ISSN Print: 2518-4245
ISSN Online: 2518-4253
Vol. 61(3), September 2024

PROCEEDINGS
OF THE PAKISTAN ACADEMY OF SCIENCES:
A. Physical and Computational Sciences

PAKISTAN ACADEMY OF SCIENCES
ISLAMABAD, PAKISTAN

Proceedings of the Pakistan Academy of Sciences: Part A Physical and Computational Sciences

Proceedings of the Pakistan Academy of Sciences A. Physical and Computational Sciences is the official flagship, the peer-reviewed quarterly journal of the Pakistan Academy of Sciences. This open-access journal publishes original research articles and reviews on current advances in the field of Computer Science (all), Materials Science (all), Physics and Astronomy (all), Engineering Sciences (all), Chemistry, Statistics, Mathematics, Geography, Geology in English. Authors are not required to be Fellows or Members of the Pakistan Academy of Sciences or citizens of Pakistan. The journal is covered by Print and Online ISSN, indexed in Scopus, and distributed to scientific organizations, institutes and universities throughout the country, by subscription and on an exchange basis.

Annual Subscription: **Pakistan:** Institutions, Rupees 8000/-; Individuals, Rupees 4000/- (Delivery Charges: Rupees 300/-)
Other Countries: US$ 200.00 (includes air-lifted overseas delivery)

HEC Recognized; Scopus Indexed

Published by **Pakistan Academy of Sciences**, 3 Constitution Avenue, G-5/2, Islamabad, Pakistan
Email: editor@paspk.org; **Tel:** 92-51-920 7140 & 921 5478; **Websites:** www.paspk.org/proceedings/; www.ppaspk.org

Printed at **Graphics Point.,** Office 3-A, Wasal Plaza, Fazal-e-Haq Road Blue Area Islamabad.
Ph: 051-2806257, **E-mail:** graphicspoint16@gmail.com

PROCEEDINGS OF THE
PAKISTAN ACADEMY OF SCIENCES: PART A
Physical and Computational Sciences

CONTENTS

Volume 61, No. 3, September 2024 **Page**

Submission of Manuscripts: Manuscripts may be submitted as an e-mail attachment at editor@paspk.org or submit online at http://ppaspk.org/index.php/PPASA/about/submissions. Authors must consult the ***Instructions for Authors*** at the end of this issue or at the Website: www.paspk.org/proceedings/ or www.ppaspk.org.

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 227-245 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)842

Pakistan Academy of Sciences

Review Article

Advancements in Word Embeddings: A Comprehensive Survey and Analysis

Khushal Das[1*], Kamlish[2*], and Fazeel Abid[3]

[1]Department of Computer Science, Modelling, Electronics and Systems Engineering, University of Calabria, Rende, Itlay

[2]Department of Computer & Software Engineering, College of Electrical and Mechanical Engineering, National University of Sciences and Technology (NUST), Islamabad, Pakistan

[3]Department of Computer Science and Information Technology, University of Lahore, Lahore, Pakistan

Abstract: In recent years, the field of Natural Language Processing (NLP) has seen significant growth in the study of word representation, with word embeddings proving valuable for various NLP tasks by providing representations that encapsulate prior knowledge. We reviewed word embedding models, their applications, cross-lingual embeddings, model analyses, and techniques for model compression. We offered insights into the evolving landscape of word representations in NLP, focusing on the models and algorithms used to estimate word embeddings and their analysis strategies. To address this, we conducted a detailed examination and categorization of these evaluations and models, highlighting their significant strengths and weaknesses. We discussed a prevalent method of representing text data to capture semantics, emphasizing how different techniques can be effectively applied to interpret text data. Unlike traditional word representations, such as Word to Vector (word2vec), newer contextual embeddings, like Bidirectional Encoder Representations from Transformers (BERT) and Embeddings from Language Models (ELMo), have pushed the boundaries by capturing the use of words through diverse contexts and encoding information transfer across different languages. These embeddings leverage context to represent words, leading to innovative applications in various NLP tasks.

Keywords: Word Embeddings, Word Representations, NLP, Contextual Embeddings, BERT, ELMo, Word2Vec, Cross-Lingual Embeddings.

1. INTRODUCTION

Words are components of any speech belonging to meaning as well as significance. Further, characters of any written word never have a bit of significant sense by themselves, which shows that characters can't present a powerful sense of the written word individually. For instance, Book and Pen tend to be related to one another, but it is most unlikely that we will evaluate or come across the value of this relevancy by using only the characters of this pair of words [1]. Word embeddings are frequently described as models for strings that fulfil the essential function of providing meaningful representations for words or phrases. Moreover, word embeddings represent words within a continuous vector space, allowing for the modelling of well-defined relationships among them. In this space, words are arranged into vectors with familiar properties, establishing meaningful connections through geometric relationships [2-4].

The influence of word embeddings largely depends on their ability to capture natural language and geometric relationships. They allow expeditious end-to-end modules by modulating an exceptional real-life representation right into an unremitting space; because of this, they are

Received: November 2023; Revised: August 2024; Accepted: September 2024
* Corresponding Authors: Khushal Das <Khushal.das@dimes.unical.it>; Kamlish <kamlish@outlook.com>

well-liked in natural language processing (NLP) subjects: they will be easy to plug into deep learning modules [5]. Sentiment analysis stated that physical object recognition and many other everyday jobs surpassed their renowned matching part with these techniques. We are primarily concerned with word embeddings learned on the trained corpus. This group of representations is creating an effort to compile a complete plain text useful dataset into an unceasing vector representation with no professional familiarity. Thus, in this survey, we accept this claim that the logic of a word enormously relies on the words neighbouring it. Other word embedding practices enhance this supposition and usage of a language model to shape contextualized word representations, such as BERT [6]. Recent advancements in areas such as sign language translation [7], healthcare signal processing [8], and vehicle detection [9] further illustrate the versatility of deep learning models in enhancing communication and accuracy across diverse applications. Finally, a distinctive technique is to build vectors using familiarity or another basis of experienced knowledge; an example is the TransE approach, as described by Cano and Morisio [10]. The process of training and using word embeddings for a machine learning objective is illustrated in Figure 1 [11].

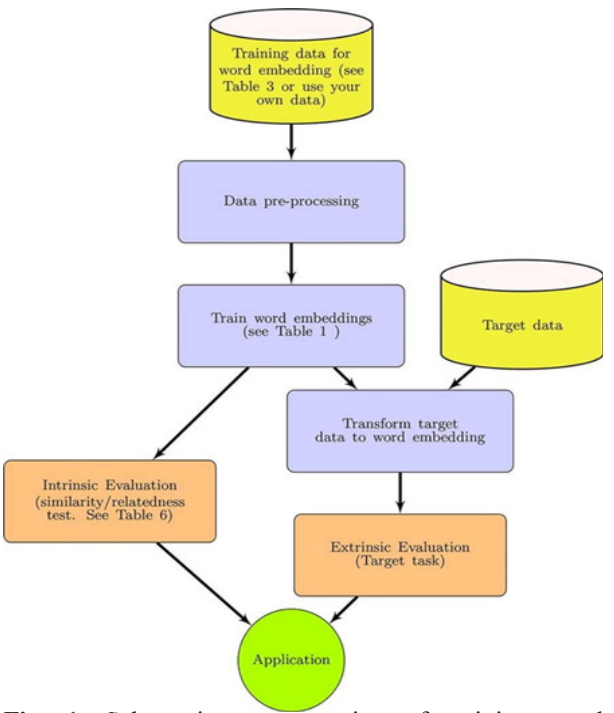

Fig. 1. Schematic representation of training word embeddings to use them for Machine learning objectives [11].

Previously, feature engineering NLP involved evolving vital mathematical functions to symbolize relevant sides of the text, such as the relation of pronouns to nouns. This method frequently required important domain information plus energy to find significant features [12]. Differently, word embeddings can be studied from the text's corpus and don't need any feature extraction or manual labelling; they are usually known in an unsupervised method [13]. So, we can say that word embeddings can be straightforwardly learned on whichever text data corpus. Word embedding is divided into two types: contextual and non-contextual word embeddings. The differentiation between these two types is that either the word embedding changes dynamically according to the context in which it appears or not.

Regardless of an excess of related works reachable on language models, word embeddings, and their advancements plus applications, no comprehensive survey collecting the detailed work done on word embeddings exists up to now. The current paper discusses the recent advances and innovations in word embeddings. In a study, Font and Costa-Jussà [14] employed a transfer translation architecture to examine incorporating two debiasing techniques using Global Vectors for Word Representation (GloVe) embeddings. The researchers put forth and assessed a scheme on the WMT English-Spanish benchmark task, observing improvements of up to one Bilingual Evaluation Understudy (BLEU) point. Regarding gender bias assessment, the researchers generated a collection of occupations and demonstrated that their system can mitigate inherent biases in the baseline system. Rezaeinia *et al*. [15] introduced Improved Word Vectors (IWV), a novel technique designed to enhance the accuracy of sentiment analysis by leveraging pre-trained word embeddings. Their methodology incorporated several approaches, including part-of-speech (POS) tagging, lexicon-based strategies, a word position algorithm, and Word2Vec or GloVe methods. Their plan's accuracy was validated using deep-learning models and benchmark datasets designed explicitly for sentiment analysis. The results of their experiment about sentiment analysis demonstrated the high effectiveness of IWV. Yao *et al*. [16] developed several intuitive evaluation methods for temporal word embeddings. Their quantitative and qualitative analyses indicate that their methodology consistently captures

evolutionary patterns. Furthermore, their approach steadily overtakes current state-of-the-art material embedding methods regarding semantic accuracy and structural quality. Zhao *et al*. [17] performed a series of intrinsic analyses, revealing several key findings. Firstly, ELMo, a language representation model, was observed to contain a significantly higher number of male objects than female objects. Secondly, the ELMo embeddings consistently incorporate gender-related information. Lastly, the encoding of gender data in ELMo was found to be uneven, with a noticeable disparity between male and female objects. Moreover, the researchers demonstrated that a prior system reliant on ELMo exhibits bias and identifies significant bias within the WinoBias dataset. Finally, the researchers examined two methodologies to mitigate gender bias and demonstrated the potential for eliminating the bias observed in the WinoBias dataset.

In recent years, the era of significant data has brought about challenges related to information overload. Addressing these challenges, Du *et al*. [18] aimed to achieve precise and automatic categorization of Internet news edition data. Recognizing the limitations of single-topic and word embedding models, they planned a novel text representation method that combined Glove models, Word2VEC, LDA, and TF-IDF. Additionally, Suhartono *et al*. [19] introduced two CNN architectures that incorporated Glove and Word2Vec word embeddings to analyze sentiment in drug reviews, utilizing deep learning methods, for instance, BERT and RoBERTa. Haller *et al*. [20] provided a comprehensive taxonomy of ways in the field, spanning classical Machine Learning to Deep Learning approaches while emphasizing the need for adaptations in Deep Learning architectures for NLP to tackle evolving challenges in ASAG tasks. Gender bias in static word embeddings was scrutinized by Caliskan *et al*. [21], revealing preferences in semantic suggestions, word frequency, parts of speech, clustered concepts, word frequency, term, parts of speech, and word meaning dimensions. Meanwhile, Tang *et al*. [22] proposed an unsupervised method to learn Dynamic Contextual Word Embeddings (DCWEs) through time-adapting a pre-trained MLM using manual and automatic templates. Alnajjar *et al*. [23] contributed to the field by creating a sentiment analysis corpus for endangered languages and Finnish. The study conducted by Yen and Jeon [24] achieved significant accuracy improvements in embedding-matching A2W systems by generating multiple embeddings and incorporating pronunciation-based embeddings. Engler *et al*. [25] introduced SensePOLAR, offering word sense-aware interpretability for contextual word embeddings. Schiffers *et al*. [26] developed word embeddings tailored for the social sciences and compared them to general language models in a domain-specific context. Lastly, Zaland *et al*. [27] comprehensively evaluated existing word embedding algorithms on extrinsic classification tasks, shedding light on how these models encode word relations. The study by Worth [28] highlights that advancements such as Word2Vec, GloVe, ELMo, and BERT embeddings rely on the idea that a word's semantic meaning is shaped by its distributional properties within a text corpus. The study by Das and Kamlish [29] shows that knowledge about words' meaning helps make summaries better and more accurate. This research benefits tasks like finding information, organizing documents, and extracting knowledge. The new method also makes summarizing text easier and reduces the need for manual work. It shows how important it is to understand language when using automated tools. This method helps deal with the vast amount of text we have today. Abro *et al*. [30] combined Word2Vec and GloVe embeddings with a neural network to improve the model. We then tested its performance using different learning rates across ten developers. The results showed that when Convolution was combined with Word2Vec embeddings, the model tended to be more accurate on average during testing.

2. REPRESENTING TEXT WITH EMBEDDINGS

This section provides a concise overview of the different types of word embeddings. We conduct a detailed analysis of a text, focusing on its word sequence to explore the contextual relationships between the embeddings.

2.1. Representation of Word Embeddings

Numerous methodologies, such as Word2Vec, GloVe, and FastText, are commonly used to examine word embeddings, each employing distinct approaches for capturing semantic relationships in a corpus. For example, Word2Vec uses neural networks to predict word context. At the same

time, GloVe captures co-occurrence statistics of words in a large corpus, and FastText considers subword information, making it more effective for morphologically rich languages [31-33]. One common approach involves utilizing a one-hot encoding technique, which assigns a distinct index within a vocabulary dictionary to each word, creating a unique representation for every word in the corpus. A comment is demonstrated by a vector consisting of all zeros except for one within an appropriate context. The process of studying context-based prediction involves the utilization of word embedding strategies. These strategies enable mapping one-hot vectors to more compact representations, often with lower dimensions than vocabulary dimensions. These representations' components capture the language data's underlying symbolic meaning. The fundamental premise is that to achieve accurate word prediction, it is necessary to enhance and refine the representations of words through learning [11].

2.1.1. Word2vec

The Word-to-Vector (Word2vec) technique under consideration relies entirely on a predictive approach that can be implemented using the skip-gram (SG) and continuous bag-of-words (CBOW) models [31]. Small neural networks are used in Continuous Bag-of-Words (CBOW) and Skip-Gram (SG) models to map words to specific points in a vector space. The distinction between these methods lies in whether the neural network endeavours to forecast a target term given its context (Continuous Bag of Words, CBOW) or vice versa. Two crucial factors determine the training of word2vec embeddings, as shown in Figure 2 [34]. Firstly, the embedding dimension is between fifty and five hundred through experimental methods. Secondly, the span of the context window refers to the number of words preceding and following the target word that is utilized as context for training the word embeddings. Additional significant hyperparameters are elaborated upon in the appendix section. The requirement for a more extensive training dataset is typically observed when training embeddings with more dimensions. It is crucial for each dimension to effectively capture a distinct aspect of meaning so that the embeddings possess the necessary capacity to differentiate between words. Paragraph2Vec and Doc2Vec are variants of the word2vec model

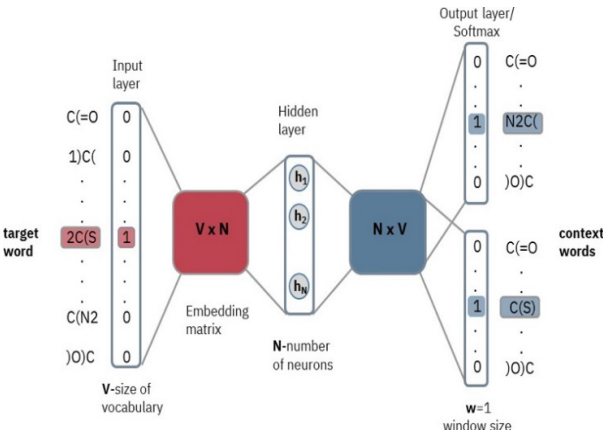

Fig. 2. Illustration of the Skip-Gram Architecture in the Word2Vec Algorithm [34].

designed to represent documents or paragraphs as vectors rather than individual words. There are two distinct types of doc2vec models: the PV-DM model, like the SG model of word2Vec, and the PV-DBOW model; both are used to distribute memory across paragraphs [35].

2.1.2. GloVe

Global Vectors for Word Representation (GloVe) model comprehends word embeddings proficiently using a word co-occurrence matrix rather than a word calculation job. A co-occurrence matrix is a VXV square matrix in which V indicates the vocabulary size. Every matrix element represents the frequency of occurrence of the specified vocabulary objects within a predetermined context window that spans the entire corpus. GloVe can comprehend vector embeddings, which facilitate the reduction of literal errors during the processing of co-occurrence statistics required by the model. Additionally, it considers the global co-occurrence statistics present in the preparation corpus. The model comprises multiple hyperparameters that must be vigilantly chosen, such as the dimension of the vector embedding and the size of the perspective window. The word vectors generated through the GloVe method exhibit epistemological equivalence to those obtained through word2vec. However, GloVe employs a count-based model as its foundation, in contrast to word2vec's predictive model [35]. GloVe, compared to word2vec, is known for its ability to capture longer-term dependencies due to its computation of statistics over more oversized context windows. However, it is essential to note that the order of these

dependencies needs to be preserved. Based on empirical observations, discernible advantage has yet to be identified for either GloVe or word2vec models. The overall reliability of these models is contingent upon various factors, such as the kind of data and the specific assessment job being measured. GloVe embeddings have proven highly effective in capturing semantic relationships in various natural language processing tasks, as demonstrated in recent studies across different application domains [36-38].

2.1.3. FastText

The FastText model extends word2vec and GloVe methods, incorporating a specific constraint. This model can lever novel, out-of-vocabulary (OOV) expressions by utilizing the word2vec skip-gram (SG) model, which includes inner subword data in character n-grams, representing sequences of adjacent characters. This approach entails constructing a vector representation for a word by considering the combination of its subword elements. This approach also enables the model to capture the structural and linguistic relationships between words and facilitates the creation of vectors for previously unanticipated words. Probabilistic FastText is a methodology used to combine FastText and Gaussian Mixture Models (GMMs). Howard and Ruder [39] did not provide any text to rewrite. The representation of each word is depicted as a Gaussian Mixture Model (GMM) consisting of n mechanisms, effectively capturing n distinct senses or meanings associated with the word. This representation can analyze the sub-word structure, distinguish between dissimilar word senses, and deliver improved representations of infrequent or hidden words. Recently, FastText has been widely applied in various fields of natural language processing, demonstrating its effectiveness in tasks such as sentiment analysis, cybersecurity, and machine learning applications [40-43].

2.1.4. ELMo

Embedding from Language Model (ELMo) is a language representation model emphasising words through character-level and word-level embeddings. Instead of employing a stable embedding for every word, ELMo evaluates the entire sentence and assigns each word an embedding [44]. The embeddings are constructed utilizing a trained bidirectional recurrent neural network (RNN) for a particular task. The embedding's bidirectional architecture is based on both preceding and subsequent words. One significant innovation of ELMo is the incorporation of task-specific weighting coefficients for the embeddings. It allows the model to be trained on one objective or task and then applied to a different task, effectively combining shared information while focusing on specific semantic aspects. Integrating ELMo word embeddings with deep learning multimodal transformers has shown promising results in enhancing image description tasks, as demonstrated in recent research by Cheng *et al*. [45]. In a recent study by Rong *et al*. [46], the advancements in multimodal deep learning, particularly integrating ELMo word embeddings with transformers, have significantly improved image description capabilities.

2.1.5. CoVe

Contextualized Word Vectors (CoVe) use a profound Long Short-Term Memory (LSTM) encoder derived from a cognitive sequence-to-sequence model specializing in machine translation [47]. This method is cast to provide word vectors with context. CoVe word embeddings are a mechanism for processing the entire input sequence. From an architectural standpoint, this model is characterized by its simplicity and lack of logic. The initial step is deleting the dual-layer, one-way LSTM encoder from the machine translation (MT) model. The process encrypts the static pre-trained GloVe embeddings used as context vectors. These context vectors are appended to the GloVe embeddings and provided as input to subsequent NLP tasks. CoVe has improved in numerous NLP charges, including sentiment analysis, question classification, entailment, and question answering [48]. Furthermore, it has brought attention to the dynamic manifestation of language.

2.1.6. ULMFit

The technique known as universal language model fine-tuning (ULMFit) was initially presented by Luong *et al*. [49]. This approach frequently employs language modelling, specifically utilizing LSTM networks to leverage extensive untagged statistics effectively. Precisely, the ULMFit model consists of three distinct phases:

- To acquire knowledge about linguistic

characteristics, a language model undergoes training using a substantial corpus of commonly used language.

- Subsequently, the model is refined by training it on a specific corpus of job-related texts, allowing it to understand job-specific language patterns better.
- Lastly, the model undergoes an advanced fine-tuning process, incorporating objective classification of job-related entities.

Moreover, the previously proposed two effective strategies, namely slanted triangular studying rates and discriminative fine-tuning, to enhance the fine-tuning process of objective-domain language models. Each neural network layer possesses distinct significance, with higher layers capturing semantic information and lower layers representing syntactic information. Hence, it is imperative to consider the unshared learning rate as the primary indicator of prejudicial fine-tuning. Ideally, the model can retain the knowledge acquired from the standard domain data while developing the latest features specific to the target domain. Additionally, the algorithm must exhibit rapid convergence towards an appropriate initial state during training, gradually refining the parameters. In pursuit of this objective, the proposal suggests using sloped triangular learning rates, characterized by a gradual initial increase followed by a subsequent linear decay. One main advantage of using a small learning rate is effectively preserving data in the pre-trained parameter. To address the issue of catastrophic forgetting, researchers proposed a gradual unfreezing approach known as the step-by-step unfreezing mechanism. This involves unfreezing the pre-trained model, starting from the last layer and progressing gradually. Additionally, three new fine-tuning tactics were introduced, which have gained popularity in subsequent research. As a result of these advancements, ULMFit has demonstrated superior performance to the present state-of-the-art models across seven text classification tasks.

2.1.7. XLNet

XLNet (eXtra Long Network) is a generalized autoregressive pretraining model for language understanding. It extends the Transformer-XL model and improves upon BERT by leveraging a permutation-based autoregressive approach to model word sequences. It allows XLNet to capture bidirectional context while maintaining the advantages of autoregressive models [50]. BERT is an autoencoding pre-training method connected to the latest autoregressive (AR) techniques used to calculate a text corpus's probability distribution using autoregressive models such as GPT and ELMo. The primary goal of BERT is to rebuild the unique data from corrupted input. Given that the compactness approximation is excluded from the objective of the BERT model, it can readily leverage bidirectional contexts for reconstruction purposes. Furthermore, this analysis aims to provide a comprehensive examination of the observed advantages of this method compared to the most recent augmented reality (AR) techniques. Nevertheless, using Artificial representations, such as [MASK], has indicated a discrepancy between the pre-training and fine-tuning processes, leading toward a need for more consistency between the two. Moreover, the BERT model adheres to the notion that the predicted tokens are independent, potentially compromising its ability to effectively capture long-range, high-order dependencies prevalent in natural language. A pre-trained non-specialized autoencoder (AE) strategy relies on transformer-xl to address such issues. This approach leverages bidirectional contexts to enhance the predictability of various factorization orders and surpasses the limitations of BERT through autoregressive preparation. Moreover, XLNet employs the permutation language modelling objective, combining the compensations of autoencoding and autoregressive approaches while mitigating their limitations [51].

2.1.8. BERT

The language representation model Bidirectional Encoder Representations from Transformers (BERT), developed by Devlin *et al*. [52], is designed to understand language by looking at both the left and right context of words in a sentence at the same time, making it more effective than earlier models [39, 44] that only considered one direction at a time. In the study, Devlin *et al*. [52] proposed a model founded upon a multilayer bi-directional transformer-encoder that serves as a contextualized word representation model. Unlike traditional sequential recurrence, this model employs parallel tending layers within the transformer neural network. The present model has undergone pre-training on two unsupervised tasks: The proposed

approach involves utilizing a covered language model, wherein approximately 15% of the tokens are unsystematically replaced with a unique "[MASK]" token. The standard is then proficient to predict the masked tickets. Additionally, a subsequent sentence prediction (NSP) task is employed, wherein the model is presented with a set of sentences and trained to identify whether the second sentence logically follows the first. This second task aims to gather more information on enduring or practical aspects. BERT is trained on a corpus of a book and text paragraphs sourced from the English language Wikipedia. The corpus contains approximately eight hundred words. Two sizes are available for pre-trained BERT models, namely BERT-base and BERT-large. BERT can be employed by directly using the pre-trained model on unannotated data or fine-tuning it on task-specific data. The pre-existing anonymized model and accompanying code for fine-tuning can be accessed through online platforms. The user's text needs more information to be rewritten academically. Numerous domain-specific iterations of BERT have undergone training or fine-tuning on text specific to a particular domain. Some examples of these iterations include:

• **BioBERT** is a modification of the BERT model specifically adapted for biomedical script analysis [53]. Its architecture has been modified and pre-trained using a large corpus of PubMed descriptions and PMC full-text snippets. The system is optimized for biomedical text mining tasks, including question answering, entity identification NER, and relation extraction.

• **ClinicalBERT** is proficient in clinical text from the publicly available mimic-iii database, which contains about 2 million clinical notes [54]. The model was introduced to the following types of messages:

i. Clinical BERT
ii. Clinical bioBERT
iii. Discharge summary BERT
iv. Discharge summary bioBERT

• **SciBERT** proficiently uses an arbitrary sample of 1.14 million semantic scholar full-text papers. SciBERT is a model that undergoes unsupervised training on various scientific publications from various fields. This pre-training helps boost its effectiveness in handling scientific NLP tasks [55]. There are four forms of Seibert:

i. The Cased (Both uppercase and lowercase vocabulary).
ii. The Uncased (Only lowercase vocabulary).
iii. Those models which are using BaseVocab.
iv. Those models are models using SciVocab and are trained from scratch.

2.1.9. MorphoRNN

Using word n-grams enables more efficient exploitation of the complex internal semantics. However, English is characterized by numerous meaningful affixes, including prefixes, roots, and suffixes. In a study, Sennrich *et al*. [56] introduced the concept of morphology to progress the learning process of word embeddings. A representation of the subword is obtained by training the fix with RNN. The embedding of the parent word is determined by considering all morphemes except those discussed by Xu and Liu [57], who focus on morphological aspects. RNN models linguistic units on a morpheme level instead of a word level. In their analysis, scholars consider the morpheme the fundamental natural language unit, conveying a unique vector to each morpheme for classification purposes. The embeddings of morphological texts are derived from the embeddings of their basic morphemes. An additional parent word embedding is derived by combining a stem and affix embedding.

2.1.10. MWE

Multi-Word Expressions (MWEs) are fixed or semi-fixed expressions that consist of multiple words but function as a single semantic unit [58]. Using word vector models to incorporate prior knowledge is a commonly employed technique for improving performance. The general practice represents a word's suffix, root, and prefix as separate tokens. The objective of MWE is to use a stylized approach in conveying the combined significance of a suffix and a prefix [57]. The model has been constructed based on the hypothesis that all meanings of morphemes in a token have equal help to the given structure of tokens, denoted as w = {w1, w2, ..., wn}. We aim to determine the meanings for each morpheme, characterized as mi, for wi (where i ranges from 1 to n). The term "mi" can be conceptually divided into three distinct components, namely "pi," "ri," and "si." These components represent the prefix denoting a collection or set, the root indicating a

group or set in addition to the suffix signifying a set of "wi." Hence, in cases where we serve as the contextual basis for wj, the altered representation of wi.

Furthermore, numerous studies focus on news at the sub-word (SW) level. In their research, Sennrich *et al*. [56] introduced the Byte Pair Encoding (BPE) approach, which involves merging frequently occurring neighbouring components within the subword to enhance word representation. Additionally, the authors demonstrated that their method outperformed alternative strategies in the milieu of neural machine translation tasks. Ustun *et al*. [59] introduced an enhanced logarithmic bilinear model (LBL) and emphasized its role in assigning morpheme labels. We have compared all pre-trained models in Table 1. The research demonstrated that this approach resulted in word embeddings that effectively preserved morphological interactions. Bian *et al*. [60] combined graphical and textual representations to enhance the effectiveness of word embeddings, demonstrating improvements through experiments involving word analogy, uniformity, and completion tasks. Their method employs a forward LSTM model to capture the prefix and root of a word and a reverse LSTM model to acquire the suffix and root, focusing on character-level information within a word. Cao and Rei [61] introduced a char2vec model using a Bidirectional Long Short-Term Memory (Bi-LSTM) network to generate embeddings for fictional representations, succeeding in morpheme boundary recovery and syntactic analogy tasks. Regarding morpheme boundary recovery, the researchers demonstrated that their morphological exploration was like that of specialized morphological investigators. Additionally, their research performed well in answering syntactic analogies. Kim *et al*. [62] introduced a novel approach that utilizes Convolutional Neural Networks (CNN) with max pooling for word embeddings. They also demonstrated that their proposed model could reduce the number of variables while enhancing performance.

2.2. Visualization of Word Embeddings

The newer word embedding techniques represent words in high-dimensional vector spaces, which allows them to learn subtle semantic relationships between words. The disadvantage of high-dimensional embeddings is that they are difficult to interpret. In most cases, such embeddings must be projected in a 2D or 3D space to facilitate critical analysis and interpretation. Another popular method of word-embedding visualization is t-SNE, which projects the embeddings into a lower dimension while trying to preserve their local structure. It has effectively shown semantic clustering, making word-embedding models more

Table 1. Comparison of pre-trained models.

Method	Architecture	Encoder	Decoder	Objective
Word2Vec	NN	No	No	Skip-gram and CBOW
GloVe	Matrix factorization	No	No	Global word-word co-occurrence Statistics
FastText	NN	No	No	Skip-gram and CBOW with sub-word information
ELMo	LSTM	Yes	Yes	Language modelling
CoVe	LSTM	Yes	No	Language modelling and word prediction
UMLFit	NN	No	No	Unsupervised machine learning of word embedding
XLNet	Transformer	Yes	No	Masked language modelling and next-sentence prediction
BERT	Transformer	Yes	No	Masked language modelling and next-sentence prediction
Morpho-RNN	NN	Yes	No	Language modelling with morphological information
MWE	NN	Yes	No	Language modelling with multi-word expression information

interpretable since related words congregate in low-dimensional spaces [63]. The effectiveness of the visualization method in many studies conducted on the performance of various word embedding algorithms, such as t-SNE. For example, t-SNE has been useful in visualizations to show clear clusters of word embeddings emanating from models such as Word2Vec and FastText [64]. These visualizations display groups of semantically similar words, offering insights into the quality of representations learned by recent research conducted by Bandyopadhyay *et al.* [65] through embeddings used in natural language understanding tasks.

These visualization techniques have also helped compare word embeddings from a classic model and those taken out of more advanced deep learning architectures, such as transformers. The research by Robinson and Pierce-Hoffman [66] have shown the importance of visualising the contextualized embeddings that transformer models create, such as BERT, where t-SNE and PCA have been used to contrast semantic similarities and differences amongst word contexts. Such a method is useful in understanding how modern embedding techniques handle word polysemy and context-dependent meanings, giving the embeddings more interpretability for downstream applications. Similarly, word embedding visualizations have discovered significant findings in health on the relationship of medical terms with their contexts. Therefore, recent efforts leveraged visualization methods to analyze embeddings derived from electronic health record data that exposed meaningful semantic relationships, helping in predictive modelling and decision-making processes [67]. The clustering patterns emerging from such visualizations have played an instrumental role in enhancing the interpretability of medical embeddings, particularly for identifying semantically related diagnostic phrases or treatment options.

Recent developments in embedding visualization have included MDS and UMAP, which are increasingly used with t-SNE for improved interpretability and scalability. These methods have showcased more intuitive visualizations of word embedding spaces, especially when dealing with large datasets or intricate models. Besides clustering words, they can also expose outliers, anomalies, and rare occurrences of words in embedding spaces, which gives further detailed insight into how different models represent such rare terms and contexts [68]. Word embedding visualization is an essential tool in embedding model analysis and interpretation, helping researchers further understand relationships and structures within data. Visualization techniques such as t-SNE, PCA, and UMAP remain vital in handling the quality of embeddings, especially in their evolution with sophisticated deep learning architectures. These visualizations allow intuition to more easily understand how embeddings encode semantic information of critical importance in developing natural language processing and beyond.

3. HISTORY OF PRE-TRAINED MODELS

Pre-training has consistently been regarded as a highly effective methodology for acquiring knowledge about the variables within deep neural networks, which are refined through fine-tuning processes for downstream tasks [69]. The year 2006 witnessed a significant advancement in deep learning, as it saw the resurgence of the acquisitive layer-wise unsupervised pre-training technique, which was subsequently combined with supervised fine-tuning [70]. In computer vision, it is a common habit to initially train models on the extensive ImageNet dataset, followed by fine-tuning on smaller datasets for specific tasks. This approach exhibits notable advantages compared to an unplanned initialization, as the model acquires comprehensive image features that can be leveraged across various vision-related tasks. In NLP, it has been demonstrated that Pre-trained Models (PTMs) trained on extensive corpora are helpful for multiple downstream NLP tasks, ranging from basic word embeddings to complex deep neural models.

3.1. Abbreviations and Acronyms

The practice of representative words as fixed-length vectors has a longstanding historical background. The concept of "modern" word embedding was first introduced in the ground-breaking research of neural network language models (NNLM). Collobert *et al*. [71] demonstrated that pre-trained word embeddings on unlabelled data can significantly enhance performance in NLP tasks. To tackle the issue of computational complexity, the researchers opted to train word embeddings using a pairwise top-ranking

job rather than relying on language modelling. This study represents the initial endeavour to acquire universal word embeddings that can be utilized for various tasks using unannotated data. According to the findings presented by Mikolov *et al*. [72], it has been revealed that deep neural networks do not yield significant benefits in developing effective word embeddings. Skip-gram (SG) and Continuous bag-of-words (CBOW) models are two shallow Architectures proposed by the authors. Despite their simplicity, these methods can acquire high-quality word embeddings that capture the underlying syntactic and semantic similarities between words. Word2Vec is widely recognized as one of the most standard NLP model implementations. It facilitates the use of pre-trained word embeddings for multiple NLP tasks. Moreover, GloVe [32] is a popular model for gaining pre-trained word embeddings. These embeddings are derived from global word-word co-occurrence facts extracted from a corpus of considerable size. While pre-trained word embeddings are valuable in NLP charges, they often lack context sensitivity and are primarily trained using shallow models. When employed in a subsequent project, the entirety of the classic must still be acquired anew. Numerous researchers also endeavour to acquire embeddings of textual elements such as reading materials, sentences, or reports during the concurrent time frame. Examples of these efforts include the utilization of paragraph vectors [73], skip-thought vectors [74], and context2vec [75]. In contrast to their contemporary counterparts, these rudimentary sentence embedding models aim to transform entered sentences into a vector representation of stable dimensions instead of generating contextual words for individual tokens.

3.2. Second-generation PTMs: Pre-trained Contextual Encoders

Given that NLP endeavours extend outside the scope of individual words, it is customary to pre-train neural encoders at the sentence level or higher. The vectors produced by neural encoders, commonly called contextual word embeddings, modify the semantic representation of texts based on their surrounding perspective. The primary successful example of PTM for NLP was introduced by Dai and Le [76]. The authors employ a language model LM or a system autoencoder to digitize extended short-term memory networks LSTMs. They observe that pre-training can enhance the guidance process and improve the inductive reasoning capabilities of LSTMs in various text classification tasks. Liu *et al*. [77] conducted pre-training of a shared LSTM encoder using a language model LM and subsequently fine-tuned it within the multi-task learning MTL framework. The authors observed that incorporating pre-training and fine-tuning techniques can significantly enhance the enactment of MTL in various text classification tasks. Please provide more information or specify what you want me to rewrite academically. It has been observed that the performance of seq2seq models can be meaningfully enriched using unattended pre-training [78]. The encoder and decoder weights are initialized using pre-trained weights from two language models. Subsequently, these weights are fine-tuned using tagged data.

Furthermore, the contextual encoder can be pre-trained with a language model (LM). In a study, McCann *et al*. [47] utilized a pre-trained deep LSTM encoder derived from an attentional sequence-to-sequence model employed in machine translation (MT). The performance of various common NLP objectives can be enhanced by utilizing the context vectors (CoVe) generated by the pre-trained encoder. The contemporary post-translational modifications (PTMs) have advanced significantly compared to their precursor PTMs. They now possess enhanced proficiency in handling extensive corpora, employ more robust and intricate architectures such as transformers, and engage in novel pre-training tasks.

The model proposed by Peters *et al.* [44] is a pre-trained two-layer LSTM encoder incorporating a bidirectional language model (BiLM). This BiLM comprises both a forward language model and a reversed language model. The contextual representations generated by the Pre-trained BiLM, ELMo (embeddings from language models), have yielded significant improvements across diverse NLP tasks. Please provide more information or specify what you want me to rewrite academically. The word meaning was determined using contextual string embeddings pre-trained with a character-level language model [79]. However, these two pre-trained models (PTMs) are commonly employed as feature extractors to generate contextual word embeddings. These embeddings are then utilized as input for the primary model in downstream tasks.

The variables in question are held constant, while the previous model's remaining variables continue to be trained from the beginning. The ULMFit (universal language model fine-tuning) approach, as described by Howard and Ruder in [39], aimed to fine-tune a pre-trained language model (LM) for the task of text classification (TC). This method achieved state-of-the-art results on six commonly used TC datasets. The ULM-fit methodology comprises three distinct phases: (i) initial training of the language model (LM) using common-domain data; (ii) subsequent fine-tuning of the LM using target-specific data; and (iii) further fine-tuning of the LM on the specific target task. ULMFit additionally incorporates several valuable techniques for fine-tuning, namely prejudicial fine-tuning, sloped triangular learning rates, and step-by-step releasing.

Lately, there has been a growing recognition of the significant capabilities of deep pre-trained models (PTMs) in acquiring universal language representations. Prominent examples include OpenAI GPT (generative pre-training) and BERT (bidirectional encoder representation from the transformer). Additionally, there is a rising trend of introducing a greater variety of self-supervised tasks to enhance the acquaintance acquisition of these pre-trained models from vast text corpora. Following the emergence of ULMFit and BERT, fine-tuning has emerged as the prevailing method for adapting pre-trained models to suit downstream tasks.

4. CROSS-LINGUAL WORD EMBEDDINGS

When observing the presence of various languages, it becomes evident that approximately seven thousand languages are currently in use. However, it is essential to note that only a limited number of languages possess abundant human-interpreted resources. The task at hand involves acquiring cross-lingual shift learning of word embeddings. We employ a model trained on languages with great linguistic resources to accomplish this. This model then maps the input embeddings of languages with limited resources onto a joint semantic space. These embeddings are commonly referred to as cross-lingual word embeddings [80]. Founded on the classification of monolingual embeddings, cross-lingual embedding learning approaches can be categorized as dynamic or static. The static method has received considerable attention in recent studies, while numerous studies are currently investigating the active process. Additionally, these approaches can be categorized into offline and online categories based on the training objective. In general, online strategies aim to optimize cross-lingual and monolingual objectives simultaneously. Conversely, offline methods involve utilizing pre-trained monolingual word embeddings from different languages as involvement and mapping them into a shared semantic space [81]. As a survey by Ruder *et al*. [80] noted, most cross-lingual word embedding models are optimized using similar objective functions, and differences in performance often stem from data requirements rather than architecture.

4.1. Static Cross-lingual Word Embeddings

When examining still embeddings, it is observed that specific methods involve learning language models for both objective and source languages. These methods then collectively enhance their respective objectives by utilizing cross-lingual goals. An approach was proposed by Klementiev *et al*. [82] to acquire bilingual word embeddings and word alignments simultaneously. Subsequently, using the monolingual skip-gram model, the researchers endeavoured to develop proficiency in bilingual embeddings, encompassing both sentence and word-level alignments. The model proposed by Lample and Conneau [83] aims to acquire bilingual embeddings that enhance the semantic coherence of sentence pairs with a specific orientation.

Guan *et al*. [84] proposed a methodology for leveraging document-aligned similar corpora to acquire bilingual embeddings. The absorption of two aligned documents made this into a pseudo-bilingual paper, which was then used to train a skip-gram model. Offline methods involve learning a projection that facilitates the transformation of the source language's vector space to the target language's vector space. The acquisition of such a matrix can be achieved through a supervised approach, wherein the objective is to minimize the squared Euclidean distance, also known as the Mean Squared Error (MSE), between the target word embedding of a translated word and the converted source word embedding. The matrix can typically be acquired by replacing the mean squared error with a max-margin hinge loss or by employing

singular value decomposition. Bengio *et al*. [85] have introduced an alternative approach for aligning word embeddings in target and source languages. This method utilizes canonical correlation analysis (CCA) to project the embeddings onto a shared space. The researchers discovered that incorporating cross-lingual embeddings into dependency analysis and comprehensive supplementary features such as lexical characteristics and word clusters yielded significant performance improvements. The authors extended their research efforts and incorporated nonlinearity into the mapping procedure.

In addition to supervised approaches to cross-lingual embedding learning, unsupervised methods have also resigned promising outcomes. The initial step involved constructing a bilingual dictionary using adversarial learning techniques, as described by Radford *et al*. [86]. Subsequently, bilingual embeddings were generated, along with a modification approach. In another study, Peters *et al*. [44] introduced a similar framework that adopts a two-step approach for acquiring multilingual embeddings. Notably, this framework considers the interdependencies among numerous languages, a factor that previous research needs to consider. To address the challenges associated with uncertainty in acquiring cross-lingual embeddings for reserved language sets, Wang *et al*. [87] introduced a resilient framework. This framework enables learning a common multilingual embedding space by iteratively incorporating additional languages into the existing space.

4.2. Dynamic Cross-lingual Word Embeddings

Many researchers have explored and shared their findings and studies on dynamic word embeddings with cross-lingual transfer, drawing inspiration from the significant advancements made in active word embeddings for monolingual applications. In a study, one of the online approaches examined by Akbik *et al*. [79] focuses on ELMo, a model that heavily on which it heavily relies. This approach aims to create a polyglot model that captures character-level information from multilingual data to generate relative representations. Lample and Conneau [83] primarily centred on BERT and its objectives, explicitly examining the utilization of cross-lingual supervision from parallel data to investigate cross-lingual language models (XMLS). This approach yielded highly favourable results on various cross-lingual tasks, establishing a new benchmark in the field. Subsequently, the researchers demonstrated that large-scale pre-trained multilingual language models significantly improved evaluating cross-lingual transfer tasks. It highlights the potential of multilingual modelling, excluding compromising the evaluation of individual language-specific outcomes.

In contrast, offline methodologies have employed linear projection to generate contextualized pre-trained embeddings [60]. The approach used in our study involved utilizing averaged contextualized embeddings as a reference point for individual words and acquiring knowledge of the shift matrix within the reference space. Wang *et al*. [87] introduced a method for directly acquiring this transformation within the given context, preserving word sense in cross-lingual dynamic embeddings. McCann *et al*. [47] evaluated current methods for dynamic cross-lingual embeddings and demonstrated their significant potential in enhancing cross-lingual dependency parsing. Additionally, they have shown that online methodologies exhibit superior encoding of cross-lingual lexical correspondence compared to offline techniques.

4.3. Multilingual Word Embeddings

In addition to the practice of transferring embedding models from resource-rich to low-resource languages through a plan, there have also been efforts to train embedding models in multiple languages simultaneously. In their study, Bengio *et al*. [85] introduced a novel language model called multi-BERT. This model was trained on a collection of mono-lingual Wikipedia corpora from a total of 104 languages. Notably, the model exhibited exceptional performance in zero-shot cross-lingual model shifts. The researchers demonstrated through a diverse set of investigative experiments that the multi-BERT model possesses the ability to seamlessly transition between languages, even in the absence of any explicit lexical cues. It is achieved by effectively capturing and understanding multilingual contexts.

In addition, Wag *et al*. [88] investigated the multi-BERT model's generalisation ability. They devised an alternative method for transferring lexical information from a monolingual model

to new languages. The outcome challenges the prevailing notion that multi-BERT exhibits strong generalization capabilities due to its utilization of a shared sub-word vocabulary and simultaneous training across multiple languages. In contrast, it was suggested that the monolingual representations should acquire abstract concepts that can be applied to various languages.

