In the Name of God

Journal of
Information Systems & Telecommunication

Vol. 12, No.1, January-March 2024, Serial Number 45

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

Manager-in-Charge: Dr. Habibollah Asghari, ACECR, Iran
Editor-in-Chief: Dr**.** Masoud Shafiee, Amir Kabir University of Technology, Iran

Executive Editor: Dr. Fatemeh Kheirkhah
Executive Manager: Shirin Gilaki
Executive Assistants: Mahdokht Ghahari, Ali BoozarPoor
Print ISSN: 2322-1437
Online ISSN: 2345-2773
Publication License: 91/13216
Editorial Office Address: No.5, Saeedi Alley, Kalej Intersection., Enghelab Ave., Tehran, Iran,
P.O.Box: 13145-799 Tel: (+9821) 88930150 Fax: (+9821) 88930157
E-mail: info@jist.ir , infojist@gmail.com
URL: www.jist.ir

Indexed by:

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

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

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

Acknowledgement

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

(A-Z)

- Afsharirad, Majid, Kharazmi University,Tehran, Iran
- Azizi, Sadoon, University of Kurdistan, Sanandaj, Iran
- Badie, Kambiz, Tehran University, Iran
- Ebadati, Omid Mahdi, Kharazmi University,Tehran, Iran
- Fathi, Amir, Urmia University, Urmia, Iran
- Farsi, Hassan, University of Birjand, South Khorasan, Iran
- Ghasemzadeh, Ardalan, Urmia University of Technology, West Azerbaijan, Iran
- Ghazalian, Reza, Babol Noshirvani University of Technology, Mazandaran, Iran
- Ghaffari, Hamidreza, Ferdous Azad University, South Khorasan Province, Iran
- Izadkhah, Habib, Tabriz University, Tabriz, Iran
- Jampoor, Mehdi, Quchan University of Technology, Razavi Khorasan, Iran
- Kumar Debnath, Sumit, National Institute of Technology Jamshedpur, India
- Mirzaei, Abbas, Islamic Azad University, Ardabil, Iran
- Mahdieh, Omid, University of Zanjan, Zanjan, Iran
- Mansoorizadeh, Muharram, Bu-Ali Sina University, Hamedan, Iran
- Munesh, Singh, Indian Institute of Information Technology Design & Manufacturing Jabalpur, India
- Ramezani, Reza, Isfahan University, Isfahan, Iran
- Rasi, Habib, Shiraz University of Technology, Shiraz, Iran
- Rahman, Saifur, Virginia Tech,Riva San Vitale, Virginia Tech Steger Center for International Scholarship Switzerland
- Rizal, Achmad, Telkom University, Bandung , Indonesia
- Roshani, Saeid, Islamic Azad University, Kermanshah Branch, Iran
- Sedghi, Shahram, Iran University of Medical Sciences, Tehran, Iran
- Shakibian, Hadi, Alzahra University, Tehran, Iran
- Soleimanian Gharehchopogh, Farhad, Islamic Azad University Urmia, Iran
- Tourani, Mahdi, University of Birjand, South Khorasan, Iran
- Tanha, Jafar, Tabriz University, Tabriz, Iran
- Yaghoobi, Kaebeh, Ale Taha Institute of Higher Education, Tehran, Iran

Table of Contents

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 1-11

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

An Efficient Sentiment Analysis Model for Crime Articles′ Comments using a Fine-tuned BERT Deep Architecture and Pre-Processing Techniques

Sovon Chakraborty[1], Muhammad Borhan Uddin Talukdar[2], Portia Sikder[3], Jia Uddin[4*]

[1.]Department of Computer science and Engineering, University of Liberal Arts Bangladesh, Dhaka, Bangladesh,
[2.]Department of Computer Science and Engineering, Daffodil International University, Savar, Bangladesh,
[3.]Department of Computer Science and Engineering, North Western University, Khulna, Bangladesh.
[4.]Department of AI and Big Data, Woosong University, Daejeon, South Korea

Received: 3 Jul 2022/ Revised: 10 Oct 2022/ Accepted: 15 Nov 2022

Abstract

The prevalence of social media these days allows users to exchange views on a multitude of events. Public comments on the talk-of-the-country crimes can be analyzed to understand how the overall mass sentiment changes over time. In this paper, a specialized dataset has been developed and utilized, comprising public comments from various types of online platforms, about contemporary crime events. The comments are later manually annotated with one of the three polarity values- positive, negative, and neutral. Before feeding the model with the data, some pre-processing tasks are applied to eliminate the dispensable parts each comment contains. In this study, A deep Bidirectional Encoder Representation from Transformers (BERT) is utilized for sentiment analysis from the pre-processed crime data. In order the evaluate the performance that the model exhibits, F1 score, ROC curve, and Heatmap are used. Experimental results demonstrate that the model shows F1 Score of 89% for the tested dataset. In addition, the proposed model outperforms the other state-of-the-art machine learning and deep learning models by exhibiting higher accuracy with less trainable parameters. As the model requires less trainable parameters, and hence the complexity is lower compared to other models, it is expected that the proposed model may be a suitable option for utilization in portable IoT devices.

