Liu, L. Qi, Y. Yuan, W. Dou, and A. X. Liu, "A Heuristic Virtual Machine Scheduling Method for Load Balancing in Fog-Cloud Computing," 2018 IEEE 4th International Conference on Big Data Security on Cloud (BigDataSecurity), IEEE International Conference on High Performance and Smart Computing, (HPSC) and IEEE International Conference on Intelligent Data and Security (IDS), May 2018, doi: 10.1109/bds/hpsc/ids18.2018.00030.
- [37] A. B. Manju and S. Sumathy, "Efficient Load Balancing Algorithm for Task Preprocessing in Fog Computing Environment," *Smart Intelligent Computing and Applications*, pp. 291–298, Nov. 2018, doi: 10.1007/978-981-13-1927-3_31.
- [38] A. Singh and N. Auluck, "Load balancing aware scheduling algorithms for fog networks," *Software: Practice and Experience*, Jun. 2019, doi: 10.1002/spe.2722.
- [39] P. P. G. Gopinath and S. K. Vasudevan, "An In-depth Analysis and Study of Load Balancing Techniques in the Cloud Computing Environment," *Procedia Computer Science*, vol. 50, pp. 427–432, 2015, doi: 10.1016/j.procs.2015.04.009.
- [40] A. Kazeem Moses, A. Joseph Bamidele, O. Roseline Oluwaseun, S. Misra, and A. Abidemi Emmanuel, "Applicability of MMRR load balancing algorithm in cloud computing," *International Journal of Computer Mathematics: Computer Systems Theory*, vol. 6, no. 1, pp. 7–20, Dec. 2020, doi: 10.1080/23799927.2020.1854864.
- [41] T. C. Hung, L. N. Hieu, P. T. Hy, and N. X. Phi, "MMSIA," *Proceedings of the 3rd International Conference on Machine Learning and Soft Computing - ICMLSC 2019*, 2019, doi: 10.1145/3310986.3311017.
- [42] A. Aghdashi and S. L. Mirtaheri, "Novel dynamic load balancing algorithm for cloud-based big data analytics," *The Journal of Supercomputing*, Aug. 2021, doi: 10.1007/s11227-021-04024-8.
- [43] J. Baek, G. Kaddoum, S. Garg, K. Kaur, and V. Gravel, "Managing Fog Networks using Reinforcement Learning Based Load Balancing Algorithm," 2019 IEEE Wireless Communications and Networking Conference (WCNC), Apr. 2019, doi: 10.1109/wcnc.2019.8885745.
- [44] S. Sharma and H. Saini, "A novel four-tier architecture for delay aware scheduling and load balancing in fog environment," *Sustainable Computing: Informatics and Systems*, vol. 24, p. 100355, Dec. 2019, doi: 10.1016/j.suscom.2019.100355.
- [45] J. Zhao, K. Yang, X. Wei, Y. Ding, L. Hu, and G. Xu, "A Heuristic Clustering-Based Task Deployment Approach for Load Balancing Using Bayes Theorem in Cloud Environment," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 2, pp. 305–316, Feb. 2016, doi: 10.1109/tpds.2015.2402655.
- [46] S. Kapoor and C. Dabas, "Cluster based load balancing in cloud computing," 2015 Eighth International Conference on Contemporary Computing (IC3), Aug. 2015, doi: 10.1109/ic3.2015.7346656.
- [47] S. Parida and Bakul Panchal, "An Efficient Dynamic Load Balancing Algorithm Using Machine Learning Technique in Cloud Environment," *International journal of scientific research in science, engineering and technology*, vol. 4, pp. 1184–1186, 2018.
- [48] M. Li et al., "Distributed machine learning load balancing strategy in cloud computing services," *Wireless Networks*, Jul. 2019, doi: 10.1007/s11276-019-02042-2.
- [49] X. Sui, D. Liu, L. Li, H. Wang, and H. Yang, "Virtual machine scheduling strategy based on machine learning algorithms for load balancing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, Jun. 2019, doi: 10.1186/s13638-019-1454-9.
- [50] B. Kruekaew and W. Kimpan, "Multi-Objective Task Scheduling Optimization for Load Balancing in Cloud Computing Environment Using Hybrid Artificial Bee Colony Algorithm With Reinforcement Learning," *IEEE Access*, vol. 10, pp. 17803–17818, 2022, doi: 10.1109/access.2022.3149955.
- [51] D. Rathod and G. Chowdhary, "Load Balancing of Fog Computing Centers: Minimizing Response Time of High Priority Requests," *International Journal of Innovative Technology and Exploring Engineering*, vol. 8, no. 11, pp. 2713–2716, Sep. 2019, doi: 10.35940/ijitee.k2171.0981119.
- [52] S. Sthapit, J. R. Hopgood, and J. Thompson, "Distributed computational load balancing for real-time applications," 2017 25th European Signal Processing Conference (EUSIPCO), Aug. 2017, doi: 10.23919/eusipco.2017.8081436.
- [53] E. C. Pinto Neto, G. Callou, and F. Aires, "An algorithm to optimise the load distribution of fog environments," 2017 IEEE International Conference on Systems, Man, and Cybernetics (SMC), Oct. 2017, doi: 10.1109/smc.2017.8122791.
- [54] S. Ningning, G. Chao, A. Xingshuo, and Z. Qiang, "Fog computing dynamic load balancing mechanism based on graph repartitioning," *China Communications*, vol. 13, no. 3, pp. 156–164, Mar. 2016, doi: 10.1109/cc.2016.7445510.
- [55] J. Jijin, B.-C. Seet, P. H. J. Chong, and H. Jarrah, "Service load balancing in fog-based 5G radio access networks," 2017 IEEE 28th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC), Oct. 2017, doi: 10.1109/pimrc.2017.8292300.
- [56] H. A. Khattak et al., "Utilization and load balancing in fog servers for health applications," *EURASIP Journal on*

- Wireless Communications and Networking, vol. 2019, no. 1, Apr. 2019, doi: 10.1186/s13638-019-1395-3.
- [57] S. Hamrioui and P. Lorenz, "Load Balancing Algorithm for Efficient and Reliable IoT Communications within E-Health Environment," GLOBECOM 2017 - 2017 IEEE Global Communications Conference, Dec. 2017, doi: 10.1109/glocom.2017.8254435.
- [58] J. L. Crespo-Mariño and E. Meneses-Rojas, Eds., High Performance Computing. Cham: Springer International Publishing, 2020. doi: 10.1007/978-3-030-41005-6.
- [59] X. Xu et al., "Dynamic Resource Allocation for Load Balancing in Fog Environment," Wireless Communications and Mobile Computing, vol. 2018, pp. 1–15, 2018, doi: 10.1155/2018/6421607.
- [60] D. Baburao, T. Pavankumar, and C. S. R. Prabhu, "Load balancing in the fog nodes using particle swarm optimization-based enhanced dynamic resource allocation method," Applied Nanoscience, Jul. 2021, doi: 10.1007/s13204-021-01970-w..
- [61] R. Beraldi, C. Canali, R. Lancellotti, and G. P. Mattia, "A Random Walk based Load Balancing Algorithm for Fog Computing," 2020 Fifth International Conference on Fog and Mobile Edge Computing (FMEC), Apr. 2020, doi: 10.1109/fmec49853.2020.9144962.
- [62] E. Batista, G. Figueiredo, M. Peixoto, M. Serrano, and C. Prazeres, "Load Balancing in the Fog of Things Platforms Through Software-Defined Networking," 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Jul. 2018, doi: 10.1109/cybermatics_2018.2018.00297.
- [63] R. Beraldi and H. Alnuweiri, "Exploiting Power-of-Choices for Load Balancing in Fog Computing," 2019 IEEE International Conference on Fog Computing (ICFC), Jun. 2019, doi: 10.1109/icfc.2019.00019.
- [64] V. R. Chandakanna and V. K. Vatsavayi, "A sliding window based Self-Learning and Adaptive Load Balancer," Journal of Network and Computer Applications, vol. 56, pp. 188–205, Oct. 2015, doi: 10.1016/j.jnca.2015.07.001.
- [65] Y. Liu, C. Zhang, B. Li, and J. Niu, "DeMS: A hybrid scheme of task scheduling and load balancing in computing clusters," Journal of Network and Computer Applications, vol. 83, pp. 213–220, Apr. 2017, doi: 10.1016/j.jnca.2015.04.017.
- [66] S. S. Patil and A. N. Gopal, "Dynamic Load Balancing Using Periodically Load Collection with Past Experience Policy on Linux Cluster System," American Journal of Mathematical and Computer Modelling, vol. 2, No. 2, pp. 60–75, 2017, doi: 10.11648/j.ajmcm.20170202.13.
- [67] V. Priya, C. Sathiya Kumar, and R. Kannan, "Resource scheduling algorithm with load balancing for cloud service provisioning," Applied Soft Computing, vol. 76, pp. 416–424, Mar. 2019, doi: 10.1016/j.asoc.2018.12.021.
- [68] D. F. Altayeb and F. A. Mustafa, "Analysis on Load Balancing Algorithms Implementation on Cloud Computing," International Journal of Innovative Research in Advanced Engineering, vol. 6, no. 2, pp. 1-32, 2016.
- [69] S. L. Chen, Y. Y. Chen, and S. H. Kuo, "CLB: A novel load balancing architecture and algorithm for cloud services," Computers & Electrical Engineering, vol. 58, pp. 154–160, Feb. 2017, doi: 10.1016/j.compeleceng.2016.01.029.
- [70] S. L. Mirtaheri, L. Grandinetti, "Optimized Dynamic Load Balancing in Distributed Exascale Computing Systems," Ph.D. Thesis, Dept. of Electronics, Informatics and Systems Engineering, Univ. of Calabria, Italy, 2016, doi: 10.13126/unical.it/dottorati/1370.
- [71] A. Bhandari and K. Kaur, "An Enhanced Post-migration Algorithm for Dynamic Load Balancing in Cloud Computing Environment," Advances in Intelligent Systems and Computing, pp. 59–73, Oct. 2018, doi: 10.1007/978-981-13-1544-2_6.
- [72] H. Naseri, S. Azizi, and A. Abdollahpouri, "BSFS: A Bidirectional Search Algorithm for Flow Scheduling in Cloud Data Centers," Journal of Information Systems and Telecommunication (JIST), vol. 3, no. 27, p. 175, Mar. 2020, doi: <https://doi.org/10.7508/jist.2019.03.002>.
- [73] M. Kaur and R. Aron, "Energy-aware load balancing in fog cloud computing," Materials Today: Proceedings, Dec. 2020, doi: 10.1016/j.matpr.2020.11.121.
- [74] R. O. Aburukba, M. AliKarrar, T. Landolsi, and K. El-Fakih, "Scheduling Internet of Things requests to minimize latency in hybrid Fog-Cloud computing," Future Generation Computer Systems, vol. 111, pp. 539–551, Oct. 2020, doi: 10.1016/j.future.2019.09.039.
- [75] J. Bisht and V. Subrahmanyam, "Survey on Load Balancing and Scheduling Algorithms in Cloud Integrated Fog Environment," Proceedings of the 2nd International Conference on ICT for Digital, Smart, and Sustainable Development, ICIDSSD 2020, 27-28 February 2020, Jamia Hamdard, New Delhi, India, 2021, doi: 10.4108/eai.27-2-2020.2303123.
- [76] A. Alarifi, F. Abdelsamie, and M. Amoon, "A fault-tolerant aware scheduling method for fog-cloud environments," PLOS ONE, vol. 14, no. 10, p. e0223902, Oct. 2019, doi: 10.1371/journal.pone.0223902.
- [77] F. Alqahtani, M. Amoon, and A. A. Nasr, "Reliable scheduling and load balancing for requests in cloud-fog computing," Peer-to-Peer Networking and Applications, vol. 14, no. 4, pp. 1905–1916, Mar. 2021, doi: 10.1007/s12083-021-01125-2.
- [78] F. Fakhar, "Investigate Network Simulation Tools in Designing and Managing Intelligent Systems," Journal of Information Systems and Telecommunication (JIST), vol. 28, no. 7, pp. 278-293, Jun. 2020, doi: 10.7508/jist.2019.04.004.
- [79] E. M. Khaneghah, S. L. Mirtaheri and M. Sharifi, "Evaluating the Effect of Inter Process Communication Efficiency on High Performance Distributed Scientific Computing," 2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing, Shanghai, China, 2008, pp. 366-372, doi: 10.1109/EUC.2008.11.

An Energy-Aware Approach to Virtual Machine Consolidation Using Classification and the Dragonfly Algorithm in Cloud Data Centers

Nastaran Evaznia^{1*}, Reza Ebrahimi¹, Davoud Bahrepor ^{1,2}

¹.Department of Computer Engineering, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

².Department of Cybersecurity and Cyberspace, Intelligent Financial Innovation Research Center, Mashhad Branch, Islamic Azad University, Mashhad, Iran.

Received: 16 Sep 2024/ Revised: 01 Dec 2024/ Accepted: 11 Feb 2025

Abstract

Nowadays, reducing energy consumption in cloud computing is of great interest due to the high operational costs and its impact on climate change. The consolidation solution is an effective method for minimizing the number of physical machines (PMs) and reducing energy consumption. The virtual machine consolidation process encounters the challenge of reducing energy consumption while effectively managing resource allocation. The aim of this paper is to address these challenges through the classification of PMs and the use of the dragonfly algorithm. The quartile parameter is utilized to classify PMs into three categories: underloaded, medium load, and overloaded. First, we identified the overloaded PMs in the overloaded category. Then, we presented a solution to select virtual machines from an overloaded PM based on resource usage. Additionally, the Dragonfly algorithm is utilized to select destinations for hosting migrant virtual machines in the medium load category. Furthermore, we identified underloaded PMs in the underloaded categories using this algorithm. The proposed solution is evaluated using the CloudSim toolkit and tested with workloads consisting of over a thousand data points from virtual machines based on PlanetLab data. The results from the simulation experiments indicate that the proposed solution, while avoiding SLA violations and minimizing additional migrations, has significantly reduced energy consumption.

Keywords: Cloud Computing; Consolidation; Quartile Parameter; Dragonfly Algorithm; SLA Violations; Migrations; Energy Consumption.

1- Introduction

Data centers in cloud computing [1-4] are physical spaces where hosts and necessary equipment are stored and accessed via the Internet to provide better services. Increasing energy consumption in cloud infrastructure can lead to higher carbon dioxide emissions and elevated operating costs [5-7]. With the advancement of virtualization, reducing energy consumption has become an important and challenging issue in the design of new systems [8-12]. To address this, a key strategy for reducing the number of active hosts is the virtual machine consolidation process, which enhances resource utilization by strategically migrating virtual machines (VMs) [13, 14]. This approach not only minimizes energy consumption but also strives to prevent violations of Service Level Agreements (SLAs) to the greatest extent possible [14-17].

Despite the significant benefits of this solution, inefficient consolidation can lead to increased costs. Research has shown that a server consumes about 70% of its energy when idle [13]. As a result, if underloaded and overloaded servers are not properly identified, they can contribute to increased migration, energy consumption, and violations of quality of service. Therefore, in this paper, we aim to answer the following question: How can the virtual machine consolidation process be conducted in a way that improves resource management and minimizes costs, while considering energy costs and migration?

Today, meta-heuristic algorithms are recognized as effective methods for solving complex problems and optimizing various fields. Their ability to explore large solution spaces and find optimal outcomes makes them invaluable for addressing diverse challenges [18]. Therefore, this paper presents a combined approach that integrates physical machines (PMs) classification with the Dragonfly meta-heuristic algorithm. In this approach, PMs are first classified using quartile criteria and categorized

✉ **Nastaran Evaznia**

Nastaran_avaznia@yahoo.com; nastaran.evaznia@iau.ir

into underloaded, medium load, and overloaded groups. A quartile divides a dataset into four equal parts, each representing a specific percentage of the data [19]. The proposed classification is based on server CPU usage, as research indicates that CPU performance significantly impacts energy consumption [13]. Consequently, this classification can effectively reduce energy consumption by accurately identifying underloaded, medium load, and overloaded PMs, while also preventing violations of quality of service standards. Therefore, in the proposed solution, we first identify the overloaded PMs within the overloaded category. Next, one or more VMs from these PMs need to be migrated to alleviate the overload condition. A multi-criteria solution based on RAM and CPU usage is proposed to determine which VMs should be migrated, thereby minimizing unnecessary migrations by selecting VMs appropriately. Additionally, a multi-criteria Dragonfly algorithm is utilized for the underloaded and medium load categories to identify the most suitable hosts. The improved Dragonfly algorithm, considering a multi-criteria fitness function, targets hosts with lower energy consumption and greater available resources to meet energy reduction goals. Thus, the main innovations of this paper are summarized as follows:

1. Providing a solution to classify PMs based on quartile parameters.
2. Selecting migrating virtual machines from overloaded hosts based on multiple criteria to avoid excessive migrations.
3. Identifying underloaded and medium-load PMs using the improved Dragonfly algorithm, employing a multi-criteria fitness function to reduce the number of active servers and overall energy consumption.

The structure of the paper is organized as follows: In Section Two, we review and critique related works. Next, the proposed method is introduced. In Section Four, we analyze the proposed method. Finally, in Section Five, we present the conclusion.