5. EMBEDDING FOR OUT-OF-VOCABULARY WORDS

The word2vec model is known for its simplicity and efficiency in learning semantic representations of words from large data files [89]. However, it has limitations in learning embeddings for OOV texts. OOV words can be categorized into terms not in the open vocabulary and words not encountered in the current corpus [90]. OOV texts can be broadly classified into three forms. (i) The dynamic lexicon, precisely online terminology, is developing continuously. (ii) Proper names refer to specific entities such as geographical locations, organizations, individuals, automated expressions, and temporal references. (iii) Investigating the Terminology of Research Fields and Professional Titles. In academic discourse, various terminologies encompass elements, such as the titles of literary works or newly created artistic pieces, including documentaries or novels. In most instances, expanding one's vocabulary is the optimal approach. In addition to linguistic processing, there is an expansion of vocabulary, enabling us to delve into the realm of OOV words, particularly those that fall within the extensive range of less favoured components. Words that are often unfamiliar are typically ignored, removed, or replaced with an 'unknown' tag (UNK), which is an insufficient solution. Addressing the challenges posed by OOV words is crucial. Recently, various neural network-based models, such as FastText, MorphoRNN, and MWE, have been developed to tackle this issue effectively.

6. DATASETS AND EVALUATION FRAMEWORKS

The measurement assessment system for current word embeddings can be categorized based on intrinsic and extrinsic evaluation. However, these evaluation approaches have faced extensive criticism in the existing literature. In this fragment, we present a brief overview of the two types of evaluations discussed in the previous quarter and direct readers to recent research studies for a thorough explanation and analysis.

6.1. Intrinsic Evaluation

Intrinsic evaluations establish relationships between words by assessing their syntactic or semantic properties, relying on artificial assessments as a basis. Through careful observation of the methods employed to acquire these assessments, it is possible to categorize such approaches into two distinct types: absolute and relative intrinsic evaluation. In the initial category, the individual reviews are gathered before victimization, serving as a reference point for word embedding methodologies. In intrinsic evaluation, the comparative approach uses accessors to assess word embeddings candidly grounded on their performance in an exact word relation objective or charge [91]. Due to its independence from human involvement or interaction, the absolute form of intrinsic evaluation is frequently employed alongside comparative inherent evaluation. In the following section, we will briefly introduce several well-known assessment techniques. The method used for assessing semantic similarity, known as similarity checking, is widely utilized due to its effectiveness in determining the relationship between word distances in human heuristic judgments and embedding space. The test sets commonly employed in current research include WordSim-353 [92], Mammals, Entities, Natural kinds (MEN) [93], and SimVerb-3500 [94].

The word analogy technique has gained significant recognition due to its integration with the well-known CBOW and Skip-gram representations. In this context, the embeddings of three words, w, x, and y, are employed to forecast the word z. The objective is to identify z in a manner that maintains the exact relationship between w and x as y and z. As an illustration, let us consider the scenario where w represents Pakistan, b represents Islamabad, and c represents India. In this case, d would correspond to Delhi. Prominent examples of trial sets of this nature include the WordRep, Microsoft Research Syntactic Analogies Dataset, and Google Analogy [95]. The synonym detection technique assesses the capacity of embeddings to accurately identify the most similar word to a given word from a pool of candidates. When considering a specific goal

word, such as "levied," one must select among options such as "imposed" (correct), "believed," "requested," and "correlated." The datasets that could be utilized in this methodology encompass the Test of English as a Foreign Language (TOEFL), English as a Second Language (ESL), and Reading and Writing for Academic Purposes (RDWP) [96].

The word embedding space in the concept categorization technique is evaluated through clustering. This task categorizes a set of specific terms into distinct subsets. For instance, the words "goat" and "dog" will be classified under the mammal category, while "oranges" and "grapes" will be categorized as fruits [97]. The identification of verb-noun pairs in textual data is facilitated by utilizing a technique known as the Sectional Preference method. Commonly used word embeddings can identify verb-noun pairs in which the noun is the subject or object of the verb. For example, the noun "humanity" is often used as the subject instead of the object of the verb "serve." Greenberg, Sayeed and Danberg (GDS) [98], and Ulrike and Pado (UP) [99] are commonly employed lexical sets.

6.2. Extrinsic Evaluation

Word embeddings are used as feedback for downstream tasks and to measure the influence of these tasks using specific metrics in extrinsic evaluations. Word embeddings have demonstrated significant applicability across various functions in the NLP domain. These embeddings can be utilized for multiple parts, as perceiving all such tasks as non-essential assessments is theoretically possible. One category of downstream tasks within this field encompasses language modelling, named entity recognition, POS tagging, chunking, machine reading comprehension, sentiment analysis, semantic role labelling, dependency parsing, machine translation, and natural language inference [100]. The assumption inherent in these non-essential evaluations is that word embeddings that yield positive results in one task will also deliver positive results in other studies. This assumption has been extensively explored and analyzed in the existing literature. Empirical observations have provided evidence that distinct NLP tasks prefer specific embeddings. Therefore, although extrinsic evaluations can help compare embeddings about a particular mission or objective, they are not mentioned as metrics for the overall review of word embeddings' excellence.

7. CONCLUSIONS

In this review, we reflect upon the evolution and impact of word embeddings within the domain of NLP. Word embeddings have formed a crucial basis for carrying out manifold tasks in NLP and have revolutionized the way text is represented; hence, semantic understanding has proficiently been achieved. Through such a detailed critical analysis, we have showcased their relative strengths and limitations while considering a host of various NLP tasks, including but not limited to sentiment analyses and machine translations. The paper has critically discussed the evolution of word embeddings from static to contextual, from the traditional Word2Vec and GloVe models to more advanced BERT and ELMo models.

Such a comparison highlights the advantages of contextualized embeddings well, which even pushed the limit of word representation further by incorporating dynamic context and improving performance on downstream tasks. However, just like any other machine learning model, embedding biases, capturing long-range dependencies, and inefficiency with out-of-vocabulary words remain critical points of concern that continue to drive research and innovation.

We also reviewed some applications of cross-lingual embedding and why multilingual models contribute to more effective language transfer and alignment across diverse languages. The emergent techniques for handling OOV words, dynamically changing embeddings, and domain-specific models have opened new vistas for applying NLP, further showcasing versatility and scalability.

Overall, word embeddings have given an effective way of encoding semantic information that furthers NLP. However, much research still needs to be carried out regarding critical challenges, such as bias, generalization toward poor-resource languages, and handling linguistic complexities. Ongoing developments of more sophisticated models and hybrid approaches are bound to shape the future of NLP, enabling highly accurate and meaningful language understanding in and out of general and specialist contexts.

8. CONFLICT OF INTEREST

The authors declare no conflict of interest.

9. REFERENCES

1. C. Liu and K.K.H. Chung. The relationships between paired associate learning and Chinese word writing in kindergarten children. *Reading and Writing* 34(8): 2127-2148 (2021).
2. P. Aceves and J.A. Evans. Mobilizing conceptual spaces: How word embedding models can inform measurement and theory within organization science. *Organization Science* 35(3): 788-814 (2024).
3. A. Berenguer, J.-N. Maz'on, and D. Tom'as. Word embeddings for retrieving tabular data from research publications. *Machine Learning* 113(4): 2227-2248 (2024).
4. F. Incitti, F. Urli, and L. Snidaro. Beyond word embeddings: A survey. *Information Fusion* 89: 418-436 (2023).
5. M. Toshevska. The Ability of Word Embeddings to Capture Word Similarities. *International Journal on Natural Language Computing (IJNLC)* 9(3): 25-42 (2020).
6. M.-C. Hung, P.-H. Hung, X.-J. Kuang, and S.-K. Lin. Intelligent portfolio construction via news sentiment analysis. *International Review of Economics and Finance* 89: 605-617 (2024).
7. K. Das, F. Abid, J. Rasheed, Kamlish, T. Asuroglu, S. Alsubai, and S. Soomro. Enhancing Communication Accessibility: UrSL-CNN Approach to Urdu Sign Language Translation for Hearing-Impaired Individuals. *CMES-Computer Modeling in Engineering and Sciences* 141(1): 689-711 (2024).
8. B. Lal, R. Gravina, F. Spagnolo, and P. Corsonello. Compressed sensing approach for physiological signals: A review. *IEEE Sensors Journal* 23(6): 5513-5534 (2023).
9. A. Baloch, T.D. Memon, F. Memon, B. Lal, V. Viyas, and T. Jan. Hardware synthesis and performance analysis of intelligent transportation using canny edge detection algorithm. *International Journal of Engineering and Manufacturing* 11(4): 22-32 (2021).
10. E. Çano and M. Morisio. Word Embeddings for Sentiment Analysis: A Comprehensive Empirical Survey. *Preprint ArXiv* 1: 1902.00753 (2019).
11. F.K. Khattak, S. Jeblee, C. Pou-Prom, M. Abdalla, C. Meaney, and F. Rudzicz. A survey of word embeddings for clinical text. *Journal of Biomedical Informatics* 100: 100057 (2019).
12. A. Agarwal, B. Agarwal, and P. Harjule. Understanding the Role of Feature Engineering in Fake News Detection. In: *Soft Computing: Theories and Applications: Proceedings of SoCTA 2021*, *Singapore* pp. 769-789 (2022).
13. R.A. Stein, P.A. Jaques, and J.F. Valiati. An analysis of hierarchical text classification using word embeddings. *Information Sciences* 471: 216-232 (2017).
14. J.E. Font and M.R. Costa-Jussà. Equalizing Gender Biases in Neural Machine Translation with Word Embeddings Techniques. *Preprint ArXiv* 2: 1901.03116 (2019).
15. S.M. Rezaeinia, R. Rahmani, A. Ghodsi, and H. Veisi. Sentiment analysis based on improved pre-trained word embeddings. *Expert Systems With Applications* 117: 139–147 (2019).
16. Z. Yao, Y. Sun, W. Ding, N. Rao, and H. Xiong. Dynamic Word Embeddings for Evolving Semantic Discovery. *Preprint ArXiv* 2: 1703-00607 (2018).
17. J. Zhao, T. Wang, M. Yatskar, R. Cotterell, V. Ordonez, and K.-W. Chang. Gender Bias in Contextualized Word Embeddings. *Preprint ArXiv* 1: 1904.0331 (2019).
18. Q. Du, N. Li, W. Liu, D. Sun, S. Yang, and F. Yue. A Topic Recognition Method of News Text Based on Word Embedding Enhancement. *Computational Intelligence and Neuroscience* 2022(1): 4582480 (2022).
19. D. Suhartono, K. Purwandari, N.H. Jeremy, S. Philip, P. Arisaputra, and I.H. Parmonangan. Deep neural networks and weighted word embeddings for sentiment analysis of drug product reviews. *Procedia Computer Science* 216: 664-671 (2023).
20. S. Haller, A. Aldea, C. Seifert, and N. Strisciuglio. Survey on Automated Short Answer Grading with Deep Learning: from Word Embeddings to Transformers. *Preprint ArXiv* 1: 2204.03503 (2022).
21. A. Çalışkan, P.P. Ajay, T. Charlesworth, R. Wolfe, and M.R. Banaji. Gender Bias in Word Embeddings: A Comprehensive Analysis of Frequency Syntax, and Semantics. *Preprint ArXiv* 1: 2206.03390 (2022).
22. X. Tang, Y. Zhou, and D. Bollegala. Learning Dynamic Contextualised Word Embeddings via Template-based Temporal Adaptation. *Preprint ArXiv* 3: 2208-10734 (2023).
23. K. Alnajjar, M. Hämäläinen, and J. Rueter. Sentiment Analysis Using Aligned Word

Embeddings for Uralic Languages. *Preprint ArXiv* 1: 2305.15380 (2023).

24. H. Yen and W. Jeon. Improvements to Embedding-Matching Acoustic-to-Word ASR Using Multiple-Hypothesis Pronunciation-Based Embeddings. *Preprint ArXiv* 2: 2210.16726 (2023).
25. J. Engler, S. Sikdar, M. Lutz, and M. Strohmaier. SensePOLAR: Word sense aware interpretability for pre-trained contextual word embeddings. *Preprint ArXiv* 1: 2301.04704 (2023).
26. R. Schiffers, D. Kern, and D. Hienert. Evaluation of Word Embeddings for the Social Sciences. *Preprint ArXiv* 1: 2302.06174 (2023).
27. O. Zaland, M. Abulaish, and M. Fazil. A Comprehensive Empirical Evaluation of Existing Word Embedding Approaches. *Preprint ArXiv* 2: 2303.07196 (2024).
28. P.J. Worth. Word Embeddings and Semantic Spaces in Natural Language Processing. *International Journal of Intelligence Science* 13(1): 1-21 (2023).
29. K. Das and Kamlish. Enhancing Automated Text Summarization: A Survey and Novel Method with Semantic Information for Domain-Specific Summaries. *Journal of Computing & Biomedical Informatics* 5(2): 102-113 (2023).
30. Z.F. Abro, S.U. Rehman, K. Das, and A. Goswami. An Analysis of Deep Neural Network for Recommending Developers to Fix Reported Bugs. *Europan Journal of Science and Technology* (24): 375-379 (2021).
31. T. Mikolov, K. Chen, G. Corrado, and J. Dean. Efficient estimation of word representations in vector space. *Preprint ArXiv* 3: 1301.3781 (2013).
32. J. Pennington, R. Socher, and C. Manning. Glove: Global Vectors for Word Representation. *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP), Doha, Qatar* (25-29 October, 2014) pp. 1532-1543 (2014).
33. P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics* 5: 135-146 (2017).
34. H. Öztürk, A. Özgür, P. Schwaller, T. Laino, and E. Ozkirimli. Exploring Chemical Space Using Natural Language Processing Methodologies for Drug Discovery. *Preprint ArXiv* 1: 2002-06053 (2020).
35. F. Torregrossa, R. Allesiardo, V. Claveau, N. Kooli, and G. Gravier. A survey on training and evaluation of word embeddings. *International Journal of Data Science and Analytics* 11(2): 85-103 (2021).
36. G. Curto, M.F.J. Acosta, F. Comim, and B. Garcia-Zapirain. Are AI systems biased against the poor? A machine learning analysis using Word2Vec and GloVe embeddings. *AI and Society* 39(2): 617-632 (2024).
37. P. Rakshit and A. Sarkar. A supervised deep learning-based sentiment analysis by the implementation of Word2Vec and GloVe Embedding techniques. *Multimedia Tools and Applications* pp. 1-34 (2024).
38. M. Greeshma and P. Simon. Bidirectional Gated Recurrent Unit with Glove Embedding and Attention Mechanism for Movie Review Classification. *Procedia Computer Science* 233: 528-536 (2024).
39. J. Howard and S. Ruder. Universal Language Model Fine-Tuning for Text Classification. *Preprint ArXiv* 5: 1801.06146 (2018).
40. A. Faruq, M. Lestandy, and A. Nugraha. Analyzing Reddit Data: Hybrid Model for Depression Sentiment using FastText Embedding. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)* 8(2): 288-297 (2024).
41. S.S. Alqahtani. Security bug reports classification using fasttext. *International Journal of Information Security* 23(2): 1343-1358 (2024).
42. D. Voskergian, R. Jayousi, and M. Yousef. Enhanced TextNetTopics for Text Classification Using the GSM Approach with Filtered fastText-Based LDA Topics and RF-Based Topic Scoring: fasTNT. *Applied Science* 14(19): 8914 (2024).
43. N.A. Nasution, E.B. Nababan, and H. Mawengkang. Comparing LSTM Algorithm with Word Embedding: FastText and Word2Vec in Bahasa Batak-English Translation. *12th International Conference on Information and Communication Technology (ICoICT)*, *Bandung, Indonesia* (7 Aug - 8 Aug, 2024) pp. 306-313 (2024).
44. M.E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer. Deep Contextualized Word Representations. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, *New Orleans, Louisiana* (1-6 June, 2018) pp. 2227-2237 (2018).
45. X. Cheng, T. Mei, Y. Zi, Q. Wang, Z. Gao, and H. Yang. Algorithm Research of ELMo Word Embedding and Deep Learning Multimodal Transformer in Image Description. *Preprint ArXiv*: 2408.06357 (2024).
46. L. Rong, Y. Ding, M. Wang, A.E. Saddik, and M.S.

Hossain. A Multi-Modal ELMo Model for Image Sentiment Recognition of Consumer Data. *IEEE Transactions on Consumer Electronics* 7(1): 3697-3708 (2024).

47. B. McCann, J. Bradbury, C. Xiong, and R. Socher. Learned in Translation: Contextualized Word Vectors. *Advances in Neural Information Processing Systems* 30: 1-12 (2017).
48. W. Khan, A. Daud, K. Khan, S. Muhammad, and R. Haq. Exploring the frontiers of deep learning and natural language processing: A comprehensive overview of key challenges and emerging trends. *Natural Language Processing Journal* 4: 100026 (2023).
49. T. Luong, R. Socher, and C.D. Manning. Better Word Representations with Recursive Neural Networks for Morphology. *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, *Sofia, Bulgaria* (8-9 August, 2013) pp. 104–113 (2013).
50. Z. Yang. XLNet: Generalized Autoregressive Pretraining for Language Understanding. *Preprint ArXiv* 2: 1906.08237 (2019).
51. A.F. Adoma, N.-M. Henry, and W. Chen. Comparative analyses of Bert, Roberta, Distilbert, and Xlnet for Text-Based Emotion Recognition. *17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP)*, *IEEE* pp. 117-121 (2020).
52. J. Devlin, M.W. Chang, K. Lee, and K. Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. *Preprint ArXiv* 2: 1810.04805 (2018).
53. K. Huang, J. Altosaar, and R. Ranganath. Clinicalbert: Modeling clinical notes and predicting hospital readmission. *Preprint ArXiv* 3: 1904.05342 (2019).
54. J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C.H. So, and J. Kang. BioBERT: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics* 36(4): 1234-1240 (2019).
55. I. Beltagy, K. Lo, and A. Cohan. SciBERT: A pretrained language model for scientific text. *Preprint ArXiv* 3: 1903.10676 (2019).
56. R. Sennrich, B. Haddow, and A. Birch. Neural Machine Translation of Rare Words with Subword Units. *Preprint ArXiv* 5: 1508.07909 (2016).
57. Y. Xu and J. Liu. Implicitly Incorporating Morphological Information into Word Embedding. *Preprint ArXiv* 3: 1701.02481 (2017).
58. T. Baldwin and S.N. Kim. Multiword expressions. In: Handbook of Natural Language Processing. 2nd Edition. N. Indurkhya and F.J. Damerau (Eds.). *Taylor and Frances Group* pp. 267-292 (2010).
59. A. Üstün, M. Kurfalı, and B. Can. Characters or Morphemes: How to Represent Words?. *Proceedings of the 3rd Workshop on Representation Learning for NLP, Melbourne, Australia* (20 July 2018) pp. 144-153 (2018).
60. J. Bian, B. Gao, and T.-Y. Liu. Knowledge-powered deep Learning for word embedding. In: Machine Learning and Knowledge Discovery in Databases. T. Calders, F. Esposito, E. Hüllermeier, and R. Meo (eds). *Springer, Berlin, Heidelberg* pp. 132–148 (2014).
61. K. Cao and M. Rei. A joint Model for Word Embedding and Word Morphology. *Preprint ArXiv* 1: 1606.02601 (2016).
62. Y. Kim, Y. Jernite, D. Sontag, and A.M. Rush. Character-Aware Neural Language Models. *Preprint ArXiv* 4: 1508.06615 (2016).
63. D. Smilkov, N. Thorat, C. Nicholson, E. Reif, F.B. Viegas, and M. Wattenberg. Embedding projector: Interactive visualization and interpretation of embeddings. *Preprint ArXiv* 1: 1611.05469 (2016).
64. S. Liu, P.-T. Bremer, J.J. Thiagarajan, V. Srikumar, B. Wang, Y. Livnat, and V. Pascucci. Visual Exploration of Semantic Relationships in Neural Word Embeddings. *IEEE Transactions on Visualization and Computer Graphics* 24(1): 553-562 (2018).
65. S. Bandyopadhyay, J. Xu, N. Pawar, and D. Touretzky. Interactive visualizations of word embeddings for k-12 students. *Proceedings of the AAAI Conference on Artificial Intelligence* 36(11): 12713-12720 (2022).
66. I. Robinson and E. Pierce-Hoffman. Tree-SNE: Hierarchical clustering and visualization using t-SNE. *Preprint ArXiv* 1: 2002.05687 (2020).
67. N. Oubenali, S. Messaoud, A. Filiot, A. Lamer, and P. Andrey. Visualization of medical concepts represented using word embeddings: a scoping review. *BMC Medical Informatics and Decision Making* 22(1): 83 (2022).
68. M. Gniewkowski and T. Walkowiak. Assessment of document similarity visualization methods. In: Human Language Technology. Challenges for Computer Science and Linguistics. LTC 2019. Lecture Notes in Computer Science. Z. Vetulani, P. Paroubek, and M. Kubis (Eds.). *Springer, Cham* pp. 348-363 (2019).
69. X. Han, Z. Zhang, N. Ding, Y. Gu, and *et al*. Pre-trained models: Past, present and future. *AI Open*

2: 225-250 (2021).

70. D. Erhan, Y. Bengio, A. Courville, P.-A. Manzagol, and P. Vincent. Why Does Unsupervised Pre-training Help Deep Learning?. *Journal of Machine Learning Research* 1: 625-660 (2010).
71. R. Collobert, J. Weston, L. Bottou, M. Karlen, K. Kavukcuoglu, and P. Kuksa. Natural Language Processing (almost) from Scratch. *Preprint ArXiv* 1: 1103.0398 (2011).
72. T. Mikolov, I. Sutskever, K. Chen, G. Corrado, and J. Dean. Distributed Representations of Words and Phrases and their Compositionality. *Preprint ArXiv* 1: 1310-4556 (2013).
73. Q.V. Le and T. Mikolov. Distributed Representations of Sentences and Documents. *Preprint ArXiv* 2: 1405-4053 (2014).
74. J. Pilault, J. Park, and C. Pal. On the impressive performance of randomly weighted encoders in summarization tasks. *Preprint ArXiv* 1: 2002.09084 (2020).
75. O. Melamud, J. Goldberger, and I. Dagan. context2vec: Learning Generic Context Embedding with Bidirectional LSTM. *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, Berlin Germany* (7-12 August 2016) pp. 51-61 (2016).
76. A.M. Dai and Q.V. Le. Semi-supervised Sequence Learning. *Preprint ArXiv* 1: 1511-01432 (2015).
77. P. Liu, X. Qiu, and X. Huang. Recurrent Neural Network for Text Classification with Multi-Task Learning. *Preprint ArXiv* 1: 1605-05101 (2016).
78. P. Ramachandran, P.J. Liu, and Q.V. Le. Unsupervised Pretraining for Sequence to Sequence Learning. *Preprint ArXiv* 2: 1611.02683 (2018).
79. A. Akbik, D. Blythe, and R. Vollgraf. Contextual String Embeddings for Sequence Labeling. *Proceedings of the 27th International Conference on Computational Linguistics, Santa Fe, New Mexico, USA* (20-26 August 2018) pp. 1638-1649 (2018).
80. S. Ruder, I. Vulic, and A. ogaard. A survey of cross-lingual word embedding models. *Journal of Artificial Intelligence Research* 65: 569-631 (2019).
81. S. Conia and R. Navigli. Conception: Multilingually-Enhanced, Human-Readable Concept Vector Representations. *Proceedings of the 28th International Conference on Computational Linguistics, Barcelona, Spain (online)* (8-13 December 2020) pp. 3268-3284 (2020).
82. A. Klementiev, I. Titov, and B. Bhattarai. Inducing Crosslingual Distributed Representations of Words. *Proceedings of COLING 2012, Mumbai, India* (December 2012) pp. 1459-1474 (2012).
83. G. Lample and A. Conneau. Cross-lingual Language Model Pretraining. *Preprint ArXiv* 1: 1901.07291 (2019).
84. J. Guan, F. Huang, Z. Zhao, X. Zhu, and M. Huang. A Knowledge-Enhanced Pretraining Model for Commonsense Story Generation. *Transactions of the Association for Computational Linguistics* 8: 93-108 (2020).
85. Y. Bengio, R. Ducharme, P. Vincent, and C. Janvin. A neural Probabilistic language model. *Journal of Machine Learning Research* 3: 1137-1155 (2003).
86. A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever. Language Models are Unsupervised Multitask Learners. *OpenAI Blog* 1(8): 9 (2019).
87. W. Wang, B. Bi, M. Yan, C. Wu, Z. Bao, J. Xia, L. Peng, and L. Si. StructBERT: Incorporating Language Structures into Pre-training for Deep Language Understanding. *Preprint ArXiv* 3: 1908.04577 (2019).
88. S. Wang, W. Zhou, and C. Jiang. A survey of word embeddings based on deep learning. *Computing* 102: 717–740 (2020).
89. L. Ma and Y. Zhang. Using Word2Vec to process big text data. *2015 IEEE International Conference on Big Data (Big Data), Santa Clara, CA, USA* (29 October-01 November 2015) pp. 2895-2897 (2015).
90. O. Kwon, D. Kim, S.-R. Lee, J. Choi, and S. Lee. Handling Out-of-Vocabulary Problem in Hangeul Word Embeddings. *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, *Online* (2021) pp. 3213-3221 (2021).
91. A. Gladkova and A. Drozd. Intrinsic evaluations of word embeddings: What can we do better?. *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP, Berlin, Germany* (12 August 2016) pp. 36-42 (2016).
92. E. Agirre, E. Alfonseca, K. Hall, J. Kravalova, M. Pasca, and A. Soroa. A Study on Similarity and Relatedness Using Distributional and Wordnet-based Approaches. *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistic*, *Boulder, Colorado* (June 2009) pp. 19-27 (2009).
93. E. Bruni, N.-K. Tran, and M. Baroni. Multimodal Distributional Semantics. *Journal of Artificial*

Intelligence Research 49: 1-47 (2014).

94. D. Gerz, I. Vulic, F. Hill, R. Reichart, and A. Korhonen. Simverb-3500: A Large-Scale Evaluation Set of Verb Similarity. *Preprint ArXiv* 4:1608.00869 (2016).
95. B. Gao, J. Bian, and T.-Y. Liu. Wordrep: A Benchmark for Research on Learning Word Representations. *Preprint ArXiv* 1: 1407.1640 (2014).
96. S. Harispe, S. Ranwez, S. Janaqi, and J. Montmain. Methods and Datasets for the Evaluation of Semantic Measures. *Semantic Similarity from Natural Language and Ontology Analysis*: 131-157 (2015).
97. B. Wang, A. Wang, F. Chen, Y. Wang, and C.-C. J. Kuo. Evaluating word embedding models: Methods and experimental results. *APSIPA Transactions on Signal and Information Processing* 8: e19 (2019).
98. C. Greenberg, V. Demberg, and A. Sayeed. Verb polysemy and frequency effects in thematic fit modeling. *Proceedings of the 6th Workshop on Cognitive Modeling and Computational Linguistics*, *Denver, Colorado* (4 June 2015) pp. 48-57 (2015).
99. S. Pado and M. Lapata. Dependency-Based Construction of Semantic Space Models. *Computational Linguistics* 33(2): 161-199 (2007).
100. F. Nooralahzadeh, L. Ovrelid, and J.T. onning. Evaluation of Domain-specific Word Embeddings using Knowledge Resources. *In Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, *Miyazaki, Japan* (7-12 May2018) (2018).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 247-255 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)841

Pakistan Academy of Sciences

Research Article

Extraction of Natural Dye using Peels of Citrus Fruits for Enhancing Color Fastness of Fabrics

Farea Noor* and Mehreen Ijaz

Department of Home Economics, Lahore College for Women University, Lahore, Pakistan

Abstract: The current study investigates the extraction of dyestuff using citrus fruits and assesses its color, light, rubbing, and perspiration fastness. For the experimental study, three types of citrus fruits—orange, lemon, and grapefruit—were chosen to extract dye from their peels. The dye was applied to two types of fabrics (100% cotton and a blended fabric made of polyester-cotton (PC) with a ratio of 50:50) using the conventional aqueous method. The results revealed the remarkable efficacy of orange dye on both fabric types, demonstrating excellent color fastness attributes, with a minor preference for PC fabric in washing fastness. In contrast, lemon dye displayed better washing fastness properties on the tested materials as well as considerable staining potential. Grapefruit dye performed exceptionally well in terms of water and perspiration fastness. Future research could focus on improving dye extraction techniques for citrus fruits to increase color absorption and penetration. Determining various solvents, time duration for dyeing, and temperature settings could enhance the dyeing performance and effectiveness of natural dyes in different industrial applications. Further studies could also conduct life cycle assessments to measure the environmental impact of using citrus peels as a natural dye source compared to synthetic dyes in the textile industry.

Keywords: Color Fastness, Staining Tests, Agricultural Waste, Citrus Fruits, Natural Dyeing, Textile Sustainability.

1. INTRODUCTION

Nature, without which life would be dull, is completely charming and attractive with its beautiful colors. Color provides us with a unique perception and understanding of how a particular object exhibits or reflects light. The use of these colors as dye is an ancient art form that predates written records. It was used as far back as the Bronze Age, where the methods of dyeing involved placing flowers on cloth and rubbing the pigments onto the surface of fabrics [1]. Some historical dyes included madder, a red-colored dye obtained from Rubia tinctorum, indigo derived from indigo leaves, and yellow extracted from saffron [2]. However, with the development of artificial dyes, the tradition of using natural materials to extract dye significantly decreased, especially after the discovery of the first synthetic dye, mauveine, by William Perkin in 1856. Synthetic dyes were cheaper to produce and available in a wider range of shades. As a result, at the start of the 20th century, natural dyes had been practically removed because of the low-cost artificial dyes and their availability [3]. Synthetic dyes are predominantly used in the fabric industry for silk, cotton, wool, nylon, and leather. However, due to the toxic effects of chemical dyes, their unsafe behavior for humans as well as for the environment revived interest in the use of natural dyestuff [4]. In contrast, natural dyes are considered an environment-friendly source compared to synthetic dyestuff, as these are extracted from renewable means such as plants, insects, herbs, and minerals instead of petrochemicals. This encourages the preservation of natural resources and lowers the carbon footprint connected to the manufacture of textiles [5].

Natural dyes extracted from plants and minerals have long been used by rural communities in textile and clothing production. This practice not only helps them to produce high-quality products but also ensures the safety of the environment [6, 7]. However, it was a challenging task to obtain bright and durable colors. The manufacturing of synthetic dyes in textile and garment industries

Received: October 2023; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Farea Noor <biyanoor30@gmail.com>

has greatly affected the surrounding environment. In contrast, natural dyes are considered as safer for our surrounding atmosphere and benefit the physical and mental health of all those involved in its production process [8]. Natural dyes are primarily obtained from plants such as madder, indigo, turmeric, henna, and saffron; insects like sepia, kermes, cochineal beetles, and lac; and minerals such as malachite, cinnabar, ferrous sulfate, ochre, lapis lazuli, and clay, without any chemical treatment [9]. In addition to these, many fruits are also used for the extraction of textile dyes that have permanent colorfastness properties, such as pomegranate, berries, most citrus fruits, and stone fruits.

Citrus fruits belong to the family Rutaceous with different forms and sizes such as oranges, citrons, grapefruits, mandarins, lemons, and limes. Furthermore, they have been used as a conventional herbal medicine in various Asian countries like China, Japan, and Korea [10]. Additionally, citrus fruits are a great source of natural dyes from which the dye is extracted and applied to fabrics thus protecting the environment from harmful chemicals and other substances. Oranges are a highly favored fruit and are harvested by more than one hundred million each year worldwide. The orange peel constitutes around 20-30% of the overall weight of the orange, making it a plentiful, affordable, and easily accessible source of biomass [11]. The peel of the orange comprises cellulose, hemicellulose, volatile essential oils, carotenoids, lignin, and phenolic components [12]. The dye extracted from the peels of the orange was used for dyeing many fabrics [13]. Lemons hold agronomic significance due to their versatile use as an ingredient in cooking. To make maximum use of its peel as a plentiful agricultural waste can be extracted as a great source of natural dyes used for textiles [14]. Grapefruit, in all tropical and subtropical regions of the world, is planted with almost 4 million metric tons of annual production [15]. It comprises many water-soluble and insoluble polymers and monomers [16, 17]. Additionally, waste peels of grapefruits are also useful for making sustainable dyestuff for coloring fabrics.

In the present study, natural dyes were extracted from peels of citrus fruits to improve color fastness, brightness, and dyeing effectiveness. The purpose is to develop an eco-friendly dyestuff that helps to reduce the negative impact of synthetic dyes used in the textile industry. Although some previous studies [18] have investigated multiple benefits of using natural dyes, there are certain challenges in improving their color fastness, efficiency, and performance. This study intends to fill this gap by identifying the unexplored potential of citrus fruits as a sustainable natural dying ingredient that can help reduce the environmental footprint.

2. MATERIALS AND METHODS

The experimental study was conducted at the Dyeing and Finishing Unit of Nishat Mills Pvt. Ltd. Three citrus fruits such as orange, lemon, and grapefruit were taken to extract the dye from their peels. The concentration of mordant, dyeing temperature, and time were determined using the conventional aqueous method.

The citrus fruits were collected from the local market. These were then washed gently using tap water and their peels were removed carefully. The peels were then sundried for 3 to 4 days. After complete drying, the peels were ground into powder using an electric beater. Subsequently, 100 g peels of orange, lemon, and grapefruit were taken in separate containers with 500 ml water. The extraction of dye was achieved by boiling the mixture at 80 °C for 1 hour. Then the mixture was cooled down and filtered; the dye extraction process is shown in Figure 1. Fabric samples for dye application were also collected from Nishat Mills. The specifications of these samples are given in Table 1.

Table 1. Construction parameters of samples.

Sample code	Fabric type	Thread count	Weave type	Mass (gm)
A	Cotton -100%	110 × 90	Plain	125
B	Polyester / Cotton Blend (50:50)	76 × 68	Plain	190

Fig. 1. Process of dye extraction using fruit peels.

The mordant used in this research was copper sulfate. It has been extensively used as a mordant during the dyeing process of cellulosic-based fibers such as cotton and cotton-polyester blends [19, 20]. It can significantly enhance the bonding of dye with the fiber thus increasing the color fastness to water, light, and washing. Moreover, copper sulfate also induces luster and depth to the dyed fabric to develop vibrant shades, as natural fibers tend to fade quickly. An adequate wastewater system was implemented to ensure its proper disposal to prevent environmental degradation. Both types of fabrics were pretreated using a mild detergent to remove the impurities that may affect dye penetration. 20 grams of $CuSO_4$ per liter was used. The pretreated fabric was immersed in the prepared mordant solution and simmered at 80 °C for approximately 45 minutes. A ratio of 1:20 was maintained between the fabric and the dye to ensure even dyeing.

The fabric samples were passed through the padding machine where the machine parameters were carefully controlled. The pressure was set to 1.5 bar which ensured maximum dye penetration. The speed of the machine was set at 7 rounds per minute. The pick-up rate of 80% was kept throughout the whole process to ensure complete saturation of each sample in a dye bath. The extra dye liquor was squeezed off and each sample was dried separately in a drying chamber at 130 °C for 2 minutes. The samples were then washed thoroughly to remove any excessive dye or other impurities from the surface.

Before testing the fabric, samples were conditioned at 65% ± 5 relative humidity and a standard temperature of 20 °C for approximately 24 hours following the guidelines of ISO 139:2005 [21] The fastness of the fabrics against acid perspiration (sweat) was tested using the AATCC 15: 2002 [22]. Specimen with the dimension of 60.3 mm × 60.3 mm was cut from each type of fabric. It was weighed on a weighing scale to record the dry weight of the specimen. A specimen was soaked in the test solution for 30 minutes and occasionally squeezed. After half an hour, it was removed from the solution, passed through the wringer, and reweighed. To start the test, the plexiglass plates were placed in the perspiration tester along with the specimen/multi-fiber strips uniformly spread between 21 plates. 8.0 lb. weight was applied on top making a total of 10.0 lb. under the pressure plate. It was then locked, and the weight was removed. Specimens in the oven were preheated at 37 °C for 6 hours. The multifiber strip from the fabric was separated and placed in a conditioning atmosphere overnight. The color change grade was noted against each grey scale for each of the tested specimens.

The fastness test against alkaline perspiration was conducted according to the ISO-E04:2013 [23]. A specimen measuring 40 mm × 100 mm is sewn with the shorter side of the multifiber adjacent fabric also measuring 40 mm × 100 mm. Then the specimen was soaked in an alkaline solution of pH 8 (±0.2) with a liquor ratio of 50:1 at room temperature for half an hour. It was pressed from time to time to remove excess water. The specimen was placed between two glass plates under a nominal pressure of 12 kPa and placed in a tester to preheat. Using the same method, the test specimen was wet

in the acid-based solution at a pH of 5.5 and then tested in a pre-heated testing oven. Specimen was placed in it for 4 hours at 37 °C, aligning it so that the specimens were either in vertical or horizontal direction. A specimen was opened out removing the stitching except at one of the short sides and dried it at a temperature of 60 °C, with only two parts in contact at the line of stitching. The change in color and shade of each of the tested specimen and the staining on the adjacent fabric swatches were assessed against grey scales.

The light fastness test was performed by AATCC Lightfastness Standard 16: 2004 [24]. The fabric was cut at least 70.0 mm × 120.0 mm while the exposing area was only 30 mm × 30 mm. The fabric was then exposed for 20 operating hours at the standardized temperature and humidity. Measured the exposed standard specimen, either visual comparison (if the changed color equaled the Xenon Reference Fabric of Fade in 20 continuous light-on operating hours then the test equipment was maintaining the correct temperature) or Instrumental Color Measurement (If the exposed standard specimen equaled 20 ± 1.7 CIELAB units of color change in 20 ± 2 continuous light-on operating hours, the test machine was providing the correct temperature). If the exposed Reference Fabric differed through visual inspection or instrument testing, after 20 light-on operating hours, it was a sign that temperature sensing units within the test chamber were not responding accurately and need readjustments.

Two fabric pieces of 50 mm × 140 mm were used for rubbing in dry and wet conditions. The specimen was tested according to ISO 105-X12:2016 [25]. The conditioned rubbing cloth was placed flat over the end of the finger with the weave parallel to the direction of the rubbing finger. At a rate of one cycle per second, rubbing was made in to and fro in a straight line for 20 times, 10 times to and 10 times fro, along a track of 104 mm long on dry specimen, with the downwards force of 9 N. Any unnecessary fibrous or other material was detached before evaluation, that might impact the results. Rubbing cloth was conditioned and thoroughly soaked in distilled water and reweighed to ensure take-up of 95% to 100%. The procedure was similar to that used for dry rubbing. The specimen was then air dried. Each test cloth was backed with three layers of rubbing cloth in white color while measuring the fastness. The gray scale was used to assess the staining of cotton. The numerical rating of each of the tested specimens was made against a grey scale.

The fabrics were evaluated for their color against water [26]. A specimen of (60 ± 2) mm × (60 ± 2) was cut using a template. It was then immersed in a prepared test solution for an hour and squeezed between the rollers to remove excess liquor. The process was repeated to achieve thorough wetting. The specimen was placed in a perspiration tester with a pressure of 9.9 lb. The heat was provided at 38 °C for 18 hours in an oven. It was then removed and dried at room temperature. It was evaluated for its color change against the grayscale.

The fabrics were evaluated for their color against washing [27]. A specimen (50 × 100 ± 2 mm) was cut using a template. The laundering machine was adjusted according to the instructions given in the test procedure. The specimen was immersed in a standardized washing solution. It was placed in a canister that was sealed and agitated in the washing solution for half an hour at 40 °C. After thorough agitation, the specimen was rinsed to remove any excess liquor and dried at room temperature. It was also evaluated for its color change against the grayscale.

3. RESULTS AND DISCUSSION

Three types of dyestuff extracted from orange, lemon and grapefruits were tested on cotton and a blend of Polyester and Cotton (PC) using copper sulfate as a pre-mordanting technique. The resulting color of the dyes is given in Figure 2. The tests for color fastness by washing, perspiration, rubbing (wet and dry), and light fastness, were performed on cotton (A) and PC (B) fabrics. The results are shown in Table 2.

Table 2 depicts that the orange dye exhibited poor washing fastness properties on cotton fabric, but its performance was slightly improved on PC fabric. One of the possible reasons is that cotton, being a natural fabric, tends to retain water, potentially leading to dye bleeding due to weaker chemical bonds. In contrast, PC fabric, being synthetic, is less absorbent and possesses stronger chemical bonds, contributing to its resistance to fading. A study showed that a blend of cotton

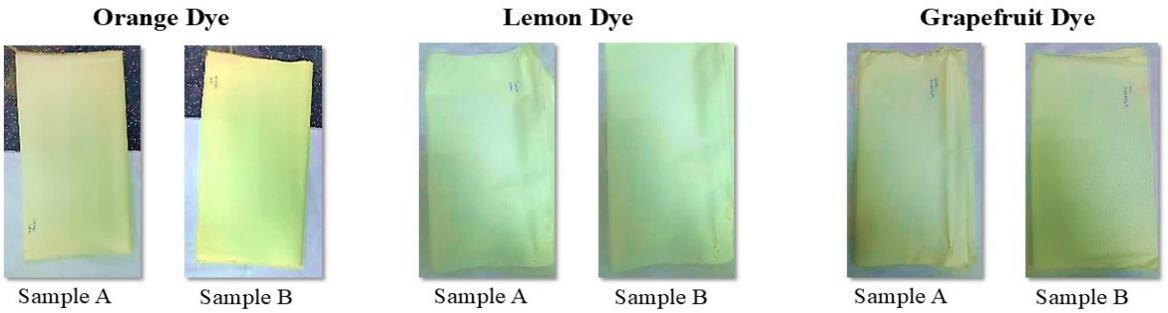

Fig. 2. Fabric dyed with orange, lemon, and grapefruit.

Table 2. Colorfastness of fabrics for each dye.

Sample	Dyestuff	Washing fastness	Water	Light fastness	Rubbing fastness		Perspiration fastness	
					Dry	Wet	Acid	Alkali
A	Orange	1-2	4	4	4	4	4	4
B	Orange	1	4	4	4	3-4	4	4
A	Lemon	0	4	4	4	3-4	4	4
B	Lemon	0	4	4	4	3	4	4
A	Grapefruit	0	4	3	4	3	4	4
B	Grapefruit	0	4	4	4	3	4	4

and polyester had better dyeability compared with 100% cotton fabrics when dyed using henna and onion against wash fastness, light fastness, acid, and alkali perspiration [28]. Notably, other fastness properties such as light fastness, rubbing fastness, and perspiration showed good results for the orange dye on both tested fabrics.

On the other hand, the results of lemon dye on both cotton and PC fabric showed good color fastnesses against rubbing, light fastness, and perspiration tests except for washing fastness. This was attributed to the dye's inability to form a strong bond with the fibers or its poor affinity for the cloth. Conversely, the study investigated the fastness and staining properties of grapefruit dye on both fabrics. The results showed that this dye had good color fastness properties against perspiration and water tests but had poor washing fastness properties for both fabrics. The obtained data showed that the orange dye had excellent results than dyes extracted from lemon and grapefruit. Furthermore, PC fabric presented better dye intake than cotton fabric due to its stronger chemical bond.

Figure 3 elucidates the comparison between samples A and B with the contrast between orange, lemon, and grapefruit dyes for the fastness tests of washing, light, rubbing, and perspiration. A descriptive analysis was made among different fruit peels used to extract dye for their colorfastness. The maximum and minimum range for each of the scores was determined and the results are depicted in Table 3.

Table 4 shows that the staining tests for the orange dye had good to excellent properties and can be used in a wide range of fabrics and other textile applications. The staining results, on the other hand, showed that the lemon dye could be applied to many fabrics with minimal variations in staining capability. The staining potential of grapefruit was moderate for all samples except for cotton in acid perspiration. Figure 4 elucidates the comparison between samples A and B with the contrast between orange, lemon, and grapefruit dyes for the staining tests along with standardized fabrics.

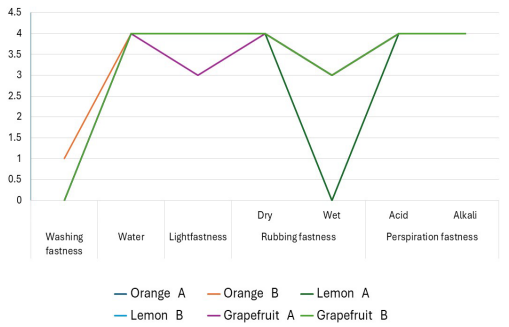

Fig. 3. Comparison of colorfastness of dye.

Table 3. Statistical analysis of colorfast test.