Keywords: BERT; BNLP; NLP; Sentiment Analysis; Bangla Sentiment Analysis

1- Introduction

An illegitimate practice punishable by law is known as a crime [1]. It has always been one of the most baffling problems around. Bangladesh has observed some talk-of-the-country crime incidents in the last few years that affected many people's lives. As the population increases with time, so does the number of reported crimes [2]. However, it turns out that the real-life crimes are not the only concern these days. Cybercrimes have also emerged as a source of great terror [3]. Along with murder, snatching, dacoity, and other crimes in day-to-day life, crimes committed using the internet are more frequent than ever. Being one of the most indispensable sectors worldwide, crimes in the E-commerce industry have also started to take place in Bangladesh. E-commerce frauds have plundered approximately 1000 Crores BDT from customers in the last couple of years [4]. A significant number of people have become destitute as a result. More often than not, people usually discuss and express their feelings on social media regarding these crimes. As a result, social media content has become a substantial source for gathering public sentiments from and analyzing these unfortunate events later on. Extensive research has been conducted lately using Twitter data related to crimes [5-8]. Despite different Machine Learning and Deep learning architectures being trained on the collected data to serve various purposes, to the best of our knowledge, yet no research has been performed on sentiment analysis using Bengali crime data. Hence, the authors put their optimum effort to contribute to this area. This study analyzes public sentiments regarding various crimes in recent times in Bangladesh. Data used in this study were collected from

✉ **Jia Uddin**
jia.uddin@wsu.ac.kr

the comment sections of various sources, including Facebook pages, online news portals, and YouTube videos. All data are accessible in the Bengali Language, and each instance is associated with either of three polarity values. Although the literature includes many endeavors in this area, Section 2 addresses only the most relevant ones to this work. Section 3 demonstrates the proposed method in pictorial form and further elaborates it. Other significant matters are also addressed. Section 4 shows and analyzes the results using effective techniques. The proposed model is also compared with other contemporary machine learning and deep learning models with higher performance. The conclusion and future research scopes regarding this work have been stated in Section 5.

1-1- Motivation

Deep learning models have been widely used in recent times for their outstanding performances in text analysis. Among the popular deep learning architectures, BERT has gained the attention of many contemporary research works regarding text classification and sentiment analysis [12-17]. A plethora of sentiment analysis works is present on datasets of different languages that exploited the BERT model to achieve the maximum outcome. The model, however, has been barely used in the case of Bengali sentiment analysis so far. Being motivated by the work of Rahman *et. al.* [12], we utilized the BERT model for sentiment analysis of public comments regarding recent crimes.

1-2 - Contributions

The main contribution of this paper can be summarized as follows:

1. A specialized dataset is built that comprises 5000 public comments regarding different crime incidents. The comments are then critically inspected to annotate with the respective polarity value.
2. A BERT-based architecture is proposed for analyzing public sentiments given the public comments. The hyperparameters of the architecture are later tuned for maximum utility.
3. The architecture turned out to yield the highest performance among other state-of-the-art models.
4. This research can be helpful in understanding the pattern in which most people are currently reacting to these kinds of incidents and the changes in this pattern over time. The findings can further be utilized for Social Science and Behavioral Science research to gain a deep insight into how people tend to react over time, where crime is ever-increasing and justice is delayed. Furthermore, the higher accuracy and the smaller number of trainable parameters make this model a good candidate to be utilized in computing devices.

2 - Literature Review

An enormous amount of research has been performed for analyzing public Sentiments from different types of data. During the Covid pandemic, extensive experiments have been regulated to understand people's sentiments worldwide. Sharmat et al. performed sentiment analysis based on Twitter data about individuals' concerns about early introduced Covid vaccines [7]. The researchers just analyzed and categorized the data in terms of Positive, Negative, and Neutral for three early introduced Covid-19 vaccines: Moderna, Pfizer, AstraZenaca. The work was confined to analyzing the collected data and discussing the outcome of polarity values, no prediction of any future outcome was attempted.

Wei Li *et al.* proposed a novel padding method for making the input data of a consistent size and making each review more useful by improving the proportion of sentiment information [9]. The researchers used two-channel CNN-LSTM and CNN-BiLSTM to predict underlying sentiments of user reviews, where several datasets were used from different sources. Although their model worked great on the Chinese Tourism Dataset, the result was not that impressive for Stanford Sentiment Treebank. Jia *et al.* applied sentiment analysis in cryptocurrency fluctuations [10]. They took customer reviews in text format and looked for reviews with positive and negative Polarity. Why the researchers did not address neutral reviews, is a matter of question here. Hemel *et al.* performed sentiment analysis based on cryptocurrency price data on individuals' concerns for early Bitcoin price prediction [11].

Rahman *et al.* used transformer-based deep learning models for Bengali text documents classification [12]. They used two popular deep learning models- BERT and ELECTRA. Three publicly available datasets were employed, which collectively contained 13 unique labels for the text documents to be classified into. Out of three datasets, the models performed very well for two datasets, but the result was unsatisfactory for the remaining one. Mrityunjay *et al.* used the BERT model on tweets composed of sentiment analysis datasets. They employed two different tweet datasets. The first one contained tweets written by people worldwide, and the second one was confined to tweets made only by the Indian population [13]. Three target-dependent variations of the BERT model were implemented by Zhengjie *et al.* Experiments with three different datasets showed that their target-dependent model performs significantly better than some commonly used others [14]. The model, however, cannot render satisfactory results for specific categories of data, and the classification accuracy is lower in that case(s). Chi *et al.* exploited the BERT model for aspect-based sentiment analysis [15]. They first formed auxiliary texts from different aspects, and then the aspect-based sentiment analysis task was converted to a sentence-pair classification task. They used a pre-trained BERT model, which was tweaked and made to produce excellent results on SemEval2013 Task 4 and SentiHood datasets.

Song *et al.* proposed a BERT-based sentiment analysis algorithm for Chinese e-commerce reviews. Apart from the outstanding performance of their proposed model, the model suffers from a major drawback. A small dataset was used in this study, and the researcher did not verify their model's strength on a larger dataset [16]. Xin *et al.* examined the efficiency of the BERT model for embedding components on End-to-End Aspect-based sentiment analysis [17]. The researchers performed experiments with BERT coupled with various neural models on two datasets. It turned out that their proposed BERT-based model for E2E-ABSA outperforms state-of-the-art works.