2- Related Works

Mustafa et al. [20] proposed an energy-optimal and SLA-aware method in the consolidation process. To achieve this, two consolidation methods are presented to select the destination for hosting migrating VMs based on the Best Fit Decreasing (BFD) method. Simulation results demonstrate improvements in energy efficiency and a reduction in SLA violations. To reduce energy and improve SLA, Dabhi and Thakor [21] addressed the destination selection mechanism for the migration VM allocation. In this framework, the performance of the destination physical machine's processor is evaluated, and hosts with an average load are selected. Furthermore, the results demonstrate the

performance improvements of the proposed approach. Researcher in [22] has presented a virtual machine consolidation algorithm aimed at optimizing the use of VMs to influence the balance between energy consumption and quality of service. This algorithm selects VMs for consolidation based on resource usage. For migration, it employs criteria such as the distance between hosts and the fulfillment of quality of service requirements. The simulation results indicate a better balance between energy consumption and service quality compared to other methods. Khalid et al. [23] focus on energy optimization through virtual machine consolidation. For VM consolidation, they employ mechanisms based on dynamic thresholds and adaptive migration of VMs. The proposed algorithm seeks to balance energy efficiency and performance by identifying overused hosts and relocating VMs to underutilized hosts. The simulation results of this paper demonstrate a reduction in energy consumption while maintaining high-quality services for users in the cloud infrastructure. Ali et al. [24] emphasize the importance of addressing energy consumption and security issues in cloud computing. To achieve this, they utilize particle swarm optimization (PSO) and colony optimization (CO) techniques. The simulation results indicate a reduction in costs and an increase in efficiency. Shaw et al. [24] present a virtual machine consolidation method using a reinforcement learning algorithm. In this paper, the reinforcement learning algorithm for the consolidation problem represents resource capacity to optimize the distribution of VMs, thereby improving resource management. The experimental results show that avoiding violations of the service level agreement enhances energy efficiency. Researchers have introduced the Modified Bird Feeding Algorithm (ModAFBA) in [25] as a solution for the VM consolidation process, aiming to enhance resource management and efficiency in cloud infrastructure. The simulation results reveal a reduction in energy consumption and the number of migrations, while preventing violations of quality of service. Patel and Bhadka [26] present two computational frameworks for allocation and migration. In this structure, a placement technique is employed to find the best location for each request based on the typical data center configuration of servers. Additionally, a list of VMs is calculated using a power model for migration, targeting those that consume more power. Furthermore, the destination is selected using Dolphin optimization, considering the server with the maximum workload. The experimental results indicate a reduction in energy consumption and the amount of migration. Manikandan and Janani [27] propose a solution that combines hybrid fuzzy and k-means clustering with black widow method optimization and fish swarm optimization for efficient resource allocation. The results of the tests demonstrate a reduction in costs and energy consumption. A summary of the studied methods is presented in Table 1.

Table 1: Summary searches in the area of cloud computing focus on energy awareness

Based on Table 1, the papers are categorized according to the steps of consolidation, clarifying which steps each paper

Method	overloaded PMs	VM selection	Destination selection	Underloaded PM
Mustafa et al. [20]	×	×	✓	×
Dabhi and Thakor [21]	✓	×	✓	✓
Kumaran et al. [22]	×	✓	×	×
Khalid et al. [23]	✓	✓	×	✓
Ali et al. [24]	×	×	✓	×
Shaw and Barrett [28]	✓	×	✓	×
Alsadie and Alsulami [25]	×	✓	✓	×
Patel and Bhadka [26]	✓	✓	✓	×
Manikand an and Janani [27]	×	×	✓	×
Proposed Method	✓	✓	✓	✓

focuses on. The proposed method, which classifies PMs and appropriately categorizes them while utilizing the multi-criteria Dragonfly algorithm, offers an effective solution for optimal resource management at each consolidation stage.

3- Proposed Method

The consolidation strategy occurs in four stages. The first step is to identify the overloaded PMs. Next, if a PM is overloaded, one or more VMs must be migrated from that host to avoid SLA violations. In the third stage, the strategy focuses on finding the destination to host the migrating virtual machines. Finally, it identifies underloaded PMs to shut down [15, 17]. In the proposed method, the quartile parameter is used to categorize PMs within the cloud infrastructure, and the Dragonfly algorithm is employed to enhance mapping and reduce the number of PMs. First, the PMs are sorted by CPU usage, as CPU usage affects the energy consumption of PMs [13, 22]. Then, the first (Q1), second (Q2), and third (Q3) quartiles are calculated based on this. The PMs are divided into three categories:

underloaded, medium load, and overloaded PMs based on the quartile parameter. Fig. 1 illustrates this classification of PMs.

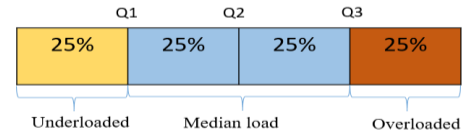


Fig. 1 Classification of PMs

According to Fig. 1, a PM whose CPU usage is less than Q1 is categorized as underloaded, while a PM whose CPU usage is in the range of Q2 is categorized as medium load. A PM with CPU usage greater than Q3 is classified as overloaded. First, we identify the overloaded PMs.

3-1- Identification of Overloaded Physical Machines

According to the consolidation steps, the first step in the consolidation phase is to identify overloaded PMs. Researchers in [29] proposed four suitable approaches to find dynamic thresholds for detecting overloaded PMs. Compared to external models, the Median Absolute Deviation (MAD) method is robust. In this phase, the MAD is used to identify overloaded PMs among those in the overloaded category. Eq. (1) provides this metric.

$$\text{If } (PM_i^{CPU} > Q3) \quad i=1, 2, \dots, N \quad (1)$$

$$\{ T_u = 1-s \cdot OC_MAD$$

In Eq. (1), PM_i^{CPU} is CPU usage of PM_i , T_u is the upper threshold, $s \in \mathbb{R}^+$, and OC_MAD is MAD in the overloaded category. N is the number of PMs.

MAD parameter uses previous knowledge to generate a new threshold value. To obtain a MAD value, it is necessary to use univariate data X_1, X_2, \dots, X_n . Eq. (2) expresses this criterion [20, 29].

$$MAD = \text{median}(|X_i - \text{median}_j(X_j)|) \quad (2)$$

Hence, if the host's CPU usage in the overloaded category is greater than T_u , that host is considered to be overloaded. In the event that the PM is overloaded, it would be necessary to migrate several VMs from that PM to prevent service quality violations. It is assumed that cloud centers include an N number of PMs and a V number of VMs. In the next step, we check what virtual machine to choose for migration from the overloaded PM.

3-2- Selection of Migrant Virtual Machines

In the previous works [20, 29], the Minimum Migration Time (MMT) policy is used to choose a VM to migrate from a host that was overloaded. The virtual machine has been selected for migration under this policy due to its reduced memory usage. In addition, merely one criterion is considered. In this policy, the amount of CPU used by the VM, which might be influential in overloading and increasing the energy consumption of the physical machine, has not been taken into account. In the proposed solution, the minimum memory maximum processor method has been presented, which aims to combine the MMT policy and the use of the virtual machine processor by considering several criteria. The purpose of presenting the desired method is to choose a virtual machine that has the lowest migration time compared to other virtual machines and uses more processors than the other virtual machines; therefore, we can reduce the number of additional migrations with this selection. Eq. (3) provides this criterion.

$$v \in V_i | \forall \alpha \in V_i, \frac{RAM_u(v)}{UtilizationOfCpu_u(v)} \leq \frac{RAM_u(\alpha)}{UtilizationOfCpu_u(\alpha)} \quad (3)$$

According to Eq. (3), $RAM_u(\alpha)$ is the recently used amount of RAM by virtual machine α . $UtilizationOfCpu_u(\alpha)$ is the value of the recently used processor by virtual machine α . V_i is a set of VMs that have been recently allocated to the host $_i$. In the proposed solution, a virtual machine with a lower ratio than that of other VMs is selected as the designated virtual machine for migration from the overloaded host. After the migration, this criterion is applied again to select the next virtual machine if the overloaded host's performance remains above the threshold.

3-3- Selection of the Destination Host

The next step in the consolidation phase, after selecting the migration VMs, is to choose a destination for hosting these migrated virtual machines.

After identifying the overloaded hosts and migrating the necessary virtual machines, the migrated virtual machines should be transferred to a destination with the required capacity and cost-effective efficiency. To select the destination for hosting the migrating virtual machines, the PM is chosen based on the proposed Dragonfly algorithm from the PMs in the medium load category. PMs in the medium load category have CPU utilization greater than Q1 and less than Q3 according to Eq. (4).

$$\begin{aligned} & \text{If } (PM_i^{CPU} > Q1 \text{ and } PM_i^{CPU} < Q3) \\ & \quad i=1, 2, \dots, N \\ & \Rightarrow \text{find PMs based on Dragonfly} \\ & \quad \text{algorithm} \end{aligned} \quad (4)$$

In Eq. (4), $Q1$ is the first quarter, and $Q3$ is the third quarter. PM_i^{CPU} shows the CPU usage of the PM_i .

The reason for choosing this category is that it mitigates the risk of overloading the PMs in the future while also avoiding classification in the underloaded category, which could prevent the shutdown of that host later. Furthermore, in the proposed Dragonfly algorithm, a multi-criteria fitness function is employed to identify the best host for allocation within this category.

In general, due to its high speed, accuracy, and capabilities, the Dragonfly algorithm [30] has been utilized alongside a multi-criteria fitness function. As a result, the steps of the proposed method are as follows:

Step 1: The Dragonfly population and food sources, along with their characteristics, are quantified. In modeling the proposed solution using the Dragonfly algorithm, dragonflies represent VMs that search for prey (PMs). PMs possess characteristics such as processors, memory, network, and bandwidth. Consequently, these resources are considered within a broad set of constraints and can be represented by three parameters: CPU, RAM, and bandwidth (BW). Therefore, if PM_i represents the i -th physical machine, the available capacity of this PM (AC (PM $_i$)) is expressed in Eq. (5).

$$AC (PM_i) = \{CPU_i, RAM_i, BW_i\} \quad (5)$$

Step 2: In this step, the fitness function for all PMs (resources) is calculated. By defining a multi-criteria fitness function and evaluating several criteria, we aim to select a host for allocation that possesses the necessary resources for hosting while avoiding the risk of overloading and violating the quality of service. Consequently, among the PMs, those that are not overloaded and meet the necessary resource requirements for hosting the virtual machine can be selected based on the proposed function. The limit is calculated using Eq. (6) and Eq. (7), respectively.

$$RR(VM_i) < AC (PM_i) \quad i=1, 2, \dots, N \quad (6)$$

In Eq. (6), the $RR(VM_i)$ indicates the resources required by the virtual machine, while $AC (PM_i)$ represents the available capacity of the i -th PM to prevent overloading. Consequently, the fitness function (FF) is calculated using Eq. (7).

$$FF = \text{MIN} \left(\frac{Energy (PM_i)}{AC (PM_i)} \right) \quad (7)$$

According to the FF in Eq. (7), Energy (PM_i) represents the amount of energy consumed by PM_i , and AC (PM_i) is the capacity of the available resources of the PM. Based on the fitness function, the lower the energy consumption ratio of the host and the capacity of the available resources (RAM, CPU, and BW), the more suitable the position would be for the prey in the proposed Dragonfly algorithm (i.e., more suitable in the proposed host method). Therefore, in this structure, the host where the fitness function criterion is minimized compared to other PMs (positions) would be the most suitable host for the destination selection process for migrant dragonflies (migrant virtual machines).

Step 3: Update the optimal position (based on the fitness function). The position of the prey (PM) is updated for all migrant dragonflies (migrant virtual machines).

Step 4: Check the termination conditions. If the immigrant Dragonfly (representing the immigrant virtual machine) is not yet finished, repeat the process from the second step to select the host. Otherwise, if all dragonflies have been mapped, the termination condition is satisfied.

The pseudocode at this stage of the consolidation process is shown in Fig. 2.

```

Input: pmList, vmList   Output: allocated Host
1. For each VM in vmList do
2.   Allocated Host ← NULL;
3.   MinFitness ← MAX;
4.   For host in HostList do
5.     If utilization of cpu( $PM_i$ ) > Q1 and utilization of cpu( $PM_i$ ) < Q3
6.       Energy ( $PM_i$ ) ← energy (host, VM);
7.       FF ←  $\frac{\text{Energy} (PM_i)}{\text{AC} (PM_i)}$ 
8.       If fitness < MinFitness then
9.         MinFitness ← fitness;
10.        AllocatedHost ← host;
11.      end if
12.    end if
13.  end for
14. return allocated Host;

```

Fig. 2 Destination PMs selection pseudocode

3-4- Identification of the Underloaded Physical Machines

To identify underloaded PMs, among the PMs whose CPU usage is less than the first quartile (Q1), the PM with the lowest fitness function metric in the Dragonfly algorithm is considered an underloaded PM for shutdown. Eq. (8) provides this feature. In Eq. (8), PM_i^{CPU} is CPU usage of PM_i , and N is the number of PMs.

$$\text{If } (PM_i^{CPU} < Q1) \quad i=1, 2, \dots, N \quad (8)$$

\Rightarrow find PMs based on fitness function in Dragonfly algorithm

The dynamic VM consolidation process periodically migrates and reallocates VMs to PMs [31]. Therefore, the proposed method periodically addresses the issue of migrating VMs and reallocating them to medium load PMs, as well as putting underloaded PMs to sleep.

In the following, the flowchart diagram to express the proposed method is presented in Fig. 3.

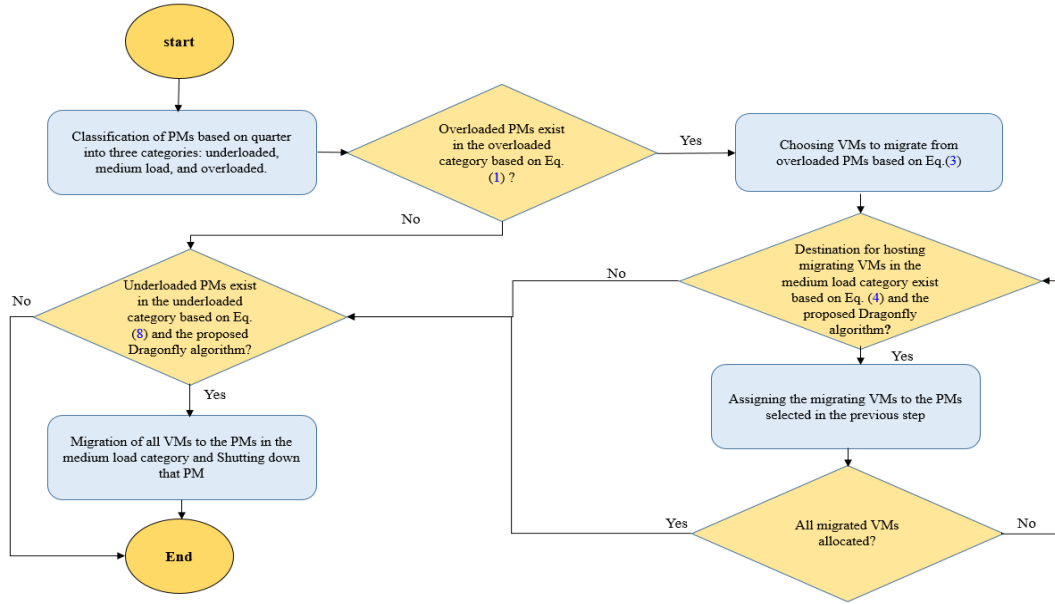


Fig. 3 Flowchart of the proposed method

According to Fig. 3, the steps of the proposed method include classifying PMs based on quartiles and applying the dragonfly algorithm in the consolidation process. First, the PMs are divided into three categories according to the quartile parameter. In the first step of consolidation, in the overloaded category (hosts whose CPU usage exceeds Q3), overloaded hosts are identified based on MAD [20]. If an overloaded host is found, it selects VMs to migrate from that host based on Eq. (3). After selecting the migrating VMs, the destination host is chosen from the medium load category using the Dragonfly algorithm. Based on the provided fitness function, the host with the minimum fitness value is selected for allocation. If no overloaded PMs are identified the method then looks for underloaded PMs in the underloaded category (hosts whose CPU utilization is below Q1 and have the minimum fitness function value in the Dragonfly algorithm for this category). This process continues until all VMs are allocated.

4- Analysis

Evaluating the proposed method in large-scale cloud data center infrastructures and real-world environments can be challenging. Therefore, to ensure the repeatability of the experiment, the CloudSim toolkit [32] has been chosen as

the simulation platform. To assess the proposed method, we used real-world workloads for a more accurate evaluation. The workload utilized is based on actual data generated by PlanetLab. Using this tool, a data center with 800 heterogeneous physical nodes was simulated. In this test environment, half of the servers are HP ProLiant ML110 G4, while the other half are HP ProLiant ML110 G5. The specifications of these servers are provided in Table 2, according to the results of the SPECpower benchmark¹.

Table 2: Specifications of physical machines

<i>Physical machine</i>	<i>Bandwidth (Gbit/s)</i>	<i>The number of cores</i>	<i>RAM (MB)</i>	<i>CPU (MIPS)</i>
HPProLiant ML110 G4	1	2	4096	1860
HP ProLiant ML110 G5	1	2	4096	2660

Each category of PMs in Table 2 has different processing speeds, memory capacities, number of cores, and bandwidths. Additionally, the specifications of the VMs used are based on real Amazon EC2 examples². In this architecture, all VMs in the dataset are single-core.

¹ http://www.spec.org/power_ssj2008

² <http://aws.amazon.com/ec2/instance-types/>

The specifications and features of the virtual machines used for evaluation are provided in Table 3.