Peels	**Orange**		**Lemon**		**Grapefruit**	
Range	Max	Min	Max	Min	Max	Min
Fabrics	A	B	A	B	A	B
Washing	1-2	1	0	0	0	0
Water	4	4	4	4	4	4
Light	4	4	4	4	4	3
Rubbing (dry condition)	4	4	4	4	4	4
Rubbing (wet condition)	4	3-4	3-4	3	3	3
Perspiration (in acid)	4	4	4	4	4	4
Perspiration (in alkali)	4	4	4	4	4	4

A previous study by Li et al. [29] has investigated that the cotton fabric dyed with an orange peel has low rubbing fastness properties both in dry and wet conditions against tannic acid used as a mordant. Another research by Taura et al. [8], in which cotton was dyed with an orange peel dye using NaOH mordant indicated that rubbing and sunlight had no significant effect on cotton fabric, but washing had increased the paleness in the overall color of the fabric. To achieve good fastness properties especially wash fastness future studies can be conducted by using NaOH and Tannic acid as mordant as they proved good mordant in the fixation of dye. A descriptive analysis was made among different fruit peels used to extract dye for their staining test. The maximum and minimum range for each of the peel are presented in Table 5. Compared to the synthetic dyes, natural dyes are eco-friendly, sustainable, non-toxic, biodegradable, and non-allergic when treated with turmeric and neem dye solutions [30]. Suri et al. [31] investigated the benefits of citrus fruit waste, highlighting their potential for recovering bioactive compounds, pectin, biofuels, essential oils, dyes, and micro and macro-nutrients. Devi and Saini [32] have also highlighted the application of an orange peel in textile industry, using it as a source of natural dye, waste absorbent, antimicrobial agent, mosquito repellent, sustainable fiber, and perfume retention agent. Taura *et al.* [8] have reported that orange

Table 4. Staining test of each dye and fabric.

	Orange								**Lemon**								**Grapefruit**							
Fabrics	**Wash**		**Water**		***ALKP**		****ACIP**		**Wash**		**Water**		**ALKP**		**ACIP**		**Wash**		**Water**		**ALKP**		**ACIP**	
	A	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**	**A**	**B**
Acetate	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5
Cotton	4	4	4	3-4	4	4	4	4	4	4	4-5	4-5	4	4-5	4	4-5	4	4	4	4	4	4	3-4	4
Nylon	4-5	4-5	4-5	4	4-5	4	4-5	4	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5
Polyester	4-5	4-5	4-5	4	4-5	4	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5
Acrylic	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5
Wool	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5	4-5

Note: *ALKP = Alkaline Perspiration, **ACIP = Acidic Perspiration

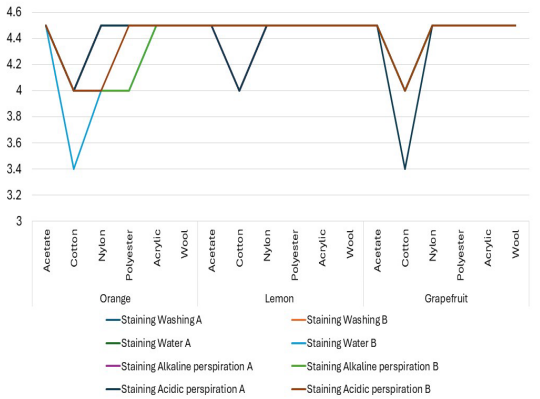

Fig. 4. Comparison of staining test.

peel is the most effective part of a plant for color extraction. It also promotes the use of sustainable and eco-friendly materials in the textile industry for various applications.

4. CONCLUSIONS

From the results of the present study, we can conclude that orange dye is more effective on both types of fabric and exhibited excellent color fastness characteristics, except for washing fastness, which slightly favored PC fabric. It demonstrated adaptability across various fabrics. On the other hand, lemon dye showed better overall fastness qualities and considerable staining potential, although it had slightly weaker washing fastness on the tested fabrics. Grapefruit dye performed poorly in washing tests but showed good results for water and perspiration fastness. These collective results highlight the potential of citrus fruits as a viable and natural source of dyes for fabrics. The findings of the study can be attributed to the high concentration of natural colorants in citrus fruits, such as flavonoids and carotenoids, which aid in the quick adherence of dyestuff to textile fibers, making them useful for various applications. The use of fruit peels as coloring agents in the textile industry holds significant potential due to their abundance, cost-effectiveness, and eco-friendly nature. Their application in other industries, such as cosmetics, food packaging, and the leather sector, could help reduce industrial waste and promote a circular economy.

5. CONFLICT OF INTEREST

The authors declare no conflict of interest.

Table 5. Statistical analysis of staining test.

Peels	**Orange**		**Lemon**		**Grapefruit**	
Range	Max	Min	Max	Min	Max	Min
Fabrics	A	B	A	B	A	B
Acetate	4-5	4-5	4-5	4-5	4-5	4-5
Cotton	4	3	4-5	4	4-5	3
Nylon	4-5	4	4-5	4-5	4-5	4-5
Polyester	4-5	4	4-5	4-5	4-5	4-5
Acrylic	4-5	4-5	4-5	4-5	4-5	4-5
Wool	4-5	4-5	4-5	4-5	4-5	4-5

6. REFERENCES

1. A. Ado, H. Yahaya, A.A. Kwalli, and R.S. Abdulkadir. Dyeing of textiles with eco-friendly natural dyes: a review. *International Journal of Environmental Monitoring and Protection* 1(5): 76-81 (2014).
2. R. Siva. Status of natural dyes and dye-yielding plants in India. *Current Science* 92(7): 916-925 (2007).
3. P. Saravanan and G. Chandramohan. Dyeing of silk with eco-friendly natural dye obtained from barks of Ficus Religiosa. L. *Universal Journal of Environmental Research and Technology* 1(3): 268-273 (2011).
4. K. Sinha, P.D. Saha, and S. Datta. Extraction of natural dye from petals of Flame of forest (Butea monosperma) flower: Process optimization using response surface methodology (RSM). *Dyes and Pigments* 94(2): 212-216 (2012).
5. L. Chungkrang, S. Bhuyan, and A.R. Phukan. Natural dye sources and its applications in textiles: a brief review. *International Journal of Current Microbiology and Applied Sciences* 9(10): 261-269 (2020).
6. L. Chungkrang, S. Bhuyan, and A.R. Phukan. Natural dyes: extraction and applications. *International Journal of Current Microbiology and Applied Sciences* 10(1): 1669-1677 (2021).
7. H.M. Helmy. Extraction approaches of natural dyes for textile coloration. *Journal of Textiles, Coloration and Polymer Science* 17(2): 65-76 (2020).
8. U.H. Taura, M.A. Abubakar, A.M. Abubakar, and M.U. Kurgiya. Extraction and characterisation

of natural dye from orange peel for textile applications. *Journal of Applied Life Sciences and Environment* 57(1): 169–181 (2024).
9. S. Kadolph. Natural dyes: a traditional craft experiencing new attention. *Delta Kappa Gamma Bulletin* 75(1): 14 (2008).
10. Chinese Pharmacopoeia Commission. A Colored Identification Atlas of Chinese Materia Medica and Plants as Specified in the Pharmacopoeia of the People's Republic of China. *Natural History Book Service (NHBS) Publishers* (2010).
11. S.M.S. Sawalha, D. Arráez-Román, A. Segura-Carretero, and A. Fernández-Gutiérrez. Quantification of main phenolic compounds in sweet and bitter orange peel using CE–MS/MS. *Food Chemistry* 116(2): 567-574 (2009).
12. C. Namasivayam, N. Muniasamy, K. Gayatri, M. Rani, and K. Ranganathan. Removal of dyes from aqueous solutions by cellulosic waste orange peel. *Bioresource Technology* 57(1): 37-43 (1996).
13. X. Hou, X. Chen, Y. Cheng, H. Xu, L. Chen, and Y. Yang. Dyeing and UV-protection properties of water extracts from orange peel. *Journal of Cleaner Production* 52: 410-419 (2013).
14. A. Bhatnagar, E. Kumar, A.K. Minocha, B.H. Jeon, H. Song, and Y.C. Seo. Removal of anionic dyes from water using Citrus limonum (lemon) peel: equilibrium studies and kinetic modeling. *Separation Science and Technology* 44(2): 316-334 (2009).
15. A. Saeed, M. Sharif, and M. Iqbal. Application potential of grapefruit peel as dye sorbent: kinetics, equilibrium, and mechanism of crystal violet adsorption. *Journal of Hazardous Materials* 179(1-3): 564-572 (2010).
16. M.R. Wilkins, W.W. Widmer, K. Grohmann, and R.G. Cameron. Hydrolysis of grapefruit peel waste with cellulase and pectinase enzymes. *Bioresource Technology* 98(8): 1596-1601 (2007).
17. S.V. Ting and E.J. Deszyck. The Carbohydrates in the Peel of Oranges and Grapefruit. *Journal of Food Science* 26(2): 146-152 (1961).
18. V.K. Gupta. Fundamentals of natural dyes and its application on textile substrates. In: Chemistry and Technology of Natural and Synthetic Dyes and Pigments. A.K. Samanta, N.S. Awwad, and H.M. Algarni (Eds.). *Intech Open eBooks* (2020).
19. V. Kaur, S. Arjunan, and I. Nanaiah, Extraction of Dyes from Plant Sources and their application on Cotton and Wool using Mordants. *Current Trends in Biotechnology and Pharmacy* 15(5): 503–506 (2021).
20. N.M. Iqbal. Dyeing of cotton with natural colorants extracted from red rose flower. *Pakistan Journal of Science* 75(03): 527–534 (2023).
21. Lithuanian Standards Board. LST EN ISO 139. Textiles–standard atmospheres for conditioning and testing. *Vilnius, Lithuania; P.A: LST* (2006). https://www.iso.org/standard/35179.html.
22. American Association of Textile Chemists and Colorists. AATCC Test Method 15. Colorfastness to Perspiration. In: AATCC Technical Manual. *P.A: AATCC* (2002). https://members.aatcc.org/store/tm15/482/.
23. International Organization for Standardization. ISO 105-E04. Textiles—Tests for Colour Fastness—Part E04: Colour Fastness to Perspiration. *P.A: ISO* (2013). https://www.iso.org/standard/57973.html#:~:text=ISO%20105%2DE04%3A2013%20specifies,the%20action%20of%20human%20perspiration.
24. American Association of Textile Chemists and Colorists. AATCC Test Method 16. Colorfastness to Light. In AATCC Technical Manual; *P.A: AATCC* (2004). https://www.scribd.com/document/669445084/AATCC-Test-Method-16-2004.
25. International Organization for Standardization. ISO 105 X12. Textiles–tests for colour fastness—Part X12: Colour fastness to rubbing. *Geneva, Switzerland; P.A: ISO* (2016). https://www.iso.org/standard/65207.html#:~:text=ISO%20105%2DX12%3A2016%20specifies,off%20and%20staining%20other%20materials.
26. American Association of Textile Chemists and Colorists. AATCC Test Method 107. Colorfastness to Water. AATCC Technical Manual. *P.A: AATCC* (2007). https://members.aatcc.org/store/tm107/519/.
27. American Association of Textile Chemists and Colorists. AATCC Test Method 61. Colorfastness to Laundering: Accelerated. AATCC Technical Manual. *P.A: AATCC* (2010). http://yiqi-oss.oss-cn-hangzhou.aliyuncs.com/aliyun/technology/275749/150428.pdf.
28. A.S. Mohamed. Dye ability of some fabric materials (cotton, polyester, and cotton/polyester blend) using some natural dyes in an economical dyeing process1. *Egyptian Journal of Agricultural Research* 87(4): 1173-1187 (2009).
29. K. Li, Q. Ding, and H. Zhang. Eco-friendly dyeing of cotton fabric using natural dye from orange peel. *Journal of the Textile Institute* 113(3): 360–366 (2021).

30. N. Harini and N. Santhi. Challenges and opportunities in product development using natural dyes. *The Scientific Temper* 14(01): 211–215 (2023).
31. S. Suri, A. Singh, and P.K. Nema. Recent advances in valorization of citrus fruits processing waste: a way forward towards environmental sustainability. *Food Science and Biotechnology* 30(13): 1601–1626 (2021).
32. O.R. Devi and H. Saini. Utilization of orange peel waste in textile industry: A review. *International Journal of Chemical Studies* 8(4): 05–08 (2020).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 257-271 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)848

Pakistan Academy of Sciences

Research Article

Selecting the Optimal Classifier for Wrapper-Based Feature Selection Methods

Farzad Zandi[1], Parvaneh Mansouri[1], and Reza Sheibani[2*]

[1]Department of Mathematics and Computer Science, Arak Branch,
Islamic Azad University, Arak, Iran
[2]Department of Computer Engineering, Mashhad Branch,
Islamic Azad University, Mashhad, Iran

Abstract: Dimensionality reduction, the elimination of irrelevant features, and the selection of an optimal subset of features are critical components in the construction of an efficacious machine learning model. Among the various feature selection methodologies, wrapper-based methods yield superior results due to their evaluation of candidate subsets. Numerous meta-heuristic methods have been employed in this feature selection process. A significant and complex issue in feature selection utilizing these methods is the selection of the most suitable classifier. In this study, we propose a novel method for selecting the optimal classifier during the feature selection process. We employ ten distinct classifiers for two swarm intelligence methods, namely Bat and Gray Wolf, and compute their results on four cancer datasets: Leukemia, SRBCT, Prostate, and Colon. Our findings demonstrate that the proposed method identifies the optimal classifiers for all four datasets. Consequently, when employing wrapper-based methods to select features for each dataset, the optimal classifier is identified.

Keywords: Feature Selection, Wrapper-based Methods, Metaheuristic Algorithms, Roulette Wheel, Optimal Classifier.

1. INTRODUCTION

Feature selection is a pivotal process in machine learning that entails the selection of the most pertinent features from the dataset that contribute significantly to the prediction variable or outcome. The objective is to eliminate irrelevant or redundant features. This can lead to a decrease in model accuracy and performance. Feature selection is instrumental in constructing an effective machine learning model for several reasons:

- **Enhances Accuracy**: Irrelevant or redundant features can adversely affect the model's performance. By utilizing only, the most pertinent features, we can construct more accurate models.
- **Mitigates Overfitting**: A model trained with irrelevant features is more prone to overfitting, where it performs well on the training data but poorly on unseen data.
- **Accelerates Training**: Less data equates to faster training times. By reducing the number of features, we can expedite the training process.
- **Boosts Interpretability**: Models with fewer features are simpler to understand and interpret.

There are several techniques for feature selection [1], each with its unique strengths and weaknesses. Here are a few commonly employed methods:

- **Filter Methods**: These methods are often univariate and consider each feature independently or in relation to the dependent variable. Examples include the Chi-Squared Test [2], Information Gain [3], and Correlation Coefficient Scores [4].
- **Wrapper Methods**: These methods perceive the selection of a set of features as a search problem. Examples include Recursive Feature Elimination [5], Forward Selection, and Backward Elimination [6].
- **Embedded Methods**: These methods ascertain which features best contribute to the accuracy of the model during the model creation process. An example includes LASSO [7].

Received: March 2024; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Reza Sheibani <reza.shni@gmail.com>

Among the various methodologies for feature selection, wrapper-based methods yield superior results. This can be attributed to the employment of a fitness function, which is utilized to evaluate each selected subset. For a considerable duration, meta-heuristic algorithms have been instrumental in resolving numerous optimization problems that are either exceedingly complex or possess an extensive problem space, rendering them unsolvable [8]. Novel meta-heuristic algorithms, encompassing population-based, evolutionary-based, and nature-inspired based methods, are continually being developed. With minor modifications, these methods can be transformed into a robust wrapper-based feature selection method [9]. These methods typically commence with a subset of features and, after evaluating them with a fitting function, they converge towards the optimal results based on the proposed algorithm. The fitting function serves as a crucial criterion that could be a supervised machine learning algorithm, calculating the suitability of candidate features. In machine learning, there exists a multitude of supervised algorithms, each with its unique advantages and disadvantages [10]. A particular classifier may yield satisfactory results with a dataset, while another classifier may produce unacceptable or even subpar results on the same dataset. This challenge is also applicable to the evaluation function of feature selection problems. Consequently, the selection of an appropriate classifier can enhance the evaluation performance. Given that it is not feasible to test individual classifiers on all datasets due to the time-consuming nature of this process, we propose a solution to this challenge, enabling the utilization of the optimal classifier for quality assessment.

Numerous studies have been conducted to predict various types of cancers, employing a range of machine learning algorithms in the process. Common algorithms such as K-Nearest Neighborhood, Support Vector Machine, Logistic Regression, Decision Tree, and Bayes have been widely utilized. More complex methods, including Random Forest, Ensemble, Boosting, Neural Network, and deep learning, have also been explored. However, these methods often encounter high time complexity due to the large data dimensions, particularly the number of genes. To address this issue, researchers have considered the use of different dimension reduction methods. Among the feature selection methods, filtering-based methods have been favored over wrapper-based and embedding-based methods. For instance, Purbolaksono *et al.* [11] employed mutual information to reduce dimensions and identify informative genes. Aydadenta and Adivijaya [12] utilized k-means and Information Gain for feature selection. Initially, k-means was used to group similar features into a cluster, thereby eliminating redundancy. Subsequently, the Relief algorithm was used to rank the elements of the clusters, and the top-ranking features of each cluster were combined for Random Forest training. Cilia *et al.* [13] concentrated on feature ranking techniques that evaluate each feature individually. In the work of Bol'on *et al.* [14], various feature selection techniques such as ReliefF, SVM-RFE, mRMR, IG, and FCBF were used for gene selection. Al-Batah *et al.* [15] employed the filter method, CFS, to eliminate redundant genes and identify informative ones. Baliarsingh *et al.* [16] used the Wilcoxon rank sum test to select relevant genes. Su *et al.* [17] introduced a gene selection method based on the Kolmogorov-Smirnov (K-S) test and CFS. Initially, the K-S test removed redundant genes and noise by comparing the distribution of two types of samples. The filtered subset was then evaluated by CFS, leaving only genes with high correlation with the class and low redundancy. Lastly, Ahmad *et al.* [18] used different filter feature selection techniques, namely SNR, FC, IG, and t-Test, to select informative genes. This comprehensive review of methods and techniques provides a solid foundation for future research in cancer prediction and gene selection.

In addition to the aforementioned methods, some researchers have employed wrapper-based methods to identify the most effective genes. For instance, Wu *et al.* [19] proposed a hybrid approach that leverages an enhanced HI-BQPSO binary quantum particle swarm optimization algorithm for feature selection. This innovative method amalgamates the benefits of filtering and random heuristic search. The process begins with the utilization of the Maximum Information Coefficient (MIC) to compute the correlation between features and class, thereby obtaining an initial feature subset. Subsequently, the enhanced BQPSO is employed to derive the optimal feature subset. This methodological approach underscores the potential of hybrid models in gene selection for cancer prediction.

Several studies have also explored the combination of filtering and wrapper-based methods. For instance, Medjahed *et al.* [20] utilized Support Vector Machines based on Recursive Feature Elimination (SVM-RFE) to eliminate 40% of features. The remaining subset was subsequently processed through Binary Dragonfly (BDF) to retain only the informative genes. In another study, Jain *et al.* [21] proposed a hybrid feature selection method that amalgamated Correlation-based Feature Selection (CFS) and Improved Binary Particle Swarm Optimization (IBPSO). The use of IBPSO enhanced the initial convergence to the local optimum of Binary Particle Swarm Optimization (BPSO). Furthermore, Shahbeig *et al.* [22] introduced a hybrid Teaching-Learning-Based Optimization - Particle Swarm Optimization (TLBO-PSO) method. This approach combines the principles of Teaching-Learning-Based Optimization (TLBO) and a mutated version of Fuzzy Adaptive Particle Swarm Optimization (PSO) algorithms. These studies highlight the potential of hybrid models in gene selection for cancer prediction.

A thorough examination of feature selection techniques and cancer prediction methodologies is presented in the study conducted by Abd-Elnaby *et al.* [23]. This comprehensive review provides valuable insights into the current state of research in this field. An important observation to note is that all the introduced wrapper-based methods inherently require a classifier. This observation prompts an intriguing question: would modifying the classifiers used in these methods influence the efficiency of the problem, either in a beneficial or detrimental manner? This potential variability could introduce complexities and pose challenges in the application of these methods.

In this study, we introduce a novel feature selection model that, in addition to selecting the optimal features, enables the determination of the most suitable classifier for use with each dataset. Initially, we selected two swarm-intelligence-based methods, Bat and Gray Wolf, from among meta-heuristic methods and adapted them into two distinct feature selection approaches. Subsequently, from the realm of machine learning algorithms, we selected ten classifiers - KNN [24], SVM [25], Bayes [26], Ridge [27], DT [28], RF [29], Bagging [30], LightGBM [31], Perceptron [32], and LDA [33] - to serve as the fitness function for both algorithms and executed the feature selection process with them. This process was repeated ten times to enhance confidence, owing to the nature of wrapper-based methods. The results were noted, and the optimal classifier was identified among them for comparison with our proposed model. In our proposed model, we incorporated a heuristic component into both algorithms to select the optimal classifier concurrently with the feature selection. This process was also executed ten times for assurance. Furthermore, to demonstrate the efficiency of the proposed method, the test was conducted on four datasets of gene expressions for Leukemia, Prostate, SRBCT, and Colon cancers. Utilizing the results obtained from the Bat and Gray Wolf, using individual classifiers, and comparing them with the proposed method, we demonstrate that the proposed method identifies the optimal classifier on the evaluated datasets. Therefore, for the feature selection process using swarm intelligence algorithms, the optimal classifier can be selected for each dataset.

Figure 1 presents our proposed model steps for selecting the optimal classifier for wrapper-based feature selection methods. To evaluate

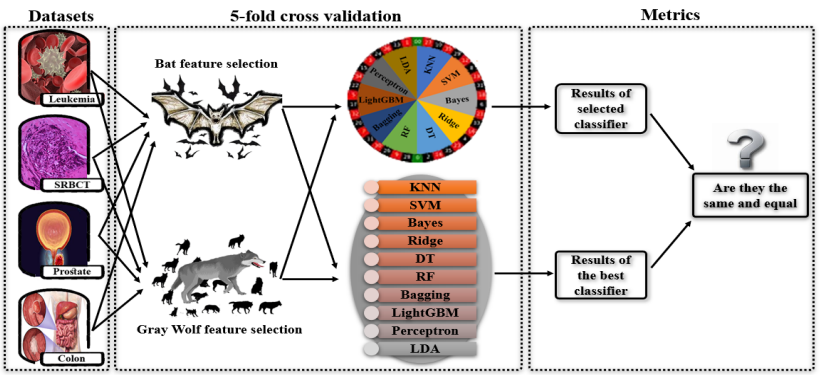

Fig. 1. Our proposed model steps for selecting the optimal classifier for wrapper-based feature selection methods.

the performance of the proposed model, four microarray datasets were employed. The Bat and Gray Wolf optimization algorithms were utilized as dimensionality reduction methods. These datasets were processed using these two meta-heuristic techniques to select the optimal subset of features. The accuracy of the selected features was then evaluated using ten distinct machine learning methods, and the results were documented for comparison. Subsequently, a roulette wheel mechanism was integrated into both the Bat and Gray Wolf algorithms. During the feature selection process, the probability of selecting each of the ten machine learning methods was determined. In the following iteration, one of these ten methods was employed to compute the fitness function via the roulette wheel mechanism. The accuracy of the selected features was recalculated, leveraging this mechanism alongside the top-performing machine learning method (highest probability), and the outcomes were recorded. A 5-fold cross-validation procedure was conducted during the feature selection process by both the Bat and Gray Wolf algorithms, with and without the roulette wheel step, ensuring that all samples were included in the training and testing phases. Finally, a comparative analysis was performed on the results obtained from the Bat and Gray Wolf feature selection methods, both with and without the roulette wheel step, alongside the selected machine learning method.

The main purpose of our study is to identify the optimal classifier to be used as a fitness function within the wrapper-based feature selection process. Selecting the optimal machine learning method tailored to the specific dataset, such as cancer diagnosis, can enhance the accuracy of computations and identify the best subset of features. Our method concentrates on determining the optimal classifier from several candidate machine learning methods. This has been implemented on four cancer datasets using two meta-heuristic algorithms, incorporating the roulette wheel mechanism.

2. MATERIALS AND METHODS

Feature selection constitutes a crucial phase in the machine learning pipeline. It serves to enhance the performance of a model, mitigate overfitting, augment interpretability, and expedite training. By comprehending and implementing suitable feature selection methods, we can construct more efficient and efficacious machine learning models. In this section, we present a novel pipeline capable of identifying the optimal classifier for utilization in wrapper-based feature selection methods.

2.1. Datasets

In this study, we utilized four datasets, as detailed in Table 1, to evaluate the proposed model. These datasets are a type of high-dimensional data structure known as microarrays, which carry genetic information from a given sample. The first dataset employed in this study carries information about Leukemia cancer, distinguishing between two classes: Acute Myeloid Leukemia (AML) and Acute Lymphoblastic Leukemia (ALL). This dataset comprises 7129 identified and quantified genes for 72 patients. We partitioned the dataset into a training set and a test set [34]. The second dataset is the Small Round Blue Cell Tumors (SRBCT), a gene expression dataset from a childhood cancer study [35]. It contains 83 samples with 2308 genes across four classes, with the following distribution: 29 cases of Ewing Sarcoma (EWS), 11 cases of Burkitt Lymphoma (BL), 18 cases of Neuroblastoma (NB), and 25 cases of Rhabdomyosarcoma (RMS). The third dataset pertains to Prostate cancer and consists of 12600 genes across two classes: 77 wild type and 52 Prostate cancer samples [36]. The final dataset is the Colon dataset, composed of 2000 genes and 62 samples taken from Colon cancer patients. Among the samples, there are 40 tumor biopsies (marked as abnormal) and 22 normal [37]. We employed the same strategy of splitting the dataset into a training set and a test set to validate the constructed model.

Table 1. The dataset utilized for evaluating our proposed model is characterized by the number of samples, genes, and classes.

Dataset	Number of samples	Number of genes	Number of classes
Leukemia	72	7129	2
SRBCT	83	2308	4
Prostate	136	12600	2
Colon	62	2000	2

2.2. Feature Selection

In order to implement the wrapper-based feature selection method, we opted for two meta-heuristic algorithms, namely Bat and Gray Wolf, which will be succinctly elucidated. The Bat and Gray Wolf algorithms necessitate an estimator, such as classifiers, to compute the performance of the optimal subset of features. In this study, we selected ten classifiers and executed each of the Bat and Gray Wolf algorithms with them on the four datasets delineated in the preceding section. Owing to the stochastic nature of swarm intelligence methods, the evaluation process is reiterated ten times. Subsequently, the method of selecting the optimal classifier, which is expounded upon in the ensuing sections using these ten classifiers, is implemented. In addition to selecting the optimal subset of data, the optimal classifier is identified on the target dataset without incurring any additional time overhead.

2.3. Bat Algorithm

Bats are intriguing creatures whose advanced echolocation capabilities have garnered the attention of researchers across various fields. Echolocation operates akin to sonar: primarily micro-bats emit a loud, brief pulse of sound, which, upon striking an object, returns as an echo after a fraction of time [38]. Consequently, bats can calculate their distance from an object [39]. Moreover, this remarkable orientation mechanism enables bats to discern the difference between an obstacle and prey, facilitating hunting even in complete darkness [40]. All bats utilize echolocation to gauge distance, and they also possess the ability to distinguish between food/prey and background barriers; A bat 'b' flies randomly with velocity 'v' at position 'x' with a fixed frequency 'f_{min}', varying wavelength 'λ', and loudness 'A_0' to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and modulate the rate of pulse emission 'R ϵ [0, 1]', contingent on the proximity of their target. Although the loudness can vary in numerous ways, Yang [41] posits that the loudness varies from a large (positive) 'A_0' to a minimum constant value 'A_{min}'.

Algorithm1. Feature selection by Bat Algorithm.

1. Initialize the bat population X_b, random number in {0, 1}, b = 1, 2, ..., nBats.
2. Initialize the loudness A_b, random number in range [1, 2], b = 1, 2, ..., nBats.
3. Initialize the pulse emission rate R_b, random number in range [0, 1], b = 1, 2, ..., nBats.
4. Initialize the velocity $V_b = 0$, b = 1, 2, ..., nBats.
5. Initialize the fitness $fitness_b = -\infty$, b = 1, 2, ..., nBats.
6. Initialize the globalFitness = -∞.
7. Initialize the maximum and minimum frequency, $f_{min} = 0$, $f_{max} = 1$.
8. For (t in maxIter):
9. For (b in nBats):
10. Find ($X_b == 1$) and make train and test subsets.
11. Calculate accuracy.
12. accuracy = (theta × accuracy + (1-theta) × (1-$\frac{sum(X_b==1)}{nFeatures}$))
13. If (rand < A_b & accuracy > $fitness_b$):
14. $fitness_b$ = accuracy
15. $A_b = \alpha A_b$
16. $R_b = R_b^0[1-\exp(-\gamma t)]$
17. [maxFitness, maxIndex] = max(fitness)
18. If (maxFitness > globalFitness):
19. globalFitness = maxFitness
20. xHat = $x_{maxIndex}$
21. For (b in nBats):
22. If (rand > R_b):
23. $X_b = X_b + \epsilon \bar{A}$
24. Convert X_b to {0, 1} by Sigmoid function
25. If (rand < A_b & $fitness_b$ > globalFitness):
26. $F_b = f_{min} + (f_{max} - f_{min})$ rand
27. $V_b = V_b + (xHat - X_b) F_b$
28. $X_b = X_b + V_b$
29. Convert X_b to {0, 1} by Sigmoid function
30. Return xHat.

Algorithm 1, shows the process of feature selection by Bat Algorithm. Initially, a population with random positions is established as candidate genes for each bat. Subsequently, the loudness, pulse emission rate, velocity, minimum and maximum frequency, and fitness are initialized. In each iteration, candidate genes are specified for each bat, and the accuracy of the selected subset is calculated. Based on the number of candidate genes, the fitness of each bat is then evaluated. Lines 13 to 20 indicate the best position and global fitness if the fitness is improved. Lines 21 to 29 update the positions of the bats. Finally, Line 30 returns the best subset of selected genes (best position) as the final solution.

2.4. Gray Wolf Algorithm

The Gray Wolf algorithm is a representative swarm-intelligence algorithm, inspired by the leadership hierarchy and hunting mechanism of gray wolves in nature. Gray wolves, recognized as apex predators, typically have an average group size of 5–12. Within the hierarchy of the Gray Wolf algorithm, the alpha (α) is considered the most dominant member of the group. The remaining subordinates, beta (β) and delta (δ), assist in controlling the majority of wolves in the hierarchy, which are considered as omega (ω). The ω wolves hold the lowest ranking within the hierarchy [42]. The mathematical model of the hunting mechanism of gray wolves comprises the following stages:

- Tracking, chasing, and approaching the prey.

- Pursuing, encircling, and harassing the prey until it ceases movement.
- Attacking the prey.

During the hunt, gray wolves encircle the prey. The hunting of prey is typically guided by α, β, and δ, who participate occasionally. The best candidate solutions, namely α, β, and δ, possess superior knowledge about the potential location of the prey. The other search agents (ω) update their positions according to the positions of the three best search agents [43].

Algorithm 2, illustrates the process of feature selection by the Gray Wolf Algorithm. Initially, a population is generated with random positions representing candidate genes for each wolf. Subsequently, the values of the parameters a, A, and C are initialized. Candidate genes are identified for each wolf, and the accuracy of the selected subset is calculated. The fitness of each wolf is then evaluated based on the number of candidate genes. Line 6 specifies the first, second, and third best wolves. During each iteration of the algorithm (lines 7 to 13), the positions and parameters of a, A, and C are updated, and both accuracy and fitness are recalculated. The positions of the top three wolves are updated accordingly, reflecting any improvements in global fitness. Finally, line 14 returns the best subset of the selected genes (the position of the best wolf) as the solution.

Algorithm 2. Feature selection by Gray Wolf Algorithm.
1. Initialize the gray wolf population X_w, random number in {0, 1}, w = 1, 2, ..., nWolves.
2. Initialize a, A and C.
3. Find (X_w == 1) and make train and test subsets.
4. Calculate accuracy.
5. fitness = (theta × accuracy + (1-theta) × (1- $\frac{\text{sum}(X_w == 1)}{\text{nFeatures}}$))
6. X_α, X_β, X_δ = The first, second and third best wolves.
7. For (t in maxIter):
8. For (w in nWolves):
9. Update the position of the current wolf.
10. Update a, A, C.
11. Calculate accuracy.
12. fitness = (theta × accuracy + (1-theta) × (1- $\frac{\text{sum}(X_w == 1)}{\text{nFeatures}}$))
13. Update X_α, X_β, X_δ.
14. Return X_α

2.5. Classifiers

In order to assess the subset of candidate features selected through the feature selection process, we employed two algorithms as described in the preceding section. Among the machine learning classification algorithms, we utilized ten classifiers: K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Naive Bayes, Ridge Regression, Decision Tree (DT), Random Forest (RF), Bagging Classifier, Light Gradient Boosting Machine (LightGBM), Perceptron, and Linear Discriminant Analysis (LDA). Each of these classifiers was independently applied in the feature selection processes of both the Bat Algorithm and the Gray Wolf Optimization Algorithm. For the KNN classifier, the number of neighbors was set to 7. For ensemble methods such as RF and LightGBM, the number of weak learners was set to 100. The parameters for all other classifiers were kept at their default settings. It should be noted that the choice of classifiers is not limited to the ones used in this study, and any other classifier can be incorporated as needed.

2.6. Selecting the Optimal Classifier

In order to identify the optimal classifier from the ten methods selected during the feature selection process, we introduced modifications and incorporated a heuristic component into the Bat and Gray Wolf algorithms. This approach ensures that while the best features are being selected, the most suitable classifier for each dataset is also identified. In both the Bat and Gray Wolf feature selection methods, we initially generate a random population. For both methods, ten agents are considered. Each agent possesses a position represented by a binary vector, with candidate features assigned a specific value within this vector. To evaluate the fitness of candidate features, we employ the ten classifiers mentioned earlier in the cost function. Initially, each agent is randomly assigned one of the classifiers, and the fitness of candidate features is evaluated. In addition to assessing the fitness of each agent, we also retain the classifier with which that agent was evaluated. Up to this point, no additional computational burden has been imposed on the algorithms, meaning the performance of the ten agents with the ten classifiers has been estimated. Meta-heuristic methods, particularly swarm intelligence algorithms, commence with an initial population, and the algorithm strives to converge towards the most favorable positions. In both of our feature selection methods (Bat and Gray Wolf), changes in agents' positions are introduced in each round to guide them towards the optimal solutions. In addition to evaluating the fitness of each agent in the current position, the fitness in new positions must also be assessed with a classifier. If these positions are superior, they are updated; otherwise, the new position is disregarded.

Prior to initiating the second iteration, we implement a Roulette Wheel using the calculated score for each classifier. The Roulette Wheel Selection algorithm, also known as Fitness Proportionate Selection, is a method employed in genetic algorithms [44] to select potentially beneficial solutions for recombination. Each potential solution (in this case, features selected by agents) in the population is assigned a fitness score based on its quality or suitability. Each classifier's fitness score is then utilized to assign a proportion of the roulette wheel. The higher the fitness score, the larger the classifier's portion of the wheel. The proportion of the Roulette Wheel assigned to each classifier is calculated as follows:

$$p_i = \frac{f_i}{\sum_{j=1}^{n} f_j} \qquad (1)$$

Where p_i represents the proportion of the i-th classifier, f_i denotes the fitness of the i-th agent as determined by the i-th classifier, f_j signifies the fitness of the j-th classifier, and n is the total number of classifiers.

In the second iteration, the implemented Roulette Wheel is utilized to assign each classifier. A random number is generated within the range of 0 to 1. The classifier whose segment encompasses this number is selected. This process is iteratively performed until all agents are selected for the second generation. The fundamental concept is that classifiers with superior fitness have a higher probability of being selected, yet there remains a possibility for less fit classifiers to be chosen. This strategy maintains diversity within the population and prevents premature convergence. From this iteration forward, each time there is a requirement to calculate the fitness of an agent, the Roulette Wheel is updated. Consequently, classifiers that have yielded superior results will likely have more selection opportunities.

Algorithm 3 illustrates the process of identifying the optimal classifier in conjunction with selecting the best features. Initially, a classifier is assigned to each agent (bat or wolf). The fitness of each agent is then evaluated using the assigned classifier. Based on the obtained fitness, the probability of selecting each classifier is initialized. A roulette wheel is constructed using these probabilities. Between lines 5 and 10, the position of each agent is updated using Algorithms 1 and 2. Subsequently, a classifier is selected for each agent using the roulette wheel, and the fitness is recalculated with the assigned classifier. The probabilities of the classifiers and the roulette wheel are then updated accordingly. Finally, in line 11, the best subset of genes (optimal positions) identified by the best classifier is returned as the final solution.

Algorithm 3. Rating the classifiers by Roulette Wheel probabilities.
1. Assign a classifier to each agent randomly.
2. Calculate the fitness of each agent by assigned classifier.
3. Calculate the classifiers selecting probabilities by obtained fitness.
4. Make a Roulette Wheel.
5. For (t in maxIter):
6. Update the Bats and Wolves positions by algorithms 1 and 2.
7. Select a classifier by Roulette Wheel for each agent.
8. Calculate the fitness of each agent by assigned classifier.
9. Update the classifiers selecting probabilities by obtained fitness.
10. Update the Roulette Wheel.
11. Return best classifier and best position.

The number of features selected by the Bat and Gray Wolf methods varies in each iteration. Therefore, to evaluate the candidate features in proportion to their numbers, we utilized the following equation:

$$\text{Fitness} = (\theta \times \text{accuracy} + (1-\theta) \times (1 - \frac{\Sigma(x_i = 1)}{\text{nFeatures}})) \qquad (2)$$

The number of Bats and Wolves was set to 10, and the termination condition was established at 50. To ensure the participation of all samples in both the training and testing processes, we employed a 5-fold cross-validation technique and computed the mean value.

3. RESULTS AND DISCUSSION

In this section, we examine the proposed method. We employed two dimensionality reduction methods, namely the Bat and Gray Wolf algorithms, to select the optimal genes from four cancer datasets. Ten classifiers were used to evaluate candidate data subsets by both the Bat and Gray Wolf methods. For the Bat method, we considered 10 bats and 50 repetitions, and for the Gray Wolf method, we considered 10 wolves and 50 repetitions. To ensure all data were included in both the training and testing processes, we utilized a 5-fold cross-validation method. For increased certainty, the feature selection process was repeated 10 times with both methods. The results are presented in Tables 2 to 9 (given in supplementary data).

As can be observed in Table 2, the best values obtained from 10 executions on the leukemia cancer dataset show that the KNN classifier achieved an accuracy of 84.76% with 3767 features out of 7129.

The SVM method achieved an accuracy of 90.29% with 3984 features. The Bayes method yielded better results, achieving an accuracy of 98.57% with 3031 features. The Ridge method achieved the same accuracy as the Bayes method, but with fewer selected features (2545 genes). The DT method achieved an accuracy of 95.81% with 5224 features. The RF method achieved an accuracy of 98.57% with 3031 features. The Bagging method, in its best performance, obtained an accuracy of 97.24% with 4536 features. The best result, 100% accuracy, was achieved by LightGBM with 4828 features. The Perceptron method achieved an accuracy of 96.00% with 3300 features, and finally, the LDA method achieved an accuracy of 91.90% with 4834 features in its best iteration.

After noting the above results, the Bat feature selection method was implemented using the same 10 classifiers, but with the proposed method. In each repetition of the Bat algorithm, after recording the obtained accuracy, the roulette wheels are updated with new values for the classifiers. Therefore, in the next iteration, the probability of choosing a classifier that has yielded better results is higher. The results of executing the above process 10 times on the leukemia cancer dataset are shown in the last part of Table 2. As can be seen, LightGBM, Bayes, Bagging, and Ridge have been selected as the best classifiers in different iterations. Among these, LightGBM, with 5260 features, achieved the best result that was discussed in the previous section, 100 percent. As a result, the best classifier has been found among the 10 used classifiers. In Figure 2(a), the change in the values of the selection probabilities of the classifiers in iterations 1, 10, 20, 30, 40, and 50 is shown. Classifiers such as KNN and SVM, due to poor results, had their values remain constant on the Roulette Wheel. In contrast, classifiers such as LightGBM, with more repetitions, saw an increase in the probability of their selection.

With respect to the SRBCT cancer dataset (as shown in Table 3), the Bat feature selection method, in its optimal performance, achieved an accuracy of 100%. Classifiers such as Bayes, Ridge, Random Forest (RF), Light Gradient Boosting Machine (LightGBM), and Perceptron yielded similar results. As can be observed in the final section of Table 3, our proposed method identified the Ridge classifier as the most effective for the feature extraction process using the Bat method, achieving the highest accuracy of 100%. Figure 2(b) illustrates the probabilities associated with selecting classifiers for the feature selection process, utilizing the Bat method on the Small Round Blue Cell Tumors (SRBCT) cancer dataset. Table 4 presents the outcomes of the aforementioned method on the prostate cancer dataset. It is evident that due to the substantial number of features (12,600 genes), the most favorable result is an accuracy of 93.44% with 9,889 features, which is associated with the Ridge classifier. The Light Gradient Boosting Machine (LightGBM) also achieved an accuracy of 93.41% with 6,968 features. Upon examining the final section of Table 4, the accuracy and efficacy of our proposed method become apparent. Among the classifiers, LightGBM has achieved the highest accuracy of 94.12% with 7,527 features, thereby establishing itself as the optimal classifier in this context. The Bayes, Random Forest (RF), and Perceptron methods maintained a constant selection probability across different iterations due to their relatively weaker performance, as illustrated in Figure 2(c). Notably, the selection probability for the Light Gradient Boosting Machine (LightGBM) increased consistently across different iterations due to its superior performance.

In our analysis of the final dataset, Colon cancer (presented in Table 5), two classifiers demonstrated superior results compared to others across various executions. The Ridge classifier, with 923 features, and the Linear Discriminant Analysis (LDA) classifier, with 853 features out of 2000 genes, both achieved an accuracy of 91.92%. In the final section of Table 5, our method identified LDA as the best classifier, selecting 914 features with an accuracy of 87.18%. Figure 2(d), similar to other Figures, displays the selection probabilities of classifiers across different iterations. To validate the effectiveness of our proposed method, we also implemented the entire process using the Gray Wolf feature selection method. The results are presented in Tables 6, 7, 8, and 9. As shown in Table 6, our Gray Wolf-based method selected the LightGBM classifier for the Leukemia dataset, achieving 100% accuracy, which matches the best result obtained from the implementation of Greg's method with individual classifiers. For the SRBCT dataset, our method selected the Bayes classifier, achieving 100% accuracy, as can be verified by referring to Table 7. This suggests that it may be the most suitable classifier for this data. Upon examining the

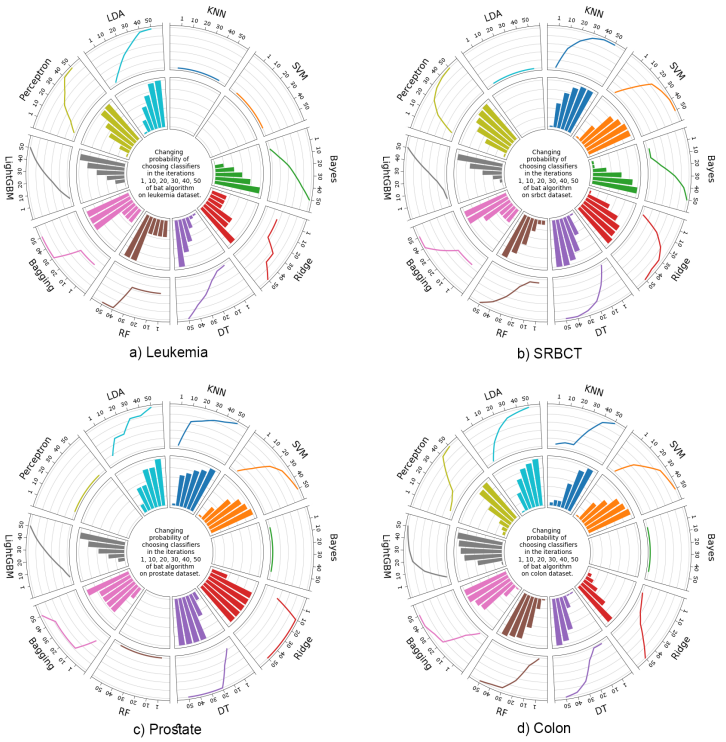

Fig. 2. Probabilities of selecting optimal classifiers for the feature selection process using the Bat algorithm on various cancer datasets.

Prostate cancer dataset with our proposed method, the LightGBM classifier selected 7747 features and achieved an accuracy of 94.13%, as can be verified by referring to Table 8. This indicates that it is the optimal classifier. As can be seen in Table 9, the proposed Gray Wolf feature selection method also identified the best classifier for the Colon cancer dataset, LDA, which achieved an accuracy of 91.92%.