Although the literature shows an extensive use and a huge success of the BERT model in the field of Natural Language Processing, the model has not been employed for Bengali sentiment analysis research. After analyzing all the related research, an attempt has been made by the authors of this paper to analyze and classify public sentiments using a fine-tuned proposed BERT deep architecture. All the data used in this study are based on the recent talk-of-the-country crimes occurred in Bangladesh**.**

3 - Proposed Methodology

The initial focus of the researchers was to identify mass reactions regarding crimes in terms of positive, negative, and neutral. To do so, data were first outsourced to various sources. Data are structured in Bengali and contain emojis and other dispensable parts. All the punctuations, special characters, digits, and stop words are removed first. Thenceforth, all the raw data are then tokenized for the development of context. Examining the sequence of words and interpreting them is another primacy of tokenization. Lemmatization is then applied to remove suffixes from each word and to make them shorter by restoring them to their root forms. Word embedding techniques are finally employed, and the data is ready for action.
The researchers propose a BERT-based model for analyzing sentiments and classifying the comments as positive, negative, or neutral. The performance of the model is evaluated using some famous and widely used performance metrics, namely- ROC Curve, F1-Score, and Heat Map.

The result obtained from the proposed model has been later compared with state-of-the-art models of this endeavor. The comparison is shown in detail in the result analysis section.

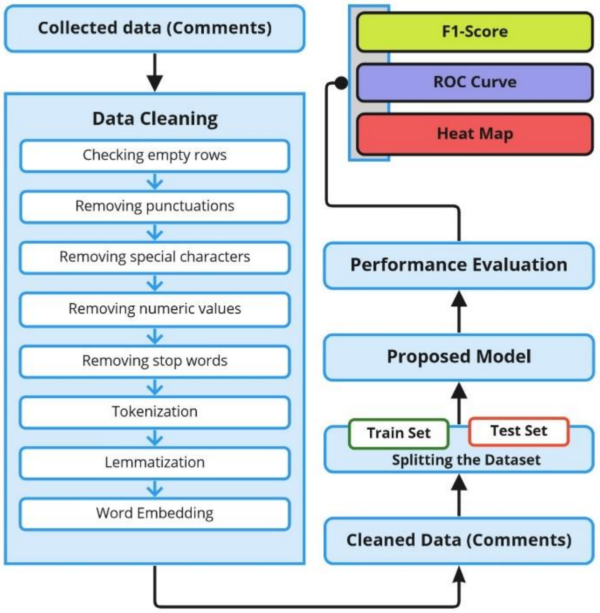


Fig. 1. Proposed Methodology

3-1 - Dataset Description

To achieve the benchmark of a developed country, Bangladesh exerts much effort into developing all sectors. While the infrastructural development curve is moving upward, some contemporary events like E-commerce fraud, murder, forced disappearances, burglary, and rape on moving public transport, on the other hand, seem to counterbalance the development. These recurrent crimes are leaving some apparent marks on the public mind. Social media, nowadays, serves as a reflection of what is going on in people's minds. When people bump into such events in the news feed, they usually share their reactions in the comment sections. In this paper, the researchers are focused on drawing an overall public consensus about these contemporary crime events.

Along the way to do so, a dataset is built, which is later analyzed to understand what most people are feeling given a crime article. This paper considers three polarities for public reactions: positive, negative, and neutral.

The dataset comprises 5000 comments, which took a month for the authors to gather and prepare the data. The authors targeted the comments sections of Bengali crime articles corresponding Facebook pages redirect the readers to a webpage where the original article resides or to a YouTube news report. Hence, some data comes from the comments section of a news article or a YouTube news report as well. Data sources, their corresponding reference URLs, and the number of followers is shown in Table 1.

The dominant reasons for selecting these sources are mentioned below.

1. The sources belong to the mainstream media of Bangladesh
2. They have gained a massive audience over time.
3. Many readers frequently interact with the articles in the comment section.
4. The audience tend to express their opinions when encountering a crime article from these above-mentioned sources.

Table 1: Data Collection Sources, Reference URLs, and Number of Followers

Sources	Reference Link	Number of Followers (in million)
The Prothom Alo	https://www.facebook.com/DailyProthomAlo	18
BBC Bangla	https://www.facebook.com/BBCBengaliService	14
Independent TV	https://www.facebook.com/independent24Television	8.2
Ekattor TV	https://www.facebook.com/ekattor.tv	5.4
The Daily Star	https://www.facebook.com/dailystarnews	3.4

Samples from the collected data are represented in Table 2.

Table 2: Sample comments of some contemporary crime events picked from the dataset

Comments	Source Information
একজনের জন্য হাজারো জন কলঙ্কিত হয়	Collected from the comment section of The Prothom Alo facebook page
পুলিশ সদস্যরা হলেন আমাদের প্রিয় বন্ধু	Collected from the comment section of The BBC Bangla facebook page
ঘটনার তদন্ত করা হোক কঠিনভাবে	Collected from the News article of The Prothom Alo
ব্যভিচারের জন্য কঠোর সাজার আইন করা দরকার	Collected from the comment section of The Ekattor TV web portal
বিষয়টি সঠিক তদন্তের দাবি জানাচ্ছি	Collected from the comment section of The Daily Star web portal

This dataset focuses on gathering comments written in Bangla using Bengali Alphabet. Approximately 20% of commenters have been found to express their viewpoint using the English Alphabet. Moreover, most of this 20% uses transliteration, where the English Alphabet is used to write in Bengali. These types of comments were not included in the dataset so that ambiguity can be avoided. Emoticons associated with the comments were preprocessed after collecting the whole dataset.

3-2- Annotation of the Collected Dataset

Annotation of a dataset is usually done for helping the NLP models to understand the key phrases that lie within a comment, accompanying the determination of the parts of speech of the comment data [18]. The annotation of the data was done cooperatively by seven students of the European University of Bangladesh and the authors. Each comment received votes to be labeled as either Positive, Negative, or Neutral. The polarity value of a particular comment was determined according to the votes count for each label.

The label with the highest votes count is accepted as the polarity value for a particular comment. Later on, the whole dataset is validated by two European University of Bangladesh scholars.

The following comment is collected from the comment section of the Prothom Alo online news portal-

“ধন্যবাদ জানাচ্ছি বাংলাদেশ পুলিশ বাহিনীকে”

This comment is then provided to the 7 participants, and their evaluation is recorded. The result is displayed in Table 3.