Table 3: Specifications of virtual machines

<i>virtual machine</i>	<i>Micro instance</i>	<i>Small instance</i>	<i>Extra-large instance</i>
CPU (MIPS)	500	1000	2000
RAM(MB)	613	1700	3750
Bandwidth (Gbit/s)	1	1	1
Size (GB)	2.5	2.5	2.5

In Table 3, the virtual machines (VMs) differ in terms of processing speed and memory capacity. The workload in the presented method is based on real data obtained from PlanetLab over ten different days. This data reflects CPU usage by more than 1,000 VMs simultaneously from servers located in over 500 locations. For this purpose, ten days were randomly selected from the workflow data collected between March and April 2011 [33]. The characteristics of the dataset used to evaluate the results are shown in Table 4 [33].

Table 4: Specifications of PlanetLab data

<i>Date</i>	<i>Number of VMs</i>
03/03/2011	1052
06/03/2011	898
09/03/2011	1061
22/03/2011	1516
25/03/2011	1078
03/04/2011	1463
09/04/2011	1358
11/04/2011	1233
12/04/2011	1054
20/04/2011	1033

The proposed algorithm and the comparison method were coded using NetBeans software and CloudSim version 3, and executed on a 64-bit system with 8 GB of RAM.

4-1- The results of the simulation

The criteria and the parameters considered for evaluating the proposed method are energy efficiency, migrations, and service level agreement violation (SLAV). The energy consumption parameter is based on processor efficiency [20]. Given that processor efficiency changes over time, the energy criterion is defined as a function of

time according to the processor's efficiency, as expressed in Eq. (9) [20, 27].

$$E_i = \int_{t_0}^{t_1} P(u(t_i))dt \tag{9}$$

According to Eq. (9), E_i , the total amount of energy used by the i -th physical machine, is calculated as the integral of energy efficiency over a period from t_0 to t_1 . $u(t_i)$ represents the utilization rate of the i -th physical machine's processor as a function of time. Additionally, the SLAV parameter, which is entirely unfavorable in cloud infrastructure, contributes to increased costs. This criterion depends on two main factors: the state of hosts being overloaded and the occurrence of additional migrations. Specifically, these factors are represented by SLAV Time per Active Host (SLATAH) and Performance Degradation due to Migration (PDM). Consequently, these criteria are examined in Eq. (10) and Eq. (11) [20, 27].

$$SLATAH = \frac{1}{M} \sum_{j=1}^M \frac{T_{si}}{T_{ai}} \tag{10}$$

Let M be the number of hosts, and T_{si} represent the total time that the i -th host experiences 100% utilization, which results in a SLAV. Furthermore, T_{ai} estimates the total time of the i -th PM in an active state. In the following section, the parameter PDM is detailed in Eq. (11) [20, 27].

$$PDM = \frac{1}{N} \sum_{i=1}^N \frac{C_{dj}}{C_{rj}} \tag{11}$$

based on Eq. (11), N indicates the number of v VMs, C_{dj} estimates the efficiency violation of the j -th VM caused by the migration, while C_{rj} represents the total capacity required by the j -th VM during its execution. Considering the equal importance of these two criteria in service quality violations, a combined criterion that accounts for both parameters is utilized for the SLAV measurement. This parameter is presented in Eq. (12) [20, 27].

$$SLAV = SLATAH * PDM \tag{12}$$

The works considered for comparison are the Energy and SLA-Aware VM Placement (ESVMP) [21] and the Blackwidow and Fish Swarm Optimization (BWFSO) [27]. These papers were chosen due to the compatibility of their methods with the simulation environment and their utilization of meta-heuristic algorithms. Fig. 4 displays the results of the energy consumption for the proposed method alongside the compared methods, based on PlanetLab data.

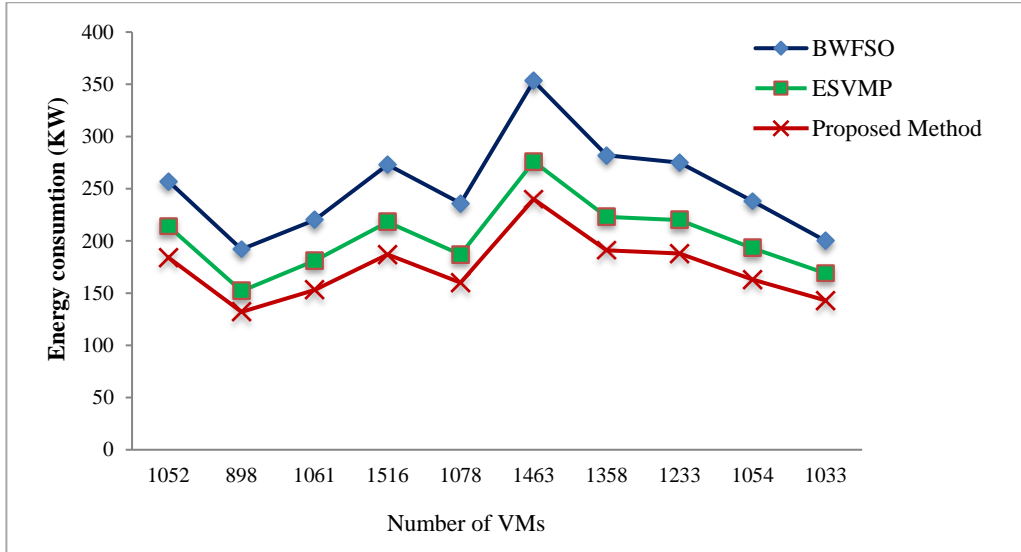


Fig. 4 Comparison of Energy Consumption

In Fig. 4, the amount of energy consumed in kilowatts at different times is displayed on the vertical axis, utilizing PlanetLab data. As illustrated, the energy consumption of the proposed method is significantly lower than that of the compared methods. This reduction is due to the classification of PMs based on the quartile parameter and the use of the Dragonfly algorithm with a multi-criteria objective function

during the consolidation stages. Our solution effectively improves and reduces costs, including energy consumption, by optimizing resource management. Specifically, energy consumption is reduced by 14% compared to ESVMP and 31% compared to BWFSO. Fig. 5 presents the number of migrations for the proposed method compared to other methods.

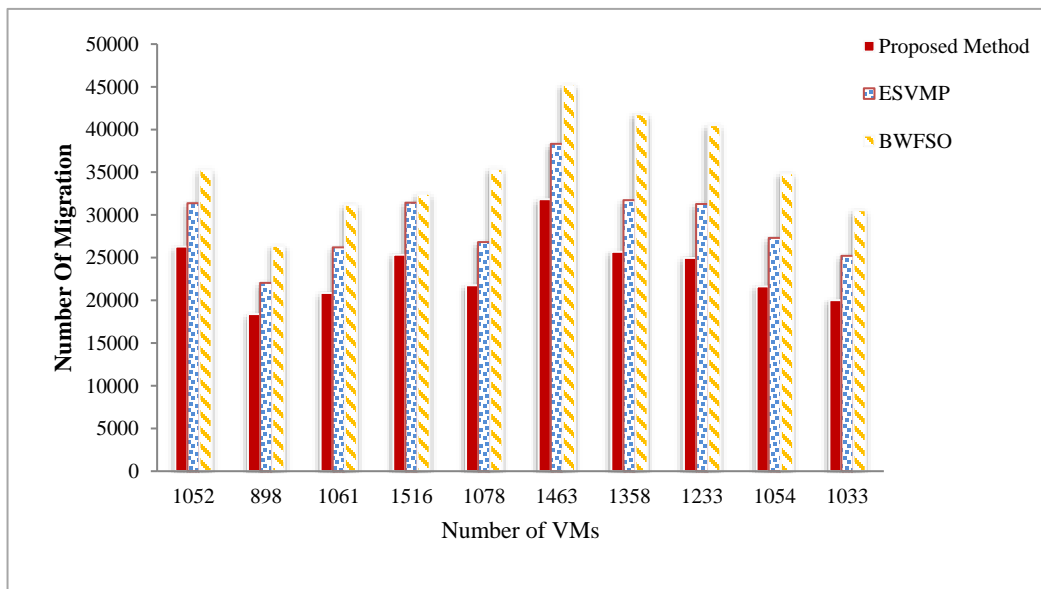


Fig. 5 Comparison of the Number of Migrations.

In Fig. 5, the number of migrations is plotted on the vertical axis using Planet Lab data. As shown, the proposed method results in fewer migrations compared to the baseline papers, with improvements of 19% compared

to ESVMP and 33% compared to BWFSO. Fig. 6 compares the SLAV of the proposed method with ESVMP and BWFSO.

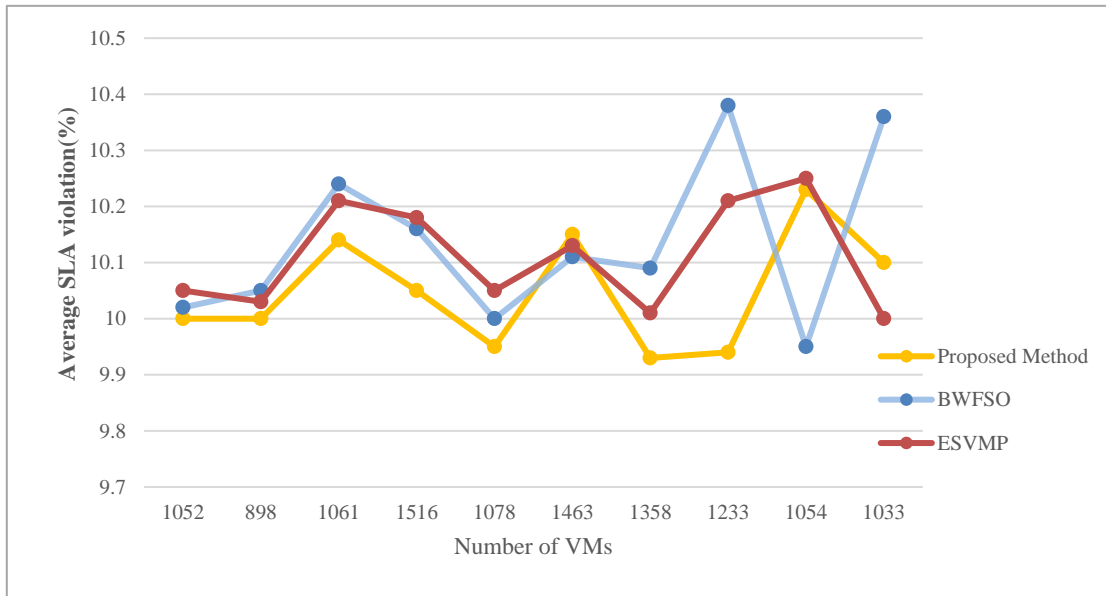


Fig.6 Comparison of the Percentage of SLAV.

In Fig. 6, the average SLAV violation is demonstrated. The results show that, by considering multiple criteria and categorization, the proposed method performs better by reducing energy consumption while avoiding violations of the service level agreement. Additionally, it has improved by 1% compared to ESVMP and 2% compared to BWFSO. The graphs indicate that increasing the number of VMs leads to higher energy consumption and more migrations. However, our proposed method, which categorizes PMs appropriately and utilizes the multi-criteria Dragonfly algorithm, demonstrates improved performance in reducing both energy consumption and the number of migrations, while also preventing an increase in SLAV.

5- Conclusions

In recent years, the growing demand for cloud services has made energy consumption optimization a critical issue. High energy usage in data centers negatively impacts operational costs and the environment. To maximize the benefits of cloud services and reduce expenses, it is essential to minimize energy consumption while adhering to Service Level Agreements. The process of virtual machine consolidation can effectively optimize energy consumption by reducing the number of active physical

machines (PMs) and shutting down idle servers. However, improper consolidation can increase energy usage and negatively affect service quality. To address these challenges, this paper proposes a hybrid solution that combines PMs classification with a meta-heuristic algorithm to optimize energy consumption and manage resources effectively. PMs are categorized based on processor utilization using the quartile parameter, as optimal processor utilization is essential for minimizing energy consumption. By accurately identifying PMs within the appropriate categories, we can achieve improved energy efficiency and more effective resource management. Additionally, by identifying migrating virtual machines based on several criteria, we can prevent unnecessary migrations that increase costs. Furthermore, the use of the Dragonfly algorithm with a multi-criteria fitness function based on energy consumption and available resources helps us find suitable destinations for hosting migrating virtual machines. Finally, we identify underloaded PMs in the underloaded category using the proposed Dragonfly algorithm and take steps to shut them down, thereby reducing energy consumption. The performance of the proposed method has been evaluated using real workloads in the CloudSim simulator. The simulation results demonstrate that, compared to the first and second papers, energy consumption decreased by 14%

relative to ESVM and by 31% compared to BWFSO. Additionally, the total number of migrations was reduced by 19% compared to ESVM and by 33% compared to BWFSO, while the SLAV was decreased by 1% and 2% respectively. For future work, it is recommended to incorporate fog computing into the proposed method to further reduce latency. Moreover, focusing on the healthcare sector and integrating this approach could effectively lower user costs.

Abbreviations

SLA	Service Level Agreement
PMs	Physical Machines
SLAV	Service Level Agreement Violation
BFD	Best Fit Decreasing
ModAFBA	Modified Feeding Birds Algorithm
PM	Physical Machine
MAD	Medium Absolute Deviation
VMs	Virtual Machines
MMT	Minimum Migration Time
SLATAH	SLAV time per active host
PDM	Performance Degradation due to Migration
ESVM	Energy and SLA-aware VM Placement
BWFSO	Black-widow and Fish Swarm Optimization
K-means	K-means refers to data classification with the aim of partitioning n data into k clusters.

References

- [1] T. Alam, "Cloud Computing and Its Role in the Information Technology," *SSRN Electron. J.*, vol. 1, no. 2, 2020, pp. 108–115, doi: 10.2139/ssrn.3639063.
- [2] M. Yenugula, S. Sahoo, and S. Goswami, "Cloud computing for sustainable development: An analysis of environmental, economic and social benefits," *J. Futur. Sustain.*, vol. 4, no. 1, 2024, pp. 59–66, doi: 10.5267/j.jfs.2024.1.005.
- [3] M. M. Sadeeq, N. M. Abdulkareem, S. R. M. Zeebaree, D. M. Ahmed, A. S. Sami, and R. R. Zebari, "IoT and Cloud computing issues, challenges and opportunities: A review," *Qubahan Acad. J.*, vol. 1, no. 2, 2021, pp. 1–7, doi: <https://doi.org/10.48161/qaj.v1n2a36>.
- [4] S. S. Fateminasab, S. Memarian, S. R. K. Tabbakh, and M. C. Romero-Ternero, "A Review on Open Data Storage and Retrieval Techniques in Blockchain-based Applications," in *2024 10th International Conference on Web Research (ICWR)*, IEEE, 2024, pp. 297–302, doi: 10.1109/ICWR61162.2024.10533356.
- [5] W. Yao, Z. Wang, Y. Hou, X. Zhu, X. Li, and Y. Xia, "An energy-efficient load balance strategy based on virtual machine consolidation in cloud environment," *Futur. Gener. Comput. Syst.*, vol. 146, 2023, pp. 222–233, doi: 10.1016/j.future.2023.04.014.
- [6] F. Tashtarian, M. F. Zhani, B. Fatemipour, and D. Yazdani, "CoDeC: A cost-effective and delay-aware SFC deployment," *IEEE Trans. Netw. Serv. Manag.*, vol. 17, no. 2, 2019, pp. 793–806, doi: 10.1109/TNSM.2019.2949753.
- [7] S. Shahryari, F. Tashtarian, and S.-A. Hosseini-Seno, "CoPaM: Cost-aware VM Placement and Migration for Mobile services in Multi-Cloudlet environment: An SDN-based approach," *Comput. Commun.*, vol. 191, 2022, pp. 257–273, doi: 10.1016/j.comcom.2022.05.005.
- [8] R. Shaw, E. Howley, and E. Barrett, "An energy efficient anti-correlated virtual machine placement algorithm using resource usage predictions," *Simul. Model. Pract. Theory*, vol. 93, 2019, pp. 322–342, doi: 10.1016/j.simpat.2018.09.019.
- [9] C. Thiam and F. Thiam, "Energy efficient cloud data center using dynamic virtual machine consolidation algorithm," in *Business Information Systems: 22nd International Conference, BIS 2019, Seville, Spain, June 26–28, 2019, Proceedings, Part I 22*, Springer, 2019, pp. 514–525, doi: 10.1007/978-3-030-20485.
- [10] D. Bahrepour, N. Evaznia, and T. Khodabakhshi, "A New Resource Allocation Method Based on PSO in Cloud Computing," *Int. J. Web Res.*, vol. 7, no. 2, 2024, pp. 13–21, doi: 10.22133/ijwr.2024.457539.1216.
- [11] N. Evaznia and R. Ebrahimi, "Providing a Solution for Optimal Management of Resources using the Multi-objective Crow Search Algorithm in Cloud Data Centers," in *2023 9th International Conference on Web Research (ICWR)*, IEEE, 2023, pp. 179–184, doi: 10.1109/ICWR57742.2023.10139192.
- [12] S. S. F. Nasab, T. Z. Marjaneh, and D. Bahrepour, "Energy Efficiency and Establishing Service Level Agreement using Fuzzification of Virtual Machine Selection Policies for Migrating in Cloud Computing," in *2023 9th International Conference on Web Research (ICWR)*, IEEE, 2023, pp. 201–207, doi: 10.1109/ICWR57742.2023.10138982.
- [13] A. Varasteh, F. Tashtarian, and M. Goudarzi, "On reliability-aware server consolidation in cloud datacenters," in *2017 16th International Symposium on Parallel and Distributed Computing (ISPDC)*, IEEE, 2017, pp. 95–101, doi: 10.1109/ISPDC.2017.26.
- [14] L. Helali and M. N. Omri, "A survey of data center consolidation in cloud computing systems," *Comput. Sci. Rev.*, vol. 39, 2021, p. 100366, doi: 10.1016/j.cosrev.2021.100366.
- [15] R. Zolfaghari and A. M. Rahmani, "Virtual machine consolidation in cloud computing systems: Challenges and future trends," *Wirel. Pers. Commun.*, vol. 115, no. 3, 2020, pp. 2289–2326, doi: 10.1007/s11277-020-07682-8.
- [16] M.-H. Malekloo, N. Kara, and M. El Barachi, "An energy efficient and SLA compliant approach for resource allocation and consolidation in cloud computing environments," *Sustain. Comput. Informatics Syst.*, vol. 17, 2018, pp. 9–24, doi: 10.1016/j.suscom.2018.02.001.