Figure 3, illustrate the probabilities associated with selecting different classifiers across various iterations, utilizing the Gray Wolf feature selection method on a range of cancer datasets. Figure 3(a) presents the probabilities obtained at iterations 1, 10, 20, 30, 40, and 50 using our proposed method for leukemia cancer. As can be seen, the probability of choosing KNN, LDA and Perceptron methods is less than other methods. Similarly, Figures 3(b), 3(c), and 3(d) illustrate the probabilities obtained for SRBCT, Prostate, and Colon cancers, respectively.The experimental procedures were conducted on a computer system equipped with a 2.40 GHz processor, 8.0 GB of RAM, and operating on the Windows 10 platform. Feature selection is crucial in machine learning methodologies, as it identifies irrelevant features, thereby enhancing accuracy and reducing computational time. Due to the superior results of wrapper-based methods compared to filter-based and embedding-based methods, meta-heuristic approaches have been employed in various feature selection problems. The fitness function that estimates the merit of the selected features by these methods is of significant importance and can greatly influence both accuracy and efficiency. The challenge lies in determining the most suitable algorithm for estimating the merit of a specific dataset. The proposed method was evaluated against a number of existing methods, with the comparative results presented in Table 10. This comparison provides a comprehensive understanding of the performance of the proposed method relative to established methodologies.

Table 10. The accuracy of our proposed method versus several existing methods for comparative analysis.

Method	Leukemia	SRBCT	Prostate	Colon
Aydadenta [12]	N/A	N/A	88.97	85.87
Cilia (NN) [13]	99.44	N/A	N/A	91.94
Wu [19]	97.81	N/A	N/A	88.36
Jian [21]	100	100	N/A	95.00
Our proposed model	100	100	94.13	91.92

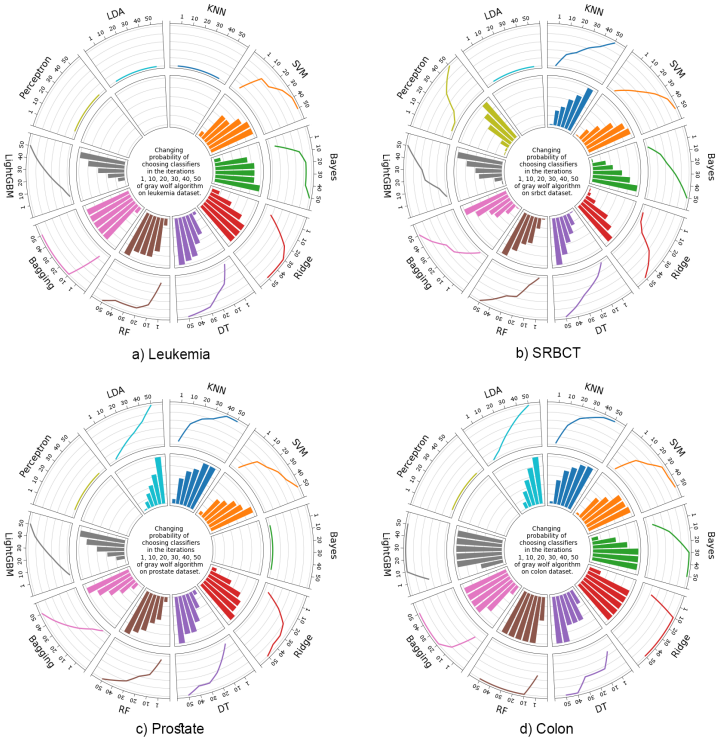

Fig. 3. Probabilities of selecting optimal classifiers for the feature selection process using the Gray Wolf algorithm on various cancer datasets.

4. CONCLUSIONS

In the present study, we introduced a simple yet practical method to select the optimal classifier in the fitness function. Our proposed method demonstrates which classifier is best suited for the studied collection. To implement the proposed model, we utilized two metaheuristic algorithms based on the swarm intelligence of the bat and the gray wolf. We then conducted the feature selection process using 10 various classifiers separately and recorded the results.

Subsequently, we integrated our proposed method into both the bat and gray wolf algorithms, performed the feature selection process with them, and noted the selected classifiers apart from the results. We analyzed Leukemia, SRBCT, Prostate, and Colon cancer datasets to evaluate our proposed method. Our findings indicate that our model identifies the best classifier for the desired dataset without incurring computational overhead while selecting the best features. Consequently, when using wrapper-based methods for feature selection, the optimal classifier can be selected for the investigated dataset.

5. DATA AVAILABILITY

The datasets examined in this study, along with the corresponding Python codes and results, are accessible to editors, reviewers, and readers via our GitHub page at the following URL: https://github.com/Farzad-Zandi/Selecting-Optimal-Classifier.

6. CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

7. REFERENCES

1. A. Jović, K. Brkić, and N. Bogunović. A review of feature selection methods with applications. *38th international convention on information and communication technology, electronics and microelectronics (MIPRO), Ieee, Opatija, Croatia, pp.* 1200-1205 (2015).
2. P.E. Greenwood and M.S. Nikulin (Eds.). A guide to chi-squared testing. *John Wiley & Sons, New York, USA* (1996).
3. B. Azhagusundari and A.S. Thanamani. Feature selection based on information gain. *International Journal of Innovative Technology and Exploring Engineering (IJITEE)* 2(2): 18-21 (2013).

4. A.G. Asuero, A. Sayago, and A. González. The correlation coefficient: An overview. *Critical Reviews in Analytical Chemistry* 36(1): 41-59 (2006).
5. X.W. Chen and J.C. Jeong. Enhanced recursive feature elimination. *6th International Conference on Machine Learning and Applications (ICMLA), IEEE, Cincinnati, OH, USA, pp.* 429-435 (2007).
6. V. Kumar and S. Minz. Feature selection. *SmartCR* 4(3): 211-229 (2014).
7. V. Fonti and E. Belitser. Feature selection using lasso. *VU Amsterdam Research Paper in Business Analytics* 30: 1-25 (2017).
8. X.S. Yang. Metaheuristic optimization. *Scholarpedia* 6(8): 11472 (2011).
9. P. Agrawal, H.F. Abutarboush, T. Ganesh, and A.W. Mohamed. Metaheuristic algorithms on feature selection: A survey of one decade of research (2009-2019). *Ieee Access* 9: 26766-26791 (2021).
10. I. Muhammad and Z. Yan. Supervised machine learning approaches: a survey. *ICTACT Journal on Soft Computing* 5(3): 946-952 (2015).
11. M.D. Purbolaksono, K.C. Widiastuti, M.S. Mubarok, and F.A. Ma'ruf. Implementation of mutual information and bayes theorem for classification microarray data. *Journal of Physics: Conference Series* 971(1): 012011 (2018).
12. H. Aydadenta and A. Adiwijaya. A clustering approach for feature selection in microarray data classification using random forest. *Journal of Information Processing Systems* 14(5): 1167-1175 (2018).
13. N.D. Cilia, C. De Stefano, F. Fontanella, S. Raimondo, and A. Scotto di Freca. An experimental comparison of feature-selection and classification methods for microarray datasets. *Information* 10(3): 109 (2019).
14. V. Bolón-Canedo, N. Sánchez-Marono, A. Alonso-Betanzos, J.M. Benítez, and F. Herrera. A review of microarray datasets and applied feature selection methods. *Information sciences* 282: 111-135 (2014).
15. M. Al-Batah, B. Zaqaibeh, S.A. Alomari, and M.S. Alzboon. Gene microarray cancer classification using correlation based feature selection algorithm and rules classifiers. *International Journal of Online & Biomedical Engineering* 15(8): 62 (2019).
16. S.K. Baliarsingh, C. Dora, and S. Vipsita. Jaya optimized extreme learning machine for breast cancer data classification. *Intelligent and Cloud Computing: Proceedings of ICICC 2019* 2: *Springer, Singapore, pp.* 459-467 (2021).
17. Q. Su, Y. Wang, X. Jiang, F. Chen, and W.C. Lu. A cancer gene selection algorithm based on the KS test and CFS. *BioMed Research International* 1: 1645619 (2017).
18. F.K. Ahmad. A comparative study on gene selection methods for tissues classification on large scale gene expression data. *Jurnal Teknologi* 78(5-10): 116-125 (2016).
19. Q. Wu, Z. Ma, J. Fan, G. Xu, and Y. Shen. A feature selection method based on hybrid improved binary quantum particle swarm optimization. *Ieee Access* 7: 80588-80601 (2019).
20. S.A. Medjahed, T.A. Saadi, A. Benyettou, and M. Ouali. Kernel-based learning and feature selection analysis for cancer diagnosis. *Applied Soft Computing* 51: 39-48 (2017).
21. I. Jain, V.K. Jain, and R. Jain. Correlation feature selection based improved-binary particle swarm optimization for gene selection and cancer classification. *Applied Soft Computing* 62: 203-215 (2018).
22. S. Shahbeig, M.S. Helfroush, and A. Rahideh. A fuzzy multi-objective hybrid TLBO–PSO approach to select the associated genes with breast cancer. *Signal Processing* 131: 58-65 (2017).
23. M. Abd-Elnaby, M. Alfonse, and M. Roushdy. Classification of breast cancer using microarray gene expression data: A survey. *Journal of Biomedical Informatics* 117: 103764 (2021).
24. O. Kramer (Ed.). Dimensionality reduction with unsupervised nearest neighbors. *Springer, Berlin, Heidelberg, Germany* (2013).
25. D.A. Pisner and D.M. Schnyer. Support vector machine. *Machine learning, Elsevier, Amesterdam, Netherlands, pp.* 101-121 (2020).
26. M.E. Glickman and D.A. Van Dyk. Basic bayesian methods. *Topics in Biostatistics* 404: 319-338 (2007).
27. C. Peng and Q. Cheng. Discriminative ridge machine: A classifier for high-dimensional data or imbalanced data. *IEEE Transactions on Neural Networks and Learning Systems* 32(6): 2595-2609 (2020).
28. Y.Y. Song and L.U. Ying. Decision tree methods: applications for classification and prediction. *Shanghai Archives of Psychiatry* 27(2): 130 (2015).
29. L. Breiman. Random forests. *Machine learning* 45: 5-32 (2001).
30. L. Breiman. Bagging predictors. *Machine Learning* 24: 123-140 (1996).
31. G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, and T.Y. Liu. Lightgbm: A highly efficient gradient boosting decision tree. *31st Conference*

on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA (2017).
32. M.C. Popescu, V.E. Balas, L. Perescu-Popescu, and N. Mastorakis. Multilayer perceptron and neural networks. *WSEAS Transactions on Circuits and Systems* 8(7): 579-588 (2009).
33. P. Xanthopoulos, P.M. Pardalos, T.B. Trafalis, P. Xanthopoulos, P.M. Pardalos, and T.B. Trafalis (Eds.). Linear discriminant analysis. In: Robust Data Mining. *SpringerBriefs in Optimization. Springer, New York* pp. 27-33 (2013).
34. T.R. Golub, D.K. Slonim, P. Tamayo, C. Huard, M. Gaasenbeek, J.P. Mesirov, and *et al*. Molecular classification of cancer: class discovery and class prediction by gene expression monitoring. *Science* 286(5439): 531-537 (1999).
35. J. Khan, J.S. Wei, M. Ringner, L.H. Saal, M. Ladanyi, F. Westermann, ... and P.S. Meltzer. Classification and diagnostic prediction of cancers using gene expression profiling and artificial neural networks. *Nature Medicine* 7(6): 673-679 (2001).
36. K. Nirmalakumari, H. Rajaguru, and P. Rajkumar. Microarray prostate cancer classification using eminent genes. In: *2021 Smart Technologies, Communication and Robotics (STCR). Sathyamangalam, India, pp.* 1-5. (2021).
37. U. Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, and A.J. Levine. Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. *Proceedings of the National Academy of Sciences* 96(12): 6745-6750 (1999).
38. D.R. Griffin, F.A. Webster, and C.R. Michael. The echolocation of flying insects by bats. *Animal Behavior* 8(3-4): 141-154 (1960).
39. W. Metzner. Echolocation behaviour in bats. *Science Progress* 75: 453-465 (1991).
40. H.U. Schnitzler and E.K. Kalko. Echolocation by insect-eating bats: we define four distinct functional groups of bats and find differences in signal structure that correlate with the typical echolocation tasks faced by each group. *Bioscience* 51(7): 557-569 (2001).
41. X.S. Yang. Bat algorithm for multi-objective optimization. *International Journal of Bio-Inspired Computation* 3(5): 267-274 (2011).
42. S. Mirjalili, S.M. Mirjalili, and A. Lewis. Grey wolf optimizer. *Advances in Engineering Software* 69: 46-61 (2014).
43. N. Mittal, U. Singh, and B.S. Sohi. Modified grey wolf optimizer for global engineering optimization. *Applied Computational Intelligence and Soft Computing* 1: 7950348 (2016).
44. S. Anand, N. Afreen, and S. Yazdani. A novel and efficient selection method in genetic algorithm. *International Journal of Computer Applications* 129(15): 7-12 (2015).

SUPPLEMENTARY DATA

Table 2. The accuracy values obtained from the Bat feature selection algorithm were evaluated using ten different classifiers on the Leukemia dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
5311	83.43	5978	87.62	3535	98.57	4835	98.57	3960	94.57	3033	98.57	4536	97.24	5501	100	4879	90.38	5420	90.38	5258	98.57	Ridge
5087	84.76	3984	90.29	4542	98.57	5700	98.57	4606	93.05	3573	98.57	4491	97.12	4828	100	3496	95.90	5191	90.48	4481	98.67	LightGBM
5649	84.76	5071	87.62	3831	98.57	4274	98.57	4295	93.05	4843	98.57	4451	96.89	5980	98.57	5330	95.90	5068	90.48	5260	100	LightGBM
3611	83.43	5201	86.19	3331	98.57	4320	98.57	5745	93.05	5791	98.57	4429	97.01	5402	100	4220	94.57	5777	90.38	5271	98.57	Bayes
5065	83.43	5345	87.62	4523	98.57	5717	98.57	4059	94.48	3031	98.57	4578	97.24	5736	98.57	3300	96.00	3390	91.81	3347	98.57	Bagging
3767	84.76	3024	86.19	3965	98.57	5681	98.57	3449	93.05	5577	98.57	4459	96.76	5483	98.57	3375	94.57	3343	90.48	2971	9857	Bagging
4554	84.76	5519	86.19	3031	98.57	5880	98.57	3758	93.05	3954	98.57	4821	97.01	5237	98.57	4563	92.00	4834	91.90	3977	98.67	LightGBM
5354	84.76	2386	86.19	3460	98.57	2545	98.57	4138	93.05	3171	98.57	4675	97.12	4971	100	4257	90.67	5713	90.38	6082	98.57	Bayes
5754	84.76	5919	87.62	3052	98.57	3778	98.57	3501	93.14	4876	98.57	4875	96.93	4949	100	5244	90.57	4617	89.05	4054	98.57	Ridge
5567	83.43	4325	86.19	5887	98.57	4064	98.57	5224	95.81	3180	98.57	4931	97.03	5780	98.57	4967	95.90	3145	89.05	5707	98.57	Bagging

Table 3. The accuracy values obtained from the Bat feature selection algorithm were evaluated using ten different classifiers on the SRBCT dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
1456	90.15	899	97.57	847	100	1387	100	1879	92.72	1498	100	1270	98.82	1621	100	1032	100	1803	78.31	983	100	Ridge
1339	87.72	1194	97.57	1892	100	1129	100	1828	93.97	1704	100	1356	98.75	1649	100	1303	98.82	1053	74.93	1244	100	Ridge
1166	87.72	1774	97.57	1473	100	1731	100	1353	91.47	1596	100	1321	98.46	1670	100	1774	98.82	1581	72.28	1130	100	Ridge
1765	90.15	1893	97.57	1539	100	976	100	1459	93.97	1678	100	1293	98.78	1691	100	1300	98.82	1183	69.85	1584	100	Ridge
1058	87.79	1109	97.57	1932	100	1416	100	1064	92.79	1579	100	1311	98.64	1673	100	1774	98.82	1290	77.13	1614	100	Ridge
1413	87.79	1425	97.57	1898	100	1762	100	1107	92.72	1564	100	1342	98.26	1752	100	1950	98.82	1618	72.43	1897	100	Ridge
1052	87.72	1567	97.57	988	100	1581	100	1746	93.97	1626	100	1281	98.01	1638	100	1389	98.82	1795	71.10	1880	100	Ridge
1778	91.32	1130	97.57	1811	100	1354	100	1860	92.79	1653	100	1330	97.76	1652	100	1378	98.82	1851	71.18	1956	100	Ridge
1866	87.65	1791	97.57	1173	100	1275	100	1564	93.97	1527	100	1262	98.37	1679	100	1877	100	1577	75.00	1468	100	Ridge
1062	87.87	1820	97.57	1224	100	1438	100	1514	92.79	1546	100	1257	97.89	1713	100	1831	100	1457	72.35	1449	100	Ridge

Table 4. The accuracy values obtained from the Bat feature selection algorithm were evaluated using ten different classifiers on the Prostate dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
9813	80.93	7538	80.19	5135	56.67	9889	93.44	10222	85.34	7606	92.72	9872	91.93	6968	93.41	8080	81.75	9530	86.11	9081	92.70	LightGBM
6722	80.19	8170	81.64	7026	56.67	9628	92.70	8138	88.20	6561	91.75	9991	91.93	7002	90.91	5341	81.75	9229	84.66	8447	93.39	LightGBM
6054	79.47	8390	80.19	8445	56.67	9477	92.72	10645	86.75	8012	91.26	9615	91.47	6898	92.49	7102	80.26	9177	86.08	8910	92.01	Bagging
5772	80.19	7116	79.47	10422	56.67	8910	91.27	9828	87.76	7316	92.41	9646	91.78	6416	92.17	9230	85.24	7512	86.11	9984	92.67	LightGBM
10252	78.73	8780	80.19	8258	56.67	6571	92.70	9159	87.92	7189	91.43	9719	90.99	7137	91.46	7068	81.64	7029	87.59	7378	92.72	RF
7986	80.90	5266	79.44	9750	56.67	8987	92.72	10573	88.09	7145	92.66	9486	90.41	6459	91.73	10058	81.69	9102	84.60	7527	94.12	LightGBM
9315	80.21	8374	79.47	4581	56.67	10182	92.72	8845	86.26	7416	91.83	9629	91.38	6743	90.78	9160	84.68	8994	85.34	9827	91.93	LightGBM
7346	80.19	8284	80.16	8415	56.67	7353	91.27	8291	85.47	6985	92.58	9813	90.12	6972	91.49	6743	82.43	9347	86.85	5867	92.70	RF
6453	80.19	10165	80.93	5003	56.67	8407	92.72	8697	86.59	7373	91.22	9687	91.28	6842	92.83	10194	81.00	7067	84.63	6079	93.44	LightGBM
5013	80.19	6380	81.67	10520	56.67	9540	91.98	10357	86.38	7842	91.68	9649	91.09	6794	93.30	8382	83.17	8204	85.34	7043	91.24	RF

Table 5. The accuracy values obtained from the Bat feature selection algorithm were evaluated using ten different classifiers on the Colon dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
838	80.64	1109	87.31	1071	70.64	1238	90.26	916	88.97	1030	85.51	1161	88.97	1178	87.18	1331	87.05	1392	91.92	1013	87.05	LDA
1431	77.56	1386	84.23	1150	67.44	872	90.26	1680	84.10	1127	83.19	1073	90.13	1150	85.64	1505	88.59	732	91.92	1263	85.51	LDA
1126	77.56	1620	84.23	1346	69.23	1350	91.92	1139	85.64	1132	84.61	1692	87.18	912	88.85	1249	86.92	1519	91.92	1598	87.18	LDA
1069	77.44	1254	84.23	1117	69.10	1615	91.92	1510	84.10	1039	85.12	1529	88.42	1415	88.72	1571	87.05	853	91.92	1661	87.18	LDA
1242	79.10	1103	84.23	762	69.10	1534	90.38	1574	87.18	1143	83.49	1743	88.19	1107	85.64	1408	86.92	1520	91.92	914	87.18	LDA
924	77.56	722	85.77	1024	72.31	1108	91.92	1454	87.05	1087	84.67	1746	88.84	1197	87.31	1189	86.92	1450	90.38	807	85.51	LDA
1331	79.10	1650	84.23	1052	70.77	1495	90.26	923	85.64	1053	85.20	1673	87.76	1041	87.18	1136	86.92	1094	91.92	1701	85.51	LDA
1518	77.56	1567	85.77	1747	69.10	1370	91.92	1161	87.44	969	84.89	1694	88.39	1512	85.77	1699	86.92	967	90.38	1285	85.51	LDA
995	80.64	1104	84.23	818	70.64	1343	91.92	1566	85.64	993	84.51	1358	87.94	1477	88.72	1418	88.59	1051	90.38	1503	85.51	LDA
1406	79.10	1388	85.77	1341	70.64	923	91.92	1383	90.26	1057	83.24	1361	88.09	1250	85.64	1378	87.05	1476	91.92	1327	85.51	LDA

Table 6. The accuracy values obtained from the Gray Wolf feature selection algorithm were evaluated using ten different classifiers on the Leukemia dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
4191	83.43	4269	87.62	4145	98.57	3585	98.57	4311	93.05	4158	98.57	4356	97.24	4265	100	4577	94.48	4472	90.38	4177	98.57	Ridge
4263	84.76	3590	86.19	3550	98.57	3542	98.57	4480	91.62	3484	98.57	4174	97.24	4572	100	4529	95.90	4299	90.48	3504	98.57	Ridge
3565	83.43	3532	86.19	4183	98.57	3588	98.57	4554	94.57	3504	98.57	4528	97.24	4493	100	4532	93.24	4448	90.48	4070	98.57	LightGBM
4517	84.76	3641	86.19	3586	98.57	3605	98.57	4523	93.05	4221	98.57	4496	95.90	4592	100	4395	94.57	4434	90.48	4206	98.57	Ridge
4439	84.76	3553	86.19	3571	98.57	3625	98.57	4284	93.05	4320	98.57	4344	97.24	4468	98.67	4276	90.57	4564	91.81	4066	98.57	Ridge
4463	84.76	4262	87.62	3591	98.57	3538	98.57	4436	93.05	4078	98.57	4470	97.24	4231	98.67	4646	95.71	4363	90.48	4196	98.57	RF
4112	83.43	4405	87.62	3548	98.57	3518	98.57	4290	91.71	4205	98.57	4346	98.67	4334	100	4240	95.90	4233	90.38	4079	98.57	Ridge
4356	84.76	3756	86.19	3919	98.57	3585	98.57	4554	91.62	4290	98.57	4491	97.24	4250	98.67	4289	94.57	4461	91.81	4088	98.57	LightGBM
4436	84.76	4022	86.19	4205	98.57	4194	98.57	4348	93.05	4171	98.57	4263	97.14	4579	100	4396	95.90	4483	90.48	4128	98.57	Ridge
4558	84.76	3743	86.19	4037	98.57	3526	98.57	4451	91.62	4238	98.57	4618	97.24	4140	100	4502	95.90	4578	91.90	4502	100	LightGBM

Table 7. The accuracy values obtained from the Gray Wolf feature selection algorithm were evaluated using ten different classifiers on the SRBCT dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
1510	87.72	1151	97.57	1337	100	1151	100	1407	93.97	1115	100	1388	100	1335	100	1411	100	1421	72.43	1206	100	Ridge
1427	87.72	1349	97.57	1397	100	1152	100	1358	92.79	1155	100	1359	100	1176	100	1449	100	1447	73.46	1126	100	Bayes
1359	88.97	1132	97.57	1308	100	1130	100	1421	93.97	1131	100	1385	100	1127	100	1382	100	1427	71.25	1159	100	Ridge
1482	90.15	1180	97.57	1321	100	1131	100	1485	92.72	1180	100	1403	98.82	1344	100	1369	100	1505	72.35	1170	100	Ridge
1356	88.90	1146	97.57	1420	100	1186	100	1416	93.97	1152	100	1376	98.82	1340	100	1341	100	1506	72.35	1182	100	Ridge
1363	87.72	1346	97.57	1364	100	1168	100	1481	93.97	1174	100	1215	100	1334	100	1409	100	1506	71.10	1171	100	Ridge
1389	88.90	1155	97.57	1384	100	1159	100	1416	92.72	1155	100	1465	100	1439	100	1412	98.82	1504	72.35	1141	100	Ridge
1450	87.72	1135	97.57	1366	100	1158	100	1352	92.79	1139	100	1152	100	1155	100	1459	100	1382	73.53	1170	100	Ridge
1476	87.79	1155	97.57	1403	100	1160	100	1459	93.97	1160	100	1509	100	1398	100	1367	100	1327	73.46	1153	100	Ridge
1408	87.72	1165	97.57	1398	100	1140	100	1505	92.79	1119	100	1520	100	1147	100	1146	100	1391	72.28	1158	100	Ridge

Table 8. The accuracy values obtained from the Gray Wolf feature selection algorithm were evaluated using ten different classifiers on the Prostate dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
7497	80.19	6686	67.62	6258	56.67	7554	92.01	7878	87.49	7925	93.41	7640	91.19	7474	94.13	7531	83.23	7841	86.11	7329	92.67	LightGBM
7886	79.47	7324	67.62	7013	56.67	7677	92.72	7875	86.03	7991	92.70	7631	90.73	7361	92.39	7778	86.19	8166	85.34	8002	92.90	Bagging
7644	80.19	6320	67.62	6345	56.67	7599	92.72	8235	87.51	7873	94.18	7631	90.16	7118	93.19	7836	86.83	7689	84.66	7510	91.98	RF
8071	80.19	7552	67.62	6328	56.67	7981	92.72	8163	85.34	7921	92.72	7199	90.72	7243	92.67	7710	81.01	8140	85.37	7837	91.96	RF
8000	80.93	7417	67.62	6393	56.67	7506	92.70	7672	86.75	7476	92.72	7618	90.49	7521	92.21	8022	84.71	6336	86.11	7747	94.13	LightGBM
7928	80.19	7325	67.62	7316	56.67	8146	92.70	7786	87.51	7824	92.72	7475	90.49	7419	92.17	8036	83.23	8061	85.40	7817	92.72	RF
7974	80.93	7411	67.62	7718	56.67	7518	92.72	7812	86.43	7639	94.17	7194	90.17	7589	92.55	7992	83.17	7976	85.37	7850	91.98	RF
7787	80.21	6336	67.62	7390	56.67	8061	92.72	7749	86.18	7782	93.78	7643	90.41	7364	93.73	6398	83.02	7866	86.08	7829	91.16	Bagging
7736	79.44	6317	67.62	6231	56.67	7876	92.72	7718	85.76	7716	93.91	7215	90.32	7008	93.58	8168	83.23	7984	84.66	7946	91.98	LightGBM
8051	80.19	6575	67.62	6240	56.67	7781	92.72	7728	86.74	7549	93.19	7821	90.94	6952	93.91	7738	86.14	7732	85.37	7797	92.65	LightGBM

Table 9. The accuracy values obtained from the Gray Wolf feature selection algorithm were evaluated using ten different classifiers on the Colon dataset.

KNN		SVM		Bayes		Ridge		DT		RF		Bagging		LightGBM		Perceptron		LDA		Proposed		
nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	nFeatures	Accuracy	Selected Classifier
1260	77.56	1003	84.23	1241	70.77	1250	91.92	1269	87.31	1277	87.18	1318	87.31	1285	87.18	1260	87.05	1194	91.92	1168	90.26	LDA
1242	77.44	1027	84.23	1221	70.64	1301	91.92	1215	84.10	1320	85.64	1257	88.85	1198	87.18	1313	86.92	1169	91.92	994	88.72	LDA
1211	77.56	1223	85.77	1268	70.64	1157	90.38	1268	85.64	1241	87.18	1286	88.72	1227	87.18	1312	86.92	1223	91.92	1201	88.72	LDA
1285	78.97	1169	85.77	1251	70.77	1209	91.92	1221	85.64	1283	88.72	1261	90.38	1257	85.64	1243	86.92	1276	91.92	1167	90.26	LDA
1014	77.56	989	84.23	1221	70.64	1284	91.92	1257	85.64	1288	88.59	1288	88.72	1206	86.92	1297	88.59	1272	91.92	1169	88.72	LDA
1237	80.64	1180	85.77	1200	70.64	1183	90.38	1268	85.51	1186	88.07	1133	88.85	1212	85.64	1017	86.92	1195	91.92	1254	90.26	LDA
1254	77.56	1028	84.23	1215	70.77	1312	90.38	1260	87.31	1147	87.92	1252	88.72	1233	87.18	1214	88.59	1277	91.92	1314	91.92	LDA
1254	79.10	1015	84.23	1201	70.64	1007	90.38	1198	85.64	1223	87.49	1255	88.59	1249	87.18	1184	87.05	1199	91.92	1266	87.31	Bagging
1009	77.56	1012	85.77	1178	70.64	1244	91.92	1305	85.64	1279	87.41	1245	87.31	1242	85.64	1226	87.05	1182	91.92	1193	88.72	LDA
1308	78.97	1012	84.23	1172	70.64	1164	91.92	1226	87.18	1257	87.19	1221	88.59	1215	87.18	1186	88.59	1178	91.92	1254	90.38	LDA

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 273-281 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)852

Pakistan Academy of Sciences

Research Article

An Efficient Four Step Fifteenth Order Method for Solution of Non-Linear Models in Real-World Problems

Sanaullah Jamali[1]*, Zubair Ahmed Kalhoro[1], and Imran Qasim Memon[2]

[1]Institute of Mathematics and Computer Science, University of Sindh, Allama I.I. Kazi Campus, Jamshoro, Sindh, Pakistan

[2]Department of Basic Sciences and Related Studies, Mehran University of Engineering and Technology, Jamshoro, Sindh, Pakistan

Abstract: Non-linear equations are fundamental to a wide range of practical applications in engineering and applied sciences. This research paper presents a novel iterative scheme—a fifteenth-order approach—designed to effectively solve non-linear problems. The numerical results of the Proposed Scheme are thoroughly compared with those of existing methods. Graphical representations and basin of attraction analysis reveal that the fifteenth-order method achieves superior accuracy and efficiency, surpassing alternative methods in the precise estimation of solutions to non-linear problems.

Keywords: Real Word Problems, Taylor Series Expansion, Order of Convergence, Fifteenth Order Methods, Basin of Attraction.

1. INTRODUCTION

The prevalence of non-linear equations, as represented by the form $\emptyset(\varkappa) = 0$, is observed across various fields in both pure and applied sciences, engineering, and computing. Recent efforts have been dedicated by numerous scientists and engineers to the addressing of non-linear equations through both numerical and analytical methods [1]. In the field of literature, various Iterative Schemes and algorithms have been developed, drawing inspiration from techniques such as interpolation, Taylor's series, quadrature formulas, and decomposition. Additionally, there are several modifications and enhancements to existing methods, along with the introduction of various hybrid iterative approaches. Xiaojian [2] introduced several variants of the Chebyshev–Halley methods that do not require the second derivative. The proposed methods were demonstrated to have a minimum convergence order of three, with each iteration involving the evaluation of two functions and one first derivative. Lakho *et al.* [3] presented a seventh ordered three step method using Lagrange interpolation technique. Naseem *et al.* [4] proposed fourth method based on forward and finite difference schemes. Abbasi *et al.* [5] proposed a ninth order method using Hermite interpolation technique. Soomro *et al.* [6] proposed a bracketing approach based on the Regula Falsi method, leading to an enhanced convergence rate. Generally, the expression of the roots of non-linear or transcendental equations in closed form or their analytical computation is found to be challenging. The calculation of approximations of the roots, typically presented as either small isolating intervals or floating-point numbers, is enabled by root-finding algorithms.

In this paper, a hybrid four-step scheme for the solution of non-linear methods is presented. In this scheme, the convergence rate is enhanced, and the number of function evaluations required per iteration is reduced through the incorporation of a weight function and interpolation techniques, respectively, contributing to the improvement of

Received: April 2024; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Sanaullah Jamali <sanaullah.jamali@usindh.edu.pk>

the efficiency index. To evaluate the effectiveness of the proposed approach, numerous examples and Real-world problems were subjected to testing and comparison with existing methods (TE1 15th, TE2 15th, and TE3 15th as mentioned by Eftekhari [7], and FS 15th as indicated by Soleymani and Sharifi [8]). The results of these comparisons are presented in various Tables and Figures, while the stability of the Proposed Scheme is further validated through the analysis of the Basin of attraction. This method is not applicable for the system of nonlinear equations.

2. DERIVATION OF PROPOSED SCHEME

For the first step of Proposed Scheme we take Newton method [9, 10].

$$v_i = \varkappa_i - \frac{\phi(\varkappa_i)}{\phi'(\varkappa_i)} \tag{1}$$

And for the second step of Proposed Scheme we take again newton method Kumar *et al.* [11], reduce the function evaluations we modify it by replacing the $\phi'(v_i)$ by $\phi'(\varkappa_i)$ in second step and introduce weight function P

$$\left.\begin{aligned} &\text{And } P = 1 + 2a + a^2 + a^3, \quad a = \frac{\phi(v_i)}{\phi(\varkappa_i)} \\ &\xi_i = v_i - P\left(\frac{\phi(v_i)}{\phi'(\varkappa_i)}\right) \end{aligned}\right\} \tag{2}$$

For the third step of Proposed Scheme we take again newton method [11], to reduce the function evaluations we replacing the $\phi'(\xi_i)$ by $H'(\xi_i)$ Hermite interpolation in third step.

$$o_i = \xi_i - \frac{\phi(\xi_i)}{H'(\xi_i)}, \tag{3}$$

In three-point formula (3) it requires five function evaluation per iteration, to reduce the number of function evaluations we approximate $\phi'(\xi_i)$ using available data. Since we have four values $\phi(\varkappa)$, $\phi'(\varkappa), \phi(v)$ & $\phi(\xi)$ approximate ϕ by its Hermite's interpolating polynomial $H_3$of degree 3 at the nodes $\varkappa, v, \xi$ and utilize the approximation $\phi'(\xi) \approx H_3'(\xi)$ in the third step of the iterative scheme (3). Hermite's interpolating polynomial of third degree has the form.

$$H_3(\eta) = a_0 + a_1(\eta - \varkappa) + a_2(\eta - \varkappa)^2 + a_3(\eta - \varkappa)^3 \tag{4}$$

And its derivative is:

$$H_3'(\eta) = a_1 + 2a_2(\eta - \varkappa) + 3a_3(\eta - \varkappa)^2 \tag{5}$$

The unknown coefficients will be determined using available data from the conditions:

$$H_3(\varkappa) = \phi(\varkappa), H_3(v) = \phi(v), H_3(\xi) = \phi(\xi), \& H_3'(\varkappa) = \phi'(\varkappa)$$

Putting $\eta = \varkappa$ into (4) & (5) we get $a_0 = \phi(\varkappa)$ and $a_1 = \phi'(\varkappa)$. The coefficients a_2 and a_3 are obtained from the system of two linear equation formed by using the remaining two conditions $\eta = v$ & $\eta = \xi$ in (4) and we obtained.

$$a_2 = \frac{(\xi-\varkappa)\phi[v,\varkappa]}{(\xi-v)(v-\varkappa)} - \frac{(v-\varkappa)\phi[\xi,\varkappa]}{(\xi-v)(v-\varkappa)} - \phi'(\varkappa)\left(\frac{1}{\xi-\varkappa} - \frac{1}{v-\varkappa}\right)$$

$$\& \quad a_3 = \frac{\phi[\xi,\varkappa]}{(\xi-v)(\xi-\varkappa)} - \frac{\phi[v,\varkappa]}{(\xi-v)(v-\varkappa)} + \frac{\phi'(\varkappa)}{(\xi-\varkappa)(v-\varkappa)}$$

By putting the values of a_1, a_2, a_3 & $\eta = \xi$ in (5) we get:

$$H_3'(\xi) = 2(\phi[\varkappa,\xi] - \phi[\varkappa,v]) + \phi[v,\xi] + \frac{v-\xi}{v-\varkappa}(\phi[\varkappa,v] - \phi'(\varkappa)) \tag{6}$$

Finally, in fourth step of Proposed Scheme we take again newton method from Kumar *et al.* [11], to reduce the function evaluations we replacing the $\phi'(o_i)$ by $j'(o_i)$ in four step.

$$\varkappa_{i+1} = o_i - \frac{\phi(o_i)}{j'(o_i)} \tag{7}$$

where $j'(o_i) = \phi[\varkappa_i,\xi_i] + (\phi[v_i,\varkappa_i,\xi_i] - \phi[v_i,\varkappa_i,o_i] - \phi[\xi_i,\varkappa_i,o_i])(v_i - \xi_i)$

see in [7].

$$\left.\begin{aligned} &\text{Step. 1 } v_i = \varkappa_i - \frac{\phi(\varkappa_i)}{\phi'(\varkappa_i)} \\ &\text{Step. 2 } \xi_i = v_i - P\left(\frac{\phi(v_i)}{\phi'(\varkappa_i)}\right) \\ &\text{Step. 3 } o_i = \xi_i - \frac{\phi(\xi_i)}{H_3'(\xi_i)} \\ &\text{Step. 4 } \varkappa_{i+1} = o_i - \frac{\phi(o_i)}{j'(o_i)} \end{aligned}\right\} \tag{8}$$

Finally, (8) is our proposed Scheme, requiring only one first derivative evaluation and four function evaluations per iteration.

3. CONVERGENCE ANALYSIS

Theorem I: Consider α belonging to the set D, and let it be a simple root of a sufficiently

differentiable function $\phi: D \subset \mathbb{R} \to \mathbb{R}$. Here, D represents an open interval that includes $\varkappa_0$ as an initial estimate of α. In such circumstances, the method outlined in equation (8) achieves a fifteenth-order accuracy. Remarkably, it required only four function evaluations and a single first derivative calculation in each complete iteration, eliminating the necessity for second or higher-order derivatives.

Proof.
The function $\phi(\varkappa_i)$ can be expressed using its Taylor series expansion as:

$$\phi(\varkappa_i) = \sum_{m=0}^{\infty} \frac{\phi^m(\alpha)}{m!}(\varkappa_i - \alpha)^m = \phi(\alpha) + \phi'(\alpha)(\varkappa_i - \alpha) + \frac{\phi''(\alpha)}{2!}(\varkappa_i - \alpha)^2 + \frac{\phi'''(\alpha)}{3!}(\varkappa_i - \alpha)^3 + \cdots$$

For simplicity, we assume that

$$C_k = \left(\frac{1}{k!}\right)\frac{\phi^k(\alpha)}{\phi'(\alpha)}, k \geq 2.$$

and assume that $\omega_i = \varkappa_i - \alpha$. Thus, we have:
For step one:

$$\phi(\varkappa_i) = \phi'(\alpha)\left(\omega_i + C_2\omega_i^2 + C_3\omega_i^3 + C_4\omega_i^4 + \cdots + O\left(\omega_i^{16}\right)\right) \tag{9}$$

and

$$\phi'(\varkappa_i) = \phi'(\alpha)\left(1 + 2C_2\omega_i + 3C_3\omega_i^2 + 4C_4\omega_i^3 + \cdots + O\left(\omega_i^{15}\right)\right) \tag{10}$$

From eq (9) and (10)

$$\frac{\phi(\varkappa_i)}{\phi'(\varkappa_i)} = \frac{\phi'(\alpha)\begin{pmatrix}\omega_i + C_2\omega_i^2 + C_3\omega_i^3 + C_4\omega_i^4 + C_5\omega_i^5 \\ + C_6\omega_i^6 + C_7\omega_i^7 + \cdots + O(\omega_i^{16})\end{pmatrix}}{\phi'(\alpha)\begin{pmatrix}1 + 2C_2\omega_i + 3C_3\omega_i^2 + 4C_4\omega_i^3 + 5C_5\omega_i^4 \\ + 6C_6\omega_i^5 + 7C_7\omega_i^6 + \cdots + O(\omega_i^{15})\end{pmatrix}} \tag{11}$$

and

$$\frac{\phi(\varkappa_i)}{\phi'(\varkappa_i)} = \omega_i - C_2\omega_i^2 + (2C_2^2 - 2C_3)\omega_i^3 + (-4C_2^3 + 7C_3C_2 - 3C_4)\omega_i^4 + (8C_2^4 - 20C_3C_2^2 + 10C_4C_2 + 6C_3^2 - 4C_5)\omega_i^5 + \cdots + O(\omega_i^{16}) \tag{12}$$

Step. 1 $v_i = \varkappa_i - \frac{\phi(\varkappa_i)}{\phi'(\varkappa_i)} = C_2\omega_i^2 + (2C_3 - 2C_2^2)\omega_i^3 + (4C_2^3 - 7C_3C_2 + 3C_4)\omega_i^4$

$$+(-8C_2^4 + 20C_3C_2^2 - 10C_4C_2 - 6C_3^2 + 4C_5)\omega_i^5 + \cdots + O(\omega_i^{16}) \tag{13}$$

Now for step two:

$$\phi(v_i) = \phi'(\alpha)\begin{pmatrix} C_2\omega_i^2 + (2C_3 - 2C_2^2)\omega_i^3 + (5C_2^3 - 7C_3C_2 + 3C_4)\omega_i^4 \\ -2(6C_2^4 - 12C_3C_2^2 + 5C_4C_2 + 3C_3^2 - 2C_5)\omega_i^5 + \cdots + O\left(\omega_i^{16}\right)\end{pmatrix} \tag{14}$$

From equation (14) and (10):

$$\frac{\phi(v_i)}{\phi'(\varkappa_i)} = \frac{\phi'(\alpha)\begin{pmatrix} C_2\omega_i^2 + (2C_3 - 2C_2^2)\omega_i^3 \\ + (4C_2^3 - 7C_3C_2 + 3C_4)\omega_i^4 + \cdots + O(\omega_i^{16})\end{pmatrix}}{\phi'(\alpha)\begin{pmatrix}1 + 2C_2\omega_i + 3C_3\omega_i^2 + 4C_4\omega_i^3 + 5C_5\omega_i^4 \\ + 6C_6\omega_i^5 + 7C_7\omega_i^6 + \cdots + O(\omega_i^{15})\end{pmatrix}} \tag{15}$$

After simplifying we get:

$$\frac{\phi(v_i)}{\phi'(\varkappa_i)} = C_2\omega_i + (2C_3 - 3C_2^2)\omega_i^2 + (8C_2^3 - 10C_3C_2 + 3C_4)\omega_i^3 + (-20C_2^4 + 37C_3C_2^2 - 14C_4C_2 - 8C_3^2 + 4C_5)\omega_i^4 + \left(48C_2^5 - 118C_3C_2^3 + 51C_4C_2^2 + 55C_3^2C_2 - 18C_5C_2 - 22C_3C_4 + 5C_6\right)\omega_i^5 + \cdots + O\left(\omega_i^{16}\right) \tag{16}$$

And weight function:

$$P = 1 + 2a + a^2 + a^3, \text{ and } a = \frac{\phi(v_i)}{\phi(\varkappa_i)} \tag{17}$$

$$P = -1 - 2C_2\omega_i + (5C_2^2 + C_2 - 4C_3)\omega_i^2 + (-11C_2^3 - 2C_2^2 + 16C_3C_2 + 2C_3 - 6C_4)\omega_i^3 + (24C_2^4 + 4C_2^3 - 48C_3C_2^2 - 7C_3C_2 + 22C_4C_2 + 12C_3^2 + 3C_4 - 8C_5)\omega_i^4 + \cdots + O(\omega_i^{16}) \tag{18}$$

$$P\frac{\phi(v_i)}{\phi'(\varkappa_i)} = C_2\omega_i^2 + (2C_3 - 2C_2^2)\omega_i^3 - 3(2C_2C_3 - C_4)\omega_i^4 + (19C_2^4 - 6C_3C_2^2 - 8C_4C_2 - 4C_3^2 + 4C_5)\omega_i^5 \ldots + O(\omega_i^{16}) \tag{19}$$

Step. 2 $\xi_i = v_i - P\left(\frac{\phi(v_i)}{\phi'(v_i)}\right) = (4C_2^3 - C_2C_3)\omega_i^4 + (-27C_2^4 + 26C_3C_2^2 - 2C_4C_2 - 2C_3^2)\omega_i^5 + \cdots + O\left(\omega_i^{16}\right)$ (20)

$$\phi(\xi_i) = \phi'(\alpha)\left((4C_2^3 - C_2C_3)\omega_i^4 + \begin{pmatrix}-27C_2^4 + 26C_3C_2^2 \\ -2C_4C_2 - 2C_3^2\end{pmatrix}\omega_i^5 + \cdots + O\left(\omega_i^{16}\right)\right) \tag{21}$$

$$H'(\xi_i) = 2(\phi[\varkappa_i, \xi_i] - \phi[v_i, \varkappa_i]) + \phi[v_i, \xi_i] + (v_i - \xi_i)\phi[v_i, \varkappa_i, \varkappa_i] \tag{22}$$

$$H_3'(\xi_i) = \phi'(\alpha)\begin{pmatrix} 1 + C_2(8C_2^3 - 2C_3C_2 + C_4)\omega_i^4 \\ -2\begin{pmatrix}27C_2^5 - 26C_3C_2^3 + 3C_4C_2^2 \\ + 2C_3^2C_2 - C_5C_2 - C_3C_4\end{pmatrix}\omega_i^5 + \cdots + O\left(\omega_i^{16}\right)\end{pmatrix} \tag{23}$$

$$\frac{\phi(\xi_i)}{H_3'(\xi_i)} = \frac{\phi'(\alpha)\begin{pmatrix}(4C_2^3-C_2C_3)\omega_i^4\\ +\begin{pmatrix}-27C_2^4+26C_3C_2^2\\ -2C_4C_2-2C_3^2\end{pmatrix}\omega_i^5+\cdots+O(\omega_i^{16})\end{pmatrix}}{\phi'(\alpha)\begin{pmatrix}1+C_2(8C_2^3-2C_3C_2+C_4)\omega_i^4\\ -2\begin{pmatrix}27C_2^5-26C_3C_2^3+3C_4C_2^2\\ +2C_3^2C_2-C_5C_2-C_3C_4\end{pmatrix}\omega_i^5+\cdots+O(\omega_i^{16})\end{pmatrix}} \tag{24}$$

$$\frac{\phi(\xi_i)}{H_3'(\xi_i)} = (4C_2^3 - C_2C_3)\omega_i^4 + \begin{pmatrix}-27C_2^4 + 26C_3C_2^2\\ -2C_4C_2 - 2C_3^2\end{pmatrix}\omega_i^5 + \begin{pmatrix}121C_2^5 - 196C_3C_2^3 + 39C_4C_2^2\\ +54C_3^2C_2 - 3C_5C_2 - 7C_3C_4\end{pmatrix}\omega_i^6 + \cdots + O(\omega_i^{16}) \tag{25}$$

Step. 3 $o_i = \xi_i - \frac{\phi(\xi_i)}{H'(\xi_i)} = \begin{pmatrix}16C_2^7 - 8C_3C_2^5 + 4C_4C_2^4\\ +C_3^2C_2^3 - C_3C_4C_2^2\end{pmatrix}\omega_i^8 + \begin{pmatrix}-216C_2^8 + 262C_3C_2^6 - 51C_4C_2^5 - 68C_3^2C_2^4 + 8C_5C_2^4\\ +40C_3C_4C_2^3 + 4C_3^3C_2^2 - 2C_4^2C_2^2 - 2C_3C_5C_2^2 - 4C_3^2C_4C_2\end{pmatrix}\omega_i^9 + \cdots + O(\omega_i^{16})$ (26)

$$\phi(o_i) = \phi'(\alpha)\begin{pmatrix}\begin{pmatrix}16C_2^7 - 8C_3C_2^5 + 4C_4C_2^4\\ +C_3^2C_2^3 - C_3C_4C_2^2\end{pmatrix}\omega_i^8\\ +\begin{pmatrix}-216C_2^8 + 262C_3C_2^6 - 51C_4C_2^5 - 68C_3^2C_2^4 + 8C_5C_2^4\\ +40C_3C_4C_2^3 + 4C_3^3C_2^2 - 2C_4^2C_2^2 - 2C_3C_5C_2^2 - 4C_3^2C_4C_2\end{pmatrix}\omega_i^9 + \cdots + O(\omega_i^{16})\end{pmatrix} \tag{27}$$

$$j'(o_i) = \phi[x_i, \xi_i] + (\phi[v_i, x_i, \xi_i] - \phi[v_i, x_i, o_i] - \phi[\xi_i, x_i, o_i])(v_i - \xi_i) \tag{28}$$

$$j'(o_i) = \phi'(\alpha)\begin{pmatrix}1 + C_2^2(4C_2^2 - C_3)C_4\omega_i^7\\ +C_2\begin{pmatrix}32C_2^7 - 16C_3C_2^5 - 27C_4C_2^4 + 2(C_3^2 + 2C_5)C_2^3\\ +34C_3C_4C_2^2 - (2C_4^2 + C_3C_5)C_2 - 4C_3^2C_4\end{pmatrix}\omega_i^8 + \cdots + O(\omega_i^{16})\end{pmatrix} \tag{29}$$

$$\frac{\phi(o_i)}{j'(o_i)} = \frac{\phi'(\alpha)\begin{pmatrix}\begin{pmatrix}16C_2^7 - 8C_3C_2^5 + 4C_4C_2^4\\ +C_3^2C_2^3 - C_3C_4C_2^2\end{pmatrix}\omega_i^8\\ +\begin{pmatrix}-216C_2^8 + 262C_3C_2^6 - 51C_4C_2^5\\ -68C_3^2C_2^4 + 8C_5C_2^4 + 40C_3C_4C_2^3\\ +4C_3^3C_2^2 - 2C_4^2C_2^2 - 2C_3C_5C_2^2\\ -4C_3^2C_4C_2\end{pmatrix}\omega_i^9 + \cdots + O(\omega_i^{16})\end{pmatrix}}{\phi'(\alpha)\begin{pmatrix}1 + C_2^2(4C_2^2 - C_3)C_4\omega_i^7\\ +C_2\begin{pmatrix}32C_2^7 - 16C_3C_2^5 - 27C_4C_2^4\\ +2(C_3^2 + 2C_5)C_2^3 + 34C_3C_4C_2^2\\ -(2C_4^2 + C_3C_5)C_2 - 4C_3^2C_4\end{pmatrix}\omega_i^8 + \cdots + O(\omega_i^{16})\end{pmatrix}} \tag{30}$$

$$\frac{\phi(o_i)}{j'(o_i)} = C_2^2(4C_2^2 - C_3)\begin{pmatrix}4C_2^3\\ -C_3C_2 + C_4\end{pmatrix}\omega_i^8 - C_2\begin{pmatrix}216C_2^7 - 262C_3C_2^5 + 51C_4C_2^4\\ +(68C_3^2 - 8C_5)C_2^3 - 40C_3C_4C_2^2\\ +2(-2C_3^3 + C_5C_3 + C_4^2)C_2 + 4C_3^2C_4\end{pmatrix}\omega_i^9 + \cdots + O(\omega_i^{16}) \tag{31}$$

Step. 4

$$x_{i+1} = o_i - \frac{\phi(o_i)}{j'(o_i)} = C_2^4(C_3 - 4C_2^2)^2C_4(4C_2^3 - C_3C_2 + C_4)\omega_i^{15} + O(\omega_i^{16}) \tag{32}$$

Finally, the proposed scheme mentioned in (8) exhibits a convergence rate of fifteen, and the efficiency index for proposed algorithm is calculated to be 1.718771928.