Table 3: Explanation of Voting Method for determining the polarity

Participant ID	Participant’s Gender	Cast Polarity
1	Male	Positive
2	Female	Positive
3	Male	Neutral
4	Male	Positive
5	Female	Neutral
6	Male	Positive
7	Male	Positive

Table 3 clearly shows the polarity determination process of a specific comment. The comment mentioned in the table received five votes for Positive, and the rest two votes were cast for Neutral. The Negative Polarity receives no votes. Since most votes advocate Positive Polarity, the final polarity value determined for this comment is Positive. The same procedure is followed for determining the Polarity of all comments. An odd number of participants (seven) is taken to eliminate the chances of a tie. The total comments count is 5000. Table 4 demonstrates the number of comments according to each polarity type.

Table 4: Frequency Count of each polarity in the dataset

Polarity	Amount of Data
Positive	1645
Negative	1750
Neutral	1605
Total	**5000**

Table 5 demonstrates how the polarity values are represented in the final dataset. The researchers employ a technique here termed One-hot encoding. This technique transforms the categorical polarity values into vectors of 0s and 1s. The length of these vectors depends on the number of categories that we want our dataset to be classified into. Since the dataset has three polarity values, hence the length of the vectors is supposed to be 3.

The Positive Polarity corresponds to the first element, the Negative corresponds to the second, and the Neutral corresponds to the third. Each of the labels having its position in the vectors contributes to either a zero or a one according to the category it falls into. If the Polarity of a comment is negative, the second position in the vector will contribute to a one, producing two 0s for the remaining two positions. So, the vector looks like [0, 1, 0]. Since the vector holds a one only in the position for which the corresponding polarity value is true, hence the technique is called One-hot.

Table 5: One-hot encoding for the representation of polarity values

Comment	Positive	Negative	Neutral
স্যালুট জানাই ৫ সদস্যের ঐ পুলিশ টিমকে।	1.0	0.0	0.0
সত্যের জয় হয় সবসময়।	1.0	0.0	0.0
মানুষ কতটা নিচে নামলে এমন কাজ করতে পারে!	0.0	1.0	0.0
সুষ্ঠু তদন্ত করে আইনানুগ ব্যবস্থা নেয়া হউক।	0.0	0.0	1.0
মানুষ দুনিয়াতে চিরস্থায়ী নয়	0.0	0.0	1.0
মাদক ব্যবসায়ী এবং মাদক সেবনকারীদের সাথে এর চাইতে ভালো ব্যবহার হয় না।	0.0	1.0	0.0
গরিব-দুঃখীদের জন্য কোনো আইন নাই	0.0	1.0	0.0

Fig. 2 demonstrates the outcome of applying Zipf's law [18] to the dataset. Zipf's law can be instrumental in observing the words under the light of their frequency counts. The rank of a particular word in the corpus should be inversely proportional to the word's frequency count. From this figure, it can be observed that the most common word has an occurrence of 823 times. The second data has appeared 802 times, and so on. The correlation among the data can also be measured by using Intraclass Correlation Coefficients (ICC). The value of the ICC for this dataset is approximately 0.74. ICC is regarded as a quantitative measurement of the units that are well organized inside the dataset.

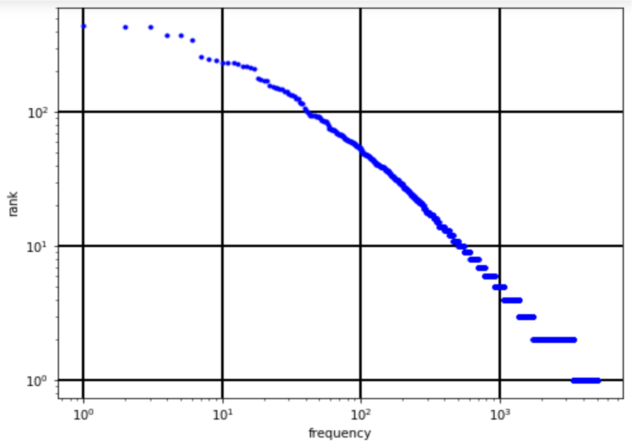


Fig. 2: The dataset under the light of Zipf’s law

ICC is also used for the appraisal of the equilibrium of available data. Principally a value of more than 0.7 is considered excellent reliability of data. The formula is stated as,

$$\boldsymbol{ICC} = \frac{\boldsymbol{Variance\ of\ interest}}{\boldsymbol{Total\ variance}} \qquad (1)$$

3-3 - Data Preprocessing

When the data collection was over, the dataset was first checked to see whether any empty row existed or not. All the empty rows are discarded from the dataset to deal with the missing values. Later on, the punctuations, special characters, numeric values, and stop words available in a text are replaced by blank spaces. The example below demonstrates how a text is transformed across the journey.

“স্যালুট জানাই ৫ সদস্যের ঐ পুলিশ টিমকে। ”

The above sentence incorporates punctuations. A space replaces punctuations. After the removal of punctuations, the sentence comes into the following shape.

“স্যালুট জানাই ৫ সদস্যের ঐ পুলিশ টিমকে”

Special characters, numeric values, and emojis are strenuous for machines to understand. A space also replaces them. The preprocessing is done with the help of BNLP and NLTK toolkit available in Python programming language.

“স্যালুট জানাই সদস্যের ঐ পুলিশ টিমকে”

Stop words from this sentence are also required to be taken care of. After the removal of stop words, the sentence can be rewritten as follows:

“স্যালুট সদস্যের পুলিশ টিমকে”

The sentence is divided into words, called tokens, using a process called tokenization. The sentence is then divided into the following tokens.

[‘স্যালুট’, ‘সদস্যের’, ‘পুলিশ’, ‘টিমকে’]

Lemmatization is applied for data normalization. This operation is performed using the NLTK toolkit [20]. Lemmatization is the process of converting the data into its root form from the dictionary, called a lemma [21]. For example, lemmatization turns the word "বাংলাদেশের" into "বাংলাদেশ".

After performing Lemmatization, the following lemmas are the outcome.