- [17] U. Arshad, M. Aleem, G. Srivastava, and J. C.-W. Lin, "Utilizing power consumption and SLA violations using dynamic VM consolidation in cloud data centers," *Renew. Sustain. Energy Rev.*, vol. 167, 2022, p. 112782, doi: 10.1016/j.rser.2022.112782.
- [18] H. F. Farimani, D. Bahrepour, S. R. K. Tabbakh, and R. Ghaemi, "A new meta-heuristic algorithm: Artificial Yellow Ground Squirrel (YGSA)," 2022, doi: 10.21203/rs.3.rs-1909482/v1.
- [19] H. F. Farimani, D. Bahrepour, and S. R. K. Tabbakh, "Reallocation of virtual machines to cloud data centers to reduce service level agreement violation and energy consumption using the FMT method," *J. Inf. Syst. Telecommun.*, vol. 4, no. 28, 2020, p. 316.
- [20] S. Mustafa, K. Bilal, S. U. R. Malik, and S. A. Madani, "SLA-aware energy efficient resource management for cloud environments," *IEEE Access*, vol. 6, 2018, pp. 15004–15020, doi: 10.1109/ACCESS.2018.2808320.
- [21] D. Dabhi and D. Thakor, "Energy and SLA-Aware VM Placement Policy for VM Consolidation Process in Cloud Data Centers," in *Sustainable Technology and Advanced Computing in Electrical Engineering: Proceedings of ICSTACE 2021*, Springer, 2022, pp. 351–365, doi: 0.1007/978-981-19-4364-5_26.
- [22] K. M., "Energy-Aware Virtual Machine Consolidation Algorithm for Enhanced QoS in Data Centers," *Int. Sci. J. Eng. Manag.*, vol. 03, no. 04, 2024, pp. 1–9, doi: 10.55041/ISJEM01696.
- [23] U. Khalid, S. Ahmad, B. Chang, M. Nisar, J. Cha, and E. Munir, "Energy Optimization in Cloud Computing Environment through Virtual Machine Consolidation." 2023. doi: 10.21203/rs.3.rs-3284176/v1.
- [24] A. Ali and T. T. Tin, "Unleashing the Power of Consolidate Cloud Computing: Secure and Energy-Efficient Virtual Machines at Your Service," 2023, doi: 10.21203/rs.3.rs-3133236/v1.
- [25] D. Alsadie and M. Alsulami, "Efficient Resource Management in Cloud Environments: A Modified Feeding Birds Algorithm for VM Consolidation," *Mathematics*, vol. 12, no. 12, 2024, p. 1845, doi: 10.3390/math12121845.
- [26] R. P. Patel and H. B. Bhadka, "Energy-Aware VMs Consolidation Computing Frameworks' of Data Center in Cloud Computing Environment," *J. Sci. Technol.*, vol. 7, no. 1, 2022, pp. 82–91.
- [27] N. Manikandan, P. Divya, and S. Janani, "BWFSO: hybrid Black-widow and Fish swarm optimization Algorithm for resource allocation and task scheduling in cloud computing," *Mater. Today Proc.*, vol. 62, 2022, pp. 4903–4908, doi: 10.1016/j.matpr.2022.03.535.
- [28] R. Shaw, E. Howley, and E. Barrett, "Applying reinforcement learning towards automating energy efficient virtual machine consolidation in cloud data centers," *Inf. Syst.*, vol. 107, 2022, p. 101722, doi: 10.1016/j.is.2021.101722.
- [29] A. Beloglazov and R. Buyya, "Optimal online deterministic algorithms and adaptive heuristics for energy and performance efficient dynamic consolidation of virtual machines in cloud data centers," *Concurr. Comput. Pract. Exp.*, vol. 24, no. 13, 2012, pp. 1397–1420, doi: 10.1002/cpe.1867.
- [30] Ç. İ. Acı and H. Gülcan, "A modified dragonfly optimization algorithm for single-and multiobjective problems using Brownian motion," *Comput. Intell. Neurosci.*, vol. 2019, no. 1, 2019, p. 6871298, doi: 10.1155/2019/6871298.
- [31] H. Xiao, Z. Hu, and K. Li, "Multi-objective VM consolidation based on thresholds and ant colony system in cloud computing," *IEEE Access*, vol. 7, 2019, pp. 53441–53453, doi: 10.1109/ACCESS.2019.2912722.
- [32] R. N. Calheiros, R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R. Buyya, "CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms," *Softw. Pract. Exp.*, vol. 41, no. 1, 2011, pp. 23–50, doi: 10.1002/spe.995.
- [33] K. Park and V. S. Pai, "CoMon: a mostly-scalable monitoring system for PlanetLab," *ACM SIGOPS Oper. Syst. Rev.*, vol. 40, no. 1, 2006, pp. 65–74, doi: 10.1145/1113361.1113374.

Enhancing the Quality of ICT Regulation in Iran: A Study on the Application of the COBIT IT Governance Framework

Ehsan Baraty¹, Akbar Nabiollahi^{2,3*}, Naser Khani¹

¹. Management Department, Najafabad Branch, Islamic Azad University, Najafabad, Iran

². Computer Engineering Faculty, Najafabad Branch, Islamic Azad University, Najafabad, Iran

³. Big Data Research Center, Najafabad Branch, Islamic Azad University, Najafabad, Iran

Received: 08 Feb 2024/ Revised: 04 Jan 2025/ Accepted: 15 Feb 2025

Abstract

This study explores the application of the COBIT (Control Objectives for Information and Related Technologies) IT governance framework to enhance the ICT Regulatory Tracker (ICTRT) scores, a tool developed by the International Telecommunication Union (ITU) to assess ICT regulatory bodies across countries. Given the absence of specific improvement strategies from the ITU, this research fills a critical gap by investigating how COBIT processes can be leveraged for ICT regulation improvement. Utilizing an Automatic Content Analysis (ACA) method, we identified significant relationships between 22 out of 37 COBIT processes and ICTRT indicators, with particular emphasis on APO09, APO11, and DSS02 processes. Focus group methodology employed to validate these findings and development of a continuous improvement plan tailored for Iran's ICT regulatory body. This plan integrates 13 COBIT processes from the identified set, providing a structured approach for implementation. The findings not only highlight effective COBIT processes but also offer actionable insights for regulatory bodies aiming to enhance their regulatory quality and advance towards a digital economy.

Keywords: ICT Regulatory Tracker; COBIT Framework; ICT Regulation Quality; Digital Transformation.

1- Introduction

The digital economy has become an essential element of the global economic landscape, with Information and Communication Technology (ICT) serving as its backbone. ICT is crucial for driving development and facilitating the digital transformation of businesses [1]. An effective regulatory framework is vital for improving business performance and promoting national growth toward sustainable development, enabling digital transformation across various sectors. Consequently, ICT regulators aim to harness digital transformation as a means to achieve the Sustainable Development Goals (SDGs) in collaboration with other sectors [2].

The ITU has analyzed data from ICT regulators in 193 countries and created the ICTRT. This tool helps countries evaluate their regulatory status and formulate policies to enhance ICT regulation, thereby supporting digital transformation and overall development. The ICTRT also

enables countries to customize their regulatory reform strategies based on local and national priorities. Notably, there is a strong correlation between ICTRT scores and the development level of the digital economy; regulatory bodies with higher scores tend to be more effective in attracting investments, fostering technological innovation, and expanding market opportunities [3].

Despite its importance, the ITU has not provided specific guidance on improving ICTRT indicators, leaving individual countries to navigate this challenge independently. This gap presents a significant research opportunity for regulators seeking effective strategies to enhance ICT regulation quality. Our comprehensive review of existing frameworks and literature highlights that the COBIT framework, developed by ISACA¹, offers a robust approach for managing Information Technology (IT) processes while ensuring alignment between IT strategies and business objectives. This framework has been widely studied and implemented

1. ISACA (the Information Systems Audit and Control Association) is a professional association focused on IT governance, risk management, and cybersecurity.

across various organizations, yielding successful outcomes that contribute to process improvements [4].

This paper aims to investigate the role of COBIT processes in enhancing ICTRT indicators through an ACA methodology.

The main contributions of this research include:

- Identifying relationships between ICTRT indicators and COBIT processes.
- Designing a continuous improvement plan for Iran's Communication Regulatory Authority (CRA) based on selected COBIT processes.

The innovation of this research is centered on the following aspects:

- Application of the COBIT IT governance framework to enhance ICTRT scores.
- Utilization of ACA to identify significant relationships between COBIT processes and ICTRT indicators.
- Development of a tailored continuous improvement plan specifically for CRA.

The structure of this paper is as follows: The second section presents a literature review that discusses the tasks and challenges of ICT regulation, elaborates on the ICTRT, and examines the application of COBIT for business process improvement. The third section outlines the research methodology, detailing the ACA process and focus group method used in this study. Finally, sections four and five present the analysis results and summarize the study's objectives, achievements, recommendations for ICT regulators, and suggestions for future research.

2- Related Works and Literature Review

In this section, the literature and related works are explained in three parts. The first part discusses the ICT regulation tasks and challenges. The ICTRT is elaborated on in the second part, and the third part covers the use of the COBIT ITGF for business process improvement.

2-1- ICT Regulation Tasks and Challenges

ICT regulation encompasses various tasks and challenges, including infrastructure management, spectrum regulation, and consumer protection. Regulatory bodies are tasked with fundamental functions such as competition oversight and internet regulation, which can differ significantly based on local conditions. For instance, Yeganeh et al. identified 25 critical measures aimed at enhancing the regulatory quality and balance within Iran's national information network [5].

Spectrum regulation is particularly crucial for national authorities. Olwal et al. examined broadband regulation initiatives in Southern Africa, proposing a framework for dynamic spectrum management to modernize outdated policies. Their findings provide valuable benchmarks for regulatory bodies [6].

The performance of ICT regulators directly impacts service quality. Danbatta and Zangina evaluated Nigeria's

Communications Commission, highlighting key outcomes such as promoting telecommunications research, addressing network vandalism, and developing e-waste management policies to improve service quality. They identified power supply issues as a significant barrier to network reliability [7].

Emerging technologies present new challenges for regulators. Suryanegara recognized 5G as a disruptive innovation, introducing challenges related to security frameworks and economic considerations linked to renewable energy [8]. Similarly, Mohlameane and Ruxwana noted that South Africa's regulatory frameworks inadequately address the complexities of cloud computing, indicating a need for updates [9].

Regulatory bodies must proactively adapt to new technologies, such as smart cities. Barden outlined various challenges faced by regulators in this context, including licensing, data interoperability, and privacy concerns [10]. Additionally, Nguyen assessed the collaboration between state and non-state actors in Vietnam's cyber regulatory framework, evaluating their roles throughout different regulatory periods [11].

2-2- The ICTRT

The ICTRT employs 50 indicators categorized into four domains: regulatory authority, regulatory mandate, regulatory regime, and competition framework, each contributing to a maximum score of 100. Countries are classified into four generations based on their scores, reflecting their regulatory maturity [3].

Research utilizing ICTRT data has been categorized into three main areas:

- 1) **Economic Impact:** Studies such as Raifuet et al. (2023) demonstrate a strong correlation between the quality of ICT regulation and financial development across 23 African nations from 2003 to 2020. This underscores the importance of enhancing ICT regulations for economic growth [12]. Additionally, Nepal's strategic adoption of ICT development strategies illustrates efforts to improve regulatory quality [13].
- 2) **Social Aspects:** Research by Adams and Akobeng on 46 African countries from 1984 to 2018 examines the relationship between ICT and inequality, utilizing indicators such as governance and regulatory quality [14]. Furthermore, Shobande and Ogbeifun link ICT regulation quality with environmental sustainability, revealing how effective regulation can mitigate climate change effects through various indirect mechanisms [15].
- 3) **Regulatory Frameworks:** Chauhan and Mathew analyze India's telecommunications and internet access regulatory environment, highlighting successful policies that support development [16]. Similarly, Nikarya et al. find a significant correlation between

ICTRT indicators and Internet Development Index (IDI) indicators, emphasizing the critical role of regulatory quality in fostering ICT industry growth [17].

Despite these insights, most studies remain descriptive without proposing specific solutions to enhance ICTRT indicators. This gap indicates a need for targeted investigations aimed at developing operational strategies for improving regulatory frameworks.

2-3- Using COBIT ITGF for Business Process Improvement

IT has fundamentally transformed business processes, necessitating a strong alignment between IT strategies and business objectives to mitigate potential disruptions [18], [19]. IT governance plays a crucial role in this alignment, enhancing management, accountability, and compliance while fostering continuous improvement [20].

The COBIT framework is a prominent IT governance framework designed to ensure effective adoption of IT governance practices. It facilitates the mapping and alignment of business and IT goals, thereby supporting organizations in achieving their strategic objectives [1], [19], [21].

Numerous studies have demonstrated the practical application of COBIT across various sectors. For instance, Kahorongo et al. highlighted COBIT's significance in Namibia Bank's efforts to achieve holistic organizational improvement [1]. Similarly, Abu-Musa's research in Saudi Arabia emphasized how COBIT enhances service organizations' understanding and management of IT governance processes, which directly impacts their success metrics [21]. In Kenya's banking sector, Chege et al. found a positive correlation between IT governance maturity and financial performance, underscoring the framework's influence on business outcomes [20].

Organizations have also leveraged COBIT to adopt emerging technologies such as the Internet of Things (IoT). Henriques et al. explored how COBIT facilitates IoT project implementation by identifying key governance enablers, including data privacy and protection measures [22]. Almusawi's study on Iraqi private banks revealed that implementing COBIT enhances the reliability and security of accounting information systems while mitigating audit risks identified by external auditors [23].

The literature indicates that COBIT serves as a strategic model for evaluating ICT performance and can assist regulatory bodies in improving ICT regulations to foster the development of the digital economy [24]. Research suggests that the maturity level of IT governance framework implementation correlates directly with business process performance across industries [20]. Furthermore, COBIT can be utilized as a continuous improvement tool for ICT regulatory bodies to adapt to emerging technologies like IoT and cloud computing [22], [25].

Recent studies have also examined COBIT's application in Enterprise Architecture scenarios, demonstrating its effectiveness in analyzing various organizational contexts, including Iran's telecommunication research center [18]. Overall, the COBIT framework is instrumental for ICT regulatory bodies in making informed investment decisions regarding IT resources and enhancing regulatory quality.

3- Research Method

This study is designed to address two questions:

- 1) How can we measure the relationship between ICTRT indicators and COBIT processes?
- 2) How can we design a continuous improvement plan for the CRA based on selected COBIT processes?

To answer these questions, we used a two-step strategy. First, we employed the ACA methodology to discover the relationships between ICTRT indicators and COBIT processes. Then, we conducted a focus group method to validate the ACA results and design a continuous improvement plan for the CRA based on selected COBIT processes. Each of the two steps is explained below.

3-1- Step 1: The ACA Process

Given that COBIT comprises 37 processes and ICTRT includes 50 indicators, this results in 1,850 potential relationships for analysis. To efficiently analyze these relationships, we employed the ACA method.

To implement the ACA method various key steps have been described in the research literature. First, define the research objectives and questions to guide the analysis. Next, data collection is performed from relevant sources, followed by preprocessing to clean and prepare the data for analysis. Finally, algorithms are applied to analyze the data and identify patterns or themes, leading to the interpretation and reporting of the findings [26], [27] and [28].

According to the literature, the ACA method designed with three steps: data collection, data analysis, and output preparation. Fig. 1 shows these steps that are explained below.

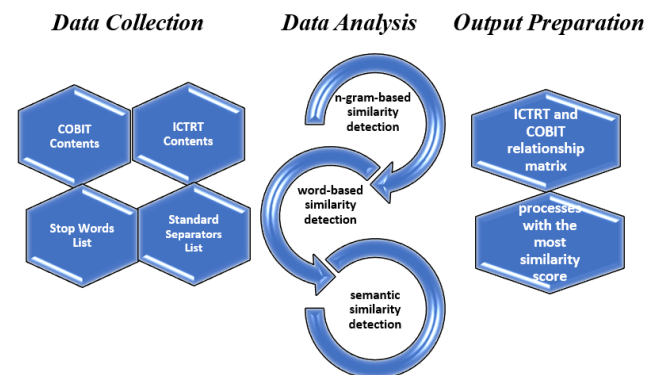


Fig. 1 The ACA method steps

Data Collection:

In this step, the following tasks conducted:

- Text contents regarding ICTRT indicators and COBIT processes were collected from ISACA publications for COBIT¹ and ITU publications for ICTRT², with the selection of content carried out through purposive sampling.
- Stop words list and standard separators for English texts were stored in a database;
- To prepare the texts for the analysis, a set of preprocessing tasks like tokenizing the text contents into words and n-grams, creating text vectors of words and n-grams, stop-words removal, keywords extraction, and semantic expansion of the keywords were performed.

Data Analysis:

In this step, 37 COBIT processes and 50 ICTRT indicators were considered as the ACA categories. Each category was assigned a set of keywords, semantic expanded items, and n-grams, and the relationship between categories was calculated using text similarity detection techniques.