4. NUMERICAL EXPRIMENTS

The problems presented below were sourced from literature [12, 13] and assessed using both the Proposed Scheme and its corresponding counterpart. Tables and graphs illustrate the Computational Order of Convergence (COC) [14], accuracy, efficiency, and consistency of the Proposed Scheme and its analogous methods. All the problems below were solved using Maple 2022 software, and line plots were generated using Origin 2021 software on my personal laptop with the following specifications: Intel® Core™ i3-4010U CPU @ 1.70 GHz and 8.00 GB RAM.

Problem 1.

$$\phi_1(x) = \sin\left(\frac{1}{x}\right) - x$$
$$\phi_2(x) = 1 + e^{x^2+x} - \cos(-x^2 + 1) + x^3$$
$$\phi_3(x) = (x-1)^2 - 1$$
$$\phi_4(x) = \sqrt{x^2 + 2x + 5} - 2\sin(x) - x^2 + 3$$
$$\phi_5(x) = 10xe^{-x^2} - 1$$
$$\phi_6(x) = \cos^2(x) - \frac{x}{5}$$

Figure 1 shows the comparison of the Computational Order of Convergence (COC) for the proposed method and its counterpart methods for Problem 1. In this figure, subfigures (1a), (1b), (1c), (1d), (1e) and (1f) display the graphical representations of $\phi_1(x), \phi_2(x), \phi_3(x), \phi_4(x), \phi_5(x)$ and $\phi_6(x)$ respectively. Above all figures are shown that proposed method converges faster as compared to counterparts.

Table 1. Value of $|x_1 - x_0|, |x_2 - x_1|, |x_3 - x_2|$ & COC of different methods of order fifteenth of problems $\phi_1(x)$ to $\phi_6(x)$.

$(\phi_n(x), x_0)$	Proposed scheme	TE1 15th	TE2 15th	TE3 15th	FS 15th
$(\phi_1(x), -0.01)$					
$\|x_1 - x_0\|$	$5.3800e-5$	$5.3800e-5$	$5.3800e-5$	$5.3800e-5$	$5.3800e-5$
$\|x_2 - x_1\|$	$1.7033e-16$	$6.3242e-16$	$2.5633e-16$	$7.2405e-16$	$2.9040e-16$
$\|x_3 - x_2\|$	$9.9051e-197$	$6.1561e-185$	$1.7937e-194$	$2.3415e-184$	$9.8668e-194$
COC	15.773	15.463	15.735	15.499	15.750
$(\phi_2(x), 0.3)$					
$\|x_1 - x_0\|$	1.2457 ...	*Diverge*	*Diverge*	*Diverge*	*Diverge*
$\|x_2 - x_1\|$	$5.4320e-2$	*Diverge*	*Diverge*	*Diverge*	*Diverge*
$\|x_3 - x_2\|$	$1.4298e-23$	*Diverge*	*Diverge*	*Diverge*	*Diverge*
COC	15.862	*Diverge*	*Diverge*	*Diverge*	*Diverge*
$(\phi_3(x), 0.1)$					
$\|x_1 - x_0\|$	1.9000 ...	*Diverge*	*Diverge*	*Diverge*	2.2048 ...
$\|x_2 - x_1\|$	$2.8218e-5$	*Diverge*	*Diverge*	*Diverge*	$3.0475e-1$
$\|x_3 - x_2\|$	$2.9192e-71$	*Diverge*	*Diverge*	*Diverge*	$3.3542e-10$
COC	13.667	*Diverge*	*Diverge*	*Diverge*	10.424
$(\phi_4(x), 1.2)$					
$\|x_1 - x_0\|$	1.1320 ...	1.1320 ...	1.1320 ...	1.1320 ...	1.1320 ...
$\|x_2 - x_1\|$	$4.4546e-14$	$1.3539e-13$	$5.0004e-14$	$4.7627e-14$	$4.3702e-14$
$\|x_3 - x_2\|$	$8.0368e-211$	$2.1381e-202$	$9.8294e-210$	$9.6175e-210$	$8.8286e-211$
COC	14.677	14.611	14.654	14.630	14.664
$(\phi_5(x), -0.1)$					
$\|x_1 - x_0\|$	$2.0103e-1$	$2.0103e-1$	$2.0103e-1$	$2.0103e-1$	$2.0103e-1$
$\|x_2 - x_1\|$	$1.3984e-16$	$1.3984e-16$	$4.8642e-16$	$1.2235e-14$	$1.7467e-16$
$\|x_3 - x_2\|$	$1.0390e-241$	$1.8213e-216$	$1.2218e-232$	$3.5682e-212$	$1.1647e-239$
COC	14.853	13.187	14.819	14.471	14.818
$(\phi_6(x), 0.1)$					
$\|x_1 - x_0\|$	2.2145 ...	*Diverge*	3.8930 ...	*Diverge*	1.4335 ...
$\|x_2 - x_1\|$	$5.6587e-3$	*Diverge*	$3.1198e-1$	*Diverge*	$4.5480e-1$
$\|x_3 - x_2\|$	$2.1410e-41$	*Diverge*	$2.0877e-14$	*Diverge*	$7.2712e-3$
COC	14.820	*Diverge*	12.019	*Diverge*	3.6026

Problem 2. The van der Waals equation see in Liu and Lee [15].

$$\phi(x) = 0.986x^3 - 5.181x^2 + 9.067x - 5.289 \tag{33}$$

Problem 3. The governs the depth of embedment equation See in Ding *et al.* [16].

$$\phi(x) = \frac{x^3 + 2.87x^2 - 4.62x - 10.28}{4.62} \tag{34}$$

Problem 4. The multipactor phenomenon See in Naseem *et al.* [17].

$$\phi(x) = x - \frac{1}{2}cos(x) + \frac{\pi}{4} \tag{35}$$

4.1. Description of Basin of Attraction

The examination of solution stability for the nonlinear function $\phi(x) = 0$, achieved through an Iterative Scheme, can be simplified using the concept of basins of attraction [20, 29]. Utilizing MATLAB R2014a, a visual depiction of all basins within the range $R = [-5 \times 5] \times [-5 \times 5]$ was created, consisting of $360{,}000$ points at a density of 600×600.

For iteration termination criteria, two conditions were set: a maximum iteration count of 10 or an error threshold of 1×10^{-10}. Each point within R initiated the iterative algorithms. If, within 10 iterations, the sequence from the iterative algorithm converged to a root x_k^* of the

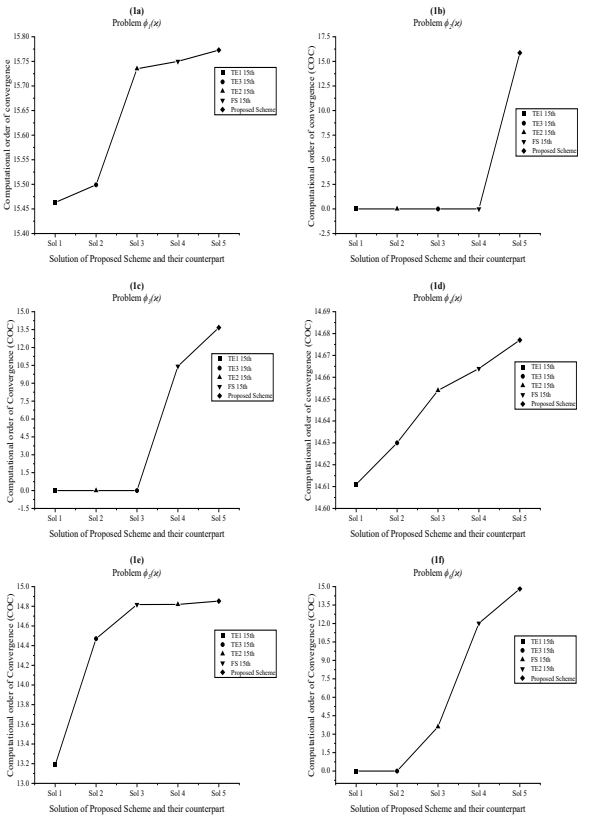

Fig. 1. COC versus solution of problem $\phi_1(x)$ to $\phi_6(x)$ by Proposed Scheme and their counterparts.

function $P_k(x)$ and met the specified tolerance, a distinct color (excluding black) was assigned to the starting point, corresponding to the convergent root. Conversely, if the iterative algorithm initiated with a point $x \in R$ exceeded the maximum iteration count of 16 without converging to any root x_k or converging to a different value p within the specified tolerance $|p - x^*| < 1 \times 10^{-10}$, it was determined that the starting point had diverged. In such instances, a black color was assigned to the starting point.

Problem 5. Below problems were taken from literature.

S. No.	Functions $(P(x))$	Roots $(x_k : k = 1, 2, 3, \ldots)$
i.	$P_1(x) = x^5 + 1$	$x_k = -1, -\frac{305}{987} \pm \frac{855}{899}i, \frac{1292}{1597} \pm \frac{4456}{7581}i$
ii.	$P_2(x) = x^3 + 1$	$x_k = 1, \frac{1 \pm \sqrt{3}i}{2}$
iii.	$P_3(x) = x^2 + 2x - \frac{1}{2}$	$x_k = \frac{-2 \pm \sqrt{6}}{2}$
iv.	$P_4(x) = x^4 + \frac{1}{64}$	$x_k = \frac{1 \pm 1i}{4}, \frac{-1 \pm 1i}{4}$
v.	$P_5(x) = x^5 - \frac{1}{2}ix^4 + \frac{1}{64}x - \frac{1}{128}i$	$x_k = \frac{1 \pm 1i}{4}, \frac{-1 \pm 1i}{4}, \frac{1}{2}i$
vi.	$P_6(x) = x^2 - \frac{1}{4}$	$x_k = \frac{1}{2}, -\frac{1}{2}$

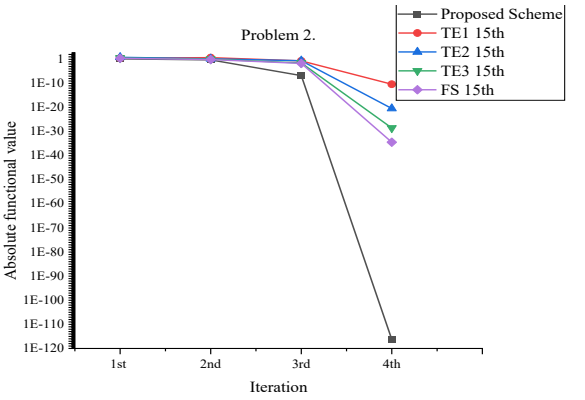

Fig. 2. Solution versus iterations of problem 2. by Proposed Scheme and its counterparts by assuming the scale $1E - 30 = 1E - 1$.

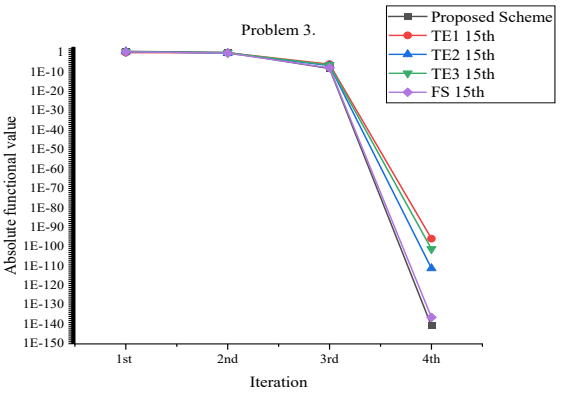

Fig. 3. Solution versus iterations of problem 3. by Proposed Scheme and its counterparts by assuming the scale $1E - 300 = 1E - 1$.

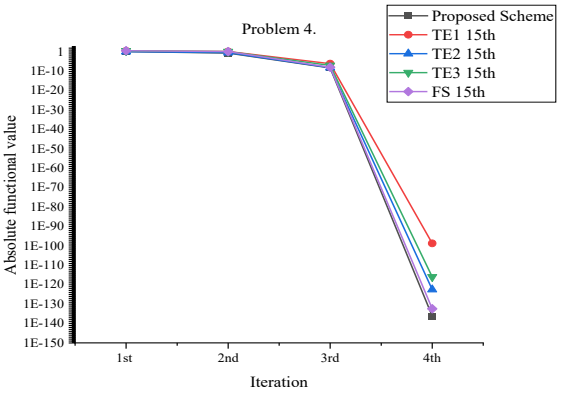

Fig. 4. Solution versus iterations of problem 4. by Proposed Scheme and its counterparts by assuming the scale $1E - 200 = 1E - 1$.

Figure 5 illustrates the basins of attraction for Problem 5, with all initial guesses within the range $R = [-5 \times 5] \times [-5 \times 5]$, as obtained by the proposed method. Subfigures (5a) and (5b) correspond to the basin of attraction for $P_1(x)$, which has five roots, with (5a) illustrating the root

Table 2. Numerical results for problem 2. for first four iterations and their absolute functional values at $\varkappa_0 = 1.5$.

Method	Root & corresponding absolute functional value	1st iteration	2nd iteration	3rd iteration	4th iteration
Proposed scheme	$\varkappa$	1.9482 ...	1.9298 ...	1.9298 ...	1.9298 ...
	$\lvert\phi(\varkappa)\rvert$	$1.7732E-3$	$1.9421E-16$	$4.1241E-222$	$7.0501E-3515$
TE1 15th	$\varkappa$	2.4548 ...	1.9373 ...	1.9298 ...	1.9298 ...
	$\lvert\phi(\varkappa)\rvert$	$3.3344E-1$	$6.8210E-4$	$1.2382E-23$	$1.8522E-319$
TE2 15th	$\varkappa$	2.1918 ...	1.9301 ...	1.9298 ...	1.9298 ...
	$\lvert\phi(\varkappa)\rvert$	$7.6603E-2$	$2.3684E-5$	$7.9528E-45$	$6.3774E-637$
TE3 15th	$\varkappa$	2.1120 ...	1.9298 ...	1.9298 ...	1.9298 ...
	$\lvert\phi(\varkappa)\rvert$	$3.9203E-2$	$1.8729E-6$	$3.1684E-61$	$8.4513E-883$
FS 15th	$\varkappa$	2.0851 ...	1.9298 ...	1.9298 ...	1.9298 ...
	$\lvert\phi(\varkappa)\rvert$	$2.9826E-2$	$1.2588E-6$	$3.2506E-67$	$1.2738E-1036$

Table 3. Numerical results for problem 3. for first four iterations and their absolute function values at $\varkappa_0 = -0.5$.

Method	Root & corresponding absolute functional value	1st iteration	2nd iteration	3rd iteration	4th iteration
Proposed scheme	$\varkappa$	−1.5417 ...	−1.5417 ...	−1.5417	−1.5417 ...
	$\lvert\phi(\varkappa)\rvert$	$5.6042E-10$	$5.6042E-166$	$4.7846E-2652$	$3.8128E-42429$
TE1 15th	$\varkappa$	−1.5417 ...	−1.5417 ...	−1.5417 ...	−1.5417 ...
	$\lvert\phi(\varkappa)\rvert$	$1.1165E-8$	$2.4491E-128$	$3.2057E-1923$	$1.8188E-28846$
TE2 15th	$\varkappa$	−1.5417 ...	−1.5417 ...	−1.5417 ...	−1.5417 ...
	$\lvert\phi(\varkappa)\rvert$	$4.2762E-10$	$1.5897E-148$	$5.6951E-2225$	$1.1695E-33371$
TE3 15th	$\varkappa$	−1.5417 ...	−1.5417 ...	−1.5417 ...	−1.5417 ...
	$\lvert\phi(\varkappa)\rvert$	$3.1795E-9$	$2.7183E-135$	$2.5914E-2026$	$1.2648E-30391$
FS 15th	$\varkappa$	−1.5417 ...	−1.5417 ...	−1.5417 ...	−1.5417 ...
	$\lvert\phi(\varkappa)\rvert$	$3.0162E-10$	$2.0774E-160$	$5.3247E-2563$	$1.8479E-41004$

Table 4. Numerical results for problem 4. for first four iterations and their absolute function values at $\varkappa_0 = 2.3$.

Method	Root & corresponding absolute functional value	1st iteration	2nd iteration	3rd iteration	4th iteration
Proposed scheme	$\varkappa$	−0.3090 ...	−0.3090 ...	−0.3090 ...	−0.3090 ...
	$\lvert\phi(\varkappa)\rvert$	$1.9167E-8$	$1.1780E-121$	$7.9496E-1820$	$2.1777E-27292$
TE1 15th	$\varkappa$	−0.3090 ...	−0.3090 ...	−0.3090 ...	−0.3090 ...
	$\lvert\phi(\varkappa)\rvert$	$4.3835E-6$	$5.0330E-88$	$3.9988E-1317$	$1.2690E-19753$
TE2 15th	$\varkappa$	−0.3090 ...	−0.3090 ...	−0.3090 ...	−0.3090 ...
	$\lvert\phi(\varkappa)\rvert$	$1.6766E-7$	$1.6490E-109$	$1.2856E-1639$	$3.0731E-24591$
TE3 15th	$\varkappa$	−0.3090 ...	−0.3090 ...	−0.3090 ...	−0.3090 ...
	$\lvert\phi(\varkappa)\rvert$	$4.0270E-7$	$5.5846E-104$	$7.5362E-1557$	$6.7531E-23350$
FS 15th	$\varkappa$	−0.3090 ...	−0.3090 ...	−0.3090 ...	−0.3090 ...
	$\lvert\phi(\varkappa)\rvert$	$4.8649E-8$	$6.7560E-118$	$9.3098E-1766$	$1.1420E-26483$

locations and (5b) depicting the number of iterations required for convergence. Subfigures (5c) and (5d) show the basin of attraction for $P_2(\varkappa)$ with three roots, where (5c) shows the root positions and (5d) represents the iteration count for each initial guess. For $P_3(\varkappa)$ with two roots, subfigures (5e) and (5f) displaying the roots and the required iterations, respectively. Subfigures (5g) and (5h) represent the basin of attraction for $P_4(\varkappa)$ with four roots, (5i) and (5j) for $P_5(\varkappa)$with five roots, and (5k) and (5l) for $P_6(\varkappa)$ with two roots. In each pair of subfigures, the left panel visualizes the location of the roots, while the right panel indicates the number of iterations required to achieve convergence to the respective root.

5. SUMMARY

To summarize, it is observed that TE1 15th diverges in problem 1 of $\phi_2(\varkappa)$, $\phi_3(\varkappa)$, and $\phi_6(\varkappa)$, TE2 15th diverges in $\phi_2(\varkappa)$ and $\phi_3(\varkappa)$, TE3 15th diverges in $\phi_2(\varkappa)$, $\phi_3(\varkappa)$and $\phi_6(\varkappa)$, while FS 15th diverges in $\phi_2(\varkappa)$, as detailed in Table 1. Both Table 1 and Figure 1 demonstrate that the Computational Order of Convergence (COC) of the Proposed Scheme exceeds that of all other methods across all test problems. Additionally, real-world problems 2-4, presented in Tables 2-4 and Figures 2-4, indicate that the Proposed Scheme achieves faster convergence compared to alternative methods. The stability analysis of the Proposed Scheme is depicted in Figure 5 through the basin of attraction.

6. CONCLUSIONS

The research paper introduces a new method for solving non-linear algebraic and transcendental equations, utilizing a weight function and interpolations. Key features include a fifteenth-order convergence rate, requiring only four function evaluations and one first derivative per iteration. Analysis of test problems reveals divergence patterns in TE1 15th, TE2 15th, TE3 15th, and FS 15th. Notably, the Proposed Scheme demonstrates superior computational Order of Convergence (COC) compared to counterparts in Table 1. Real-world problems 1-3 exhibit the proposed scheme's rapid convergence relative to other methods, and stability analysis in Figure 5 confirms its stability. Overall, the Proposed Scheme proves efficient, fast convergent, stable, and consistent, representing a significant advancement in the literature.

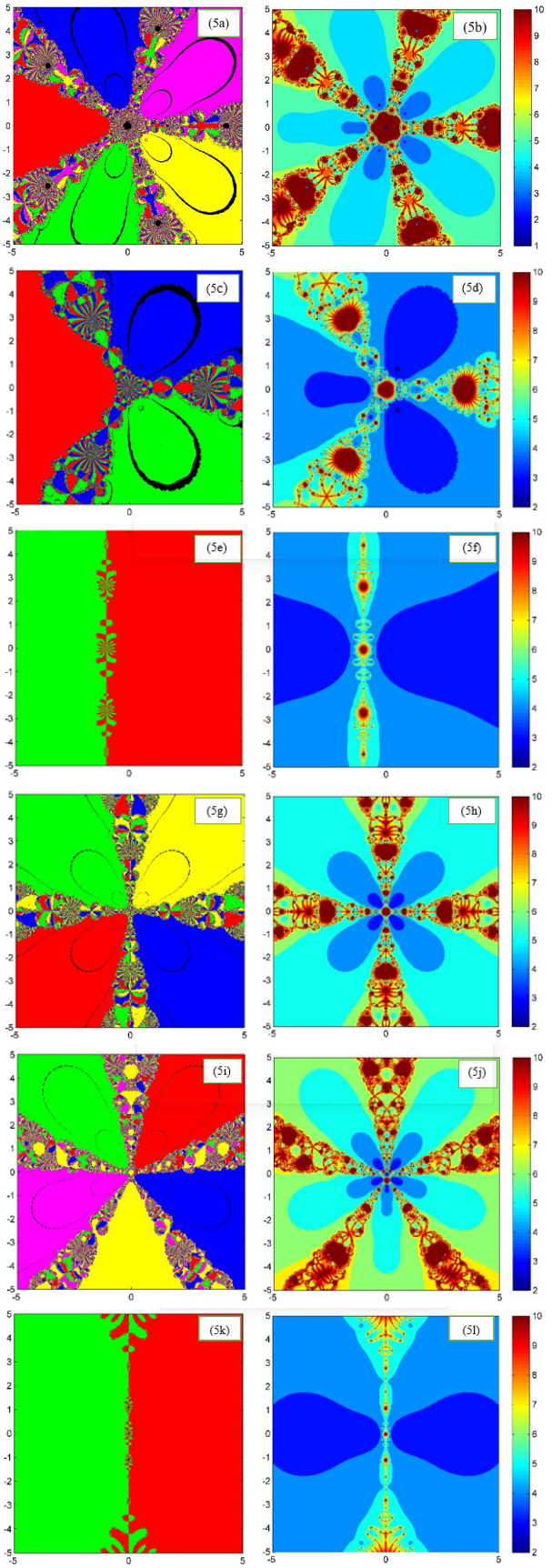

Fig. 5. Basin of attraction of $P_1(\varkappa)$ to $P_6(\varkappa)$ of problems 5 obtained by the proposed method.

7. ACKNOWLEDGEMENT

The authors express their gratitude to Inistitute of Mathematics and Computer Science, University of Sindh, Jamshoro, Sindh, for facilitating the completion of this research.

8. CONFLICT OF INTEREST

The authors declare no conflict of interest.

9. REFERENCES

1. S. Thota and P. Shanmugasundaram. On new sixth and seventh order iterative methods for solving non-linear equations using homotopy perturbation technique. *BMC Research Notes* 15(1): 1-15 (2022).
2. Z. Xiaojian. Modified Chebyshev-Halley methods free from second derivative. *Applied Mathematics and Computation* 203(2): 824-827 (2008).
3. F.A. Lakho, Z.A. Kalhoro, S. Jamali, A.W. Shaikh, and J. Guan. A three steps seventh order iterative method for solution nonlinear equation using Lagrange Interpolation technique. *VFAST Transactions on Mathematics* 12(1): 46-59 (2024).
4. A. Naseem, M.A. Rehman, and J. Younis. A New Root-Finding Algorithm for Solving Real-World Problems and Its Complex Dynamics via Computer Technology. *Complexity* 2021: 1-10 (2021).
5. Z. Abbasi, Z.A. Kalhoro, S. Jamali, A.W. Shaikh, and O.A. Rajput. A novel Approach for Real-World Problems Based on Hermite Interpolation Technique and Analysis Using Basins of Attraction. *The Sciencetech.* 5(3): 112-126 (2024).
6. M.I. Soomro, Z.A. Kalhoro, A.W. Shaikh, S. Jamali, and O. Ali. Modified Bracketing Iterative Method for Solving Nonlinear Equations. *VFAST Transactions on Mathematics* 12(1): 105-120 (2024).
7. T. Eftekhari. A new family of four-step fifteenth-order root-finding methods with high efficiency index. *Computational Methods for Differential Equations* 3(1): 51-58 (2015).
8. F. Soleymani and M. Sharifi. On a class of fifteenth-order iterative formulas for simple roots. *International Electronic Journal of Pure and Applied Mathematics* 3(3): 245-252 (2011).
9. S. Jamali, Z.A. Kalhoro, A.W. Shaikh, and M.S. Chandio. An iterative, bracketing & derivative-free method for numerical solution of non-linear equations using stirling interpolation technique. *Journal of Mechanics of Continua and Mathematical Sciences* 16(6): 13-27 (2021).
10. Z. Kovacs. Understanding convergence and stability of the Newton-Raphson method. *Teaching Mathematics and Statistics in Sciences*: 1-12 (2011). http://www.model.u-szeged.hu/data/etc/edoc/imp/ZKovacs/ZKovacs.pdf.
11. M. Kumar, A.K. Singh, and A. Srivastava. Various Newton-type iterative methods for solving nonlinear equations. *Journal of the Egyptian Mathematical Society* 21(3): 334-339 (2013).
12. S. Jamali, Z.A. Kalhoro, A.W. Shaikh, M.S. Chandio, and S. Dehraj. A novel two point optimal derivative free method for numerical solution of nonlinear algebraic, transcendental Equations and application problems using weight function. *VFAST Transactions on Mathematics* 10(2): 137-146 (2022).
13. P. Sivakumar and J. Jayaraman. Some New Higher Order Weighted Newton Methods for Solving Nonlinear Equation with Applications. *Mathematical and Computational Applications* 24(2): 1-16 (2019).
14. M. Grau-Sánchez, M. Noguera, À. Grau, and J.R. Herrero. On new computational local orders of convergence. *Applied Mathematics Letters* 25(12): 2023-2030 (2012).
15. C.S. Liu and T.L. Lee. A New Family of Fourth-Order Optimal Iterative Schemes and Remark on Kung and Traub's Conjecture. *Journal of Mathematics* 2021: 5516694 (2021).
16. H. fei Ding, Y. xin Zhang, S. fu Wang, and X. ya Yang. A note on some quadrature based three-step iterative methods for non-linear equations. *Applied Mathematics and Computation* 215(1): 53-57 (2009).
17. A. Naseem, M.A. Rehman, and T. Abdeljawad. Computational methods for non-linear equations with some real-world applications and their graphical analysis. *Intelligent Automation and Soft Computing* 30(3): 805-819 (2021).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 283-291 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)865

Pakistan Academy of Sciences

Research Article

Spatio-Temporal Monitoring and Risk Mapping of Glacial Lake Outburst Flood in Hunza Valley, Pakistan

Nausheen Mazhar[1], Tehreem Fatima[1], Muhammad Nasar-u-Minallah[2*], Asif Sajjad[3], and Sohail Abbas[4]

[1]Department of Geography, Lahore College for Women University, Lahore, Pakistan
[2]Institute of Geography, University of the Punjab, Lahore, Pakistan
[3]Department of Environmental Sciences, Quaid-e-Azam University, Islamabad
[4]College of Geography and Environmental Science: Key Research Institute of Yellow River Civilization and Sustainable Development Collaborative Innovation Center on Yellow River Civilization of Henan Province, Henan University, Kaifeng, China

Abstract: Glacial lake outburst flood (GLOF) disasters are serious and potentially increase huge risks to livelihoods and infrastructure in the mountain regions of the world. The northern highland regions of Pakistan are home to some of the biggest alpine glaciers. In this investigation, the Hunza Valley of Pakistan has undergone remote sensing-based risk assessment for Glacial Lake Outburst Floods. Borith and Passu Lakes were chosen to identify flood risk in the downstream areas. Landsat images were used from 1990-2020. Different spectral indices such as the Normalized Difference Snow Index (NDSI), Normalized Difference Glacier Index (NDGI) and Normalized Difference Water Index (NDWI) were applied to evaluate snow cover changes. Furthermore, these Lakes were digitized to evaluate any variation in Lake Areas, over the study period. Also, built-up areas close to lakes were digitized to identify total risk. The Land Surface Temperature (LST), NDSI, NDGI, NDWI, Digital Elevation Model (DEM) and Slope variables were given weights using the Analytic Hierarchy Process (AHP) method. In the analysis of flood risk mapping, maximum weight was assigned to Land Surface Temperature, and minimum weight was assigned to the slope. The result revealed that the settlements located in the Ghulkin, Gulmit, Husseini, Passu, Zarabad, and Khorramabad are at moderate risk while settlements located near Hunza River such as Karimabad, Khanna Abad, and Aliabad are at high risk. The outcome also showed that Borith Lake's area expanded, going from 0.059 km^2 in 1985 to 0.074 km^2 in 2020 and Passu Lake's area also grew, going from 0.074 km^2 in 2005 to 0.077 km^2 in 2020. In last, Buffer analysis was performed to identify areas that are likely to be affected by the flood. The result of the study can help carry out a downstream risk assessment and better preparedness for future flood hazards.

Keywords: Glacial Lake, Climate Change, Outburst Flood, Risk Mapping, Analytic Hierarchy Process, Hunza Valley.

1. INTRODUCTION

Glaciers are retreating and increasing the occurrence of glacier-related hazards. Since 1880, the world's temperature has risen by 0.85 °C [1]. The average temperature in the Indian Sub-continent can be increased by 3.5-5.5 °C by 2100 [2]. Glacier meltwater increases the river discharge but in the longer term, it reduces it [3]. The Indus irrigation system is an important part of Pakistan's economy [4], and the main source of the Indus River is glacial meltwater [5]. The world's largest glaciers can be found in Pakistan's northern mountain ranges, the Karakorum, Himalaya, and Hindu-Kush [1]. Siachen is the longest glacier outside the Polar Regions, about 75 km long located in the Karakorum Mountains. Hispar Glacier is 61 km in length [6]. Karakorum glaciers are highly influenced by the Indian monsoon. The glaciers of Karakorum have steep hill slopes and mostly are

Received: June 2024; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Muhammad Nasar-u-Minallah <nasarbhalli@gmail.com>

debris-covered [3]. Mostly precipitation occurs in spring and winter in the form of snow on the glaciers of Karakorum because they are at extremely high altitudes. The climate of Karakorum is altered by Tibetan anticyclones and westerlies which increase snowfall in winter [7]. The average annual precipitation in Baltoro Glacier is 2500 mm at 8000 m. The highest precipitation occurs in the summer monsoon in the Hunza basin [8]. Global warming caused a mass loss of 10% in high mountains of Asia which increased by almost 3% sea level [9].

Glacier advance and retreat are caused by climate change and monsoon rains raise the possibility of a lake bursting [10]. The Karakorum, Himalaya and Hindukush (HKH) region is also influenced by climate change and GLOFs came out as a looming threat in this region. Glacier area variations are a serious menace in the Karakorum region [11]. Natural disasters like the Glacial Lake Outburst Flood (GLOF) can become more severe due to climate change, endangering the population [10]. The impact of a GLOF event downstream is quite extensive in terms of damage to roads, bridges, trekking trials, villages, and agricultural lands as well as the loss of human life and other infrastructure [12]. When a glacier melts, glacial lakes typically form at the snout of the glacier. HKH is the most affected region of GLOF. Critical glacial lakes can destroy the CPEC route along the Karakorum Highway [13]. More attention is required to monitor extension in lakes created by debris-covered glaciers because of their potential burst. The glacial hazard of GLOFs can destroy the farmlands and downstream settlements [14]. Increased sea levels, flooding, erosion, and frequent GLOFs are the results of this. There is a significant spatial variation in the stable, advancing, and retreating glaciers found in the Karakoram region [15].

The glaciers of Karakorum have been retreating since 1990 but surging in glaciers has also been noticed. Climate is the most important factor of change in the glacial behaviour of Karakorum [16]. Some glaciers of Karakorum are surging such as Hassan Abad Glacier [06]. In the last 15 years, Passu Glacier has undergone a forceful retreat and increased danger in Atta Bad Lake which can burst soon [17]. Passu glacial lake experienced two outbursts in two decades destroying houses and a bridge on Karakorum Highway, debris flowed downstream consuming the structures on the way [18]. To study outbursts, there is a need to identify critical lakes to promote preventive measures [19]. In 1980, the International Karakoram Project conducted a survey along the Karakoram Highway in the Hunza Valley and identified 339 catastrophic incidents [10]. A glacial decline in Karakorum resulted in glacial lakes which have a high risk of GLOF.

Hassan Abad Hunza experienced five GLOF events by the surging of Shishper Glacier and the most recent was recorded in May 2020 [13]. When the water of the Ghizer River was obstructed and burst after a few months, it led to the creation of Khalti Lake in 1999, which caused a significant loss in villages downstream. Between 2007 and 2009, two GLOF episodes from the Ghulkin Glacier were documented, both of which shut down the Karakorum Highway [20]. In Huaraz (Peru) 4500 people died in 1941 of the GLOF event [21]. Two GLOF events occurred in Gupis Valley in 1994 and 1999 causing a huge loss of land and infrastructure [20]. Hunza River Basin has a previous record of severe flooding [22]. Land located in Gulmit is used as agricultural land and the maximum glacial retreat is recorded during ablation season. The glacier area of Gulmit (in Hunza Nagar) is about 11-kilometre squares [2]. The glaciers Ghulkin and Gulmit are situated south of Batura Glacier, both are debris-covered and the primary source of water to the Hunza Nagar district and Hunza River. Glacial lakes are formed by the meltwater of glaciers.

The volume of water in lakes increases due to an increase in glacier melting, and lakes that abruptly burst might damage towns downstream. This event is referred to as the Glacial Lake Outburst Flood (GLOF). In the past few years, many glacial lakes have developed in Pakistan's HKH region. Gilgit Baltistan has experienced many GLOF events. GLOF can take the lives of humans and livestock and cause damage to roads, bridges, and farms land. However, global warming has had a substantial impact on Pakistan's glaciers [23]. The first decade of the twenty-first century is regarded as the planet's hottest decade. Lake located at the terminus of Passu Glacier had records of outbursts in the last two decades causing a huge loss to the downstream settlements in Hunza [22]. Five GLOF events occurred in Hunza Basin between 2007-2008 and posed a severe threat

to near settlements of the Karakorum Mountain Range [4]. According to Amin *et al*. [20], Sosot village experienced a GLOF event in 1994 and two GLOF events were recorded from Ghulkin Glacier. In the past 200 years, the Karakorum Range has seen 35 outburst episodes. Due to the lack of mitigation, management, and government policies, a huge loss of land and infrastructure and because of poor livelihood, people are not able to respond quickly and effectively to any flood situation [13]. Therefore, there's a need to constantly monitor the glacial lakes and the factors affecting them. The study's primary goal is to locate regions that may be at risk of GLOF flooding downstream.

2. MATERIALS AND METHODS

2.1. Study Area

The study area includes Gilgit Baltistan, as well as Hunza Nagar District and Upper Hunza, a sizable, glaciated region in North Pakistan. The Hunza lies between 36.3167° N and 74.6500° E at an elevation of 2,438 meters. The Hunza Valley contains the Passu and Borith lakes that have been selected. These lakes were chosen to identify potential risk for the study areas as shown in Figure 1. The elevation of Borith Lake is 2600 meters above sea level. It lies about 2 km to the north of Gulmit. A sizable body of water known as a lake is located beneath the settlement of Hussaini. South of the tongue of the Batura Glacier, close to the Passu Glacier, there is a lake called Passu Lake.

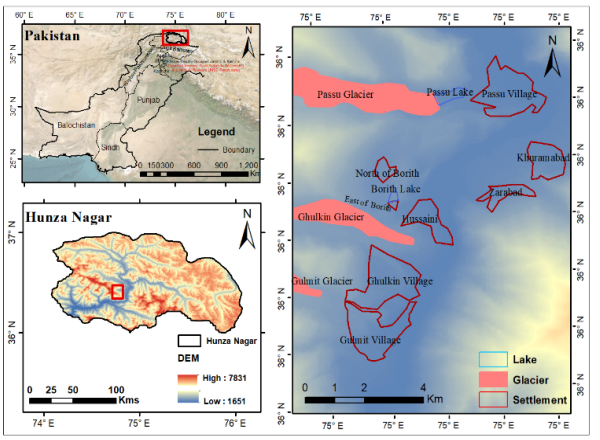

Fig. 1. Location of the study area.

2.2. Data and Its Sources

For this investigation, the Digital Elevation Model (DEM) and multi-temporal Landsat datasets were utilized. Since Landsat satellites have provided data since 1972, therefore various Landsat series datasets were used. For this investigation, the cloud-free Landsat images for April, September, and December were gathered. Landsat (TM/ETM+/ OLI) images in total 12 were obtained from the US Geological Survey (USGS) as illustrated in Table 1. The Study area has been covered in total four Landsat satellite tiles with paths and rows 149-34, 149-35, 150-34, and 150-35. In specific, Landsat images for the years 1993, 2001, 2010 and 2020 were acquired for comprehensive analysis.

2.3. Data Analysis

The spatial overlay technique was applied to the resulting layers of Normalized Difference Water Index (NDWI), Normalized Difference Glacier Index (NDGI), Normalized Difference Snow Index (NDSI), Land Surface (LST), and slope and flow accumulation (Table 2 and Figure 2). Based on this analysis, the risk areas were calculated for settlements around these selected lakes and surrounding glaciers.

For risk area identification, different indicators were studied and while performing weighted overlay analysis different weights were assigned to variables by using the Analytical Hierarchy Process (AHP) technique [24]. Maximum weight was assigned to Land Surface Temperature which was about 38% because it is the driving force behind the glacial behaviour, snow cover and expansion rate of lakes.

Table 1. Details of used landsat data.

Satellite and sensors	Bands used	Spatial resolution (m)	Month	Year
Landsat 5-TM	2,3,4,5,6	30	Nov, Dec	1993
Landsat 7-ETM+	2,3,4,5,6	30	Nov	2001
Landsat 5-TM	2,3,4,5,6	30	Nov, Dec	2010
Landsat 8-OLI/TIRS	3,4,5,6,10	30	April	2020

Table 2. Source and utility of spectral indices used in the study.