[‘স্যালুট’, ‘সদস্য’, ‘পুলিশ’, ‘টিম’]

Word Embedding is then performed to represent the words in lower dimension space. This process converts all the texts and documents into numerical space. Later on, the data is fed to the proposed model.

3-4 - The Proposed Model

BERT is a deep learning model where all the inputs are connected to all the outputs using transformers to determine higher accuracy. The weights associated with the hidden layer nodes are randomly allocated and constantly improved. BERT is widely used for masked language modeling and prediction for the next sentence. In this article, the authors propose a BERT-based architecture for sentiment analysis. After the cleaning of raw data, the transformers are fed with formatted data. The transformer is a model that can predict data in any order. Unlike other deep learning models, transformers can train vast amounts of data in a tiny amount of time. The base layer of the transformer is frozen. Three trainable layers have been added. The model is applied after proper parameter tuning. Figure 3 shows how a comment is processed along the way to receiving a polarity value using the proposed model. ELECTRA is a method that can be used to pre-train transformer networks. ELECTRA models are trained to discriminate between actual input tokens and fake input tokens. They contain two transformer models: the generator and the discriminator, similar to the Generative Adversarial Networks (GANs).

We have used Keras sequential API, BNLP and NLTK toolkits for the implementation of this research. BNLP toolkit is mainly used for the tokenization of texts in the Bengali language. Two of the major functionalities of BNLP includes constructing neural model and embedding Bengali words. NLTK toolkit is a python package that is used to make human language usable to computers. Once the preprocessing is done, the cleaned data is passed on to the proposed BERT-based model.

The model begins with 256 transformer layers. The transformer architecture consists of an encoder and a decoder. Multiple identical layers are attached to the encoder. Each layer incorporates two sublayers. The sublayers are denoted as:

i) Multi-head Self-attention pooling
ii) Position-wise feed-forward network.

For attention mechanism that runs several times in parallel for achieving the highest accuracy. The outputs are independent and finally concatenated and linearly transformed into an expected dimension. Two dense layers are available, known as position-wise-feed-forward networks, as the outputs are placed in a sequence. This is applied in the last dimension [24].

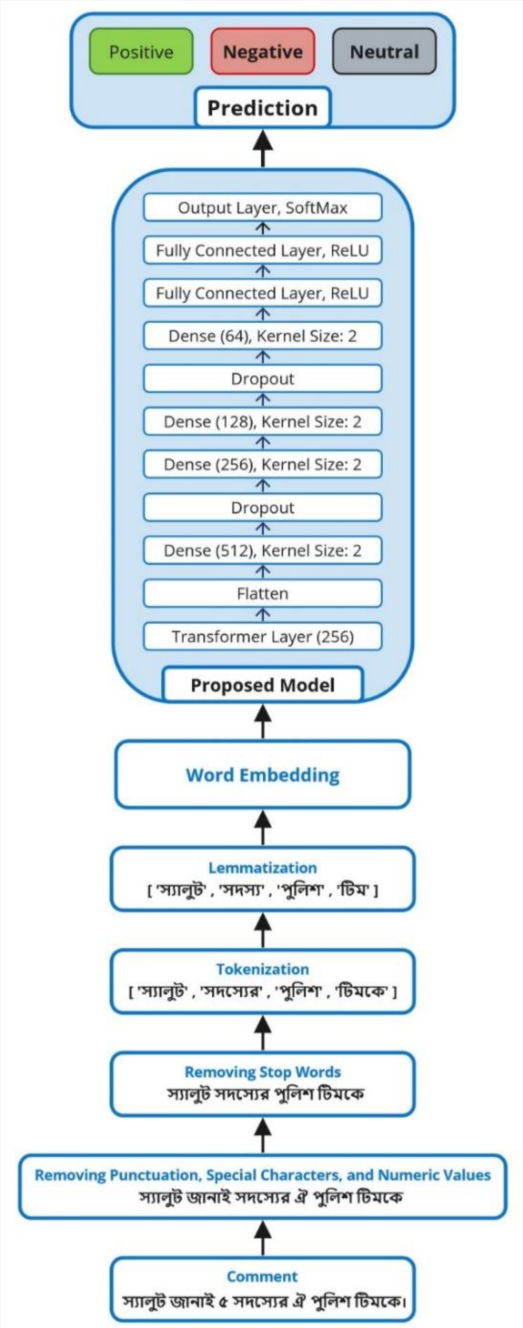

Positive
Negative
Neutral
Prediction
Output Layer, SoftMax
Fully Connected Layer, ReLU
Fully Connected Layer, ReLU
Dense (64), Kernel Size: 2
Dropout
Dense (128), Kernel Size: 2
Dense (256), Kernel Size: 2
Dropout
Dense (512), Kernel Size: 2
Flatten
Transformer Layer (256)
Proposed Model
Word Embedding
Lemmatization
['স্যালুট' , 'সদস্য' , 'পুলিশ' , 'টিম']
Tokenization
['স্যালুট' , 'সদস্যের' , 'পুলিশ' , 'টিমকে']
Removing Stop Words
স্যালুট সদস্যের পুলিশ টিমকে
Removing Punctuation, Special Characters, and Numeric Values
স্যালুট জানাই সদস্যের ঐ পুলিশ টিমকে
Comment
স্যালুট জানাই ৫ সদস্যের ঐ পুলিশ টিমকে।

Fig. 3 Employing the Proposed Model to predict the Polarity of a comment

There is a residual connection provided around both sublayers. In the case of a decoder, along with two sublayers, there is another sublayer added. Individual position in the decoder ensures the prediction. The prediction depends on output tokens that have been generated. In this research, 256 transformer layers are deployed. All the data are then converted into a 1D array. This task is performed in the flatten layer. The main purpose of flattening is to convert multi-dimensional data into one-dimensional data. Multi-dimensional data are arduous and costly to perform any mathematical operations.