Different methods have been proposed for similarity detection between two contents in ACA research. ACA researchers in similar studies used NLP and Text similarity detection techniques such as terms frequency, Latent Semantic Analysis (LSA), and Hyperspace Analog to Language (HAL) to find major themes in the ACA process; each of them tries to find the most critical concepts in content and aid in rapid understanding of unfamiliar domains and content exploration [29], [30], [31], [32].

In this study, we measured the similarity between categories using three similarity detection methods: n-gram-based similarity detection, word-based similarity detection, and semantic similarity detection. Each relationship was assigned a score between 0 and 1, where 1 indicates the strongest relationship and 0 indicates the weakest relationship. Eq. (1), Eq. (2) and Eq. (3) respectively shows the equations used to measure n-gram-based similarity, word-based similarity, and semantic similarity between categories and the final score for each relationship was calculated by averaging these three similarity scores.

$$\frac{ngramsCosineSimilarity(I_x, P_y)}{|I_x| \times |P_y|} = \frac{\sum_{i=1}^n I_i \times P_i}{\sqrt{\sum_{i=1}^n I_i^2} \times \sqrt{\sum_{i=1}^n P_i^2}} \tag{1}$$

$$\frac{WordsCosineSimilarity(I_x, P_y)}{|I_x| \times |P_y|} = \frac{\sum_{i=1}^n I_i \times P_i}{\sqrt{\sum_{i=1}^n I_i^2} \times \sqrt{\sum_{i=1}^n P_i^2}} \tag{2}$$

1. Selected parts of the COBIT supplemental tools and materials and the COBIT 5 toolkit documents were used.

2. Selected parts of the ITU GSR (Global Symposium for Regulators) and Global ICT Regulatory Outlook (GIRO) Reports were used.

$$SemanticSimilarity(I, P) = \frac{Sim(IW, PW_1) + Sim(IW, PW_2) + Sim(PW, IW_1) + Sim(PW, IW_2)}{4} \tag{3}$$

Output Preparation:

In this step, the relationship matrix was prepared. **Error! Reference source not found.** shows the relationship matrix schema. In this matrix, rows are ICTRT indicators (named I1, I2, ..., I50) and columns are COBIT processes (named P1, P2, ..., P37), while each cell shows the relationship score between related indicators and related processes.

Table 1: The relationship matrix schema

	P1	P2	P3	...	P37
I1	0.08	0.62	0.17	...	0.91
I2	0.10	0.14	0.09	...	0.14
I3	0.07	0.21	0.06	...	0.16
I4	0.30	0.13	0.08	...	0.30
I5	0.10	0.14	0.14	...	0.30
I6	0.32	0.03	0.19	...	0.87
I7	0.08	0.71	0.10	...	0.13
I8	0.23	0.49	0.13	...	0.13
...
I50	0.11	0.17	0.14	...	0.28

In this step, also, the list of processes with the most similarity score was prepared. Table 2 shows these processes.

Table 2: Top 10 discovered relationship scores

Rank	ICTRT Indicator ID	COBIT Process Name	Relationship Score
1	11	APO09	0.909
2	16	APO09	0.872
3	26	APO11	0.844
4	47	APO09	0.689
5	50	DSS02	0.68
6	48	APO09	0.679
7	49	APO09	0.679
8	37	DSS05	0.677
9	38	DSS05	0.653
10	20	APO01	0.634

3-2- Step 2: The Focus Group Process

The focus group method involves several key stages in the literature. First, researchers define clear objectives and questions to guide the discussions. Next, participants are

selected based on specific criteria relevant to the research topic, ensuring a diverse range of insights. A structured discussion guide is then created to facilitate the session, which is conducted by a trained moderator in a comfortable environment, typically lasting 60 to 120 minutes. After recording and transcribing the sessions for analysis, thematic analysis is performed to identify patterns and insights [33], [34], [35].

In this study, the focus group method employed with the following objectives:

- To assess the accuracy and validity of the relationship matrix.
- To develop a continuous improvement plan for the CRA using COBIT processes.

After defining the research objectives, six participants were selected from experts within the CRA. In the selection process, emphasis was placed on expertise relevant to ICT regulation, encompassing areas such as IT, mobile communications, fixed communications, postal services, finance, administration, legal affairs, and management. This diverse selection aimed to ensure a comprehensive understanding of the various aspects of ICT regulation during the focus group discussions.

Finally, we conducted a two-hour focus group meeting to discuss our objectives and systematically review each topic with the participation of experts in the field. During these discussions, we randomly assessed and confirmed the validity and reliability of the identified relationships between ICTRT indicators and COBIT processes.

Additionally, we examined several reports derived from the relationship matrix, including those highlighting the highest-scoring relationships and the domains and processes that significantly impact ICTRT indicators.

We also reviewed the CRA's latest status in the ITU annual assessments and prepared a prioritized COBIT processes list for the CRA enhancement and based on this list, a plan for continuous improvement of regulation quality in four steps with a cyclic strategy was developed that is shown in Fig. 2.

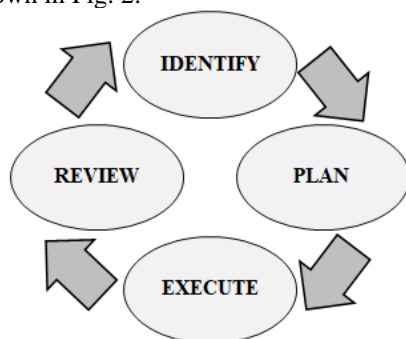


Fig. 2 The CRA continuous improvement plan

Step 1- Identify: In this step, in addition to the prioritized processes list, new organizational and environmental status and the feedback from the last cycle are identified.

Step 2- Plan: In this step, first the prioritized processes list is updated using the identified data and the focus group method; then, the top one/multiple process in the processes list is/are selected to be executed.

Step 3- Execute: In this step, the selected process/processes is/are implemented.

Step 4- Review: In this step, the performance and improvements in ICTRT indicators and regulation quality are evaluated and the feedback is applied to the next cycle. The presented continuous improvement plan enables the CRA to make key decisions to improve ICT regulation quality, achieve higher scores in ICTRT, and facilitate digital transformation and sustainable development in different sections of the economy in a step-by-step program.

4- Discussion

In the previous section, the methodology of the study was outlined in two parts: the ACA process and the focus group process. This section will discuss the results of the conducted study in four parts:

- Analyzing the highest relationship scores.
- Analyzing the most effective COBIT processes.
- Analyzing the most effective process domains of the COBIT.
- Analyzing the CRA continuous improvement plan.

4-1- Analyzing the Highest Relationship Scores

Table 2 shows the top 10 relationships with the highest scores within the relationship matrix. There are noteworthy comments about this list that are explained below:

- Five relationships out of 10, related to the APO09 process, that could help ICT regulators to improve indicators 11, 16, 47, 48, and 49. The implementation of the APO09 process named "Manage Service Agreements" could play a crucial role in enabling regulatory bodies to maintain oversight and ensure that service providers meet the required standards in the rapidly evolving technologies. It also could help ICT regulators to establish service standards, manage service level agreements, performance monitoring, risk identification and management, and continuous improvement.
- Two relationships out of 10, related to the DSS05 process, that could help ICT regulators to improve indicators 37 and 38. This process named "Manage Service Requests and Incidents" enables ICT regulators to manage incidents efficiently, handle service requests from operators, end users, and other stakeholders, and manage documentation and reporting tasks. It also

supports ICT regulatory bodies in maintaining effective governance and ensuring that IT services meet both operational needs and regulatory requirements.

- Three remaining processes in the list are APO11, DSS02, and APO01 which have strong effects on ICTRT indicators 26, 50, and 20 respectively.

4-2- Analyzing the Most Effective COBIT Processes

In the relationship matrix, the average of 50 scores associated with each COBIT process effectively represents the overall impact of that process on enhancing ICTRT indicators. This indicates that the implementation of that process will play a more significant role in improving the overall ICTRT indicators compared to other processes. Table 3 shows the top 5 processes with the highest correlation with all indicators.

Table 3: Top 5 processes with the highest correlation with all indicators

Rank	Process Name	Process Title
1	DSS02	Manage Service Requests and Incidents
2	APO09	Manage Service Agreements
3	DSS01	Manage Operations
4	DSS05	Manage Security Services
5	EDM01	Ensure Governance Framework Setting and Maintenance

4-3- Analyzing the Most Effective Process Domains of the COBIT

In this section, we will present a comprehensive and holistic overview of the effect of COBIT processes on improving ICTRT indicators, which is summarized in Table 4. As we see, in the first row of the table there are five processes in the EDM domain within the governance key area. Out of these five processes, two (40%) have a significant impact on four ICTRT indicators, which represents 8% of the total 50 indicators, and the other rows of the table can be interpreted in the same way, too.

Table 4: COBIT key areas and process domains

Key area	Domain name	Num. of processes	Num. of effective processes (%)	Num. of affected indicators (%)
Governance	EDM	5	2 (%40)	4 (%8)
Management	APO	13	9 (%69)	24 (%48)
	BAI	10	5 (%50)	8 (%16)
	DSS	6	4 (%67)	11 (%22)
	MEA	3	2 (%67)	3 (%6)
Total:		37	22 (%59)	50

This table provides an overview of the relationship between the COBIT framework and the ICTRT, offering insights that can be valuable for regulators in the ICT sector when designing a continuous improvement plan. For example, as indicated in the table, a total of 22 unique processes out of the 37 COBIT processes contribute to the improvement of the 50 ICTRT indicators. This suggests that regulators should prioritize implementing processes from this set of 22 to achieve maximum improvement in ICTRT indicators. Additionally, the table highlights that the processes within the APO, DSS, and BAI domains play the most significant roles in improving ICTRT indicators, which should be a focal point for ICT regulators.

4-4- Analyzing the Continuous Improvement Plan of the CRA

This section analyzes the continuous improvement plan prepared for the CRA.

Analyzing the process list for the CRA improvement: According to the latest ICTRT report, the CRA achieved a score of 86 out of 100 and was classified among G4 group countries. Table 5 presents the details of CRA's most recent scores. As indicated in this table, the CRA received scores of 20 (100%) in the Regulatory Authority dimension, 19 (86%) in the Regulatory Mandate dimension, 28 (93%) in the Regulatory Regime dimension, and 19 (68%) in the Competition Framework dimension. Conversely, the CRA lost scores of 0 (0%), 3 (14%), 2 (7%), and 9 (32%) in the same dimensions respectively. This indicates that the CRA should focus more on improving indicators within the Competition Framework domain.

Table 5: The CRA's latest status in the ICTRT

ICTRT Dimension	Number of indicators (%)	The CRA achieved scores (%)	The CRA lost scores (%)
Regulatory Authority	10 (%20)	20/20(%100)	0/20 (%0)
Regulatory Mandate	11 (%22)	19/22(%86)	3/22 (%14)
Regulatory Regime	15 (%30)	28/30(%93)	2/30 (%7)
Competition Framework	14 (%28)	19/28(%68)	9/28 (%32)
Total:	50	86/100 (%86)	14/100(%14)

According to the ITU report, the CRA needs to enhance 12 indicators. Table 6 presents a list of these 12 indicators along with the top three COBIT processes that could contribute to their improvement. For instance, in the first row, the processes BAI07, DSS04, and APO04, with correlation scores of 0.204, 0.187, and 0.185 respectively, can be utilized to improve the indicator "New Mandate: Entity in Charge of Broadcasting (Radio and TV Transmission)."

Table 6: Process list to improve 12 indicators for the CRA

Indicator needed to be improved	Top 3 effective processes	score
New mandate: entity in charge of broadcasting (radio and TV transmission)	BAI07	0.204
	DSS04	0.187
	APO04	0.185
New mandate: entity in charge of broadcasting content	BAI04	0.298
	DSS04	0.218
	APO11	0.179
New mandate: entity in charge of Internet content	BAI04	0.298
	DSS04	0.207
	MEA02	0.188
Number portability available to consumers and required from fixed-line operators	DSS06	0.493
	DSS01	0.393
	DSS04	0.360
Level of competition in IMT (3G, 4G, etc.) services	DSS05	0.653
	EDM01	0.231
	APO01	0.210
Level of competition in International Gateways	EDM01	0.271
	APO01	0.210
	MEA02	0.164
Status of the main fixed line operator	APO01	0.294
	EDM01	0.274
	DSS06	0.253
Foreign participation/ownership in facilities-based operators	APO01	0.217
	EDM01	0.205
	DSS01	0.197
Foreign participation/ownership in spectrum-based operators	APO01	0.217
	EDM01	0.208
	DSS01	0.197
Foreign participation/ownership in local service operators/long-distance service operators	APO09	0.689
	DSS02	0.677
	DSS01	0.398
Foreign participation/ownership in international service operators	APO09	0.679
	DSS02	0.668
	DSS01	0.398
Foreign participation/ownership in Internet Service Providers (ISPs)	APO09	0.679
	DSS02	0.677
	DSS01	0.325

In Table 6, 36 processes are listed to improve 12 indicators. Some processes appear multiple times, indicating that they play more significant role in enhancing various indicators. Since the CRA requires a concise and prioritized list of processes, an aggregation of data in Table 6 resulted in a shortlist of 13 processes, which is displayed in Table 7. This table includes the number of indicators that each process can improve, along with the average correlation scores for all indicators.

Table 7: Process list for the CRA improvement

Process name	number of indicators can be improved	Average of correlation scores
DSS01	6	0.318
EDM01	5	0.238
APO01	5	0.230
DSS04	4	0.243
APO09	3	0.682
DSS02	3	0.674
MEA02	2	0.176
DSS06	2	0.373
BAI04	2	0.298
APO04	1	0.185
BAI07	1	0.204
APO11	1	0.179
DSS05	1	0.653

In Table 7, the process that can improve the largest number of indicators is DSS01 which is capable of improving six indicators; and the process with the maximum score of correlation is APO09 which is noted for its potential to improve three indicators.

Analyzing the prioritization list and the continuous improvement plan: Since the implementation of each COBIT process involves specific complexities and requires significant time and resources, it is essential to prioritize the compiled list. This prioritization should focus on identifying the processes that play the most critical role in improving ICTRT indicators and enhancing regulation quality, taking into account various dimensions. The focus group employed for process list prioritization, presented the final priority list, shown in Table 8.

Table 8: Process prioritized list

Priority	Process name	Process key area
1	DSS01: Manage Operations	Management
2	EDM01: Ensure Governance Framework Setting and Maintenance	Governance
3	APO01: Manage the IT Management Framework	Management

Priority	Process name	Process key area
4	DSS04: Manage Continuity	Management
5	APO09: Manage Service Agreements	Management
6	DSS02: Manage Service Requests and Incidents	Management
7	DSS06: Manage Business Process Controls	Management
8	BAI04: Manage Availability and Capacity	Management
9	MEA02: Monitor, Evaluate, and Assess the System of Internal Control	Management
10	DSS05: Manage Security Services	Management
11	BAI07: Manage Change Acceptance and Transitioning	Management
12	APO04: Manage Innovation	Management
13	APO11: Manage Quality	Management

Processes listed in this table can be incorporated into the CRA continuous improvement plan, as outlined in the cycles depicted in Fig. 2, to continuously enhance regulatory quality and facilitate digital transformation. It is considerable that IT governance is a continuous process, and the mere implementation of its processes is not sufficient to gain the maximum value, and needs to be monitored and evaluated continuously, so, the CRA should provide continuous monitoring and evaluation mechanisms. Furthermore, full adoption of the COBIT takes years and is a too large and complex process and step-by-step implementation of processes needs to be considered [37] and [38].

5- Conclusion

This study was designed to address two questions:

1) How can the relationship between ICTRT indicators and COBIT processes be measured?

To answer this question, a two-step approach was employed. Initially, the ACA methodology was applied to investigate the relationships between ICTRT indicators and COBIT processes. The results of ACA process validated using focus group methodology.

2) How can a continuous improvement plan for the CRA be designed based on selected COBIT processes?

In response to this question, a focus group was convened to develop a continuous improvement plan for the CRA, utilizing the identified COBIT processes.

The findings from this study provide a foundational roadmap for enhancing the quality of ICT regulation, supporting both development and digital transformation. The adaptable nature of our case study allows other ICT regulatory bodies to tailor our research outcomes to formulate their own improvement strategies, considering their unique local and national contexts.

The ICTRT functions as a comprehensive framework that assists countries in enhancing their ICT regulatory quality

in the face of the challenges posed by an ever-evolving digital landscape. This research highlights the multifaceted role of the ICTRT, emphasizing its significance beyond mere ranking purposes. The findings derived from this study provide valuable insights that can guide regulatory bodies in their efforts to achieve improved compliance and better alignment with the ICTRT as well as other frameworks established by the ITU. By leveraging these insights, countries can strategically navigate the complexities of ICT regulation and foster an environment conducive to digital transformation and growth.

Future Works:

In this study, we used the ACA and focus group methodologies. Future studies are encouraged to apply alternative methodologies for discovering relationships between ICTRT and other frameworks.

Acknowledgment:

This study was conducted in collaboration with the Enterprise Architecture Laboratory of IAUN and the Communications Regulatory Authority of the Islamic Republic of Iran. We express our gratitude for their support and contributions to this research.