Spectral indices	Utility
Normalized Difference Snow Index (NDSI)	Maps that show where there is snow and where there isn't [28]
Normalized Difference Glacier Index (NDGI)	Identifying and mapping the snow-ice and ice-mixed debris class [29]
Normalized Difference Water Index (NDWI)	Mapping surface water [30]

2.3.1. Calculation of TOA (Top of Atmospheric) spectral radiance

The thermal infrared band is converted to the Top of Atmospheric (TOA) spectral radiance through rescaling factors as mentioned in the metadata file of the Landsat satellite and measured through the equation below:

$$L_\lambda = \left(\frac{LMAX\lambda - LMIN\lambda}{QCALMAX - QCALMIN}\right) - (QCAL - QCALMIN) + LMIN\lambda$$

2.3.2. Land surface temperature (LST)

In the final step thermal infrared data can be obtained from spectral radiance to measure land surface temperature [25] to degree Celsius (°C) by utilizing the equation given below:

$$LST = (K2/(In(K1/L) + 1)) - 273.15$$

2.3.3. Normalized difference snow index (NDSI)

Snow is recognized using NDSI [26]. For mapping the snow extent, NDSI has been widely employed. The following formula was used for calculating NDSI.

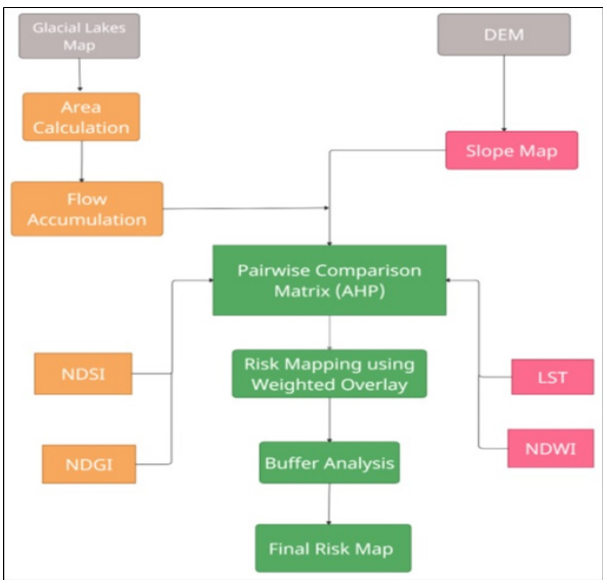

Fig. 2. Methodological framework.

$$NDSI = \frac{Green\ Band\ Reflectance - SWIR\ Band\ Reflectance}{Green\ Band\ Reflectance + SWIR\ Band\ Reflectance}$$

For Landsat 5 & 7: $NDSI = \frac{B2-B5}{B2+B5}$

For Landsat 8: $NDSI = \frac{B3-B6}{B3+B6}$

2.3.4. Normalized difference water index (NDWI)

NDWI is a method that was created primarily to designate open water features, improving their visibility while removing soil and vegetation elements [27]. The following formula is applied for calculating NDWI.

$$NDWI = \frac{Green-NIR}{Green+NIR}$$

Where NIR is near-infrared band reflectance. The same formula was used with the different band number for Landsat 5, 7 and 8.

For Landsat 5-TM: $NDWI = \frac{b2-b4}{b2+b4}$

For Landsat 7-ETM+: $NDWI = \frac{b4-b5}{b4+b5}$

For Landsat 8 (OLI): $NDWI = \frac{b3-b5}{b3+b5}$

2.3.5. Normalized difference glacier index (NDGI)

A numerical indicator called the NDGI uses the green and red spectral bands to assist find and track glacial ice. The primary remote sensing uses of NDGI are for the detection and monitoring of glaciers. The following formula was applied for calculating NDGI.

$$NDGI = \frac{Green\ band - Red\ band}{Green\ band + Red\ band}$$

For landsat 5 & 7: $NDGI = \frac{b2-b3}{b2+b3}$

For landsat 8 (OLI): $NDGI = \frac{b3-b4}{b3+b4}$

2.4. Analytical Hierarchy Process (AHP)

When one or more alternatives are compared to one or more pertinent criteria, a decision is made. The criteria is weighed according to their relative value to prioritize some of these criteria above others. To manage complex choice issue constellations, risk management and decision guidelines have been developed and put into use recently. These guidelines place a strong emphasis on risk communication and stakeholder involvement. One of the most effective and well-known methods for evaluating weight is the analytical hierarchy process (AHP), which was developed by Saaty [24]. Several natural hazard studies have employed the AHP approach, which estimates the eigenvalues of the components matrix and enables the evaluation of the judgments' consistency [31]. It has already been described as one of the most promising strategies for weight improvement.

In this study, through a review of the literature, some variables affecting GLOF events as referred to in Table 3, were shortlisted. These variables encompassed the glacial ice, snow cover, debris cover and climatic domains.These variables were assigned weights according to the pairwise comparison matrix suggested by Saaty [24]. Weights were assigned to gauge the impact of these variables on GLOF events by using the AHP technique. The AHP uses math and psychology to organize and analyze complicated decisions. It helps in assigning weights to indicators [12]. The weights for this study were finalized after taking expert guidance from field experts i.e. Pakistan Meteorological Department (PMD) and the Surface Water Hydrology Project (SWHP). Maximum weight was given to LST almost 38% and the minimum weight was given to slope almost 6% depending upon the variable which triggers most of the GLOF event.

3. RESULTS AND DISCUSSION

GLOF risk identification involves early and continuous identification of flood events that if they occur will have severe impacts on physical infrastructure and human settlements. For risk area identification different analyses were performed to identify at-risk settlements in the study area, also considering site factors such as climate.

3.1. Expansion in Area of Lakes

The lake's rate of growth is crucial since it raises the amount and severity of flooding that could occur. Using Google Earth (GE) images, lake boundaries were digitally recorded over two years. Figure 3 shows the boundary of Borith and Passu Lake in two different years. The change in the boundary of lakes can be easily detected. The result shows that the Borith Lake area has increased from 1985 to 2020 and the Passu Lake has also increased from 2005 to 2020. The Passu Lake was identified as high potential GLOF by the Pakistan Meteorological Department in 2015. It is because the melting of glaciers brought on by the recent increase in global temperatures led to the expansion of lakes [32].

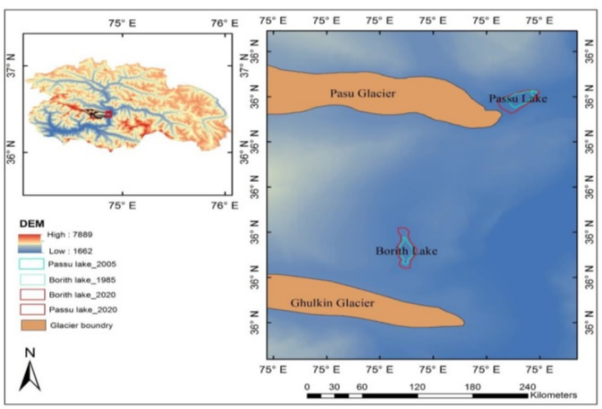

Fig. 3. Expansion in area of lakes.

Table 3. Pairwise comparison matrix of GLOF variables.

Indicators	LST	NDSI	NDGI	NDWI	Slope	SUM	n^{th} root	Priority vector (PV)	%
LST	1.00	2.00	3.00	4.00	5.00	15.00	0.68	0.39	38.76
NDSI	0.50	1.00	2.00	3.00	4.00	10.50	0.48	0.27	27.13
NDGI	0.33	0.50	1.00	2.00	3.00	6.83	0.31	0.18	17.66
NDWI	0.25	0.33	0.50	1.00	2.00	4.08	0.19	0.11	10.55
Slope	0.20	0.25	0.33	0.50	1.00	2.28	0.10	0.06	5.90
Sum	2.28	4.08	6.83	10.50	15.00	38.70	1.76	1.00	100
Sum PV	0.885	1.108	1.207	1.108	0.885	5.192	0.24	0.13	

Figure 4 shows the graphical representation of lake area changes. In 2005 the area of Passu Lake was 0.0742 sq km which increased in 2020 and is about 0.0778 sq km. The same is true for Borith Lake, whose area expanded from 0.0595 sq km in 1985 to 0.0742 sq km in 2020.

3.2. Flow Accumulation

Figure 5 shows the high value of flow accumulation across the selected lakes of Hunza Valley. Furthermore, the result shows that some tributaries of the Hunza River are passing through some villages namely, Hussaini, Ghulkin, Gulmit and Passu. As a result, these villages are at high risk if any GLOF flood occurs.

3.3. GLOF Risk Mapping

Figure 6 shows the risk mapping applied to identify the potential of GLOF flood risks in the selected study area, the weighted overlay analysis was performed to extract the risk areas. The result shows that almost all the settlements across selected villages are in the moderate risk zone. But the settlements along the river Hunza valleys are at high risk.

3.4. Buffer Zone Identification

To identify risk zones around glaciers, a buffer of 10 km was applied around the Passu, Ghulkin and Gulmit glaciers. Khadka *et al.* [12] used the same

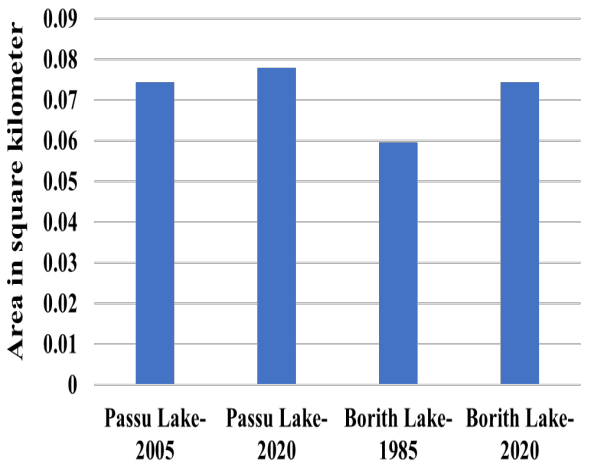

Fig. 4. Increase in area of lakes.

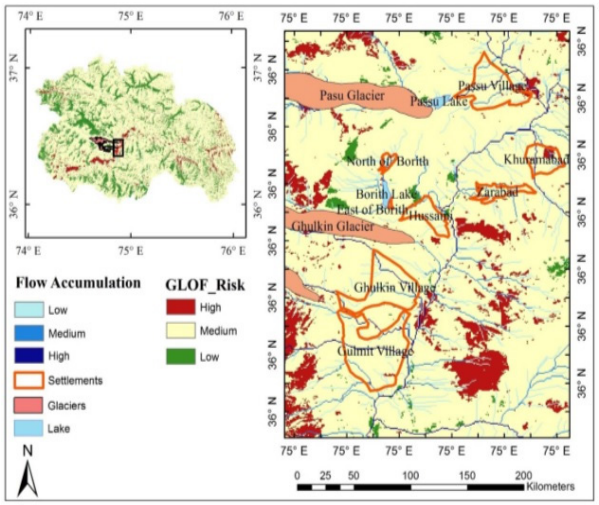

Fig. 6. GLOF risk map of Hunza.

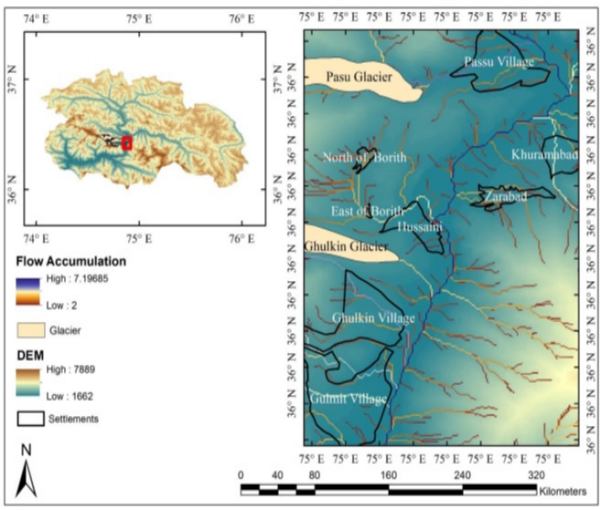

Fig. 5. Flow accumulation of Hunza.

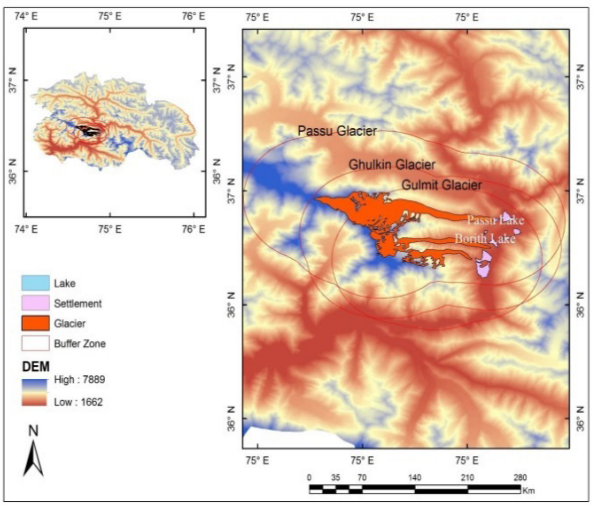

Fig. 7. Buffer zone identification.

distance to identify zones around glaciers in the Mahalangur Himalaya region. Figure 7 shows that the selected lakes and settlements under the buffer zone have extreme flood vulnerability that can be very destructive if any GLOF occurs.

4. CONCLUSIONS

In the present study, GLOF risk was calculated using datasets from 1990-2020 in the selected lakes. Different weights were assigned to different variables to calculate risk. The Pakistan Meteorological Department (PMD) and Surface Water Hydrology Project (SWHP) were consulted as an expert in this regard to assist us in weighting the variables. AHP is essential in this study as it allows for systematic weighting of various risk factors, leading to a more precise and informed GLOF risk evaluation and has been used in similar research [12, 31]. LST was given the most weight because it is the primary factor contributing to the rise in temperature. The study concludes that the settlements along river valleys are at high risk and settlements away from the river are at low risk. The Areas of selected lakes also increased over the past decade. The outcome also showed that Borith Lake's area expanded, going from 0.059 km^2 in 1985 to 0.074 km^2 in 2020. Passu Lake's area also grew, going from 0.074 km^2 in 2005 to 0.077 km^2 in 2020. Further studies have revealed that the volume and area of glacial lakes particularly Borith and Passu are shown to be increasing signalling a growing risk of GLOF events [33-35]. This increase in areas has the potential to be a significant threat to the GLOF and the settlements and other properties around selected lakes. The extremely vulnerable locations are found close to lakes and rivers, such as some communities along the Hunza River downstream of Passu Lake. The likelihood of Passu Lake bursting is low due to its natural drainage, but the projected losses in the event of GLOF are enormous. Hunza's Shimshal Valley has seen flooding from glacier lakes in the past. GLOF episodes have been documented in the Hunza valley, hence the glaciers of Hunza Nagar valley need to be watched for their lakes. 16000 people of Karimabad, 2005 people of Aliabad, 5000 people of Gulmit and people of surrounding settlements are at high risk if a GLOF event occurs. The study has taken LST, NDSI, NDGI, NDWI, and slope as the key parameters triggering the GLOF events in the major cryosphere areas, as the change in these variables due to climate changes, is making the region even more vulnerable. This study has provided a mechanism and application for the accurate risk mapping, damage assessment, and monitoring of glacier lakes and GLOFs. The results of this study will help build early warning systems in sensitive areas and reduce the negative effects of future GLOF events in the Hunza watershed. Future GLOF studies should investigate the long-term impacts of climate change on glacial lake dynamics and develop sustainable land-use practices to reduce vulnerability in high-risk areas.

5. CONFLICT OF INTEREST

The authors declare no conflict of interest.

6. REFERENCES

1. M. Shafique, B. Faiz., A.S. Bacha, and S. Ullah. Evaluating glacier dynamics using temporal remote sensing images: a case study of Hunza Valley, northern Pakistan. *Environmental Earth Sciences* 77(162): 162 (2018).
2. S.R. Bajracharya, P.K. Mool, and B.R. Shrestha. Global climate change and melting of Himalayan Glaciers. In: Melting glaciers and rising sea levels: Impacts and Implications. P.S. Ranade (Ed.). *The Icfai's University Press, India* pp. 28-46 (2008).
3. D. Scherler, B. Bookhagen, and M.R. Strecker. Spatially variable response of Himalayan glaciers to climate change affected by debris cover. *Nature Geoscience* 4(3): 156-159 (2011).
4. M. Akhtar, N. Ahmad, and M.J. Booij. The impact of climate change on the water resources of the Hindukush–Karakorum–Himalaya region under different glacier coverage scenarios. *Journal of Hydrology* 355(1-4): 148-163 (2008).
5. Z. Abid and S. Zia. Mapping and Analysis of Baltoro Glacier and Shigar River. *Pakistan Geographical Review* 74: 20-32 (2019).
6. P.A. Mayewski and P.A. Jeschke. Himalayan and Trans-Himalayan glacier fluctuations since AD 1812. *Arctic and Alpine Research* 11(3): 267-287 (1979).
7. D. Farinotti, W.W. Immerzeel, R.J. de Kok, D.J. Quincey, and A. Dehecq. Manifestations and mechanisms of the Karakoram Glacier anomaly. *Nature Geoscience* 13(1): 8-16 (2020).
8. W.W. Immerzeel, F. Pellicciotti, and A.B. Shrestha. Glaciers as a proxy to quantify the spatial distribution of precipitation in the Hunza basin. *Mountain Research and Development* 32(1): 30-38 (2012).

9. K. Wu, S. Liu, Z. Jiang, Y. Zhu, F. Xie, Y. Gao, and S. Muhammad. Surging Dynamics of Glaciers in the Hunza Valley under an Equilibrium Mass State since 1990. *Remote Sensing* 12(18): 2922 (2020).
10. A. Ashraf, R. Naz, and R. Roohi. Glacial lake outburst flood hazards in Hindukush, Karakoram, and Himalayan Ranges of Pakistan: implications and risk analysis. *Geomatics, Natural Hazards, and Risk* 3(2): 113-132 (2012).
11. F. Begum. Risk assessment of Shishper Glacier, Hassan Abad Hunza, north Pakistan. *Journal of Himalayan Earth Sciences* 52(1): 1-11 (2019).
12. N. Khadka, X. Chen, Y. Nie, S. Thakuri, G. Zheng, and G. Zhang. Evaluation of Glacial Lake Outburst Flood Susceptibility using multi-criteria assessment framework in Mahalangur Himalaya. *Frontiers in Earth Science* 8: 601288 (2021).
13. G. Khan, S. Ali, X. Xiangke, J.A. Qureshi, M. Ali, and I. Karim. Expansion of Shishper Glacier Lake and recent glacier lake outburst flood (GLOF), Gilgit-Baltistan, Pakistan. *Environmental Science and Pollution Research* 1(9): 20290-20298 (2021).
14. N. Gilany and J. Iqbal. Geospatial analysis and simulation of glacial lake outburst flood hazard in Shyok Basin of Pakistan. *Environmental Earth Sciences 79*(6): 1-11 (2020).
15. F. Muneeb, S.U. Baig, J.A. Khan, and M.F. Khokhar. Inventory and GLOF susceptibility of glacial lakes in Hunza River Basin, Western Karakorum. *Remote Sensing* 13(9): 1794 (2021).
16. K. Hewitt (Ed.). Karakoram glaciers and climate change. In: Glaciers of the Karakoram Himalaya: Glacial Environments, Processes, Hazards and Resources. *Springer, Dordrecht Heidelberg New York* pp. 291-326 (2014).
17. N. Mazhar, A.I. Mirza, S. Zia, Z.S. Butt, M.G. Shahid, and A.I. Mirza. An analysis of glacial retreat and resultant vegetation expansion in the Karakorum: a case study of Passu glacier in Hunza valley. *Biologia (Pakistan)* 64(1): 135-145 (2018).
18. G. Rasul, Q. Chaudhry, A. Mahmood, K. Hyder, and Q. Dahe. Glaciers and glacial lakes under changing climate in Pakistan. *Pakistan Journal of Meteorology* 8(15): 14-24 (2011).
19. D. Yongjian and L. Jingshi, Glacier Lake outburst flood disasters in China. *Annals of Glaciology* 16: 180-184 (1992).
20. M. Amin, D. Bano, S.S. Hassan, M.A. Goheer, A.A. Khan, M.R. Khan, and S.M. Hina. Mapping and monitoring of glacier lake outburst floods using geospatial modelling approach for Darkut valley, Pakistan. *Meteorological Applications* 27(1): 1877-1888 (2020).
21. D.R. Gurung, N.R. Khanal, S.R. Bajracharya, K. Tsering, S. Joshi, P. Tshering, L.K. Chhetri, Y. Lotay, and T. Penjor. Lemthang Tsho glacial Lake outburst flood (GLOF) in Bhutan: cause and impact. *Geoenvironmental Disasters* 4(1): 17 (2017).
22. A.Z. Zaidi, Z. Yasmeen, and M.D. Siddiqui. Glacial lake outburst flood (GLOF) Risk mapping in Hunza River Basin (Pakistan) using geospatial techniques. *Paper presented at the 6th International Conference on Recent Advances in Space Technologies (RAST), IEEE, 12th June 2013* pp. 191-195 (2013).
23. F. Zhang, C. Zeng, R.R. Pant, G. Wang, H. Zhang, and D. Chen. Meltwater hydrochemistry at four glacial catchments in the headwater of the Indus River. *Environmental Science and Pollution Research* 26(23): 23645-23660. (2019).
24. R.W. Saaty. The analytic hierarchy process-what it is and how it is used. *Mathematical Modelling* 9(3-5): 161-176 (1987).
25. M. Nasar-u-Minallah. Retrieval of land surface temperature of Lahore through Landsat-8 TIRS data. *International Journal of Economic and Environmental Geology* 10(1): 70-77 (2019).
26. V.V. Salomonson and I. Appel. Estimating fractional snow cover from MODIS using the normalized difference snow index. *Remote Sensing of Environment,* 89(3): 351-360 (2004).
27. S.K. McFeeters. The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features. *International Journal of Remote Sensing, 17*(7): 1425-1432 (1996).
28. N. Mazhar, D. Amjad, K. Javid, R. Siddiqui, M. A. Nawaz, and Z. S. Butt. Mapping Fluctuations of Hispar Glacier, Karakoram, using Normalized Difference Snow Index (NDSI) and Normalized Difference Principal Component Snow Index (NDSPCSI). *International Journal of Economic and Environmental Geology* 11(4): 48-55 (2020).
29. A. Keshri, A. Shukla, and R. Gupta. ASTER ratio indices for supraglacial terrain mapping. *International Journal of Remote Sensing* 30(2): 519-524 (2009).
30. A. Shukla and I. Ali. A hierarchical knowledge-based classification for glacier terrain mapping: a case study from Kolahoi Glacier, Kashmir Himalaya. *Annals of Glaciology* 57(71): 1-10 (2016).
31. L. Mohanty and S. Maiti. Probability of glacial lake outburst flooding in the Himalayas. *Resources, Environment and Sustainability* 5: 100031 (2021).
32. M. Shafique, B. Faiz, and A. Bacha. Evaluating glacier dynamics using temporal remote sensing

images: a case study of Hunza valley, northern Pakistan. The International Archives of the Photogrammetry, *Remote Sensing and Spatial Information Sciences* 42: 1781-5 (2019).

33. A. Nisar, S. Mahmood, and M. Shakoor. Spatio-temporal analysis of snow cover change in Hunza Valley, Karakoram Region. *Advanced Remote Sensing* 3(2): 58-68 (2023).
34. S. Mustafa, F. Rehman, A.S. Rana, A. Masood, T. Azeem, and J. Siddique. A Comparative Analysis of Selected Glaciers of Hunza Basin, Pakistan since 1972–2018: Varied Responses to Climate Change. *Doklady Earth Sciences* 514(2): 360-373 (2024).
35. M.F. Moazzam, J. Bae, and B.G. Lee. Impact of Climate Change on Spatio-Temporal Distribution of Glaciers in Western Karakoram Region since 1990: A Case Study of Central Karakoram National Park. *Water* 14(19): 2968 (2022).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 293-302 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)866

Pakistan Academy of Sciences

Research Article

Numerical Study on Failure Mechanism of Rock Slope Formed by Mudstone at Girdu, Pakistan

Zulkifl Ahmed[1], Sumra Yousuf[2*], Muhammad Rizwan[3], Muhammad Yousaf Raza Taseer[2], Muhammad Qasim Sultan[1], Mahwish Zahra[4], and Anum Aleha[4]

[1]Department of Civil Technology, Mir Chakar Khan Rind University of Technology, DG Khan, Pakistan

[2] Department of Building and Architectural Engineering, Faculty of Engineering and Technology, Bahauddin Zakariya University, Multan, Pakistan

[3]Department of Metallurgical Engineering, NED University of Engineering and Technology, Karachi, Pakistan

[4]Department of Architecture Design, National Fertilizer Corporation Institute of Engineering and Technology, Multan, Pakistan

Abstract: Failure analysis of rock slope is a core topic in rock mechanics and geotechnical engineering. This paper simulates the failure mechanism of a rock slope formed by mudstone at Girdu, Pakistan. Initially, site surveys and laboratory experiments were carried out to measure slope geometry and input shear strength properties of the material. The geotechnical structure and model analyzing computer program (GeoSMA^{3D}) was used to determine the stability of the main slippery blocks. A shear strength reduction method was selected in a three-dimensional distinct element code (3DEC) to examine the failure mechanism, the factor of safety (*FS*), shear strain concentration, and displacement of slope at various monitoring points. Furthermore, the effect of the rock slippery block, slope angle, and slope height on failure modes of slope were assessed. The present results showed that the studied section is prone to big planer failure at both maximum and minimum strength properties. Slope angle and height significantly influenced the failure characteristics of rock slope . It is inferred that these parameters must be precisely considered during slope reinforcement design.

Keywords: Rock Slope, Stability Analysis, Factor of Safety, GeoSMA^{3D}, 3DEC.

1. INTRODUCTION

Landslides are one of the most common geological natural hazards. Every year, many disastrous landslides occur all over the world, causing loss of property and life [1, 2]. This causes significant fatalities, economic losses injuries, and property damages. In the past decades, many methodologies have been proposed to reduce the risk of harm caused by rock failure. For highway planners slope failure hazards are the major concern for vehicle transportation and economic opinion [3, 4]. Slope failure mechanisms mostly depend on material strength, geological conditions, climatic conditions, hydrological conditions, slope geometry, and discontinuity characterization [5, 6]. Slope failure includes lateral spread, erosion, toppling, wedge sliding, creep, rotational landslides, tension cracking, translational landslides, squeezing, rock-fall, and sliding after loss of toe support [7].

Slope failure is a type of shear failure that causes detachment of unsupported rock material. Sometimes, rock failure can be naturally under the influence of gravity, like freeze-thaw cycles slide [8], or seismic exertion [9]. Therefore, the motion

Received: June 2024; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Sumra Yousuf <sumra.yousafrm@gmail.com>

of rock fall is mostly dependent on the volume of the sliding surface, its geometry, the shape of the sliding blocks, slope inclination angle, and surface irregularities [10]. Sometimes slope failure occurs due to the effect of heavy machine movement and blasting [11]. However, the displacement of rounded-shaped blocks was more than compared to square-shaped blocks [12]. The height of the slope, its dip angle, and the volume of the sliding surface categorize the motions of the rock slope. In previous studies, four types of rock motion have been discussed, such as sliding, bouncing, free fall, and rolling [13]. The simulation of the rock failure mechanism is a complex process that plays an important role in understanding rock-fall hazards. During rock slope stability analysis, estimating safety factors is a difficult and important task. For this purpose, a great effort has been made to develop a suitable computer code that can directly and efficiently simulate slope stability factor and displacement. In previous studies, Wang *et al*. [14] established GeoSMA^{3D} software to judge the stability of rock blocks. Liu *et al*. [15] applied the UDEC simulator to study the effect of blasting on jointed rock slopes. Sarkar *et al*. [16] assessed slope stability and landslide hazard by finite element analysis. Singh *et al*. [17] selected finite element computer code to simulate the stability of road-cut cliffs in basaltic rock mass. Zhao *et al*. [18] used unmanned aerial vehicle (UAV) photogrammetry to investigate the stability of rock cliff faces and multistep rock slopes. Kou *et al*. [19] used the article hydrodynamics method to study the progressive failure process of jointed rock slope. Gautam and Mehndiratta [20] applied FLAC^{2D} to analyze the rock slope stability considering an infilled planer. These studies computed the stability of rock considering joints ignoring key slippery blocks. Also, no one simulated the failure mechanism of rock slopes formed by mudstone.

In the present study, geotechnical structure and model analyzing (GeoSMA^{3D}) software was first applied to simulate the stability of rock slippery blocks. Then, 3DEC was used to simulate the stability of a rock slope formed by mudstone at station point 5 + 800, Girdu, Pakistan. Finally, the effect of several parameters on the slope failure profile was studied using 3DEC software.

2. STUDY AREA AND GEOLOGY

Girdu Hill Station is located in the Suleiman Mountains near Fort Munro, Pakistan. The Suleiman Mountains are a mountain range in Pakistan, located in the central part of the country as shown in Figure 1. These are the southern part of the Hindu Kush Mountains and extend into the Baluchistan, Khyber Pakhtunkhwa, and Punjab Provinces. The Girdu hilly area separates the Punjab province from the Baluchistan province. The eastern face of the Girdu hills dip steeply to the Indus River and its altitude is 1800 meters. The National Highway N70 transects the Girdu Hill Station, making it a pivotal route for travelers and goods transport. It is 85 (Km) away toward the west of the City. The National Highway N70 connects the Punjab province to the Baluchistan province. The geological formation of the rock is mudstone. Physical analysis of rock slope shows three dominant discontinuity sets J_1, J_2 and J_3 dipping at different angles, as shown in Figure 2. Discontinuity sets have incessant lengths in the range of several meters. The discontinuity J_1 dips towards the road and J_1, and J_2 dips in the direction of the hill. The orientation of the discontinuities was judged during the field investigation as presented in Table 1. The true dip and dip direction of discontinuity were measured by a geological compass. The height and the length of the slope are 20 (m) and 500 (m), respectively, which is measured with a total station.

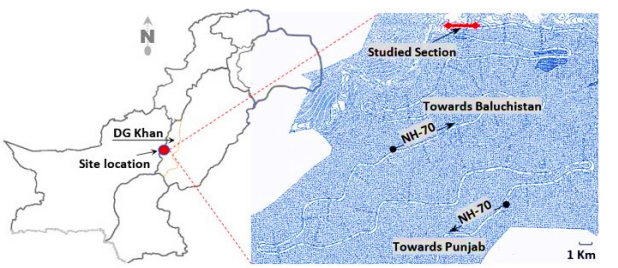

Fig. 1. Site location map (a) and road alignment in red color (b).

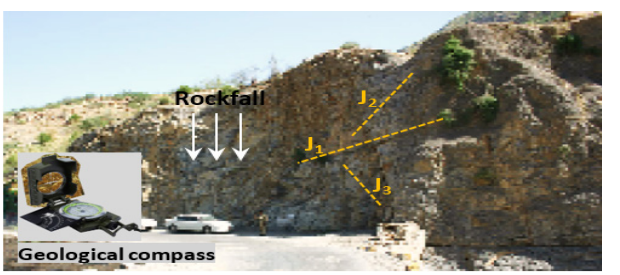

Fig. 2. Topographical, joint distribution, and rockfall at the station (5 + 800).

Table 1. Discontinuity orientation at the study area.

Discontinuity	True dip (°)	Dip direction (°)
JI	41-7	302-3
J2	66-2	279-9
J3	91-1	173-4
Flaws	13-5	069-2

3. METHODOLOGY

3.1. Laboratory Experiments

A uniaxial test is performed to estimate the mechanical properties of mudstone at various axial stresses (σ). Figure 3 shows the mudstone's results and deformation modes. Density, cohesion (c), and internal friction (ϕ) were determined by the uniaxial test (Table 2). The normal and shear stiffness of joints were estimated from the rock mass modulus, intact rock modulus, and joint spacing, assuming that the deformability of rock layers is due to the deformability of the intact rock and the deformability of the joints.

3.2. Identification of Slippery Blocks

During the field survey, two types of failure surfaces of the slippery blocks were noticed (Figure 4). The first type was formed due to the intersection of the slope surface with the slope shoulder, and the other type was formed by the slope face and discontinuity intersection. The dip angles of these surfaces are θ_1 and θ_2, respectively. The coordinates of the slope shoulder are (x_1, y_1, z_1) and (x_2, y_2, z_2). Several blocks are formed on the slope face due to the presence of joints. The presence of the rock blocks on the face of a slope increases the instability resulting in slope failure [20]. Joint surfaces were traversed with the slope shoulder. The slope coordinates are governed by the following Equation (1).

$$\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1} = \frac{z - z_1}{z_2 - z_1} \quad (1)$$

The radius of the slope is r, the center of coordinates is x_0, y_0, z_0 and the dip direction is α. So, with the help of a normal vector (l, m, n) the equation of the joint plane can be expressed in Equation (2) as:

$$l(x - x_0) + m(y - y_0) + n(z - z_0) = 0 \quad (2)$$

Equation (3) will be tested if there is a factual resolution from Equations (1) and (2) as:

$$\begin{cases} x_1 \le x \le x_2 \\ y_1 \le y \le y_2 \\ z_1 \le z \le z_2 \\ (x - x_0)^2 + (y - y_0)^2 + (z - z_0)^2 \le r^2 \end{cases} \quad (3)$$

If the attained factual result satisfies Equation (3), this means that the slope shoulder and joint plane intersect with each other. So, both Equations (4) and (5) can be applied to judge the joint plane i and j respectively, as:

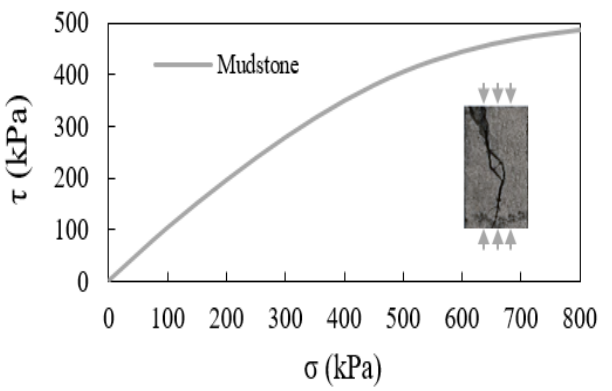

Fig. 3. Experimental results and deformation of mudstone sample.

Table 2. Mechanical properties of cut slope material maximum and minimum values.

Material	Density (g/cm³)	c (MPa)	ϕ (°)	Kn (GPa/m)	Ks (GPa/m)
Rock	2.03	3.54	32	25.27	6.1
Mudstone	1.61	0.08	22.9	22.67	5.73

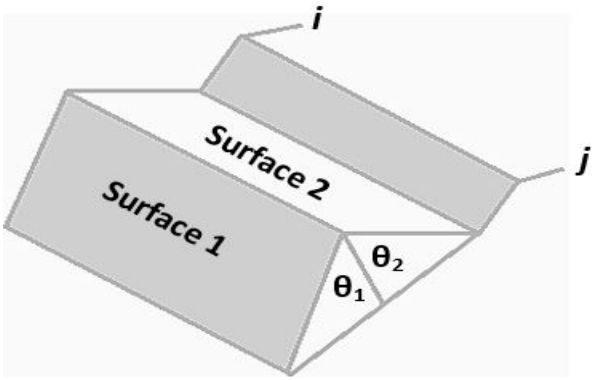

Fig. 4. Sliding model showing the failure mechanism for double failure surface.

$$Aix + Biy + Ciz + Di = 0 \quad (4)$$

$$Ajx + Bjy + Cjz + Dj = 0 \quad (5)$$

For the top of the cut slope, the plane equation is:

$$Ax + By + Cz + D = 0 \quad (6)$$

If $\Delta = \begin{vmatrix} Ai & Bi & Ci \\ Aj & Bj & Cj \\ A & B & C \end{vmatrix} = 0$, then Equations (4), (5), and (6) have no factual solutions, and discontinuity planes *i* and *j* seem on the crest of the cut slope. In contrast, the factual solution is explained as:

$$x = -\frac{\Delta x}{\Delta},\ y = -\frac{\Delta y}{\Delta},\ z = -\frac{\Delta z}{\Delta} \quad (7)$$

$$\Delta x = \begin{vmatrix} Di & Bi & Ci \\ Dj & Bj & Cj \\ D & B & C \end{vmatrix}\ \Delta y = \begin{vmatrix} Ai & Di & Ci \\ Aj & Dj & Cj \\ A & D & C \end{vmatrix} \text{ and } \Delta = \begin{vmatrix} Ai & Bi & Di \\ Aj & Bj & Dj \\ A & B & D \end{vmatrix} \quad (8)$$

Then, within the joint planes *i* and *j*, Equation (9) has been used to evaluate the corresponding joint planes.

$$\begin{cases} (xi - x)^2 + (yi - y)^2 + (zi - z)^2 \le ri^2 \\ (xj - x)^2 + (yj - y)^2 + (zj - z)^2 \le rj^2 \end{cases} \quad (9)$$

Where x_i, y_i, and z_i are coordinates, x_j, y_j, *and* z_j are midpoints of discontinuity planes, and r_i and r_j are the radii. Results can lead to the ultimate judgment of obtainable slippery blocks.

Equation (5) satisfies Equation (6) and Equation (7) based on discontinuity planes, coordinates, and radius of the failure surface. This solution leads to the final judgment of sliding surfaces, sliding blocks, and the corresponding radius of the slip circle of the slope [21].

3.3. Slippery Block Stability Calculation

After getting the coordinates, dip direction (α), and dip angle (β) of the sliding plane, the cosine of the inclination (θ) is calculated from the spatial geometry of the sliding surface. In the sliding mechanism, two sliding surfaces detached from the parent rock at the same time, when the coordinates of two points *A* and *C* are (x_a, y_a, z_a) and (x_c, y_c, z_c), respectively. The angle between the two surfaces was θ_1 and θ_2, which satisfies Equation (8). So, the *FS* of the double sliding model can be elaborated as Equation (9). The energy equation of the model can be shown as Equation (10).

$$\cos\theta = \frac{n}{\sqrt{l^2+m^2+n^2}} \quad (10)$$

Where $l = \sin\alpha\sin\beta, m = \sin\alpha\cos\beta, n = \cos\alpha$, β is the dip of structure and α is the inclination of structure. A block moves smoothly across the surface, and its shape is illustrated in Figure 5. In this model, the *FS* of the block can be calculated by Equation (11) as:

$$FS = \frac{N1tan\emptyset1+N2tan\emptyset2+c1S1+c2S2}{GSin\theta} \quad (11)$$

Where, ϕ_1, ϕ_2 and c_1, c_2 are friction and cohesion of the cut slope material, respectively, and θ is the dip of the intersection of the slipping plane S_1 and S_2. In Equation (11) N_1, N_2 is the normal force of the slipping planes and fulfills Equations (12) and (13). Where θ_1 and θ_2 are the angles between the upper normal of the plane.

$$\sin\theta = \frac{z_2 - z_1}{\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2 + (z_1 - z_2)^2}} \quad (12)$$

$$N = G\cos\theta \quad N_1 = \frac{G\cos\theta\sin\theta_1}{\sin(\theta_1+\theta_2)} \quad N_2 = \frac{G\cos\theta\sin\theta_2}{\sin(\theta_1+\theta_2)} \quad (13)$$

Finally, the factor of safety for each slippery block can be calculated by using Equation (14).

$$f_{si} = \frac{N_1\tan\varphi_1 + N_2\tan\varphi_2 + c_1S_1 + c_2S_2}{G\sin\theta} \quad (14)$$

In Equation (14) f_{si} is the safety factor of the block. ϕ_1, ϕ_2, and c_1, c_2 are friction and cohesion of the cut slope material, respectively, and θ is the dip of the intersection of the slipping plane S_1 and S_2. N_1, N_2 is the normal force of the slipping planes. θ_1 and θ_2 are

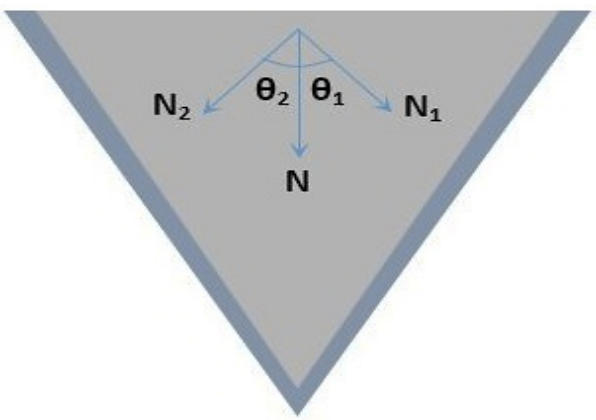

Fig. 5. Double sliding surface model.

the angles between the upper normal of the plane and *G* is the gravity of the block.

4. SLIPPERY BLOCKS CLASSIFICATION

4.1. GeoSMA^{3D} Computer Code

Wang *et al*. [14] developed a 3D numerical slope stability analysis computer program (GeoSMA^{3D}) for identifying rock sliding blocks. This program can simulate a discontinuity network, the number of slippery blocks, its sliding planes, the volume of blocks, and the corresponding *FS* [21]. The flow chart in Figure 6 shows the scheme of analysis and the development of the program.

4.2. Sliding Blocks Stability Simulation

Rock block theory and a three-dimensional numerical slope stability analysis computer program GeoSMA^{3D} are used in the current study to simulate the *FS* of each slippery block. The main analysis procedures include three-dimensional discontinuity network simulation of slope, closed block identification, removable and slippery block identification, *FS* calculation, joint plane simulation, and volume and sliding faces of blocks determination through this software [10, 21, 22].

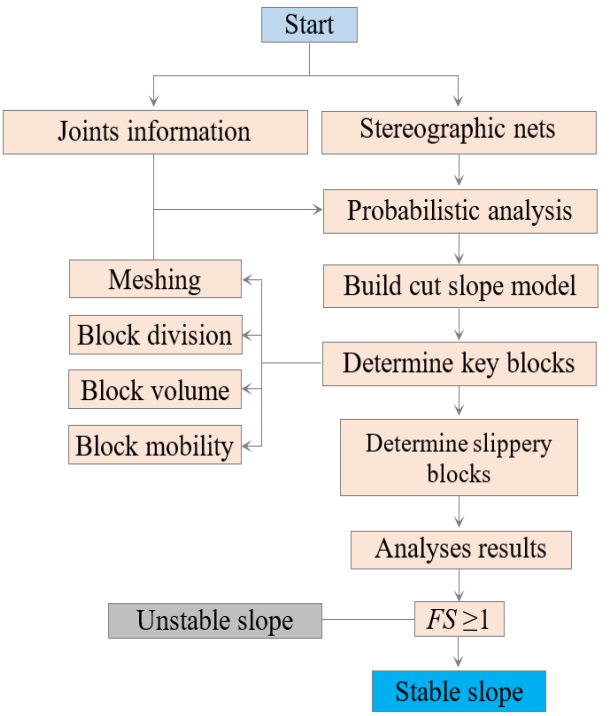

Fig. 6. Flowchart scheme showing the stability analysis procedure in GeoSMA^{3D}, after Wang and Ni [21].

The slope was modeled with a length of 60 m, width of 80 m, height of 40 m, and crest of 20 m in GeoSMA^{3D} software as shown in Figure 7. The dip angle was determined as 60°. Then the value of internal friction angle and cohesion were taken from Table 2.

The method presented by Wang and Ni [21] is selected within the framework of GeoSMA^{3D} to compute the stability of each slippery block and the results are shown in Figure 8. Joints are indicated by the trace lines. The pattern of the slippery blocks yields the shapes of all wedge blocks. Most of the slippery blocks were tetrahedrons or hexahedrons. The distribution, shape, and position of all the slippery blocks of the rock slope. The safety factor, volume of slippery blocks, and the number of sliding faces of each slippery block were determined. The slippery block whose *FS* is smaller than one harms the stability of the whole slope.

5. STABILITY SIMULATION

5.1. Models Parameters and Failure Criterion

According to slope geometry, a possible model is created in 3DEC software (Figure 9). The material strength parameters are assigned according to Table 2. Mohr-Coulomb yield criterion is selected to model the material behavior. The stability of the whole slope is simulated via the shear strength reduction method. In this paper. The rock slope model is divided into hexahedral blocks composed of hexahedrons 4 m in size. Then this slope model is imported into the three-dimensional distinct element code (3DEC) as a numerical model. The number of blocks and apexes in the numerical model were

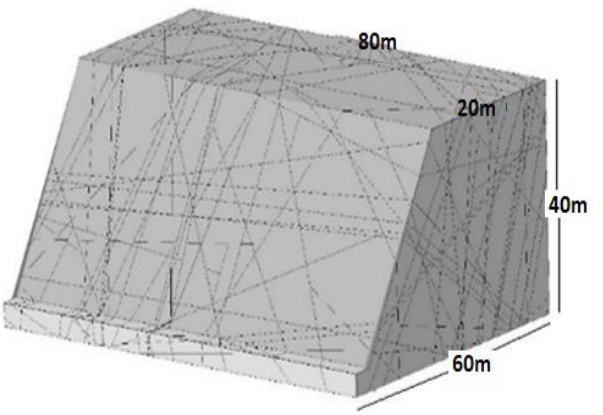

Fig. 7. Slope model with joint planes.