Table 6: Parameters details of the Proposed Model

Hyperparameters	Values
Batch Size	32
Learning Rate	0.00001
Number of Epochs	10
Decay	0.001
Optimizer	Adam, Stochastic Gradient Descent
Loss Function	Binary Cross-Entropy
Number of hidden states	724
Number of transformer blocks	12
Activation Function	ReLU, Softmax
Number of trainable parameters	17,342

After the flatten layer, individual data is passed on to a dense layer that consists of 512 hidden layers. For preventing the model from encountering overfitting, dropout is applied after the dense layer. Data is passed on to another two dense layers where the numbers of hidden layers are consecutively 256 and 128. Now data dropout is applied again, and the data is fed to another dense layer. The kernel size 2 is maintained in each dense layer. Finally, two fully connected layers are deployed in the final stage of the model which leads to an output layer. In the fully connected layers, ReLU activation function is applied where at the output layer Softmax activation function has been used by the researchers. The hyperparametric details of the proposed model are shown in Table 6.

3-5- Evaluation Metrics

The performance of the proposed model is evaluated using F1-Score, ROC Curve, and Heat Map. Once the model is trained properly, the above-mentioned techniques are used to evaluate the performance of the model.

The ROC Curve has long been used in medical decision-making and evaluating the performances of different machine learning algorithms [22]. A Heat Map is a two-dimensional data visualization technique where colors are used to represent different values. Heat Map depicts values in a matrix of a fixed dimension. The authors have used Cluster Heat Map for their interest [23]. Finally, the formula to measure F1-Score is stated below-

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad [2]$$

4 - Result Analysis

The F1-Score rendered by the proposed architecture is shown in Table 7 for each Polarity.

Table 6 demonstrates that the proposed model has achieved the highest score in the case of positive Polarity. Conversely, the model gets confused while identifying some of the negative and neutral polarity values. Sometimes the negative and the neutral comments are difficult for the model to discriminate between. For example-

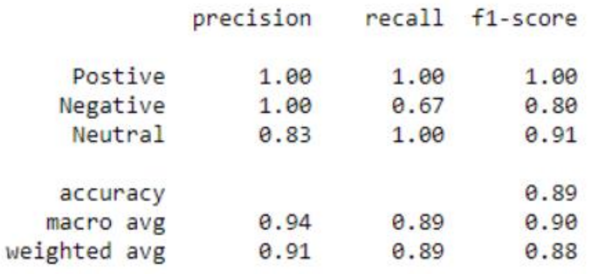
“জীবনে অনেক কষ্ট করতে হয়”

The comment mentioned above can be categorized as both negative and neutral. In such cases, the model gets confused and shows less accuracy in identifying the proper Polarity.

Table 7: F1- Score Analysis for each Polarity

Name of the Metric	Positive	Negative	Neutral
F1-Score	100%	80.23%	89.71%

Fig. 4 demonstrates how good the proposed model is at the classification task. Although the model performs just perfectly in the case of the Positive Polarity, the performance regarding the Negative and Neutral polarities is not that great. The reason why the model performs poorer for Neutral polarity, has been discussed earlier.



```
              precision    recall  f1-score

     Postive       1.00      1.00      1.00
    Negative       1.00      0.67      0.80
     Neutral       0.83      1.00      0.91

    accuracy                           0.89
   macro avg       0.94      0.89      0.90
weighted avg       0.91      0.89      0.88
```

Fig 4: Classification Summary

Fig. 5 demonstrates the changes in Accuracy Percentiles, for both the training stage and the validation stage, with the rising number of epochs. The accuracy tends to increase with the number of epochs. The maximum accuracy for both training and validation stages is recorded at epoch 10.

The performance of the proposed model is presented using a Heat Map in Fig. 6. The three target polarity classes are represented with numeric values. ‘0’ represents the positive Polarity**,** ‘1’ represents the negative Polarity and ‘2’ denotes the neutral Polarity. Taking all validation data into account, it is apparent that the model recognized all the positive polarity data properly. While 2 data from negative Polarity and 8 data from neutral Polarity are not properly identified by the proposed model.

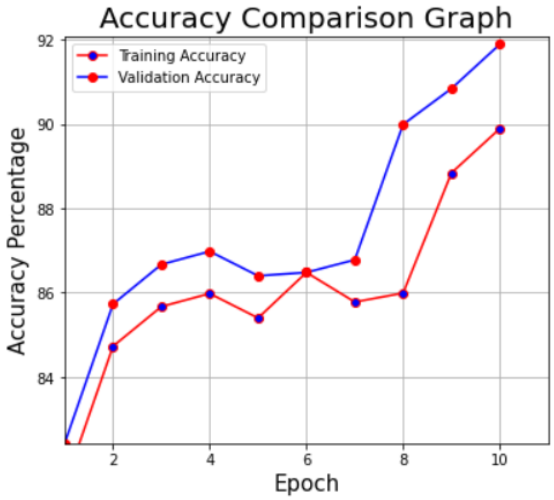


Fig. 5: Accuracy Comparison Graph for Training and Validation Phase

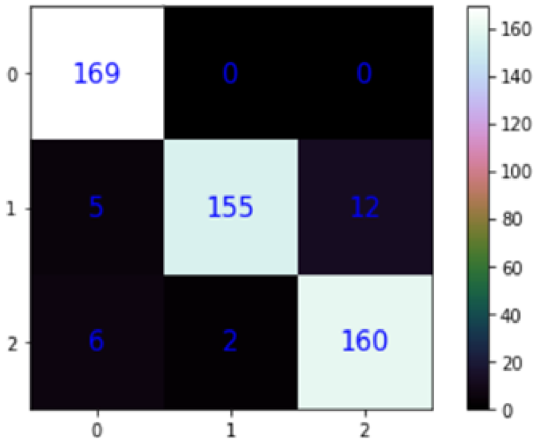


Fig. 6: Heat map for the proposed model

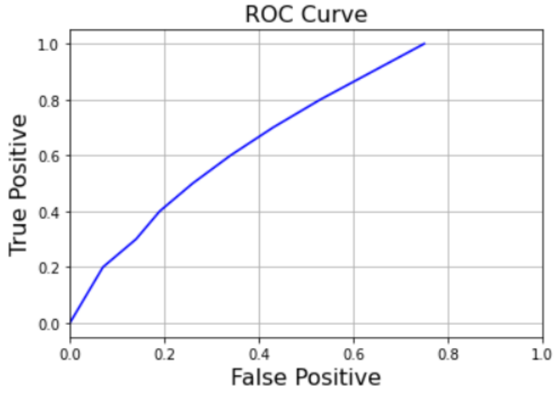


Fig 7: ROC curve of the proposed model

Fig. 7 demonstrates the ROC curve for the proposed model. The false-positive rate for a particular epoch is plotted along the X-axis, whereas the true positive rate is plotted along the Y-axis. A model is said to classify well if the curve is positioned on the top left side. From Figure 7, it can be observed that the proposed BERT-based model can classify the True positive values correctly.