References

- [1] T. C. T. Kahorongo, N. Reddy, and A. M. Karodia, "The Adoption of Information Technology in the Governance System of the Bank of Namibia," *Business and Management Studies*, vol. 1, no. 2, 2015, doi: 10.11114/bms.v1i2.876.
- [2] ITU Publications, *The impact of policies, regulation, and institutions on ICT sector performance*. Geneva: ITU, 2021.
- [3] GIRO, *Global ICT Regulatory Outlook 2020 (GIRO 2020)*. Geneva: ITU, 2020.
- [4] ISACA, "COBIT 2019 Framework: Introduction and Methodology," <https://www.isaca.org/resources/cobit>, ISACA.
- [5] H. Yeganeh, A. S. Mortazavi Kahangi, and A. Hadizadeh, "Identifying and analyzing macro actions in order to achieve regulatory goals in the country's national information network," *Iranian Journal of Information Processing and Management*, 2023.
- [6] T. Olwal, M. Masonta, L. Mfupe, and M. Mzyece, "Broadband ICT policies in Southern Africa: Initiatives and dynamic spectrum regulation," in *2013 IST-Africa Conference and Exhibition, IST-Africa 2013*, IEEE Computer Society, 2013.
- [7] K. B. Danbatta and T. Zangina, "Performance analysis of telecommunications regulations in Nigeria: A quality of service approach," *Dutse Journal of Pure and Applied Sciences*, vol. 8, no. 1a, 2022, doi: 10.4314/dujopas.v8i1a.17.
- [8] M. Suryanegara, "5G as disruptive innovation: Standard and regulatory challenges at a country level," *International Journal of Technology*, vol. 7, no. 4, 2016, doi: 10.14716/ijtech.v7i4.3232.
- [9] M. Mohlameane and N. Ruxwana, "Exploring the impact of cloud computing on existing South African regulatory

- frameworks,” *SA Journal of Information Management*, vol. 22, no. 1, 2020, doi: 10.4102/sajim.v22i1.1132.
- [10] G. Maccani, N. Connolly, S. McLoughlin, A. Puvvala, H. Karimikia, and B. Donnellan, “An emerging typology of IT governance structural mechanisms in smart cities,” *Gov Inf Q*, vol. 37, no. 4, 2020, doi: 10.1016/j.giq.2020.101499.
- [11] H. N. Nguyen, “Regulating Cyberspace in Vietnam: Entry, Struggle, and Gain,” *Columbia Journal of Asian Law*, vol. 35, no. 2, 2022, doi: 10.52214/cjal.v35i2.10028.
- [12] I. A. Raifu, I. A. Okunoye, and A. Aminu, “The effect of ICT on financial sector development in Africa: does regulatory quality matter?,” *Inf Technol Dev*, 2023, doi: 10.1080/02681102.2023.2233458.
- [13] S. Shrestha and D. Ram Adhikari, “Telecommunications Infrastructures and Services Development and Challenges in Nepal,” *International Journal of Internet, Broadcasting and Communication*, vol. 9, no. 2, pp. 27–36, 2017, [Online]. Available: <https://doi.org/10.7236/IJIBC.2017.9.2.27>
- [14] S. Adams and E. Akobeng, “ICT, governance and inequality in Africa,” *Telecomm Policy*, vol. 45, no. 10, 2021, doi: 10.1016/j.telpol.2021.102198.
- [15] O. A. Shobande and L. Ogbeifun, “Has information and communication technology improved environmental quality in the OECD? —a dynamic panel analysis,” *International Journal of Sustainable Development and World Ecology*, vol. 29, no. 1, 2022, doi: 10.1080/13504509.2021.1909172.
- [16] P. Chauhan and J. Mathew, “Evolution and Regulation of Telecommunication and Internet in India: A Study of the Policy governing the development of telecommunication and Internet in India,” *Revista de Direito, Estado e Telecomunicacoes*, vol. 15, no. 1, pp. 225–255, May 2023, doi: 10.26512/lstr.v15i1.45322.
- [17] M. Nikarya, M. Mazoochi, A. M. Montazeri, and F. Ayazi, “Investigating the Relationship between Regulatory Promotion Indicators and ICT Development,” *Iranian Journal of Information Management*, vol. 6, no. 2, pp. 47–66, 2021, doi: 10.22034/aimj.2021.132967.
- [18] M. Fasanghari, M. S. Amalnick, R. T. Anvari, and J. Razmi, “A robust data envelopment analysis method for business and IT alignment of enterprise architecture scenarios,” *Journal of Information Systems and Telecommunication*, vol. 1, no. 2, 2013, doi: 10.7508/jist.2013.02.004.
- [19] K. Bamary, M. R. Behboudi, and T. Abbasnejad, “An ICT Performance Evaluation Model based on Meta-Synthesis Approach,” *Journal of Information Systems and Telecommunication*, vol. 10, no. 39, 2022, doi: 10.52547/jist.16445.10.39.229.
- [20] S. Chege, G. Wanyembi, and C. Nyamboga, “The Relationship Between the Business-IT Alignment Maturity and the Business Performance for the Banking Industry in Kenya,” *International Journal of Scientific and Technical Research in Engineering*, vol. 3, no. 7, 2018.
- [21] A. Abu-Musa, “Exploring the importance and implementation of COBIT processes in Saudi organizations: An empirical study,” *Information Management & Computer Security*, vol. 17, no. 2, 2009, doi: 10.1108/09685220910963974.
- [22] D. Henriques, R. Almeida, R. Pereira, M. M. da Silva, and I. S. Bianchi, “How IT governance can assist iot project implementation,” *International Journal of Information Systems and Project Management*, vol. 8, no. 3, 2020, doi: 10.12821/ijispm080302.
- [23] I. G. Almusawi, “USING COBIT FRAMEWORK FOR REDUCING THE AUDIT RISKS OF ACCOUNTING INFORMATION SYSTEMS,” *Akkad Journal Of Contemporary Accounting Studies*, vol. 1, no. 1, 2022, doi: 10.55202/ajcas.v1i1.18.
- [24] A. Ilmudeen and B. H. Malik, “A Review of Information Technology Governance, Business Strategy and Information Technology Strategy,” 2016.
- [25] A. Badran, “Developing Smart Cities: Regulatory and Policy Implications for the State of Qatar,” *International Journal of Public Administration*, vol. 46, no. 7, 2023, doi: 10.1080/01900692.2021.2003811.
- [26] H. F. Hsieh and S. E. Shannon, “Three approaches to qualitative content analysis,” *Qual Health Res*, vol. 15, no. 9, 2005, doi: 10.1177/1049732305276687.
- [27] S. E. Baker and R. Edwards, “How many qualitative interviews is enough?,” *National Centre for Research Methods Review Paper*, 2012, doi: 10.1177/1525822X05279903.
- [28] K. A. Neuendorf, *The Content Analysis Guidebook*. 2020. doi: 10.4135/9781071802878.
- [29] P. Stockwell, R. M. Colomb, A. E. Smith, and J. Wiles, “Use of an automatic content analysis tool: A technique for seeing both local and global scope,” *International Journal of Human-Computer Studies*, vol. 67, no. 5, 2009, doi: 10.1016/j.ijhcs.2008.12.001.
- [30] M. Scharkow, “Content Analysis, Automatic,” in *The International Encyclopedia of Communication Research Methods*, Wiley, 2017, pp. 1–14. doi: 10.1002/9781118901731.iecrm0043.
- [31] Janusz Kacprzyk, *Standardisierte Inhaltsanalyse in der Kommunikationswissenschaft – Standardized Content Analysis in Communication Research*. Warsaw, Poland: Springer Publications, 2023. doi: 10.1007/978-3-658-36179-2.
- [32] H. Ahangarbahan and G. A. Montazer, “A fuzzy approach for ambiguity reduction in text similarity estimation (Case study: Persian web contents),” *Journal of Information Systems and Telecommunication*, vol. 3, no. 4, 2015, doi: 10.7508/jist.2015.04.002.
- [33] E. F. McQuarrie and R. A. Krueger, “Focus Groups: A Practical Guide for Applied Research,” *Journal of Marketing Research*, vol. 26, no. 3, 1989, doi: 10.2307/3172912.
- [34] M. Bloor, J. Frankland, M. Thomas, and K. Robson, *Focus Groups in Social Research*. 2012. doi: 10.4135/9781849209175.
- [35] V. Braun and V. Clarke, “Using thematic analysis in psychology,” *Qualitative Research in Psychology*, *J Chem Inf Model*, vol. 3, no. 2, 2008.
- [36] J. Grimmer and B. M. Stewart, “Text as data: The promise and pitfalls of automatic content analysis methods for political texts,” *Political Analysis*, vol. 21, no. 3, 2013, doi: 10.1093/pan/mps028.
- [37] S. Harguem, “A conceptual framework on IT governance impact on organizational performance: A dynamic capability perspective,” 2021. doi: 10.36941/ajis-2021-0012.
- [38] A. C. Amorim, M. Mira da Silva, R. Pereira, and M. Gonçalves, “Using agile methodologies for adopting COBIT,” *Inf Syst*, vol. 101, 2021, doi: 10.1016/j.is.2020.101496.

Design and Analysis of a Secure Intelligent Blood Management Information System

Hessah Alhajri¹, Mostafa Abd El-Barr², Kalim Qureshi^{1*}

¹.Department of Information Science, College of Life Sciences, Kuwait University, Kuwait

².Department of Electrical Engineering, College of Engineering, Badr University in Cairo, Egypt.

Received: 15 Feb 2024/ Revised: 04 Nov 2024/ Accepted: 11 Dec 2024

Abstract

The need for blood and the blood donors are on the rise continuously. Poor communication between blood banks and hospitals results in improper management and wastage of available blood inventory and can cause life threats. Therefore, there is an urgent need for coordination between blood banks, hospitals, and blood donors. The Design of a Secure Intelligent Blood Bank Information System (SIBBIS) is a way to align blood banks and hospitals with the help of the Internet. SIBBIS is a web application through which registered hospitals can check the availability of required blood, send a request for blood to the nearest blood bank or donor that matches the blood requirements, and order blood online as requested. A blood bank can also send a request to another blood bank in case of unavailable blood. The person willing to donate blood can find the nearest blood banks using SIBBIS. The location of the blood bank can be traced using maps. The life of the hardware, software, refrigerator cleaning of refrigerator and vaccination of employees are monitored intelligently. For this system, we developed a standard process-oriented information System analysis methodology using use case, activity, system sequence, entity relationship, and class diagrams. The usability of the developed system was evaluated, and it was 93.39%.

Keywords: Blood Bank; Information System; Security of Information System; Blood Donation.

1- Introduction

The need for blood is increasing due to the increase in the population. Humans can suffer from serious and significant health issues and even die from blood deficiency. Blood cannot be produced in the laboratory by medical science, but it can be transferred from one person to another with the guidance of medical science [1]. An ailing patient can go through several serious emergencies that cause bleeding and, hence, blood loss, which could be accidents, anemia attacks, surgical operations, and pregnancy, to name a few.

The traditional NACO blood bank management system requirement of paperwork is inefficient and error-prone, which cannot be tolerated in the time of emergency. Existing systems were designed to meet the incredibly huge and urgent demand for blood. Modern blood storage systems have been constructed to bridge the gap between blood recipients in need of blood and blood donors. It was observed that most patients who are in urgent need of

blood do not get blood on time due to the lack of communication between the blood recipients and blood donors. Improper management between the blood banks, blood donors, and hospitals lead to waste of the available blood inventory. There is a critical need for an efficient, real-time management system for communication and synchronization among the different constituents of the blood system stage players. This serious issue prompted us to devise a plan to build a secured intelligent blood bank information system to ensure that blood is always available in emergencies and that the appropriate donor is available [2]. Through the proposed system, the blood seeker should be able to view all information they need. Such information could be related to the donor, blood bank, and hospital. When the new user accesses the system. When a new user logs in to the system as a seeker or donor, they must provide valid information to prove their identity, such as their civil ID, driver's license, to authenticate the blood type [3].

The most serious concern in our proposed system is security. Sensitive information, such as patient information, must be kept secure and private when stored in a database. Users that use the proposed system are identified and their identities should be properly verified. Communication within the proposed system should be treated with confidentiality and not intentionally corrupted.

1-1- Objectives

In cases of epidemics, disasters, and wars, the need for large quantities of blood is critical. Large quantities of blood cannot be stored, so the presence of a system for preserving the data of all donors is required. Such a system should facilitate communication with these donors in the event of a need for blood donation [4]. Hence, this growth motivates build a system able to acquire the following objectives and innovations:

- *track* the quantities of blood and track the requests.
- *analyze* the standards from the World Health Organization regarding blood donation.
- *monitor* the devices specified for storing blood.
- *match* the suitable blood types automatically.
- *allow* the administrator to monitor the health/life of all connected equipment.
- *monitor* the health of all staff and the health status of all registered employees.
- *work* according to the guidelines of the World Health Organization for all blood transfusions.
- *allow* the staff to know all related information about the packages of blood.
- *manage* the components of blood and display all related information.
- *track* donors and contact them easily (after getting their acceptance).
- *check* all documents that certify that the donated blood is clear of viruses.

The Research Questions (RQs) of our research work are set to be the following:

RQ1: What are the current functional requirements of building a blood bank information system?

RQ2: What are the current non-functional requirements of building a blood bank information system?

RQ3: How to ensure the availability of a blood bank information system's equipment?

RQ4: How to manage the hygiene level of a blood bank information system employee?

RQ5: How to manage the blood donor's availability in a blood bank information system?

This article is organized as follows: Section 1 includes an introduction, motivation, and an outline of the article. Section 2 gives an overview of the current related work that investigates the efforts of other researchers to develop SIBBIS application. Section 3 provides an analysis of the proposed system and determines the system requirements. We provide a table of other related system features compared to existing and innovative features, in addition to the intelligence of the system and functional requirements. We follow the WHO (World Health Organization) guidelines to build our system in this part. Moreover, we provide illustrative design models of the proposed IS application and applied different models such as analysis diagrams and design diagrams.

Table 1: Summary of the key contributions of the related researchers' work.

Reference	Blood Bank Development Contributions
[2], [24]	Donors, patients, and hospitals can register into the system, and donors can access information about the various blood banks on the system and blood donation campaigns held by blood banks.
[9], [25]	This system brings voluntary blood donors and patients need blood into the same platform—together with Raspberry Pi to send messages to the corresponding blood donor via GSM modem
[13], [26]	Hospitals can check the availability of blood and send requests for blood to the nearest blood bank or donor matching the blood requirement
[22], [27]	Users can instantly view nearby hospitals and blood banks and hospitals online by tracing their location using GPS. An alert system for severe guiding ambulance to the patient's destination
[23], [28]	The system lists of records donors and blood group information, where contact details will appear in alphabetical order, finds a matching blood type, and reaches the nearby by city/area.

2- Related Work

In this part, we investigate the efforts of other researchers to develop similar nature systems. Recent references were chosen, and a brief comparison was made. Two tables

were provided: one presents a summary of the key contributions of the related researchers's work (Table 1). The other Table presents the key features of the three systems that we had chosen to present some similar systems to determine their functions and their roles in blood bank information systems.

In [2], [5], the authors aimed to provide a list of donors in the neighboring city/region when an urgent blood transfusion is needed. The user's contact information appears alphabetically on the screen and promptly connects them with a specific or related blood group via the Blood Bank Website. Another project component is an Android-based location-based app that will assist users in accessing blood donors' phone numbers for immediate assistance.

Online Blood Bank System (OBBS) was proposed in [6]. [9]. It provides a central repository for all available blood deposits and the accompanying information. Blood type, location, and storage date are all included in the report. This web-based system allows users to see if their specific category is available in the blood bank. Additional features include patients' names and contact information, booking, and even a requirement for a specific blood group listed on the website to locate willing donors in case of a blood emergency.

The work in [7], [13] granted the administrator access to all donor-related information in the system. The blood donor/recipient can promptly locate blood banks or hospitals matching a specific blood type or group, where they can request specific amounts of blood.

Creating a cloud-based blood bank system is the primary goal for the work of [4] to ensure that those in need have access to blood as quickly as possible, especially during times of emergency. As a part of this initiative, a mobile Android app has been utilized, containing all donor and adjacent hospital information, besides GPS capacity to locate blood banks and hospitals nearby. Health checkup drives, blood donation camps, and other similar facilities will be advertised to all registered users.

In [8], the authors aimed to create cloud storage connecting all blood banks. This cloud should deliver live information about blood supply availability. If there is insufficient blood available, the system will list blood donors' names and contact information belonging to different blood groups.

An automated blood bank system, using Raspberry Pi, was built by [9]. This system was used in the proposed project to send a message to the appropriate blood donor through a GSM modem when the user enters the requisite blood group details, bridging the donor-recipient communication gap via a low-cost and low-power Raspberry Pi kit as a communication bridge.

In [10], [22], a blood bank mobile application was created that sources a list of blood banks near the user, linking blood banks with potential donors, displaying maps, and

tracking locations while also estimating the time to reach the recipient.

In [11], [23], the Ublood System was built, an online blood donation system. It offers a quick and easy way to connect with donors and find nearby organ donors in emergencies like car accidents. A web application and an Android mobile application are both being considered. Table 2 shows the cross-list features with the proposed SIBBIS.

Table 2: Existing and innovation features

<i>Existing features from other researchers' work</i>	<i>Added innovative features in SIBBIS</i>
<ul style="list-style-type: none"> • Create user profile • Login • Define-blood banks/hospitals • Submit blood request • Find nearby donors • Send notifications • Search for donors • Map navigation • Blood type matching • Monitor blood storage and validity • Validate-donors availability 	<ul style="list-style-type: none"> • Real-time notification for blood donation • Search for nearby donors • Smart matching process • Control blood donation attempts • Life of hardware checking • Employees' health monitoring in the blood bank

The related work provided an overview of recent developments in this field. It was discovered that improper management between blood banks, blood donors, and hospitals resulted in the waste of available blood inventory. There is a critical need for an efficient, real-time management system for communication and synchronization among the different constituents of the blood system stage players. This serious issue motivated us to plan to create a secured intelligent blood bank information system to ensure that blood is always available in emergencies and that the appropriate donor is available [2].