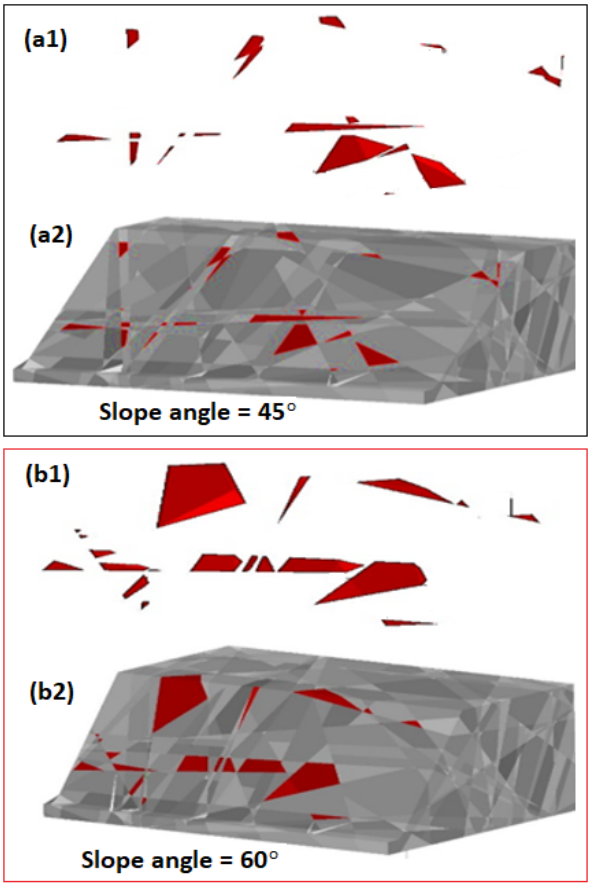

Fig. 8. All Slippery blocks in red color. Shapes of slippery blocks (a1, b1) and location of slippery blocks (a2, b2).

18389 and 112464, respectively. Each hexahedron is assumed as a rigid block in the numerical model. The input properties for the jointed rock slope and discontinuities are presented in Tables 1 and 2. It should be noted that the parameters for the virtual joints formed by the interfaces between blocks need to be carefully determined based on the lithology of the rock mass. In Table 1, γ is the unit weight, c is cohesion, ϕ is the internal friction angle, G is the shear modulus, B is the bulk modulus, k_n is the normal stiffness of the discontinuity, and K_s is shear stiffness of the discontinuity.

3DEC uses the shear-strength reduction method (SRM) to perform the stability analysis of geotechnical structures. Several simulations were executed by increasing the trial value of reduction factor (f) until slope failure. The safety factor (FS), at failure, equals the trial value of f [23]. In the strength reduction technique, the φ and c are reduced at different trial values of SRF until cut slope failure occurs as:

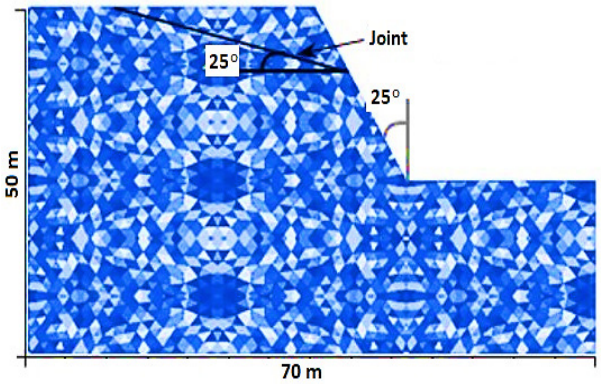

Fig. 9. Geometry of slope and joint position.

$$Cf = c/f \tag{15}$$

$$\varphi f = arctan(\frac{tan\varphi}{f}) \tag{16}$$

Where c_f is reduced cohesion and φ_f reduced internal friction angle.

The failure of the slope has two obvious characteristics in the 3DEC modeling process: (i) there is no slippery block, and most of the blocks on the entire slope are nearly static; although a few blocks are displaced, these blocks are not moved out of the boundary of the model, as shown in Figure 10(a). (ii) A small number of blocks will slip out of the model boundary, which also triggers the movement of some other blocks. Some blocks will become unstable and eventually induce large-scale sliding (Figure 10b).

6. RESULTS AND DISCUSSION

In the current study, rock slope stability was simulated through numerical modeling. Rock block theory within the network of GeoSMA^{3D} software was used to identify slippery blocks and the stability of each slippery block. The relationship between the number of slippery blocks and the corresponding FS of each slippery block was assessed. It has been observed that most of the slippery blocks have a value of FS less than one, and only three blocks have an FS of more than 1.1; these three blocks have distinct key blocks [24]. For that reason, this can be decided that the proper dressing of the rock slope is favorable to improve the stability along the road.

To study the cut slope stability, two different slope angles of 45° and 60° were selected, because

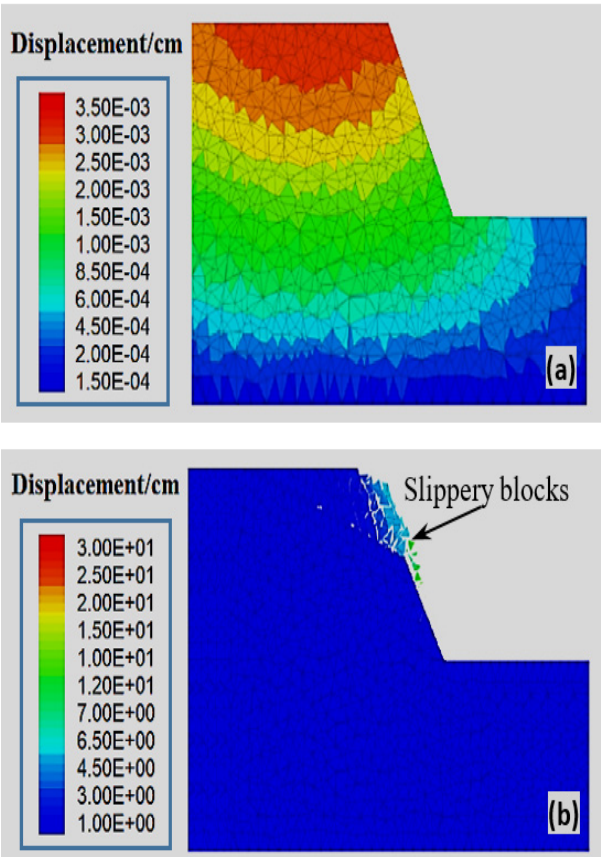

Fig. 10. (a) Zero slippery block, and (b) few slippery blocks.

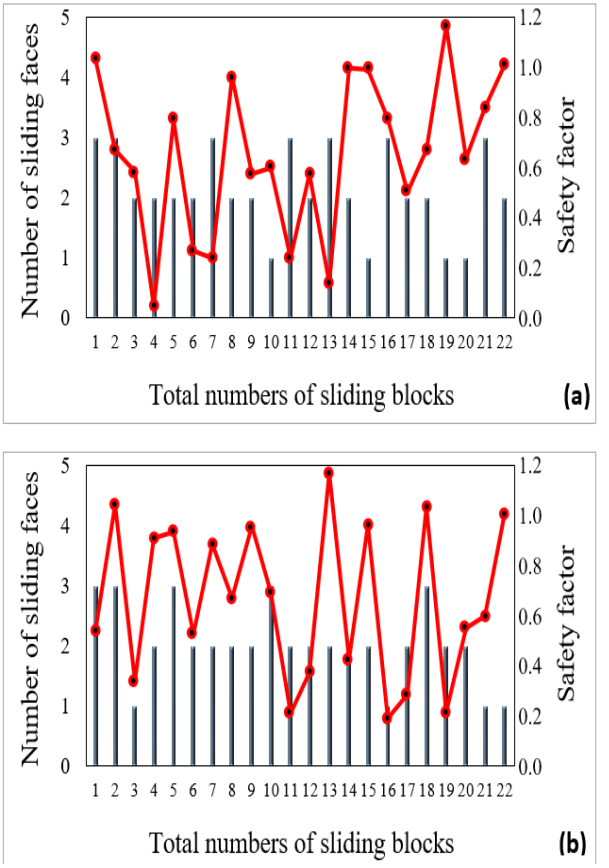

Fig. 11. Relationship diagram between block sliding faces and corresponding *FS*. (a) The slope angle is 45°, and (b) the slope angle is 60°.

the slope angle ranges from 45° to 60°. Figure 11 shows a schematic diagram of the slippery blocks searched at slope angles of 45° and 60°. The volume of block and *FS* was largely affected by the slope angle [25]. The safety factor decreased as the number of block sliding faces increased. This can be seen that as the slope angle becomes larger, the number of slippery blocks maintains a certain range, but the overall sliding surfaces of the block and the volume of each block have increased on a large scale [26]. Meanwhile, the safety factors of block number 1, 19, and 22 are larger than 1.1, when the slope dip is considered as 45° (Figure 11a). In addition, the *FS* of block numbers 2, 13, and 18 are more than 1.1 when the slope dip is considered as 60° (Figure 11b). These blocks could improve the stability and reinforcement measurement quality of rock slopes.

The concentration of shear stress in different zones of slope is examined, as shown in Figure 12. The maximum concentration of shear stress of 5.945×10^3 MPa is observed at the upper part of the

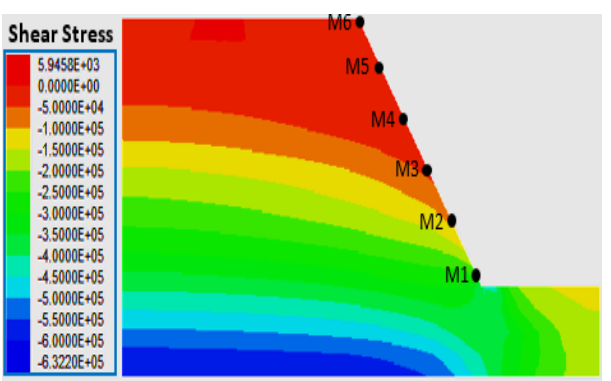

Fig. 12. Contour of shear stress and six various monitoring points.

slope. At the slope toe, the maximum shear stress of -3.00×10^5 is recorded. At six monitoring points, the displacement history of the slope is studied via 3DEC considering two different slope angles and heights. The *x*-displacements were rising in monitoring locations near to slope crest as shown in Figure 13. When the slope angle was 45°, the maximum *x*-displacement of 14 cm at monitoring location M6 was recorded.

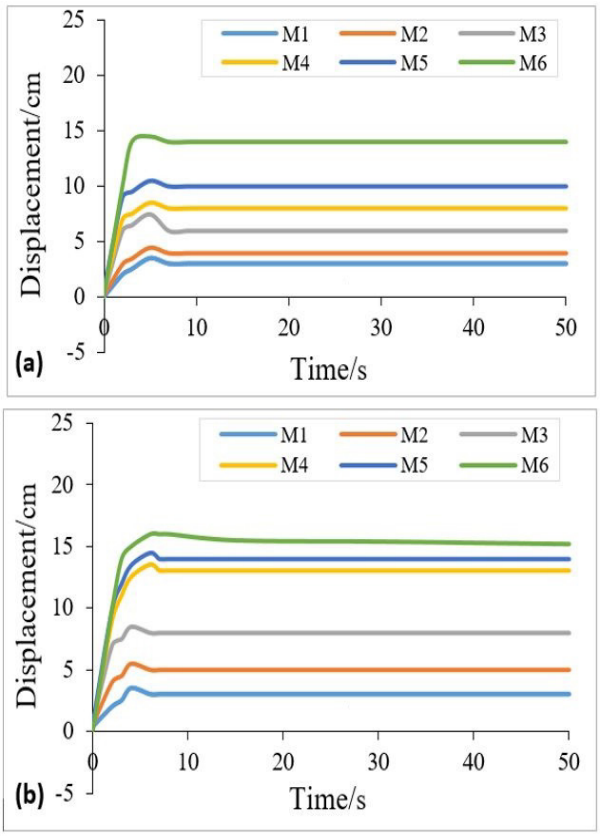

Fig. 13. Displacement history plot at various monitoring points. (a) The slope angle is 45°, and (b) the slope angle is 60°.

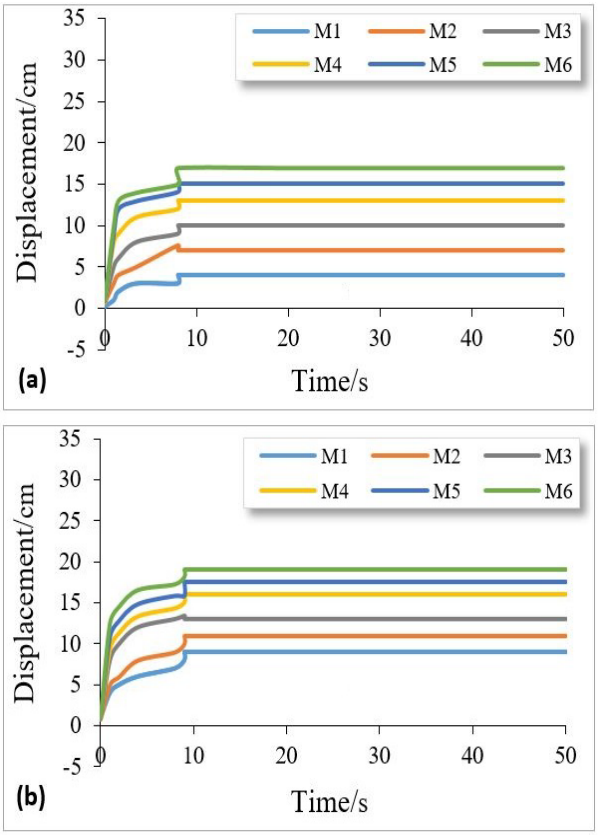

Fig. 14. Displacement history plot at various monitoring points. (a) Slope height is 40 m, and (b) slope height is 60 m.

The maximum *x*-displacement of 16 cm at monitoring location M6 was observed when the slope angle was 60°. It means, that as the slope angle rose the *x*-displacement also rose significantly. This is because of a higher angle, which produces more gravity resulting in a higher value of displacement. At monitoring locations M1, M2, and M3, the *x*-displacement was lower than that of the monitoring locations M4, M5, and M6. The higher value of displacement around the crest can cause big planner failure of the slope. So, the upper part of the slope needs more attention during slope reinforcement design.

Considering different heights of slope, the *x*-displacement history plot is presented in Figure 14. Slope height has a remarkable effect on slope. At slope heights of 40 m and 60 m, the maximum x-displacement of 16.4 cm and 21 cm was recorded, respectively. The horizontal displacement of the slope first suddenly increased at a time of 10, and after that, it maintained the same value. Almost similar observations were noted in all cases. The magnitude of displacement in the first case was higher as compared to the second case. Based on numerical investigations, strain concentration, and displacement history plots indicate that the mud rock slope is most likely to fail by the big planner at the top.

7. CONCLUSIONS

In this research, GeoSMA^{3D} and 3DEC software were used to analyze the stability of rock slope form by mudstone. From the numerical simulations, the following conclusions are drawn:

1. Rock block theory is adopted within the framework of GeoSMA^{3D} software to search key slippery blocks. Block volume, block sliding faces, and corresponding *FS* were determined by the numerical tool. Results suggest that most of the blocks have *FS* less than one.
2. The progressive failure of the cut slope was examined by the strength reduction method within the system of 3DEC. This was computed from the shear strain and displacement contour

plots at different slope angles and heights. Increasing the slope angle and height increase in *x*-displacement is recorded. Numerical outcomes showed that a big planner failure can take place at the upper portion of the slope.

3. Numerical results suggest that slope angle and height are important factors that affect the failure modes of slope. Results also indicate that the slope is unstable. Therefore, it needs to take proper slope reinforcement measures.

This study only considered static loads, so in future research dynamic loads and rainfall intensity must be taken into account for better understanding.

8. ACKNOWLEDGEMENTS

This work was conducted with the support of Mir Chakar Khan Rind University of Technology, DG Khan; Bahauddin Zakariya University, Multan; the National Highway Authority (NHA) of Pakistan; and Fort Munro Development Authority, DG Khan.

9. CONFLICT OF INTEREST

The authors declare no conflict of interest.

10. REFERENCES

1. S. Wang, Z. Ahmed, and P. Wang. Study of critical failure surface influencing factors for loose rock slope. *SN Applied Sciences* 3: 65 (2021).
2. H. Wang, J. Zou, X. Wang, P. Lv, Z. Tan, L. Cheng, W. Qiang, Q. Binli, and G. Zhengchao. Analysis of deformation mechanism of rainfall-induced landslide in the Three Gorges Reservoir Area: Piansongshu landslide. *Scientific Report* 14: 10005 (2024).
3. L.R. Alejano, B. Pons, F.G. Bastante, E. Alonso, and H.W. Stockhausen. Slope geometry design as a means for controlling rockfalls in quarries. *International Journal of Rock Mechanics and Mining Sciences* 44: 903-921 (2007)
4. Y. Zheng, R. Wu, C. Yan, R. Wang, and B. Ma. Numerical study on flexural toppling failure of rock slopes using the finite discrete element method. *Bulletin of Engineering Geology and the Environment* 83: 111 (2024).
5. Z. Chuhan, O. Pekau, J. Feng, and W. Guanglun. Application of distinct element method in dynamic analysis of high rock slopes and blocky structures. *Soil Dynamics and Earthquake Engineering* 16: 385-94 (1997).
6. R. Bhasin and A. Kaynia. Static and dynamic simulation of a 700-m high rock slope in western Norway. *Engineering Geology* 71: 213-26 (2004).
7. L.M. Highland and P. Bobrowsky. The landslide handbook: A guide to understanding landslides. Circular 1325. *US Geological Survey, Reston, Virginia* (2008). https://pubs.usgs.gov/circ/1325/pdf/C1325_508.pdf
8. N. Matsuoka and H. Sakai. Rockfall activity from an alpine cliff during thawing periods. *Geomorphology* 28: 309-328 (1999)
9. V. Solonenko. Landslides and collapses in seismic zones and their prediction. *Bulletin of the International Association of Engineering Geology* 15: 4-8 (1977).
10. S. Wang, Z. Ahmed, M.Z. Hashmi, and W. Pengyu. Cliff face rock slope stability analysis based on unmanned aerial vehicle (UAV) photogrammetry. *Geomechanics and Geophysics for Geo-Energy and Geo-Resources* 5(4): 333-344 (2019).
11. F. Agliardi and G. Crosta. High-resolution three-dimensional numerical modeling of rockfalls. *International Journal of Rock Mechanics and Mining Sciences* 40: 455-471 (2003).
12. S. Vijayakumar, T. Yacoub, and J. Curran. On the effect of rock size and shape in rockfall analyses. *Proceedings of the US Rock Mechanics Symposium (ARMA) San Francisco CA, USA* (2011).
13. D. Varnes. Slope movement types and processes. *Special Report* 176: 11-33 (1978).
14. S. Wang, P. Ni, and M. Guo. Spatial characterization of joint planes and stability analysis of tunnel blocks. *Tunnelling and Underground Space Technology* 38: 357-67 (2013).
15. Y. Liu, H. Li, J. Zhao, J. Li, and Q. Zhou. UDEC simulation for the dynamic response of a rock slope subject to explosions. *International Journal of Rock Mechanics and Mining Sciences* 41: 474 (2004).
16. S. Sarkar, K. Pandit, N. Dahiya, and P. Chandna. Quantified landslide hazard assessment based on finite element slope stability analysis for Uttarkashi–Gangnani Highway in Indian Himalayas. *Natural Hazards* 106: 1895-1914 (2021).
17. P. Singh, A. Wasnik, A. Kainthola, M. Sazid, and T. Singh. The stability of road cut cliff face along SH-121: a case study. *Natural Hazards* 68: 497-507 (2013).
18. L. Zhao, C. Jin, B. Zhao, D. Huang, Z. Zhu, and S. Zuo. A method for setting up passive protective nets for rockfalls based on unmanned aerial vehicle (UAV) photogrammetry. *Bulletin of Engineering Geology and the Environment* 83: 340 (2024).

19. H. Kou, Z. Shi, C. Xia C, Y. Zhou, and S. Meng. Dynamic simulation and failure analysis of intermittently jointed rock cells and slopes based on a novel spring-based smoothed particle hydrodynamics method. *Bulletin of Engineering Geology and the Environment* 83: 135 (2024).
20. B. Gautam and S. Mehndiratta. Analysis of a rock slope with an infilled planar joint using deterministic and probabilistic approaches. *Indian Geotechnical Journal* inpress (2024). https://link.springer.com/article/10.1007/s40098-024-00925-6
21. S. Wang and P. Ni. Application of block theory modeling on spatial block topological identification to rock slope stability analysis. *International Journal of Computational Methods* 11: 1350044 (2014).
22. F. Wang, S. Wang, M.Z. Hashmi, and Z. Xiu. The characterization of rock slope stability using key blocks within the framework of GeoSMA-3D. *Bulletin of Engineering Geology and the Environment* 77: 1405-1420 (2018).
23. J. Xiao, W. Gong, J.R. Martin, M. Shen, and Z. Luo. Probabilistic seismic stability analysis of slope at a given site in a specified exposure time. *Engineering Geology* 212: 53-62 (2016).
24. S. Wang, Z. Zhang, C. Wang, C. Zhu, and Y. Ren. Multistep rocky slope stability analysis based on unmanned aerial vehicle photogrammetry. *Environmental Earth Sciences* 78: 260 (2019).
25. Q.H. Zhang and G.H. Shi. Discussion on Key Issues in the Application of Block Theory in Rock Engineering. *Rock Mechanics and Rock Engineering* 57: 2017-2033 (2024).
26. S. Dong, Q. Zhang, Z. Mai, and H. Zhang. A Limit Equilibrium Method for Analyzing Multi-sliding-Plane Block Stability and Its Application in the Optimal Design of a Gravity Dam Foundation. *Rock Mechanics and Rock Engineering* 57: 4107-4128 (2024).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 303-310 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)679

Pakistan Academy of Sciences

Research Article

Green Synthesis of Trimetallic Oxides (CuO-ZnO-MnO) Nanoparticles using *Ocimum basilicum* Aqueous Leaves Extract: Characterization and Antibacterial Activity

Fouzia Majeed, Syeda Shaista Gillani*, and Iram Bashir

Department of Chemistry, Lahore Garrison University, DHA Phase-VI, Lahore 54792, Pakistan

Abstract: Metal oxides nanoparticles (NPs) are becoming more and more significant day by day in a variety of fields, due to their unique physical, biological and chemical properties that have made them popular. In the current study, we developed a green chemistry method that is easy to use, cost-effective, safe for the environment, and produces trimetallic oxide (CuO-ZnO- MnO) nanoparticles using an aqueous leaves extract of *Ocimum basilicum*. The sequential reduction of zinc chloride, manganese chloride, and copper chloride solutions resulted in the production of trimetallic oxide (ZnO-MnO-CuO) nanoparticles within 5 minutes at 35 °C, indicating a faster reaction rate than the chemical techniques used before for such synthesis.Trimetallic oxide nanoparticles (NPs) were charecterized by FTIR, SEM-EDX, XRD, and UV-visible spectropic techniques. A characteristic Surface Plasmon Resonance (SPR) band to confrm trimetallic oxide nanoparticles (NPs) synthesis was observed at ~450 nm of λ light by UV–Visible spectrophotometer. There were noticeable peaks in the FTIR spectra which confirmed the presence of variuos functional groups in the trimetallic oxide NPs. SEM results showed the shape and polydispersive nature of the NPs, the existence of Mn, Cu, Zn, and O atoms was confirmed by EDX. The findings of the XRD results also confirmed the synthesis of nanoparticles. Synthesized trimetallic oxide (CuO-ZnO-MnO) nanoparticles were screened for antibacterial activity against *Escherichia coli* (Gram-negative) and *Staphylococcus aureus* (Gram-positive). The agar well diffusion assay revealed that trimetallic oxide (CuO-ZnO-MnO) nanoparticles have the highest efficacy against *Staphylococcus aureus*. As a result, trimetallic oxide nanoparticles may be effective antibacterial agents in the pharmaceutical sector.

Keywords: Trimetallic Oxide Nanoparticles, Green Synthesis, Structural Characterization, Antibacterial Activity.

1. INTRODUCTION

Nanoscience is a research endeavor aimed at gaining full control over atomic and molecular surfaces. This branch of science describes molecular engineering to create functional structures (at nanoscale). Significant progress has been made in the last 20 years, even though this scientific field is still in its early stages and is expected to grow quickly in the future. The importance of nanoparticles has increased due to the advancement in chemical sensors, microelectronics, and magnetic data storage devices [1]. Planning, synthesis, implementation, and understanding of the basic phenomena and physical properties of nanomaterials are all part of nanotechnology [2, 3]. Nanoparticles are particles that range from 1 to 100 nanometers for laboratory uses and that increase from 1000 nanometers in size are used for industrial purposes [4, 5]. Individual atoms or molecules, as well as bulk materials, are sufficiently distinct from nanoparticles.

Nanomaterials have exceptional chemical, magnetic, and optical properties because of their greater surface area to volume proportion, making them extremely useful [6]. Because of their possible properties in plasmonic, metal oxide nanoparticles have attracted the interest of researchers [7]. Numerous methods were reported for the synthesis of nanomaterials with desired properties [8-13]. Since the properties of nanomaterials are dependent upon their size, hence a suitable method for controlling the nanoparticle's size is needed. Nanoparticles are synthesized by the chemical,

Received: July 2023; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Syeda Shaista Gillani <shaistaaligillani@gmail.com>

physical, and green approaches. Green chemistry is the key to design chemical products that eliminates the use of hazardous chemicals [14]. Besides various chemical and physical approaches, green synthesis for the production of nanomaterial is gaining popularity. As a reducing and precipitation agent, plant leaves, stem or root extract, or any other biomaterials are used in the green process. Plant used in current study is *Ocimum basilicum* (Figure 1). The common name is Basil, Sweet Basil in English, as Babuli Tulsi in Bengali and Hindi. *Ocimum basilicum* has a lot of medicinal values as pain-relieving activity, anti-swelling action, anti-microbial activity, anti-oxidant action, anti-ulcer activity, cardiac stimulating action, chemo modulatory activity, hepatoprotective activity, and hypoglycemic action.

Metal nanoparticles are classified as monometallic, bimetallic, and trimetallic. Bimetallic and trimetallic nanoparticles have stronger catalytic activity and better performance in various fields than monometallic nanoparticles, as trimetallic nanoparticles can be used as heterogeneous nano-catalysts in a variety of organic processes [15, 16]. Trimetallic nanoparticles have been recently explored, because of their novel physicochemical features caused by the synergetic properties of their monometallic complements [17]. Trimetallic nanoparticles have gained familiarity due to their exceptional characteristics, strong features, as well as creative applications [18-21] in cancer therapy and diagnosis, catalytic applications, antimicrobial activities and active food packing. Nano-scientists and researchers worked for nano-size as well as fine geometry of multi-metallic nanoparticles as a heterodimer, core, and amalgams to improve their catalytic performance [22-24].

The present study aimed the synthesis of trimetallic oxide (CuO-ZnO-MnO) nanoparticles by using green method. Significant antimicrobial activity was also explored against various bacterial stains.

2. MATERIALS AND METHODS

The chemicals utilized were all of analytical grade and purchased from reputable vendors. The plant (*Ocimum basilicum)* was obtained from Jinnah Garden, Lahore, Pakistan.

Fig. 1. *Ocimum basilicum* Plant.

2.1. Preparation of Extract

To prepare plant extract, plant powder (12 g) was mixed to distilled water (100 mL) with continuous stirring at 40 °C for 45 minutes. Prepared plant extract was chilled and filtered.

2.2. Synthesis of Trimetallic Oxide Nanoparticles

Leaf extract was mixed with 0.25 M equimolar salt solution of Zinc chloride, Copper chloride and Manganese chloride in 1:1 ratio. In the reaction mixture 100 mL of leaf extract was added in 100 mL of salt solution at 35 °C. Distilled water was used to wash the synthesized NPs. Reaction medium was left for 30 minutes at 25 °C to settle down nanoparticles. The pellets of nanoparticles were attained after centrifugation at 2000 rpm. Nanoparticles were dried at 45 °C in oven for 23 hours. Dried NPs were calcined in muffle furnace at 550 °C for 1 hour. The flow sheet diagram for the synthesis of trimetallic oxide nanoparticles is shown in Figure 2.

2.3. Characterization Techniques

Different instruments were used for the synthesis and chareterization of trimetallic oxide (CuO-ZnO-MnO) nanoparticles: Electronic balance (FA2004). pH-meter (Cyberscan $500^{PH),}$ Desktop constant temperature drying oven (WHL25A), Centrifuge machine (GY773), Digital autoclave (HICVAVETTI HVE-50), Muffle furnace (SX-2.5-10), UV-Visible (Ultra-3000) spectrophotometer, FTIR (Cray 360 FTIR) spectrophotometer, X-Ray diffractometer (AXS SMART APEX-I), and Scanning Electron Microscope (HITACHI S-3400N).

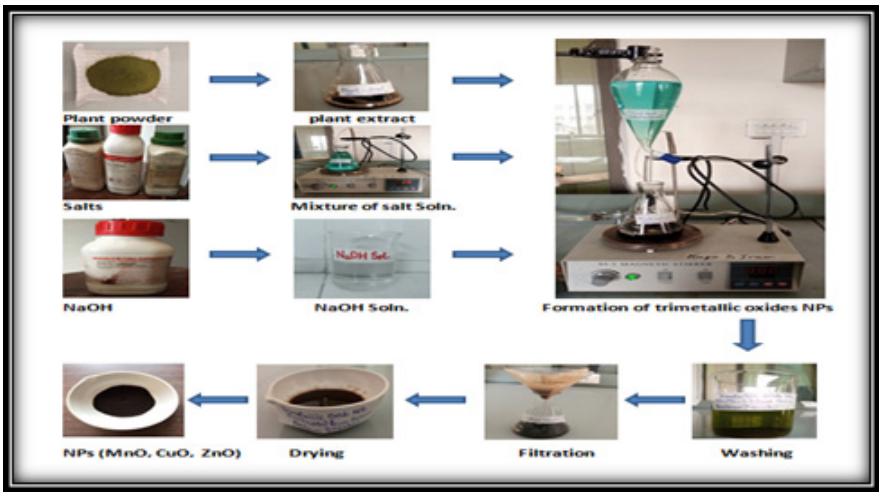

Fig. 2. Flow sheet diagram representing green synthesis of trimetallic oxide nanoparticles.

3. RESULTS AND DISCUSSION

Various spectroscopic techniques, such as UV-Visible, FTIR, XRD, SEM-EDX were employed to characterize the synthesized trimetallic oxides (CuO-ZnO-MnO) nanoparticles.

3.1. UV-Visible

Trimetallic oxide (CuO-ZnO-MnO) NPs were characterized by UV-Vis spectrophotometer. This technique was also used to monitor the production of trimetallic oxide nanoparticles. The colour of reaction medium was changed from dark brown to colloidal light brown, demonstrating the surface plasmon resonance effect (SPR) of trimetallic oxide nanoparticles. After the formation of nanoparticles, the clear aqueous extract solutions transformed into colloidal solutions. The results showed the absorption maximum at 280-370 nm that matches with ranges reported (CuO-ZnO-MnO) [25]. The absorption peak given in the Figure 3, represents the surface plasmon resonance phenomenon that occurs when nonbonding electrons were excited to an upper energy state, from the ground state as indicated by the shift in colour from dark brown to light brown. The reasonably strong absorption peaks of trimetallic oxide NPs specify the NP distribution's monodispersed character [22, 23].

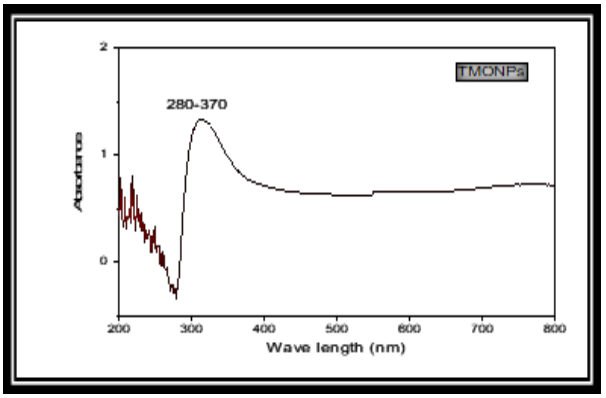

Fig. 3. UV-Visible spectrum of trimetallic oxide nanoparticles.

3.2. FTIR

FTIR has been employed to confirm the different functional groups present in plant extracts that are responsible for bio-reduction of metal ion to trimetallic oxide NPs and consequently their stabilization. Figure 4 and 5 showed FTIR spectra of studied plant extracts (*Ocimum basilicum*) as well as trimetallic oxide nanoparticles derived from the extract, respectively. The O-H stretching vibrations for alcohols and phenols at 3500-3200 cm^{-1} appeared as broad band in extract [25], while the same appeared as narrowed band in nanoparticles. The peak at 1640 cm^{-1} confirmed the stretching vibrations of C=O in carboxyl and bending vibrations of C=N present in the amide functional group of flavonoids, phenolic acids, and other compounds [26]. The bands around 424, 568, and 635cm^{-1} are assigned to trimetallic bonding with oxygen (ZnO, CuO [27], and MnO [28] bands from hydroxyl groups [29, 30]. The FTIR spectrum of the trimetallic oxide nanoparticles attained from leaf extracts showed the occurrence of a number of functional groups (Table 1) connected with active phytochemicals like as flavonoids, aromatic

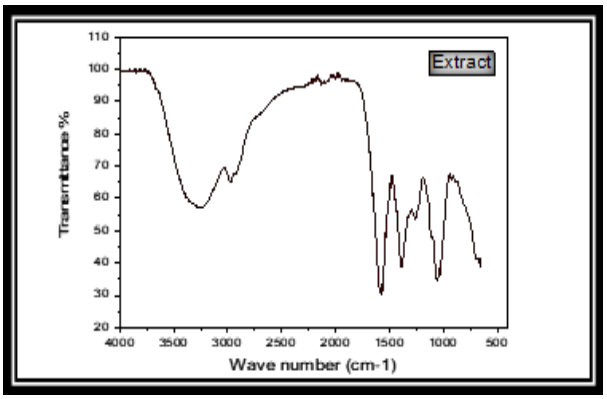

Fig. 4. FTIR spectrum of Plant Extract.

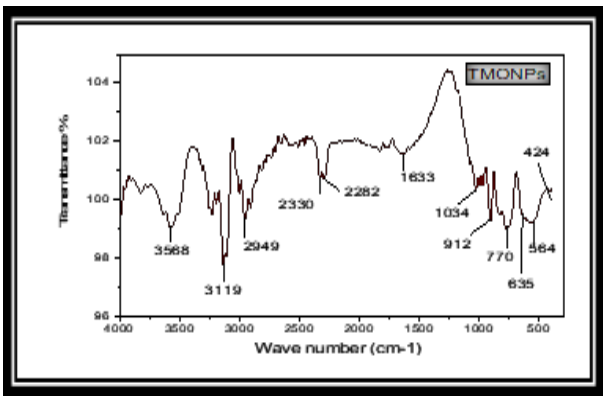

Fig. 5. FTIR spectrum of trimetallic oxide nanoparticles.

Table 1. FTIR peaks of trimetallic oxide nanoparticles.

Wavenumber (cm^{-1})	Functional groups
3568	O-H stretching
3119	Aromatic C-H stretch
2949	C-H_3 stretch
2330	Stretching of CO_2, CH
1633	C=N, C=O stretching vibration in carboxyl
912	O-H Bending
424, 564, 635	Cu-O, Zn-O, Mn-O

compounds, and phenolic acids, among others, which were most probably to be responsible for bio reduction of metal precursor and capping of synthesized trimetallic nanoparticles.

3.3. XRD

This technique was used to measure the crystallinity of nanoparticles. The XRD measurements was carried out by using Bruker AXS D8 Discover diffractometer at 40 kV, with 35 mA at room temperature. The diffracted angle 2θ was selected from 10° to 80° (Figure 6). The interplanar spacing between atoms of nanoparticles was studied by using Bragg's law:

$$n\lambda = 2d \sin\theta \qquad (1)$$

Where, θ is diffraction angle, λ is wavelength of X-ray radiations of Cu K-α which is 1.5406 Å, d is interplanar spacing and n = 1, 2, 3, 4….

The XRD peaks of green synthesized trimetallic oxide (CuO, MnO, ZnO) nanoparticles of *Ocimum basilicum* leaf, at diffracted angles 20.18°, 35.6°, 39.26° and 56.22°were indexed as (100), (021), (111) and (160) planes for MnO, respectively (JCPDS card No. (04-0326)), which indicated Orthorhombic symmetry for MnO nanoparticles. For ZnO nanoparticles, diffracted peaks at 31.77°, 35.6°, 47.16°, 56.22°, and 67.43° were indexed as (100), (101), (102), (110) and (112) indexes planes, respectively. These peaks showed resemblance with those described in JCPDS card No. (36-1451) which indicated Hexagonal ZnO nanoparticles Barzinjy and Azeez [32]. For CuO nanoparticles, diffracted peaks at 31.77°, 35.6°, 39.26°, 47.16°, 49.88° and 67.43° were indexed as (-110), (002), (200), (-112), (-202) and (113) indexes planes, respectively. These peaks showed resemblance with those described in JCPDS card No. (45-0937) which indicated Monoclinic shape for CuO nanoparticles crystal.

Moreover, the XRD spectrum showed few impurity peaks in addition to distinctive CuO, ZnO and MnO nanoparticles peaks, which indicated the presence of impurities in synthesized nanoparticles.

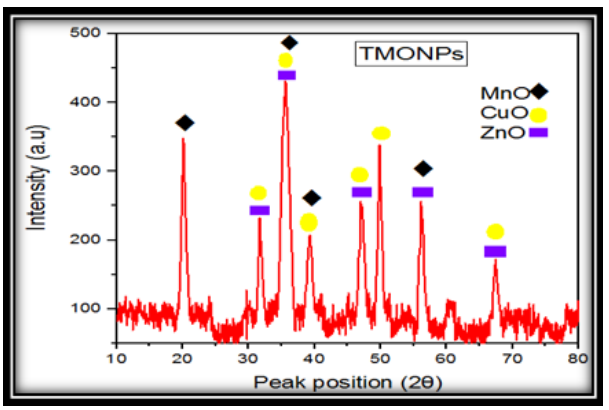

Fig. 6. XRD spectrum of trimetallic oxide nanoparticles.

Moreover, the robust and relatively broad diffraction peak showed that some other moieties were also present [30, 31].

3.4. SEM-EDX

The monoclinic (CuO), orthorhombic (MnO) and hexagonal (ZnO) shapes of trimetallic oxide nanoparticles are shown by SEM (Figure 7). The SEM results were in good agreement with XRD results. From the EDX spectra (Figure 8) Copper, zinc, manganese, and oxygen are found in trimetallic nanoparticles synthesized from *Ocimum basilicum* leaf extracts. However, trace components such as potassium, sulphur, and chlorine have also been found, which were more likely due to the presence of minerals in the soil in the form of metals and transported into various parts of the plant such as the leaf. In contrast to zinc and manganese, the resulting weight percentage for copper is lower than the feed stoichiometric ratio Table 2). Furthermore, the proportion of oxygen is considerable (about 66.36% in atomic and 44.71% in weight percent composition in trimetallic oxide nanoparticles

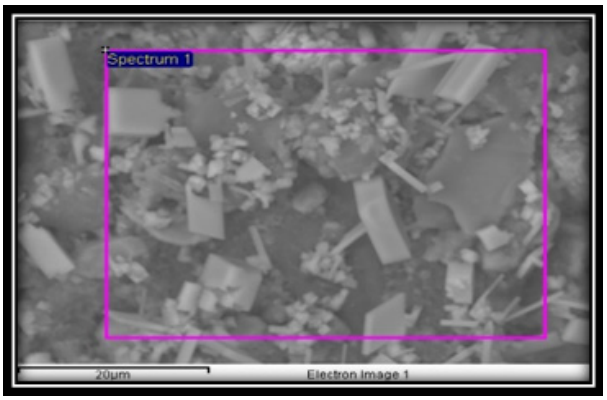

Fig. 7. SEM of nanoparticles.

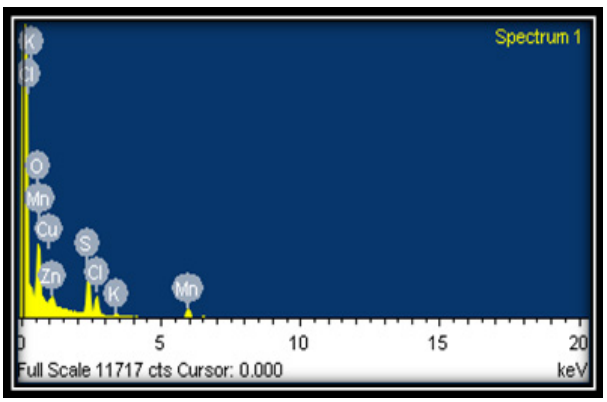

Fig. 8. EDX of nanoparticles.

Table 2. Results of EDX analysis of nanoparticles.

Element	Weight%	Atomic%
Oxygen	41.71	66.36
Manganese	18.48	5.96
Sulfur	14.46	2.48
Chlorine	10.06	2.76
Zinc	8.98	1.20
Copper	4.90	1.14
Potassium	1.42	0.93

produced from leaf extracts), that was most likely owing to its absorption from the reaction media throughout the bio-reduction procedure.

3.5. Antimicrobial Activity

Ocimum basilicum extracts have long been used for their powerful antibacterial and antifungal activities, both traditionally and commercially [9]. The varied chemical profile of *Ocimum basilicum* and high concentration of phenolic (-OH) components present in it, indicated that the therapeutic capabilities of plants were dependent on the precise chemical groups separated. Tannins, terpenoids, quinones, polysaccharides, phenols, and flavonoids are examples of antimicrobial compounds. In the present work, the green production of Trimetallic oxide NPs capped with these essential phytochemicals adds to the phytochemical mixture's improved characteristics. This is especially important for overcoming antimicrobial resistance with higher benefits and lesser harm to people.

3.5.1. Inhibitory effects of trimetallic oxide NPs on the growth of bacterial strains

The clinical pathogenic strains used in the study were E. coli (gram-nagative) and S. aureus (gram-positive). The antibacterial activity of trimetallic oxides (ZnO, CuO, MnO) nanoparticles against the studied pathogenic strains is shown in Figure 9. The antimicrobial activity of synthesized trimetallic nanoparticles (100µg/mL) was studied against pathogenic strains and their zone of inhibition was found as 25 mm for nanoparticles and 28 mm for standard (sulfamethoxazole) against E. coli. Zone of inhibition against S. aureus was 33 mm for both nanoparticles and standard (Table 3). Solvent

Table 3. Zone of Inhibition (ZOI) NPs against *E. coli* and *S. aureus*.

Clinical pathogenic strains	Zone of inhibition (mm)		
	Standard (Sulfamethoxazole) positive control	Trimetallic oxides (CuO, ZnO, MnO) NPs	Solvent (Double distilled water) negative control
Escherichia coli	28	25	0
Staphylococcus aureus	33	33	0

Table 4. Minimum Inhibitory Concentration (MIC) of NPs against *E. coli* and *S. aureus*.

Clinical pathogenic strains	Minimum inhibitory concentration (µg/mL))	
	Standard (Sulfamethoxazole) positive control	Trimetallic oxides (CuO, ZnO, MnO) NPs
Escherichia coli	5	10
Staphylococcus aureus)	8	12

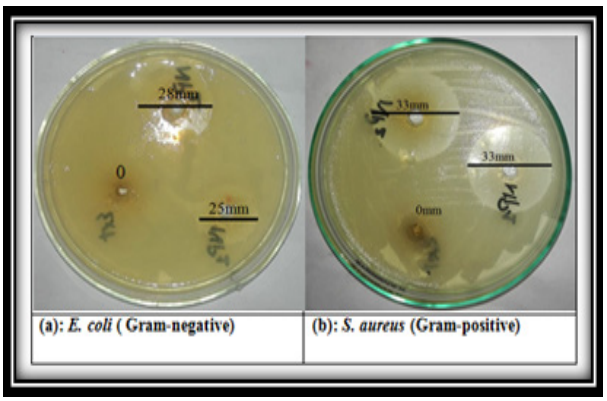

Fig. 9. Zone of inhibition of growth by NPs against E. coli (a) and S. aureus (b).