4 -1- Performance Comparison

Fig. 9 compares the performance of the proposed model with other existing machine learning models. The model achieved an F1 score of 89% whereas the second and third maximum figures were 85% and 77 % in the case of machine learning models. The figure shows the other state-of-the-art results rendered by using Maximum Entropy, Random Forests, and K-nearest neighbors.

Sentiment Analysis in the Bengali language using deep learning models received very few research attempts. Fig. 10 compares the accuracy of the proposed model with other deep learning models [26] employed for sentiment analysis, and achieved the maximum accuracy. It turns out that the proposed model yields the highest accuracy, whereas the second maximum accuracy achieved was 88%.

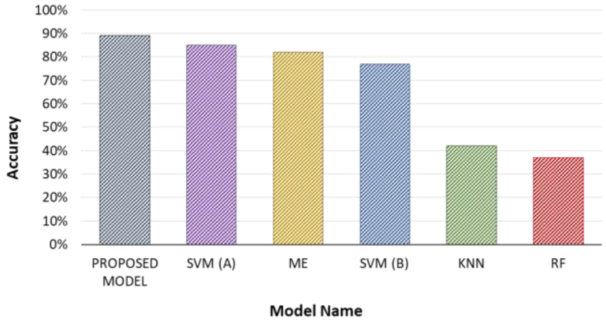


Fig. 9 Accuracy comparison of the proposed model with state-of-the-art machine learning models

Fig. 10 demonstrates another comparative analysis of the proposed model with other deep learning-based models proposed in other research articles. It shows that the proposed model beats other deep learning models proposed by Munna *et al.* [27].

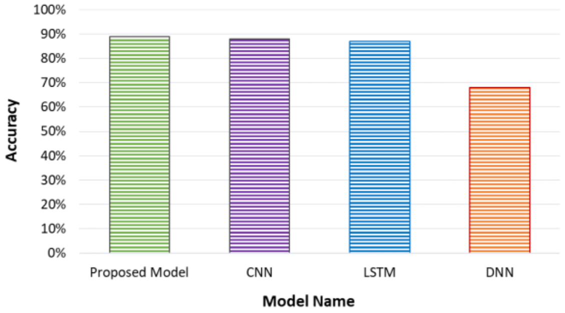


Fig. 10 Accuracy comparison of the proposed model with existing deep learning models

From the previous studies [27-29] addressed in Fig.10, it is evident that the proposed model shows higher validation accuracy.

Fig. 11 shows how good the proposed model is in terms of computational complexities. The number of trainable parameters for the proposed model is the lowest among the other state-of-the-art architectures, yet the accuracy achieved is the highest.

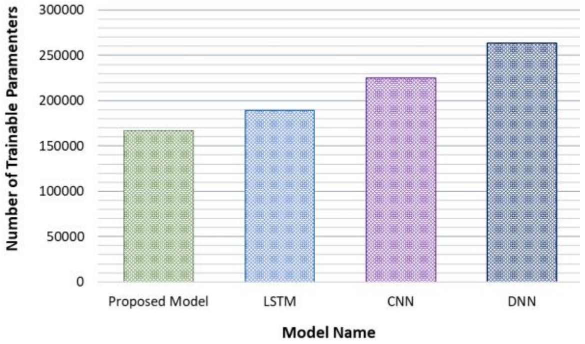


Fig. 11: Comparison of the proposed model with other deep learning architectures in terms of trainable parameters.

5 - Conclusion and Future work

This paper presents a BERT-based model to analyze the public sentiment towards recent crimes in Bangladesh. In this study, the model has validated with a specialized dataset, which consists of 5000 comments from various online portals. Once the model is trained, it works satisfactorily for all polarity types with an average of 89% F1-score. The proposed model is compared with other state-of-the-art deep learning models and turned out to produce the better outcomes. In the near future, the authors aim to exploit the model in sentiment analysis from public comments related to recent price hikes and inflation in Bangladesh.

Acknowledgement: This research is funded by Woosong University Academic Research in 2024.

References

[1] S. R. Bandekar and C. Vijayalakshmi, "Design and Analysis of Machine Learning Algorithms for the reduction of crime rates in India," Procedia Computer Science, vol. 172. Elsevier BV, pp. 122–127, 2020. doi: 10.1016/j.procs.2020.05.018.

[2] M. Pavel Rahman, A. K. M. Ifranul Hoque, Md. Faysal Ahmed, I. Iftekhirul, A. Alam, and N. Hossain, "Bangladesh Crime Reports Analysis and Prediction," 2021 International Conference on Software Engineering &amp; Computer Systems and 4th International Conference on Computational Science and Information Management (ICSECS-ICOCSIM). IEEE, Aug. 2021. doi: 10.1109/icsecs52883.2021.00089.

[3] A. H. Mohd Hanif, N. Maarop, N. Kamaruddin, and G. N. Samy, "Machine Learning Approach in Predicting Fraudulent Job Advertisement," International Journal of Academic Research in Business and Social Sciences, vol. 14, no. 1. Human Resources Management Academic Research Society (HRMARS), Jan. 12, 2024. doi: 10.6007/ijarbss/v14-i1/20532

[4] A. Alzubaidi, "Measuring the level of cyber-security awareness for cybercrime in Saudi Arabia," Heliyon, vol. 7, no. 1. Elsevier BV, p. e06016, Jan. 2021. doi: 10.1016/j.heliyon.2021.e06016.