3- Secure Intelligent Blood Bank Information System (SIBBIS)

SIBBIS is a web-based information system. Its main functions include users (blood donors and blood recipients) who can create their profile, search and find the requested blood type/amount, and request/donate blood. The system's transparency is evident in its tracking tool, enabling users to follow and monitor the process and condition of their requests. Additionally, the

system automatically directs the request to the right user. All records are maintained in a centralized database to ensure fast information retrieval and accuracy. Figure 1 displays a summary of the proposed system features, whether they exist in similar systems or innovation features.

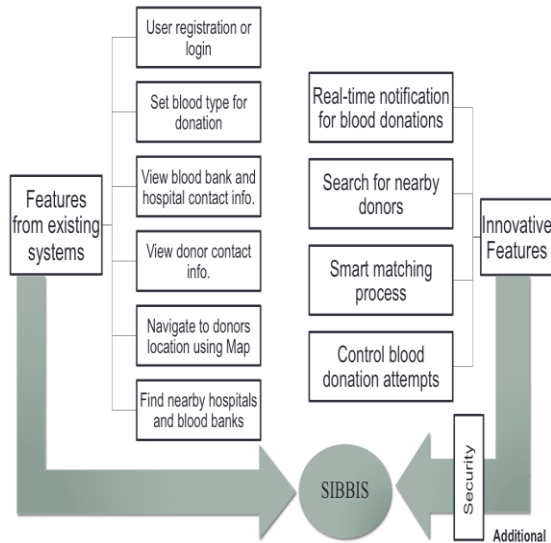


Fig. 1 SIBBIS component contract diagram

3-1- Functional Requirements of SIBBIS

Below is the list of functions that are needed to be support by the proposed SIBBIS system. The proposed SIBBIS system supported innovative features are listed in Table 2 (see above).

1. Users' registration or login
2. Finding nearby hospitals and blood banks
3. Viewing donor's contact information
4. Viewing hospitals and blood banks' contact information
5. Setting blood type for donation
6. Navigating to donor's location using maps

3-2- Comparison with similar Blood Information Management Systems

The purpose of presenting some similar systems is to know their functions and their roles in blood bank information systems. This should facilitate the designing of the SIBBIS in this research, projected to be utilized in the blood banks in Kuwait and other countries.

System-1 [12]: One of the software systems that are used in blood banks is called the Blood Donation Management System, comprising web and mobile

applications to help patients and donors communicate. People who want to donate blood must create an account by entering basic information. Google Map is connected with this application to locate a donor's exact location. An appointment is set only after a donor agrees to donate. After that, the system will notify the donor 12 hours before the donation is scheduled to take place.)

System-2 [13]: This is another similar system called Blood Bank Management System. It is a web application enabling registered hospitals to check the availability of requested blood and send blood requests to the nearest blood bank or donor matching the blood criteria. The blood bank could also be found via maps, and the mobile application is only available to donors.

System-3 [2]: CBBR Blood Bank System is another system that stores and manages donated blood that has been collected through a donation or a blood-collecting program. Donors and other recipients, such as patients and hospitals, can register in the Centralized Blood Bank Repository (CBBR). System administrators will have access to critical information like the type of blood available and locating the nearest blood center. A summary of the key features of all the three systems is shown in Table 3. Table 3: Key features of all the three systems [12,13, 2]. The table clearly shows that essential features are still missing in the literature and that more innovative work is needed in the current literature. This, in addition to what we have been through the difficult crisis of COVID-19 urges the need for innovative and intelligent blood management tools in the field. As a result of our research's aspects and studies discussed beforehand, we came out with a list of features that we will incorporate into our application.

A comparison was made between some of the highest-rated blood bank information systems, presenting their features, functions, and differences. Previous systems have deficiencies despite their several features: There is no connection between a blood donor and blood recipients, between hospitals, and hospitals with blood donors. There is no user (blood donor, blood recipient) role in the system, so the staff adds their information to the system manually. Our proposed system will bridge a connection between them to improve the donation process.

Table 3: Key features of all the three systems [12,13, 2]

System's Features	System 1 [12]	System 2 [13]	System 3 [2]
User registration and login	✗	✗	✓
Donors add blood type for donation	✗	✓	✗
Blood seekers search for nearby donors	✓	✗	✗
Finding nearby blood banks and hospitals	✗	✗	✗
Navigating directions through Google map	✗	✓	✓
Smart matching between blood types	✓	✗	✗

4- Secure Blood Bank Information System (SIBBIS)

The proposed SIBBIS aims to acquire, transmit, store, and manage various user-related information and the activities of the health departments that use it. There are various security mechanisms in health information systems to protect data access, collection, transmission, and storage [17]. The security measures in our system help in achieving the following objectives:

- Identify the system's users and their access rights.
- Maintain an operating environment for authorized users and the processing system by protecting the system from different attacks.
- Protect the information in the systems' database and during transmission in internal and external networks.

4-1- Access Controls

Access controls represent a vital security aspect as they limit access to the system's resources to specified users [16]. A health system that enables electronic and remote access sharing of medical data must employ users' identity proofing and authentication procedures before allowing any individual to perform activities, securing the system's safeguards against unauthorized users and operations. Our system has multiple user groups, and each of them will use the system to perform predefined roles. Therefore, it is essential to correctly identify users before assigning them a role in

the system. In the following, we elaborate on our system's security mechanisms:

4-1-1- User Identification

Identifying the users that our system is targeting is the first step in the development of access controls [15]. In the case of the proposed SIBBIS, our targets are the following:

- System Administrator: A person who manages the operation of a computer system
- Blood Donor: A person who is willing to donate blood
- Blood Recipient: A person who is seeking blood
- Staff: A person who works in blood banks or hospitals
- After identifying the system's users, we start the process of developing the access controls to prohibit users from accessing unnecessary resources.

4-1-2- User Registration

To access the system, users must complete the registration process by providing their identification information. To correctly identify new users' registration requests, the system requests them to provide the following credentials:

- Soft copy of a government-issued ID
- Demographic information including first and last names, birth date, nationality, gender, marital status, phone number, current address, and email address
- Indication of the group they belong to (blood donor, blood recipient, system administrator, staff)
- For blood recipient: provide information about blood type and medical history
- For staff: Provide information about specialty, certifications, years of experience, department, and location.

4-1-3- Identity Proofing

After receiving a complete registration request, the system checks the new account requests against existing ones to avoid duplications. After that, the

system administrators complete processing requests by:

- Verifying valid IDs
- Verifying that submitted demographics are valid and accurate

- Verifying that healthcare providers' documents are certified and valid
- Verifying eligible requests and notifying the system

4-1-4- User Authentication

After proofing eligible users' identities, the system generates users' IDs. The system requests said users to provide authentication information, which is a critical process as it verifies that a user is correctly identified and that a user can access the system and is eligible to perform specific functions [16]. Having a secure and efficient authentication secures the system. A user will not be able to access the system without using a valid authentication factor assigned by the system or chosen by them. Identity authentication is done by using one or more of these factors:

- Something you know, such as password, PIN, and secret answers
- Something you have, such as ID cards, mobile device, or a cryptographic key
- Something you are, such as a piece of biometric data.

Choosing a strong authentication factor is necessary for preventing unauthorized access attempts to information systems [15]. Most security breaches occur due to weak or stolen authentication information, leading to the leak of sensitive information and, in turn, leading to huge financial costs [16]. In our system, we have incorporated a Two-Factor Authentication process (2FA) to overcome the risks associated with using a single factor. Through a 2FA system, users must provide two authentication factors to access the system, thus, significantly reducing phishing attacks by adding an extra protection layer to the system. For our system, we have chosen an efficient and friendly 2FA

method. The first authentication factor is the most common factor, which is the password. For the second authentication factor, the authentication manager in our system will generate a One-Time Password (OTP) to authenticate users further. The user can receive the OTP via SMS or email. The generated password will be valid for one time only:

- Create a password.
- Choose the method of the second authentication step (SMS, email).
- Fill up the required information.
- Confirm the request.

When a user attempts to access the system, the system starts by checking the inserted ID in the user's database.

If the ID exists, the system retrieves the corresponding password in a decrypted form. Next, the user is requested to insert their password. The provided password will be compared against the retrieved password. If the entered password is correct, the system generates the OTP code and sends it to the desired user method for the second authentication step. The user receives the code and inserts it to access the system.

4-1-5- Access Control List

Authenticating eligible users alone is not enough to protect information systems. Internal users can cause threats to the system by exploiting the system's resources for illegal acts [15]. Therefore, it is crucial to define the system's access controls. The users of the system and subjects perform different actions on the system's objects. These objects of a system include files, hardware devices, network connections, and processes. Depending on the user role and the object, we define the access mode. Access modes include, but are not limited to, read, write, delete, create, and modify. Creating an access control list is the most convenient method of preventing subjects from performing unauthorized activities [16]. With this list, unwanted access will be immediately denied. Groups must be assigned roles that apply the following standards:

- Roles are designed and implemented based on the principle of least privileged.
- Permissions are to be defined based on role authority and responsibilities within a job function.
- A user account is assigned to a role that allows it to perform only what is required for that role.
- A user can only access an object based on an assigned role.
- The object is only to be concerned with the user's role and not the user.

The access rights lists define the access rights to the system's resources allowed or denied for authorized users [15]. When a user attempts to access any resource, the system checks this list to determine the user's type. The accesses provided for users are as follows:

- Create: The user can create new entities and upload folders into the system.
- Delete: The user can delete entities and folders in the system.
- Read: The user can view the entities and folders in the system.

- Write: The user can edit and modify the contents of entities and folders in the system.

When an authenticated user provides the correct authentication mean (ID, password, and OTP), the system grants the user access to the system resources. However, there is no full access to the system's resources for all users. The system checks the permission level of each user by checking the access rights list and then grants the appropriate access to each user.

4-2- Data Security Mechanism

The sensitive data incorporated in our system is projected to be diverse. The attractive nature of the health information system makes it more exposed to threats like eavesdropping, forgery, and manipulation [14]. These threats could lead to huge damages and, in some cases, life threats. If attackers gain access to health systems, they might modify the stored data or change the system's configuration. This would lead to threatening the lives of patients and low-quality services causing financial losses. Hence, it is important to consider the system's information security when stored in the database and when it is in the transmission phases.

4-2-1 Database Security

Storing tremendous amounts of data in a central database server bears many threats and risks that must be considered [19]. The system's database must be sufficiently protected. In addition to access controls, we incorporate other mechanisms to increase the system's security significantly. Via encryption, we prevent interceptors or intruders from accessing the plaintext form of data in the system's databases. Additionally, authorized users will be prevented from accessing unauthorized resources through the application of cryptography architectures to the stored data is a primary method to keep the database secure. In cryptography, data would be altered to a format that unauthorized users cannot view. Using it is very effective in protecting files and sensitive data in the database. The database of our system will be encrypted using different sets of rules that would be defined in the system's configuration phase [18].

4-2-2- Backup and Recovery

Planning for database backup and recovery are adequate safeguards against data loss and software errors. It is an important security measure to protect the system's data against crashes and network and disk failures. With it, the process of reconstructing the system's database would be faster and easier. Therefore, making copies and archives of the system's data and processes is crucial for a successful backup [17].

The transaction logging technique would be incorporated into our database server to back up the system. With it, we keep track of the updates to the system's database. Modifications on the stored data are recorded with the details of who, when, and how the update was performed. By this, we create an audit trail of all updates to the database that can trace any error or suspicious activity. This technique is important in safeguarding the system's security by keeping evidence of the source of changes. Hence detecting unauthorized actions that authorized users could conduct. Moreover, this logging can assist in detecting the source of error if there is a failure in the system or a part of it.

4-2-3 Data Transmission Security

In our proposed blood bank information system, data would travel, whether via wired or wireless means, between the different components of our system and to other health systems, if necessary. Along with the transmission medium, unauthorized users can intercept this data. Data in transmission levels targets attract many threats such as spying, altering information, interrupting communication, sending extra signals to block the base station, and networking traffic. As in database security, the most effective solution is to encrypt information during transmission to preserve confidentiality [20, 21]. The transmission of some of the system's data is over insecure public networks, such as the Internet. Therefore, encryption would be necessary to ensure confidentiality. This involves the use of algorithms and secret keys. Many protocols, such as Internet Protocol Security (IPsec), Transport Layer Security (TLS), and Secure Socket Layer (SSL), have been proposed to achieve protected communication data over insecure channels. In the system implementation phase, we will select the most appropriate protocol to achieve the system's security needs.



Fig..2 SIBBIS use case diagram

5- The Intelligence of the SIBBIS

- The following are the intelligent features supported by SIBBIS:
- Real-time notification for blood donation
- Searching for nearby donors
- Smart matching process
- Controlling blood donation attempts
- Life of hardware checking
- Employees' health monitoring in the blood bank (Blood banks and hospitals are required to ensure all employees have the needed certification (such as a COVID-19 vaccination certificate), and that safety and hygiene are always considered.
- Our proposed system asked staff members to upload the COVID-19 vaccine certification or a PCR test that ensures they are not infected as an alternative. In addition, medical history was required.

6- Analysis and Design Models of SIBBIS

In this section, we provide illustrative analysis and design models of the proposed system. Analysis diagrams include use case, sequence diagrams, activity diagram, entity relationship diagrams (ERD), and class diagrams. However, in this article, we will explain only the use case diagram, general activity diagram, and sequence diagrams.

6-1-Use Case Diagram

The use case diagram of the SIBBIS is shown in Figure 2. We have focused on the main actors of our system, which are users and system administrators. On the other hand, we have one database responsible for storing all the data. The use case diagram identifies how users interact with the system.

6-2- Activity Diagram of SIBBIS

As shown in Figure 3, the user starts by logging in to the system, then verifies the existence of the user in the system. Based on the success of the verification, the user will then be able to navigate proposed features such as creating a request for blood type, searching for donors, and navigating to the location through the map and finally through system usage.

6-3-Sequence Diagrams of SIBBIS

Figure 4 illustrates the sequence diagram of, which explains how objects and predefined actors are functioning starting from login as a base function to system usage until interacting with the backend to implement the required function.

6-4-ER Diagram of SIBBIS

In Figure 5, an entity–relationship diagram demonstrates the correlations between system objects or components. The ER model is composed of entity types and specifies relationships that can exist between these system-related entities. Some of these relations are described as one-to-one, many-to-many, and many-to-one. The system comprises seven entities: donors, donors' history, donation requests, recipients, recipients' history, blood centers or banks, and, finally, the map location details.

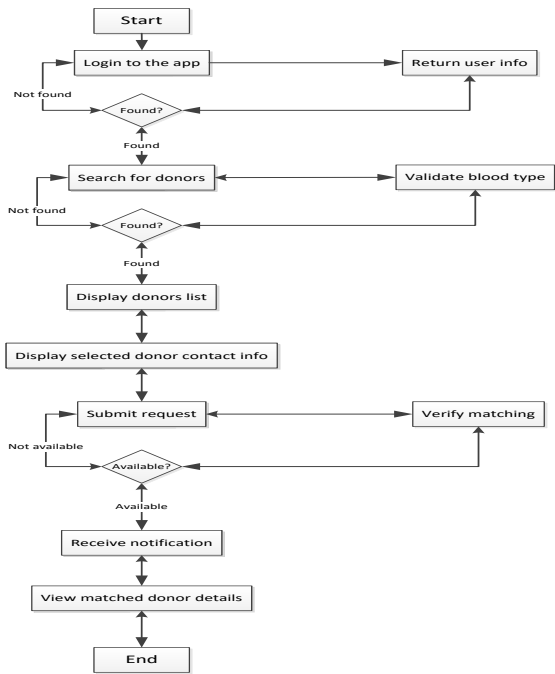


Fig. 3 General activity diagram

6-5-Class Diagram of SIBBIS

The class diagram is a type of static diagram that describes the structure of the system by visually representing the main classes defined in the system, objects, related attributes, and the relationships between these objects. As shown in Figure 6, the

application includes mainly five objects: donors, which includes information about the donors, this object is correlated to the object of location which includes the information about the location of each donor, which is also correlated with the object blood bank that includes the information of blood banks and their locations, and, finally, the recipient object, which includes the basic information of users projected to receive blood donations. As evident, the recipient object has no direct relationships with any other objects since they can search for donors and request blood donations, which can be implemented in the application without the need to build a backend relationship.

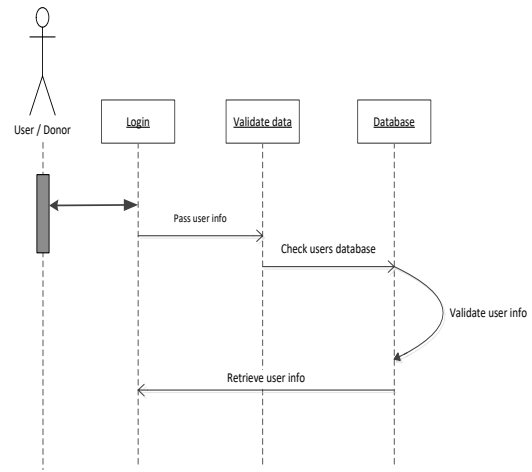


Fig. 4 Sequence diagram

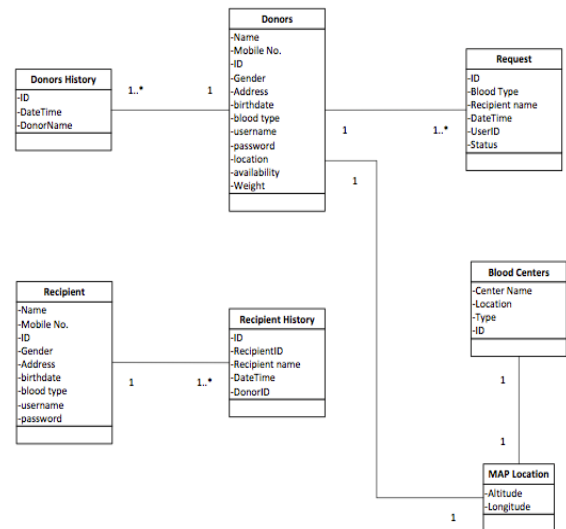


Fig. 5 Entity Relationship diagram

7- Implementation

This section is dedicated to explaining the proposed system user interface and SIBBIS usability evaluation.