(double distilled water) showed no antimicrobial activity. Minimum Inhibitory Concentration (MIC) of nanoparticles for *E. coli* was 12 µg/mL and for *S. aureus* was 8 µg/mL. MIC of sulfamethoxazole was 10µg/mL for *E. coli* and 5µg/mL for *S. aureus* (Table 4).

4. CONCLUSIONS

Green synthesis method was used to effectively synthesize trimetallic oxides (CuO, MnO, ZnO) NPs using *Ocimum basilicum* leaf extract, NaOH solution, and equimolar precursor salt solution ($ZnCl_2$, $MnCl_2$, and $CuCl_2$). The sequential reduction of metal ions resulted in the production of nanoparticles within 5 minutes at 35 °C, indicating a faster reaction rate than the chemical techniques used before for such synthesis. UV-Visible spectroscopy, SEM-EDX, XRD, and FTIR confirmed the prominent functional groups and verified the existence of stable, monoclinic (CuO), orthorhombic (MnO) and hexagonal (ZnO) nanoparticles with well-defined dimensions. Moreover, nanoparticles showed significant antibacterial action against *Escherichia coli* and *Staphylococcus aureus*.

5. ACKNOWLEDGEMENT

The authors are thankful to the Lahore Garrison University, DHA Phase-VI, Lahore 54792, Pakistan.

6. CONFLICT OF INTEREST

The authors declare no competing conflict of interest.

7. REFERENCES

1. T.A. Orshiso, E.A. Zereffa, and H.C.A. Murthy. One-Pot Biopreparation of Trimetallic ZnO–MgO–CuO Nanoparticles: Enhanced Cytotoxicity, Antibacterial Activities and Molecular Docking Studies. *Chemistry Africa* 7: 1963–1980 (2024).
2. B. Munirathinam and L. Neelakantan. Role of crystallinity on the nanomechanical and electrochemical properties of TiO2 nanotubes. *Journal of Electroanalytical Chemistry* 770: 73-83 (2016).
3. K. Roy, C. Sarkar, and C. Ghosh. Photocatalytic activity of biogenic silver nanoparticles synthesized using potato (Solanum tuberosum) infusion. *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy* 146: 286-291 (2015).
4. A. Khaleel and R.M. Richards. Ceramics. In: Nanoscale Materials in Chemistry. K.J. Klabunde

(Ed.) *John Wiley & Sons Publication, New York* pp. 85-120 (2001).

5. D. Knopp, D. Tang, and R. Niessner. Review: Bioanalytical applications of biomolecule-functionalized nanometer-sized doped silica particles. *Analytica Chimica Acta* 4:14-30 (2009).
6. S. Sun, C.B. Murray, D. Weller, L. Folks, and A. Moser. Monodisperse FePt Nanoparticles and Ferromagnetic FePt Nanocrystal Superlattices. *Science* 287: 1989-1992 (2000).
7. S.A. Maier, M.L. Brongersma, P.G. Kik, S. Meltzer, A.A. Requicha, and H.A. Atwater. Plasmonia route to nanoscale optical devices. *Advanced Materials* 13: 1501-1505 (2001).
8. Z.H. Pak, H. Abbaspour, N. Karimi, and A. Fattahi. Eco-friendly synthesis and antimicrobial activity of silver nanoparticles using Dracocephalum moldavica seed extract. *Applied Sciences* 6: 69 (2016).
9. H. Kumar and R. Rani. Structural characterization of silver nanoparticles synthesized by micro emulsion route. *International Journal of Engineering and Innovative Technology* 3: 344-348 (2013).
10. S. Lee, I.S. Cho, J.H. Lee, D.H. Kim, D.W. Kim, J.Y. Kim, H. Shin, J.K. Lee, H.S. Jung, N.G. Park, K. Kim, M.J. Ko, and K.S. Hong. Two-step sol−gel method-based TiO_2 nanoparticles with uniform morphology and size for efficient photo-energy conversion devices. *Chemistry of Materials* 22: 1958-1965 (2010).
11. X. Liu and P.X. Liu. Bimetallic nanoparticles: kinetic control matters. *Angewandte Chemie International Edition* 51: 3311-3313 (2012).
12. L. Nadaf, K. Venkatesh, M. Gadyal, and M. Afzal. Polyaniline-tin oxide nanocomposites: synthesis and characterization. *IOSR Journal of Applied Chemistry* 9: 55-61 (2016).
13. P.D.G. Silva, A.T. de Figueiredo, and E. Longo. Synthesis and characterization of core-shell ZrO2@ TiO_2. *XIV Brazil MRS Meeting - Rio de Janeiro, (27 September - 01 October, 2015* (2015).
14. P.T. Anastas and J.C. Warner (Eds.). Principles of green chemistry (Chapter 4). Green Chemistry: Theory and Practice. *Oxford University Press* (1998).
15. N. Basavegowda, K. Mishra, and Y.R. Lee. Trimetallic FeAgPt alloy as a nanocatalyst for the reduction of 4-nitroaniline and decolorization of rhodamine B: A comparative study. *Journal of Alloys and Compounds* 701: 456-464 (2017).
16. W. Yu, M.D. Porosoff, and J.G. Chen. Review of Pt-based bimetallic catalysis: from model surfaces to supported catalysts. *Chemical Reviews* 112: 5780-5817 (2012).
17. J. H. Park and H.S. Ahn. Electrochemical synthesis of multimetallic nanoparticles and their application in alkaline oxygen reduction catalysis. *Applied Surface Science* 504: 144517 (2020).
18. R. Ferrando, J. Jellinek, and R.L. Johnston. Nanoalloys: from theory to applications of alloy clusters and nanoparticles. *Chemical Reviews* 108: 845-910 (2008).
19. N. Toshima. Capped bimetallic and trimetallic nanoparticles for catalysis and information technology. Paper presented at the *Macromolecular Symposia* 270(1): 27-39 (2008).
20. Z. Vaseghi, O. Tavakoli, and A. Nematollahzadeh. Rapid biosynthesis of novel Cu/Cr/Ni trimetallic oxide nanoparticles with antimicrobial activity. *Journal of Environmental Chemical Engineering* 6: 1898-1911 (2018).
21. H. Gu, Z. Yang, J. Gao, C. Chang, and B. Xu. Heterodimers of nanoparticles: formation at a liquid− liquid interface and particle-specific surface modification by functional molecules. *Journal of the American Chemical Society* 5:1234-35 (2005).
22. Y. Liu, E. Koep, and M. Liu. A highly sensitive and fast-responding SnO_2 sensor fabricated by combustion chemical vapor deposition. *Chemistry of Materials* 17: 3997-4000 (2005).
23. J.-G. Mattei, P. Grammatikopoulos, J. Zhao, V. Singh, J. Vernieres, S. Steinhauer, and F. Djurabekova, Gas-Phase synthesis of trimetallic Nanoparticles. *Chemistry of Materials* 31: 2151-2163 (2019).
24. S. Talam, S.R. Karumuri, and N. Gunnam. Synthesis, characterization, and spectroscopic properties of ZnO nanoparticles. *ISRN Nanotechnology* 2012: 372505 (2012).
25. S.S. Gillani, S.A. Khan, R. Nazir, and A.W. Qurashi. Green Synthesis of Mixed Metal Oxide (MnO, CuO, ZnO) Nanoparticles (NPs) using Rose Petal Extract: An investigation of their Antimicrobial and Antifungal Activities. *Scientific Inquiry and Review* 5(4): 55-69 (2021).
26. K.S. Sindhura, T. Prasad, P.P. Selvam, and O. Hussain. Synthesis, characterization and evaluation of effect of phytogenic zinc nanoparticles on soil exo-enzymes. *Applied Nanoscience* 4: 819-827 (2014).
27. D. Paul, S. Mangla, and S. Neogi, Antibacterial study of CuO-NiO-ZnO trimetallic oxide nanoparticle. *Materials Letters* 271: 127740 (2020).
28. V. Kumar, K. Singh, S. Panwar, and S.K. Mehta. Green synthesis of manganese oxide nanoparticles

for the electrochemical sensing of p-nitrophenol. *International Nano Letters* 7:123-131(2017).

29. N. Bala, S. Saha, M. Chakraborty, M. Maiti, S. Das, R. Basu, and P. Nandy. Green synthesis of zinc oxide nanoparticles using Hibiscus subdariffa leaf extract: effect of temperature on synthesis, anti-bacterial activity and anti-diabetic activity. *RSC Advances* 5: 4993-5003 (2015).
30. A.A. Barzinjy and H.H. Azeez. Green synthesis and characterization of zinc oxide nanoparticles using Eucalyptus globulus Labill. leaf extract and zinc nitrate hexahydrate salt. *SN Applied Sciences* 2: 991(2020).
31. W. Muhammad, N. Ullah, M. Haroon, and B.H. Abbasi. Optical, morphological and biological analysis of zinc oxide nanoparticles (ZnO NPs) using *Papaver somniferum* L. *RSC Advances* 9: 29541-29548 (2019).

Proceedings of the Pakistan Academy of Sciences: A Physical and Computational Sciences 61(3): 311-316 (2024)

ISSN (Print): 2518-4245; ISSN (Online): 2518-4253
http://doi.org/10.53560/PPASA(61-3)680

Pakistan Academy of Sciences

Research Article

Kinetics and Thermodynamic Study of Cellulase Embedded Metal Organic Frameworks

Kainat Zahra[1], Hina Zain[*1], Nazia Kanwal[1], Jamal Ahmad[1], Aysha Bukhari[2], Ammara Nazir[2], and Athar Hussain[3]

[1]Department of Biological Sciences, Superior University, Lahore, Punjab, Pakistan
[2]Department of Chemistry, Minhaj University, Lahore, Pakistan
[3]Shaheed Benazir Bhutto University of Veterinary and Animal Sciences, Karachi, Sindh, Pakistan

Abstract: Cellulase is an important enzyme used for many purposes in different industrial sectors. Even though cellulase has so many applications, it easily denatures with a little change in pH and temperature, which causes low stability, usability, and activity. To enhance its activity and stability, immobilization of porous materials is the best way to enhance its activity, stability, and life span. For immobilization, a Metal-organic framework (MOF) is considered as a potential candidate. Cellulase@Zn- benzene 1-4 di-carboxylic acid (BDC) by hydrothermal method and Zn-cellulase-benzene 1-4 dicarboxylic acid (BDC) by *de novo* approach were prepared, and their activities were analyzed. Zn-cellulase-benzene 1-4 dicarboxylic acid (BDC) produced by the *de novo* approach, shows higher activity, stability, catalytic performance, and life span than the free enzyme, and cellulase@Zn- benzene 1-4 di-carboxylic acid (BDC) produced by the hydrothermal method.

Keywords: Cellulase, Immobilization, Metal-organic Framework, Kinetics, De-novo Synthesis.

1. INTRODUCTION

In many biotechnology processes, enzymes are used as highly effective catalysts. Cellulase, one of these enzymes, is essential for biotechnology applications. Cellulases are produced by a wide variety of organisms, including bacteria, fungi, protozoans, and some animals. Cellulase enzymes have shown potential in various industries, such as the production of biofuels, pulp, paper manufacturing, food processing, textiles, detergents, and food. Cellulase is commonly used to break down 1, 4 glycosidic bonds in cellulose crystals to create glucose [1]. This hydrolysis breaks down the cellulose into smaller oligosaccharides and ultimately glucose, which can be used as a source of energy by the organism. Despite the potential of cellulase enzymes in cellulose hydrolysis, their practical application is limited by their low stability, reusability, and activity [2, 3]. Enzyme immobilization is a potent and straightforward strategy for addressing several of difficulties connected with utilizing enzymes as industrial catalysts, such as enzyme inhibition, selectivity, stability, and recyclability, by chemicals in the reaction media [2, 4]. Enzymes can be more reliably preserved during storage and working settings if they are immobilized so that they can be easily isolated from the reaction medium. Immobilization facilitates the separation of enzymes from their reaction media and increases their stability during storage and operation [5].

Nanomaterials have gained significant interest as potential substrates for protein immobilization in recent times, owing to their exceptional features, including remarkably favorable mass transfer resistance, high surface area, and easy diffusibility [6]. Metal-organic frameworks (MOFs) stand out among other Nano-materials due to their astounding surface-to-volume ratio and structural variety. Since MOFs have inherent structural,

Received: September 2023; Revised: August 2024; Accepted: September 2024
* Corresponding Author: Hina Zain <hina.zain@superior.edu.pk>

catalytic, electrical, optical, and magnetic properties that make them suitable for use in a variety of applications, including biological processes, industrial catalysis, and sensing, their potential has been expanded beyond catalysis using MOFs as immobilization platforms for enzymes. Due to their simplicity in functionalization, availability of appropriate hydrophilic/hydrophobic groups, and robust electrostatic interactions with proteins, MOF-enzyme platforms offer a variety of benefits, including high enzyme-substrate ratios, water stability, and reusability [7].

MOF is a particular kind of porous crystal that is composed of a network of metal atoms and a network of organic ligands. Since MOFs may be designed with a wide range of cavities, porosity, topology, surface area, and pore volume characteristics, they have the potential to be used in many different fields. The unique structural properties of MOFs enable them to be tailored for specific applications, making them highly versatile and attractive materials for various fields of research [8]. In addition, MOFs have become widely used as a substrate for immobilizing enzymes [9]. Cellulase-embedded MOFs are a promising avenue for the efficient conversion of cellulose into biofuels and other value-added products [10]. The *de novo* approach for the creation of cellulase-embedded MOF without the need of pre-existing structures guarantees the repeatability and comprehension of the experimental setup with specific reaction conditions. In contrast, the hydrothermal approach suggests using high temperatures and high pressures to synthesize the MOF embedded with cellulase [11].

In the present study both the hydrothermal process and the *de novo* methods are used to construct a cellulase-embedded MOF. This study aims to evaluate the catalytic efficacy, stability, and activity of the cellulase-embedded MOF made using these techniques with that of a conventional cellulase. A thorough comparison between the cellulase-embedded MOFs made by the hydrothermal method and the *de novo* approach has been carried out.

2. MATERIALS AND METHODS

Cellulase (*Aspergillus niger*) (C1184), zinc chloride (7646-85-7) and benzene 1-4 dicarboxylic acid (4612-26-4) was purchased from Sigma Aldrich, H_2O_2, aqueous ammonia, and rice husk were purchased from local market. Rice husk was dried for one day and then ground to fine powder.

2.1. Preparation Cellulase@Zn- benzene 1-4 di-carboxylic acid (BDC) MOF by Hydrothermal Method

2.1.1. Zinc-benzene 1-4 dicarboxylic acid (BDC) preparation

Zinc chloride and benzene 1-4 di-carboxylic acid were prepared separately in distilled water. 1.65 g of benzene 1-4 di-carboxylic acid (BDC) and 1.36 g of zinc chloride were separately dissolved in 100 ml of distilled water. 30 ml of zinc chloride solution was added dropwise to BDC solution with constant stirring for one hour. Mixture was transferred to a hydrothermal reactor and kept in an oven at 110°C overnight, solution's volume was reduced by heating. After reduction of volume, it was transferred into a china dish and dried to remove any moisture. Dried powdered of Zn-BDC MOF was obtained [12].

2.1.2. Preparation of Cellulase@Zn-benzene 1-4 di-carboxylic acid (BDC)

1 g of cellulase was dissolved in 100 ml of distilled water with constant stirring for 5-6 hours to dissolve the cellulase. Zn- BDC MOF was mixed with cellulase with constant stirring for 5–6 hours. Enzyme was properly embedded on the MOF. It was then filtered and dried in a desiccator for a few days.

2.1.3. Preparation of Zn-cellulase-BDC MOF by de novo method

In the *de novo* approach, preparation of MOF and embedding of enzyme was done simultaneously. 1.36 g of zinc chloride, 1.65 g of BDC, and 1 g of cellulase were dissolved separately each in 100 ml of water. Then 30 ml of cellulase solution was mixed with 30 ml of BDC solution. The 30 ml of zinc chloride was added dropwise into the above-mentioned solution with continuous stirring. After that, it was kept for 24 hours at room temperature. It was then centrifuged at 4000 rpm for 10–15 minutes, washed it 3 times with distilled water, and finally left it to dry for a few days [13].

2.2. Characterization

2.2.1. Fourier transforms infrared spectroscopy (FTIR)

FTIR JASCO 4100 was used to record the FTIR spectrum within the range of 4000cm to 400 cm.

2.2.2. Scanning electron microscopy (SEM)

Using SEM technique, size, and morphology of enzyme embedded MOF was examined.

2.3. Measurement of Cellulase Activity

2.3.1. Pre-treatment with H_2O_2 and aqueous ammonia

Initially, three beakers were taken; in each beaker, 2 g of rice husk was introduced, and then 3% of H_2O_2 was added in each beaker. A hot plate stirrer was then taken, and each sample was heated at 85°C for 5 to 6 hours at a speed of 150 rpm. Next, the slurry was filtered with a vacuum filter and washed with distilled water. Then, each sample was treated with 20% ammonia for 5 hours at 100°C on a hot plate stirrer. After that, each solution was washed with distilled water and filtered with the help of a vacuum filter.

2.4. Enzymatic Hydrolysis

2.4.1. Rice husk hydrolysis with standard cellulase

Standard cellulase, Zn-cellulase-BDC MOF and Cellulase@Zn BDC MOF were taken in three different Erlenmeyer flasks. Pre-treated rice husk was introduced in each flask. Enzyme substrate ratio was 1:10. Each flask was placed on a hot plate and stirred at 50 °C at 200 rpm for 6 hours for complete hydrolysis. Each flask was cooled and solution was filtered through a vacuum filter. During the stirring, the sample was aliquoted after every 15 minutes for two hours, and the reactivity was checked by UV spectrophotometry. The absorbance of the treated sample was measured with a UV spectrophotometer at 270 nm. The blank for the experiment was the pre-treated rice husk without enzyme treatment [14].

2.5. Standard Curve of Glucose

For the preparation of the stock solution, 200 mg of glucose was dissolved in 20 ml of distilled water, and dilutions were prepared ranging from 0.5 mg/ml to 4.5 mg/ml. The absorbance was measured on a UV spectrometer at 270 nm.

3. RESULTS AND DISCUSSION

3.1. Characterization

3.1.1. SEM analysis

Figure 1 shows that SEM images of Zn-cellulase-BDC MOF and Cellulase@Zn BDC MOF are similar. Both have thin layers with broken and intertwined edges. Particles are slightly spherical and have uniform growth in all directions.

3.1.2. FTIR analysis

An FTIR chromatogram (shown in Figure 2) proved that both cellulase embedded MOF similar. It confirmed the functional group of the bond generated during the enzyme embedded MOF synthesis procedure. Broad absorption bands

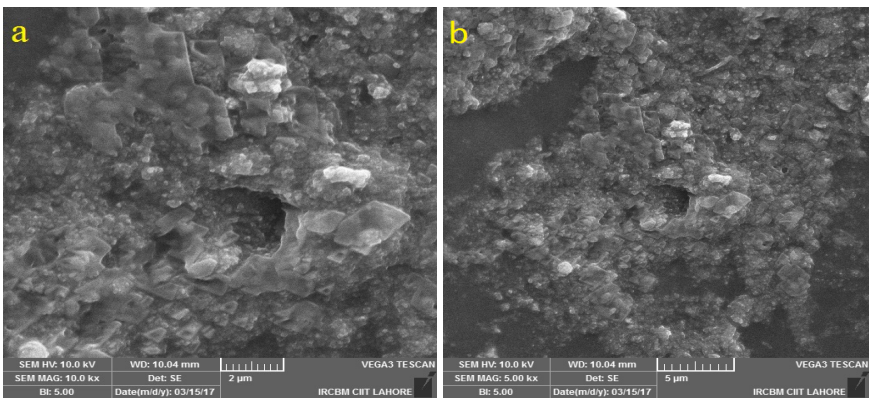

Fig. 1. SEM Images of enzyme embedded MOF, (a) Zn-cellulase-BDC MOF and (b) Cellulase@Zn BDC.

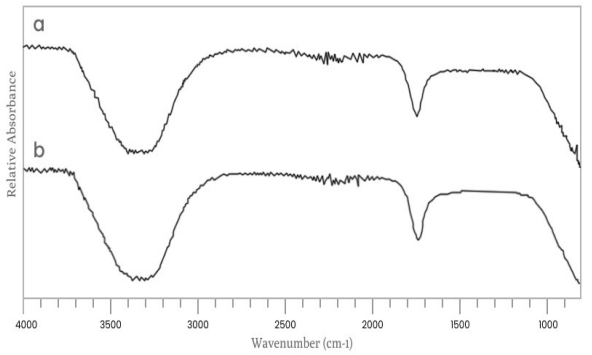

Fig. 2. FTIR of cellulase embedded MOFs, (a) Zn-cellulase-BDC MOF and (b) Cellulase@Zn BDC.

seen at approximately 3000–3500cm^{-1} generally attributed to water molecules' stretching vibrations. Strong peaks were seen in MOFs, which are associated with the C=O stretching mode at around 1710 cm^{-1}.

3.2. Absorbance Measured on UV Spectrophotometer After Enzymatic Hydrolysis

Activity of cellulase@Zn BDC MOF and Zn-cellulase-BDC MOF was measured and compared with the activity of standard cellulase. Graph was plotted against absorbance and reaction time. It has been observed that in all three experiments, enzymes active reached equilibrium after 60 min. Standard cellulase showed maximum absorbance at 0.26 nm in Figure 3. cellulase@Zn-BDC showed maximum absorbance at 0.34 nm (Figure 4), and Zn-cellulase-BDC showed maximum absorbance at 0.56 nm (Figure 5).

3.3. Amount of Glucose Released After Enzymatic Hydrolysis

Rice husk was used as substrate to check the activity of enzyme embedded on MOF. Rice husk was treated with both MOF based cellulases and results were compared with the standard cellulase. Activity of each enzyme was measured by calculating the amount of glucose released from the rice husk after treating with cellulase. Standard curve of glucose was established for calculating the amount of glucose as shown in Figure 6. Absorbance measures at 270 nm and was converted into glucose concentrations using the standard curve and graphed versus time. Standard cellulase released 0.6 mg of glucose, Cellulase@Zn-BDC MOF released 0.8 mg, and Zn-cellulase-BDC MOF released 1.2 mg of

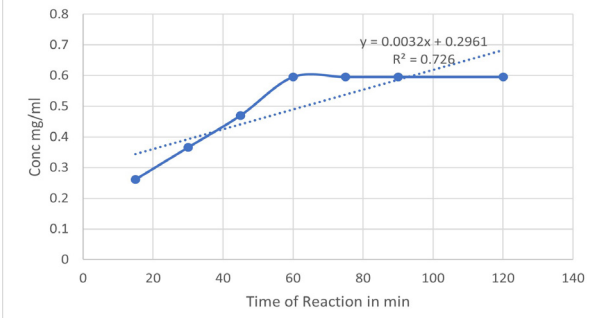

Fig. 3. Absorbance of standard cellulase with reaction time.

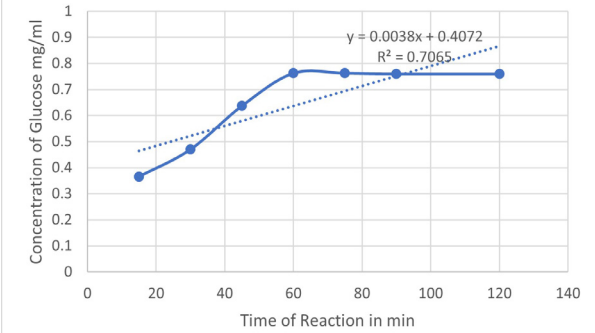

Fig. 4. Maximum absorbance of cellulase@Zn-BDC at 0.34 nm.

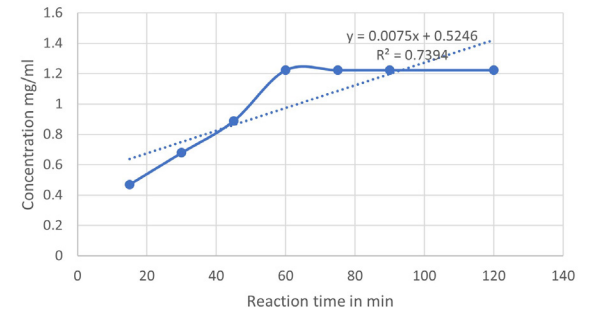

Fig. 5. Maximum absorbance of Zn-cellulase-BDC at 0.56 nm.

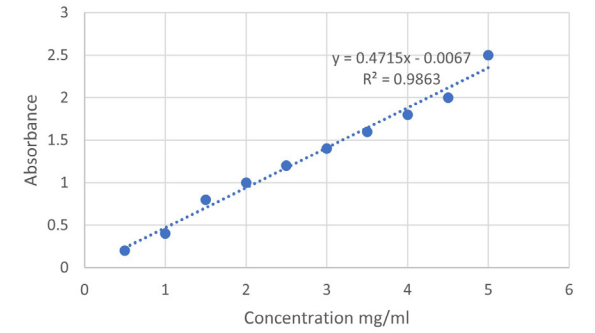

Fig. 6. Standard curve plotted by measuring the concentration of glucose at 270 nm.

glucose after enzymatic hydrolysis. The rate of the reaction is also measured in Figure 7. It indicated that Zn-cellulase-BDC *(de novo*) has high rate of reaction than standard cellulase and Cellulase@ Zn-BDC (hydrothermal). This represents, that Zn-cellulase-BDC which is synthesized by the *de novo* method, showed better activity than standard and Cellulase@Zn-BDC.

3.4. Calculated Reaction Velocity

Kinetic study of both the cellulase embedded MOF has been carried out and compared with the standard. From Figure 8 it has been observed that the kinetic activity of Zn-cellulase-BDC MOF (*de novo*) was much higher as compared to Celllase@ Zn-BDC MOF and standard cellulase. The kinetic study showed that activity of enzymes became stable over time as more product and less substrate was present in the reaction and stable after 60 minutes.

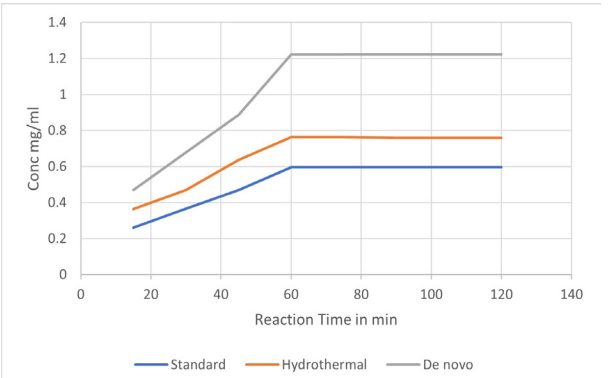

Fig. 7. Amount of glucose released by enzymatic hydrolysis of rice husk was calculated from the standard curve and plotted against time to calculate the rate of reaction.

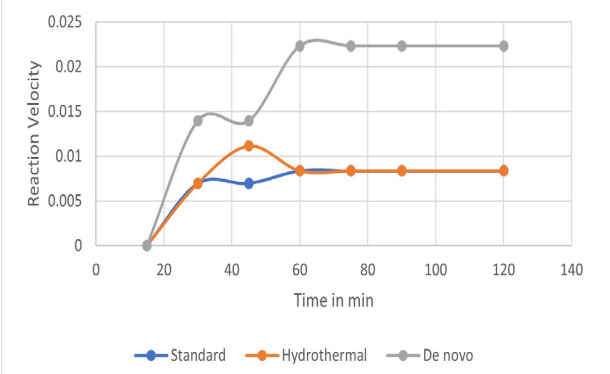

Fig. 8. Graph of reaction velocity for quantity of glucose produced vs time.

4. DISCUSSION

The essential and enhanced spatial arrangement of the enzyme within MOFs accounts for the high activity and stability of the MOF to meet the needs of industrial applications and to get a deeper understanding of the catalytic behaviour [15, 16]. To increase catalytic performance and decrease resistance to reagent mass transfer, enzymes are embedded in the MOF and the synthesis process is carefully regulated [17, 18]. When Zn-Cellulase-BDC MOF was put in comparison with Cellulase@ Zn-BDC MOF and ordinary cellulase, the reaction velocity was consistently higher. The robustness and stability of the Zn-Cellulase-BDC MOF are indicated by its long-term high activity and the stabilization of the response velocity. The distinct characteristics of the MOF structure are responsible for the Zn-Cellulase-BDC MOF's reported improved performance. Catalytic efficiency is predicted to rise because of the better and crucial spatial arrangement of the enzyme within the MOF, which improves substrate accessibility to the active sites [19, 20]. In addition, Zn-Cellulase-BDC MOF's catalytic ability appears to be much enhanced by the *de novo* synthesis technique used in its creation [21]. By using a controlled synthesis method, it is possible to minimize resistance to reagent mass transfer, maximize catalytic performance, and guarantee accurate enzyme embedding on the MOF.

5. CONCLUSIONS

In this study, cellulase was immobilize on MOF by two different methods, *de novo* and hydrothermal method. The kinetic activity of the enzymatic hydrolysis of rice husk after pre-treatment with H_2O_2 and aqueous ammonia were studied. It has been observed that enzyme embedded on MOF show increased activity and high rate of reaction. Kinetic study also proved that *de novo* method is best for the synthesis of enzyme embedded MOF.

6. CONFLICT OF INTEREST

The authors declare no conflict of interest.

7. REFERENCES

1. K. Khoshnevisan, F. Vakhshiteh, M. Barkhi, H. Baharifar, E.P. Akbar, N. Zari, H. Stamatis, and A.K. Bordbar. Immobilization of cellulase enzyme onto

magnetic nanoparticles: Applications and recent advances. *Molecular Catalysis* 442: 66-73 (2017).
2. C. Mateo, J.M. Palomo, G. Fernandez-Lorente, J.M. Guisan, and R. Fernandez-Lafuente. Improvement of enzyme activity, stability and selectivity via immobilization techniques. *Enzyme and Microbial Technology* 40(6): 1451-1463 (2007).
3. T. Hirasawa, T. Ookawa, S. Kawai, R. Funada, and S. Kajita. Production technology for bioenergy crops and trees. In: Research Approaches to Sustainable Biomass Systems, S. Tojo and T. Hirasawa (Eds.). *Academic Press, Oxford, UK* pp. 51-106 (2014).
4. R.C. Rodrigues, Á. Berenguer-Murcia, D. Carballares, R. Morellon-Sterling, and R. Fernandez-Lafuente. Stabilization of enzymes via immobilization: Multipoint covalent attachment and other stabilization strategies. *Biotechnology Advances* 52: 107821 (2021).
5. H. Jørgensen and M. Pinelo. Enzyme recycling in lignocellulosic biorefineries. *Biofuels, Bioproducts and Biorefining* 11(1): 150-167 (2017).
6. R. Ahmad and M. Sardar. Enzyme immobilization: an overview on nanoparticles as immobilization matrix. *Biochemistry and Analytical Biochemistry* 4(2): 1000178 (2015).
7. J. Mehta, N. Bhardwaj, S.K. Bhardwaj, K.H. Kim, and A. Deep. Recent advances in enzyme immobilization techniques: Metal-organic frameworks as novel substrates. *Coordination Chemistry Reviews* 322: 30-40 (2016).
8. P. Silva, S.M. Vilela, J.P. Tome, and F.A.A. Paz, Multifunctional metal–organic frameworks: from academia to industrial applications. *Chemical Society Reviews* 44(19): 6774-6803 (2015).
9. C. Doonan, R. Riccò, K. Liang, D. Bradshaw, and P. Falcaro. Metal–organic frameworks at the biointerface: synthetic strategies and applications. *Accounts of Chemical Research* 50(6): 1423-1432 (2017).
10. I. Ijaz, A. Bukhari, E. Gilani, A. Nazir, and H. Zain. Compositing of MOFs with ceramic and nanoparticles for efficient and rapid adsorptive desalination of artificial seawater or NaCl solution. *RSC Advances* 12(46): 29793-29804 (2022).
11. H.N. Abdelhamid and A.P. Mathew. Cellulose–metal organic frameworks (CelloMOFs) hybrid materials and their multifaceted Applications: A review. *Coordination Chemistry Reviews* 451: 214263 (2022).
12. A.D.R.P. Molo, E. Susanti, and S. Wonorahardjo. Application of silica rice husk ash for cellulase immobilized by sol-gel entrapment: *Scientific Study & Research*. *Chemistry & Chemical Engineering, Biotechnology, Food Industry* 22(1): 47-55 (2021).
13. W. Efrinalia, N. Novia, and E. Melwita. Kinetic model for enzymatic hydrolysis of cellulose from pre-treated rice husks. *Fermentation* 8(9): 417 (2022).
14. S.P. Edor, O.P. Edogbanya, and J.R. Kutshik. Cellulase activity of Aspergillus niger in the biodegradation of rice husk. *MOJ Biology and Medicine* 3(2): 49–51 (2018).
15. B. Qi, J. Luo, and Y. Wan. Immobilization of Cellulase on a Core-Shell Structured Metal-Organic Framework Composites: Better Inhibitors Tolerance and Easier Recycling. *Bioresource Technology* 268: 577-82 (2018).
16. M. Irfan, M. Ghazanfar, A. Rehman, and A. Siddique Strategies to reuse cellulase: Immobilization of enzymes (Part II). In: Approaches to Enhance Industrial Production of Fungal Cellulases. M. Srivastava, N. Srivastava, P.W. Ramteke, P.K. Mishra (Eds.). *Springer Cham* pp. 137-51 (2017).
17. X. Zhan, R. Tu, Z. Lu, J. Peng, C. Hou, and Z. Wang. Hierarchical mesoporous metal–organic frameworks encapsulated enzymes: Progress and perspective. *Coordination Chemistry Reviews* 443: 214032 (2021).
18. W. Xu, L. Jiao, Y. Wu, L. Hu, W. Gu, and C. Zhu. Metal–organic frameworks enhance biomimetic cascade catalysis for biosensing. *Advanced Materials* 33(22) 2005172 (2021).
19. M. Zhou, X. Ju, L. Li, L. Yan, X. Xu, and J. Chen Immobilization of cellulase in the non-natural ionic liquid environments to enhance cellulase activity and functional stability. *Applied Microbiology and Tiotechnology* 103: 2483-92 (2019).
20. K. Kamyar, A.K. Bordbar, D. Zare, D. Davoodi, M. Noruzi, M. Barkhi, and M. Tabatabaei. Immobilization of cellulase enzyme on superparamagnetic nanoparticles and determination of its activity and stability. *Chemical Engineering Journal* 171(2): 669-73 (2011).
21. S.S. Nadar, L. Vaidya, and V.K. Rathod. Enzyme Embedded Metal Organic Framework (Enzyme–MOF): *De Novo* Approaches for Immobilization. *International Journal of Biological Macromolecules* 149: 861-76 (2020).

Instructions For Authors

Manuscript Writing

The manuscript may contain a Title, Abstract, Keywords, INTRODUCTION, MATERIALS AND METHODS, RESULTS, DISCUSSION (or RESULTS AND DISCUSSION), CONCLUSIONS, ETHICAL STATEMENT (if applicable), ACKNOWLEDGEMENTS, CONFLICT OF INTEREST and REFERENCES, *and any other information that the author(s) may consider necessary*.

Title (Bold and font size 16): The title should be expressive, concise, and informative to the entire readership of the journal. It may include common terms, to make it more identifiable when people search online. Please avoid the use of long pervasive terms and non-standard or obscure abbreviations, acronyms, or symbols.

Abstract (font size 10, max 250 words)**:** Must be self-explanatory**,** stating the rationale, objective(s), methodology, main results, and conclusions of the study. Abbreviations, if used, must be defined on the first mention in the Abstract as well as in the main text. Abstracts of review articles may have a variable format.

Keywords (font size 10)**:** Provide five to eight keywords consisting of words and phrases that are closely associated with the topic depicting the article.

INTRODUCTION (font size 11)**:** Provide a clear and concise statement of the problem, citing relevant recent literature, and objectives of the investigation. Cite references in the text by number in square brackets**,** the reference must be cited in a proper English sentence [1]. or "... as previously described [3, 6–8]". For a single author: Bednorz [2] investigated the environmental pollution ... When there are only two authors: Bednorz and Allan [2] investigated the environmental pollution ... and for three or more authors: Bednorz *et al.* [2] investigated the environmental pollution ..; and list them in the REFERENCES section, in the order of citation in the text.

MATERIALS AND METHODS (font size 11)**:** Provide an adequate account of the procedures or experimental details, including statistical tests (if any), concisely but sufficiently enough to replicate the study. Relevant references to methodology must be cited.

RESULTS (font size 11)**:** Be clear and concise with the help of appropriate Tables, Figures, and other illustrations. Data should not be repeated in Tables and Figures but must be supported with statistics. The data presented in Tables and Figures must be elaborated in the main text.

DISCUSSION (font size 11)**:** Provide interpretation of the RESULTS in the light of previous relevant studies, citing published references.

CONCLUSIONS (font size 11)**:** Briefly state the implication of your study findings, and carefully address the study questions. Confine your conclusions according to the objectives of your study and the aspects covered in the abstract. Discuss both positive and negative findings.

ETHICAL STATEMENT (font size 10)**:** The statement of ethical approval by an appropriate ethics committee or review board must be included in the manuscript (if applicable), as per the Journal's policy.

ACKNOWLEDGEMENTS: (font size 10)**:** In a brief statement, acknowledge the financial support and other assistance.

CONFLICT OF INTEREST (font size 10)**:** State if there is any conflict of interest.

REFERENCES (font size 10)**:** References must be listed in numerical order as listed in the main text. Only published (and accepted for publication) journal articles, books and book chapters, conference proceedings, online reports, a degree thesis, and materials available on the website qualify for REFERENCES.

Declaration: Provide a declaration that: (i) the results are original, (ii) the same material is neither published nor under consideration for publication elsewhere, (iii) approval of all authors has been obtained, and (iv) in case the article is accepted for publication, its copyright will be assigned to the *Pakistan Academy of*

Sciences. Authors must obtain permission to reproduce, where needed, copyrighted material from other sources and ensure that no copyrights are infringed upon.

Manuscript Formatting

Manuscripts must be submitted in Microsoft Word (Latest Version .doc or .docx format); pdf files are not acceptable. Figures can be submitted separately in TIFF, GIF, JPEG, EPS, or PPT. Manuscripts, in *Times New Roman*, 1.15 spaced (but use single-space for Tables, long headings, and long captions of tables and figures). The Manuscript sections must be numbered, i.e., **1. INTRODUCTION, 2. MATERIALS AND METHODS,** and so on… (a) **Title** of the article **(**Capitalize the initial letter of each main word, font-size 16, **bold**), max 160 characters (no abbreviations or acronyms), depicting article's contents; (b) Author's complete name (font size 12, **bold**), and professional affiliation (i.e., each author's Department, Institution, Mailing address, and Email and Contact number, but no position titles) (font size 12); (c) Indicate the corresponding author with *; and (d) **Short running title**, max 50 characters (font size 10).

Headings and Subheadings (font size 11): All flush left

LEVEL-1: **ALL CAPITAL LETTERS; Bold**

Level-2: **Capitalize Each First Letter (**Except prepositions**); Bold**

Level-3: Capitalize the first letter only (Sentence case)***; Bold, Italic***

Level-4: *Run-in head*; *Italics*, *in the normal paragraph position. Capitalize the first letter only and end in a colon (i.e., **:**)*

A list of REFERENCES must be prepared as under:

a. Journal Articles *(Name of journals must be stated in full)*

1. J. Rashid, A. Ahsan, M. Xu, I. Savina, and F. Rehman. Synthesis of cerium oxide embedded perovskite type bismuth ferrite nanocomposites for sonophotocatalysis of aqueous micropollutant ibuprofen. *RSC Advances* 13(4): 2574-2586 (2023).
2. A. Fayyaz, N. Ali, Z.A. Umar, H. Asghar, M. Waqas, R. Ahmed, R. Ali, and M.A. Baig. CF-LIBS based elemental analysis of Saussurea simpsoniana medicinal plant: a study on roots, seeds, and leaves. *Analytical Sciences* 40(3): 413-427 (2024).
3. W. Bialek and S. Setayeshgar. Cooperative sensitivity and noise in biochemical signaling. *Physical Review Letters* 100: 258–263 (2008).

b. Books

4. W.R. Luellen (Ed.). Fine-Tuning Your Writing. *Wise Owl Publishing Company, Madison, WI, USA* (2001).
5. U. Alon and D.N. Wegner (Eds.). An Introduction to Systems Biology: Design Principles of Biological Circuits. *Chapman & Hall/CRC, Boca Raton, FL, USA* (2006).

c. Book Chapters

6. M.S. Sarnthein, J.E. Smolen, and J.D. Stanford. Basal sauropodomorpha: historical and recent phylogenetic developments. In: The Northern North Atlantic: A Changing Environment. P.R. Schafer and W. Schluter (Eds.). *Springer, Berlin, Germany* pp. 365–410 (2000).
7. S. Brown and L.A. Boxer. Functions of Europhiles. In: Hematology, (4th ed). W.J. Williams, E. Butler, and M.A. Litchman (Eds.). *McGraw Hill, New York, USA* pp. 103–110 (1991).

d. Reports

8. M.D. Sobsey and F.K. Pfaender. Evaluation of the H_2S method for Detection of Fecal Contamination of

Drinking Water. Report No.-WHO/SDE/WSH/02.08. *Water Sanitation and Health Programme, WHO, Geneva, Switzerland* (2002).

e. Online References

These should specify the full URL for reference, please check again to confirm that the work you are citing is still accessible:

9. UNESCO. Global Education Monitoring Report 2024/5: Leadership in education−Lead for learning. *United Nations Educational, Scientific and Cultural Organization, Paris, France* (2024). https://digitallibrary.un.org/record/4066661?ln=en&v=pdf

10. L.M. Highland and P. Bobrowsky. The landslide handbook−A guide to understanding landslides. Circular 1325. *US Geological Survey, Reston, Virginia* (2008).

    https://pubs.usgs.gov/circ/1325/pdf/C1325_508.pdf

f. Conference Proceedings

11. M. Khalid, A.B. Majid, F. Mansour, and C.R. Smith. Word Representations with Recursive Neural Networks for Morphology. *27th European Conference on Signal Processing, (2nd - 6th September 2021), Madrid, Spain* (2021).

g. A Degree Thesis

12. M. Afzal. Investigation of structural and magnetic properties of nanometallic Fe-Mn Alloys. Ph.D. Thesis. *Quaid-i-Azam University, Islamabad, Pakistan* (2023).

Tables: Insert all tables as editable text, not as images. Number tables consecutively following their appearance in the text. A concise but self-explanatory heading must be given. Tables should be numbered according to the order of citation (like **Table 1.**, **Table 2.** (font size 10)). *Do not* abbreviate the word "Table" to "Tab.". Round off data to the nearest three significant digits. Provide essential explanatory footnotes, with superscript letters or symbols keyed to the data. Do not use vertical or horizontal lines, except for separating column heads from the data and at the end of the Table.

Figures: In the main text write Figure, not Fig. Figures may be printed in two sizes: column width of 8.0 cm or page width of 16.5 cm; In the Figure caption, number them as **Fig. 1.**, **Fig. 2.** Captions to Figures must be concise but self-explanatory (font size 10). Laser-printed line drawings are acceptable. Do not use lettering smaller than 9 points or unnecessarily large. Photographs must be of high quality. A scale bar should be provided on all photomicrographs. All Figures should have sufficiently high resolution (minimum 300 dpi) to enhance the readability. Figures as separate files in JPG or TIFF format may be provided.

SUBMISSION CHECKLIST

The following list will be useful during the final checking of an article before submission to the journal.

1. Manuscript in MS Word format
2. Cover Letter
3. Novelty Statement
4. Copyright Form
5. Figures in JPG or TIFF format

In case of any difficulty while submitting your manuscript, please get in touch with:

Editor-in-Chief

Pakistan Academy of Sciences
3-Constitution Avenue,
G-5/2, Islamabad, Pakistan
Email: editor@paspk.org
Tel: +92-51-920 7140
Websites: http://www.paspk.org/proceedings/; http://ppaspk.org/

PROCEEDINGS OF THE
PAKISTAN ACADEMY OF SCIENCES: PART A
Physical and Computational Sciences

CONTENTS

Volume 61, No. 3, September 2024 **Page**

Review Article

Research Articles

PAKISTAN ACADEMY OF SCIENCES, ISLAMABAD, PAKISTAN

HEC Recognized; Scopus Indexed

Websites: http://www.paspk.org/proceedings/; http://ppaspk.org