7-1 Build System Screenshots

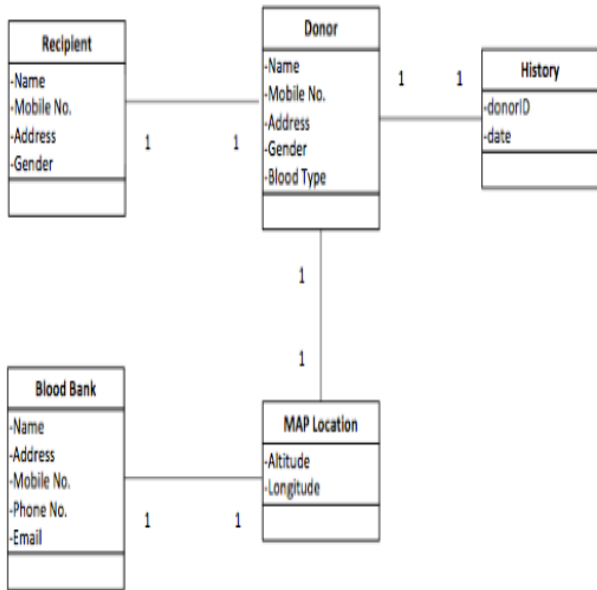
The screenshots of the developed SIBBIS are shown in Figures 8–17.

8- SIBBIS Usability Evaluation

To evaluate the usability of the developed system, we conducted a survey using the Computer System Usability Questionnaire (CSUQ), consisting of 19 questions

randomly distributed among 50 users. The user used the developed system and recorded their evaluation on the question paper. The user evaluates the system usability by answering the following items:

1. Overall, I am satisfied with how easy it is to use this system.



2. It was simple to use this system.
3. I can effectively complete my work using this system.
4. I am able to complete my work quickly using this system.
5. I am able to efficiently complete my work using this system.
6. I feel comfortable using this system.
7. It was easy to learn to use this system.
8. I believe I became productive quickly using this system.
9. The system gives error messages that clearly tell me how to fix problems.
10. Whenever I make a mistake using the system, I recover easily and quickly.
11. The information (such as online help, on-screen messages, and other documentation) provided with this system is clear.
12. It is easy to find the information I needed.
13. The information provided for the system is easy to understand.
14. The information is effective in helping me complete the tasks and scenarios.
15. The organization of information on the system screens is clear.
16. The interface of this system is pleasant.
17. I like using the interface of this system.

18. This system has all the functions and capabilities I expect it to have.
19. Overall, I am satisfied with this system.

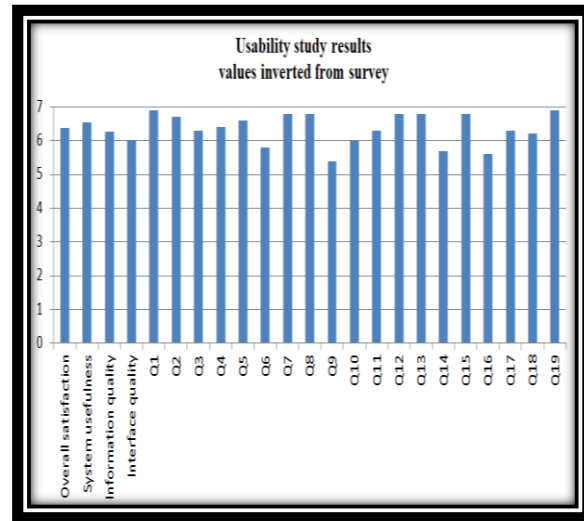


Fig. 6 Class Diagram

Fig. 7 CSUQ Results

Figure 7 shows the scores for general categories, together with the averages for all of the questions. While the study concluded that SIBBIS software recipients in the whole country by 93.39% (93.3%)

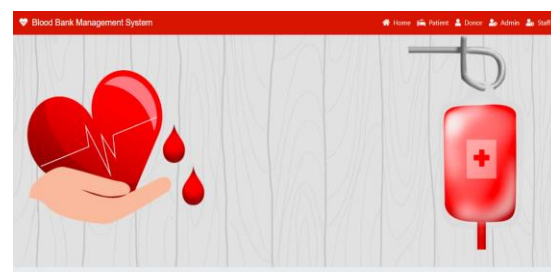


Fig. 8 Main page

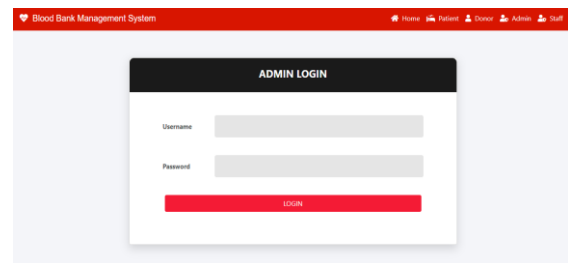


Fig. 9 Admin login page

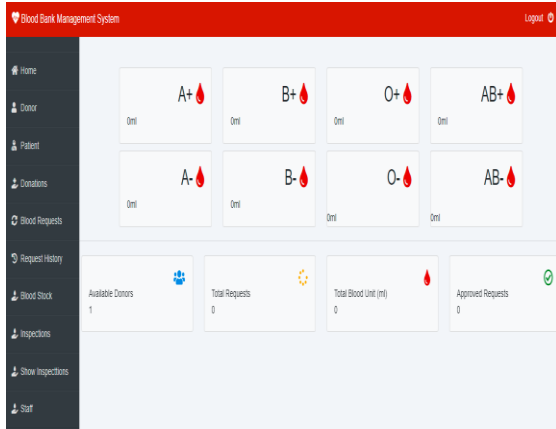


Fig. 10 Admin main page.

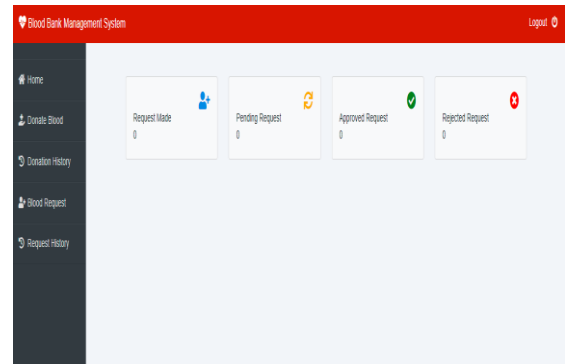


Fig. 13 Donor login page.

Fig. 14. Donor main page

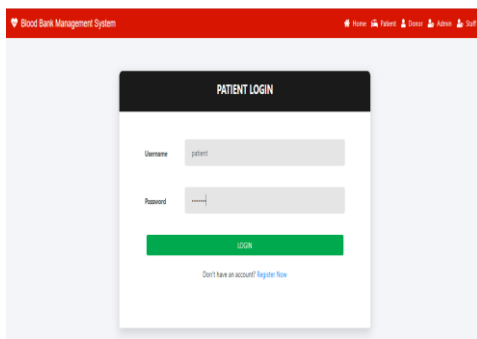


Fig. 11 Patient login page.

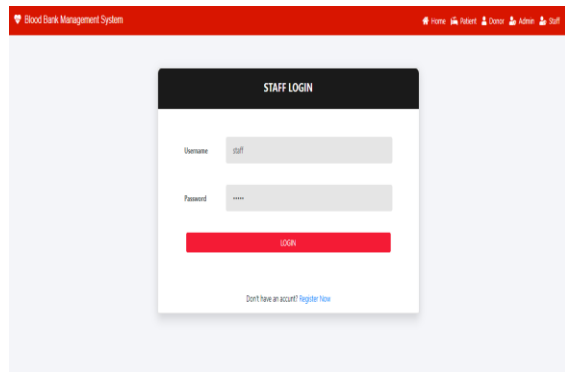


Fig. 15 Staff login page

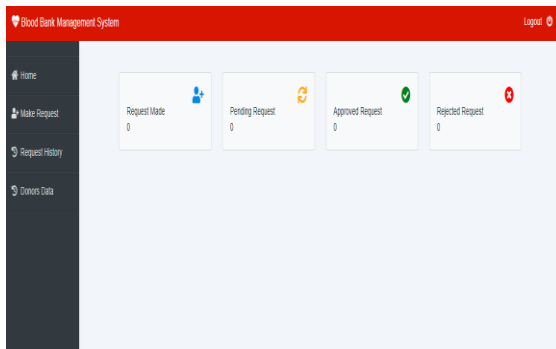


Fig. 12 Patient main page.

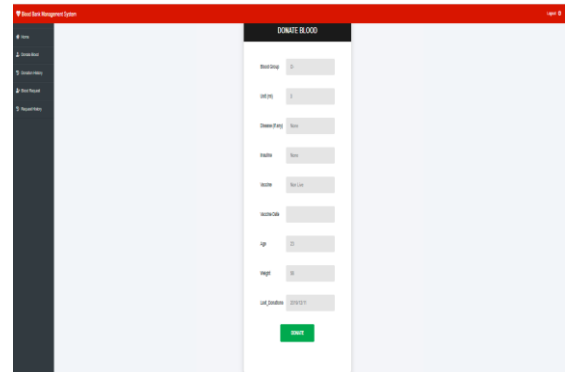


Fig. 16. Donor donating blood

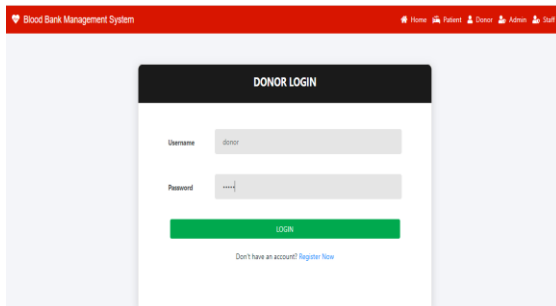


Fig. 17 Patient requesting blood

9- Conclusion

In conclusion, technology plays a vital part in improving the wellness and quality of life of people. This article presents the design and analysis of a standard process-oriented information System analysis methodology using use case, activity, system sequence, entity relationship, and class diagrams. SIBBIS is developed under the guidelines of the WHO. The proposed system is intelligent in terms of the real-time notification for blood donation, search for nearby donors, smart matching process, control of blood donation attempts, life of hardware checking, and employees' health monitoring in the blood bank. The system is an attempt to complement and add up what is missing in the current literature in terms of innovation and intelligence. The system was tested, and its usability was 93%. In addition, the system will be beneficial for blood banks, hospitals, clinics, and blood donors and blood seekers (patients). In the future, we have a plan to add more functions for the users and also make the system more intelligent and secure.

References

- [1] Advisera Expert Solutions Ltd., "How to Implement NIST Cybersecurity Framework using ISO 27001", 2017, [Online] Available:http://www.infomania-services.fr/control/file/181206101951000000_5892133709_1544091591.pdf
- [2] Akar, I. F., Mohammad, T. A., & Ismail, M. "CBBR Centralized Blood Bank Repository", ResearchGate. Retrieved November 9, 2021, from https://www.researchgate.net/publication/319372139_CBBR_Centralized_Blood_Bank_Repository
- [3] Ramachandran, P. Girija, N., Bhuvaneshwari, T., "Classification Blood Donors Using Data Mining Techniques", International Journal of Computer Science and Engineering, Vol.1, Issue 1, 2011, pp.10-13.
- [4] Chaudhari, S., Walekar, S., Ruparel, K., and Pandagale, V., "A Secure Cloud Computing Based Framework for the Blood bank", International Conference on Smart City and Emerging Technology (ICSCET), Mumbai, India, 2018, pp. 1-7.
- [5] Madan, T., Bharwai, N., Kumar, N., "Blood Donation Management with Modern Engineering", International Computational Engineering and Networking, 2016, vol. 9(8), pp.27-31.
- [6] Chetan Masram, Arshad Mulani, Rasika Bhitale, Jidnesh Koli, "Online Blood Bank Management System", International Research Journal of Engineering and Technology, Vol. 8, issue 06, 2021, pp.4220-4226
- [7] S Periyanaagi, A Manikandan, M Muthukrishnan, M Ramakrishnan, "Bdoor App-Bood Donation Application using Android Studio", Journal of Physics: Conference Series, 2021, pp.1-12.
- [8] AL-Kalbani, S. I. A. Kazmi and J. Pandey, "IoT Based Smart Network for Blood Bank," 2018 7th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, 2018, pp. 732-736.
- [9] A. C. Adsul, V. K. Bhosale and R. M. Autee, "Automated blood bank system using Raspberry PI," 2018 2nd International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 2018, pp. 252-255.
- [10] Fathima, M., A. Valarmathi, " Blood Bank Mobile Application", International Journal of Scientific Research in Computer Science, Engineering and Information Technology, vol. 2, issue 2, 2017, pp. 1126-1130.
- [11] Chawla, S., Dalal, T. "UBlood: Utilize every cell of Blood- A Proposed Mobile based Application Framework", International Journal of Computer Science and Information Technologies, Vol, 8(2), 2017, pp.292-294.
- [12] Akkas Ali, K. M., Jahan, I., Islam, A., & Parvez, "Blood Donation Management System", American Journal of Engineering Research (AJER), 2015, Vol4(6), pp.123-136. [13] Raut, P., Parab, P., Suthar, Y., Narwani, S., & Pandey, "Blood Bank Management System", International Journal of Advance
- [14] Appari, Ajit, Johnson, "Information Security and Privacy in Healthcare: Current State of Research", International Journal of Internet and Enterprise Management, 2020, Vol. 6. Pp.279-314.
- [15] Athanase Nkuzimana, Boiko, Andrii & Shendryk, Vira, "System Integration and Security of Information Systems", Procedia Computer Science, 2017, Vol.104. pp.35-42.
- [16] Boriev, Z & Sokolov, S & Nyrkov, "Review of modern biometric user authentication and their development prospects", IOP Conference Series Materials Science and Engineering. 2015, Vol. 91, pp.1-12.
- [17] Leila Rikhtechi, Vahid Rafe, Afshin Rezakhani, "Secure Access Control in Security Information and Event Management System", Journal of Information Systems and Telecommunication, 2021, pp. 67-78.
- [18] Pattanashetti, M. A., & Pilli, G. S, "Novel Transfusion parameters in Blood bank for Thalassemia patients", Indian

- Journal of Pathology and Onco-logy, 2017, vol. 4(4), pp.396-399.
- [19] M. Puppala, T. He, X. Yu, S. Chen, R. Ogunti and S. T. C. Wong, "Data security and privacy management in healthcare applications and clinical data warehouse environment," IEEE-EMBS International Conference on Biomedical and Health Informatics (BHI), Las Vegas, NV, USA, 2016, pp. 5-8
- [20] Zukime, M. & Mat Junoh, Mohd Zukime & Osman, Abdullah & Ab. Halim, M. Suberi & Halim, Ab & Safizal Abdullah, M.. Data Security: Issues And Challenges for Disaster Management In The New Millennium. International Journal of Scientific & Technology Research. 2014, 3.8.2277-8612.
- [21] Siddiq Iqbal, Sujatha B. R. "Secure Key Management Scheme for Hierarchical Network Using Combinatorial Design", Journal of Information Systems and Telecommunication, Vol. 10, No.1, January-March 2022, pp. 20-27.
- [22] Ashita Jain, Amit Nirmal, Nitish Sapre, Prof Shubhada Mone, "Online Blood Bank Management System using Android", International Journal of Innovative Studies in Sciences and Engineering Technology, Volume: 2 Issue: 2, 2016, pp.55-58.
- [23] Mohit, "Review on Blood Bank Management Systems", International Research Journal of Modernization in Engineering Technology and Science, Volume:03/Issue: 04/April-2021, pp. 172-174.
- [24] Ben Elmir, W., Allaoua Hemmak, A., and Senouci, B., Smart Platform for Data Blood Bank Management: Forecasting Demand in Blood Supply Chain Using Machine Learning", Information January 2023, 14(1):1-31. <https://doi.org/10.3390/info14010031>
- [25] Sri A., Pravallika, M. Kumar, O., Sridevi, K., Balaji, K., "A Systematic Review on Blood Bank Information Systems", International Journal of Scientific Research & Engineering Trends, Volume 10, Issue 2, Mar-Apr-2024, ISSN (Online): 2395-566X. <https://doi.org/10.22214/ijraset.2023.49843>
- [26] Varghese A., Thilak, K., Saritha., M., "Technological advancements, digital transformation, and future trends in blood transfusion services", International Journal of Advances in Medicine, March-April 2024, Vol 11, Issue 2, pp. 147-152.
- [27] Ghouri AM, Khan HR, Mani V, Ul Haq MA, De Sousa, ABL., "An Artificial-Intelligence-Based Omnichannel Blood Supply Chain., 2023;1-58. DOI:10.22214/ijraset.2023.49843
- [28] Mohammed R., Al-zebari, A., "Proposed A Web-Based Intelligent System to Manage the Blood Bank in Zakho District, Vol 8 No.3, June 2023, PP. 1267-1284. DOI:10.25212/ifu.qzj.8.5.46