[5] S. Lal, L. Tiwari, R. Ranjan, A. Verma, N. Sardana, and R. Mourya, "Analysis and Classification of Crime Tweets," Procedia Computer Science, vol. 167. Elsevier BV, pp. 1911–1919, 2020. doi: 10.1016/j.procs.2020.03.211.

[6] A. A. Biswas and S. Basak, "Forecasting the Trends and Patterns of Crime in Bangladesh using Machine Learning Model," 2019 2nd International Conference on Intelligent Communication and Computational Techniques (ICCT). IEEE, Sep. 2019. doi: 10.1109/icct46177.2019.8969031.

[7] F. M. J. Mehedi Shamrat et al., "Sentiment analysis on twitter tweets about COVID-19 vaccines usi ng NLP and supervised KNN classification algorithm," Indonesian Journal of Electrical Engineering and Computer Science, vol. 23, no. 1. Institute of Advanced Engineering and Science, p. 463, Jul. 01, 2021. doi: 10.11591/ijeecs.v23.i1.pp463-470.

[8] S. Aghababaei and M. Makrehchi, "Mining Social Media Content for Crime Prediction," 2016 IEEE/WIC/ACM International Conference on Web Intelligence (WI). IEEE, Oct. 2016. doi: 10.1109/wi.2016.0089.

[9] W. Li, L. Zhu, Y. Shi, K. Guo, and E. Cambria, "User reviews: Sentiment analysis using lexicon integrated two-channel CNN–LSTM family models," Applied Soft Computing, vol. 94. Elsevier BV, p. 106435, Sep. 2020. doi: 10.1016/j.asoc.2020.106435.

[10] Rahman, S., Hemel, J. N., Anta, S. J. A., Al Muhee, H., & Uddin, J. (2018, June). Sentiment analysis using R: An approach to correlate cryptocurrency price fluctuations with change in user sentiment using machine learning. In 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR) (pp. 492-497). IEEE.

[11] S. Rahman, J. N. Hemel, S. J. A. Anta, H. Al Muhee, and J. Uddin, "Sentiment analysis using R: An approach to correlate cryptocurrency price fluctuations with change in user sentiment using machine learning," In Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2nd International Conference on Imaging, Vision & Pattern Recognition (icIVPR), 2018, pp. 492-497.

[12] M. M. Rahman, Md. Aktaruzzaman Pramanik, R. Sadik, M. Roy, and P. Chakraborty, "Bangla Documents Classification using Transformer Based Deep Learning Models," 2020 2nd International Conference on Sustainable Technologies for Industry 4.0 (STI). IEEE, Dec. 19, 2020. doi: 10.1109/sti50764.2020.9350394.

[13] M. Singh, A. K. Jakhar, and S. Pandey, "Sentiment analysis on the impact of coronavirus in social life using the BERT model," Social Network Analysis and Mining, vol. 11, no. 1. Springer Science and Business Media LLC, Mar. 19, 2021. doi: 10.1007/s13278-021-00737-z.

[14] Z. Gao, A. Feng, X. Song, and X. Wu, "Target-Dependent Sentiment Classification With BERT," IEEE Access, vol. 7. Institute of Electrical and Electronics Engineers (IEEE), pp. 154290–154299, 2019. doi: 10.1109/access.2019.2946594.

[15] C. Sun, L. Huang, and X. Qiu, "Utilizing," Proceedings of the 2019 Conference of the North. Association for Computational Linguistics, 2019. doi: 10.18653/v1/n19-1035.

[16] S. Xie, J. Cao, Z. Wu, K. Liu, X. Tao, and H. Xie, "Sentiment Analysis of Chinese E-commerce Reviews Based on BERT," 2020 IEEE 18th International Conference on Industrial Informatics (INDIN). IEEE, Jul. 20, 2020. doi: 10.1109/indin45582.2020.9442190.

[17] Biswas, A., Chakraborty, S., Rifat, A. N. M. Y., Chowdhury, N. F., & Uddin, J. (2020, August). Comparative Analysis of Dimension Reduction Techniques Over Classification Algorithms for Speech Emotion Recognition. In International Conference for Emerging Technologies in Computing (pp. 170-184). Springer, Cham.

[18] S. Thurner, R. Hanel, B. Liu and B. Corominas-Murtra "Understading Zipf's law of word frequencies through sample space collapse in sentence formation," Journal of The Royal Society Interface, vol. 12, no. 108, The Royal Society, p. 2-150330, Jul 2015, doi: 10.1098/rsif.2015.0330

[19] S. Nakagawa, P. C. D. Johnson, and H. Schielzeth, "The coefficient of determination R 2 and intra-class correlation coefficient from generalized linear mixed-effects models revisited and expanded," Journal of The Royal Society Interface, vol. 14, no. 134. The Royal Society, p. 20170213, Sep. 2017. doi: 10.1098/rsif.2017.0213

[20] H. Jing, C. Wang, L. Cheng, J. Qi, S. Jiang, and X. Zhang, "Automatic Development of Knowledge Graph Based on NLTK and Sentence Analysis," 2021 3rd International Conference on Natural Language Processing (ICNLP). IEEE, Mar. 2021. doi: 10.1109/icnlp52887.2021.00015.

[21] S. Ezhilarasi and P. U. Maheswari, "Depicting a Neural Model for Lemmatization and POS Tagging of Words from Palaeographic Stone Inscriptions," 2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS). IEEE, May 06, 2021. doi: 10.1109/iciccs51141.2021.9432315.

[22] G. Y. Annum, “A Basic Strategy for Incorporating Lecture Notes with Audio-Visuals of Practical Activities to Foster Online Electronic Learning Implementation in Studio or Laboratory-Based Institutions,” Creative Education, vol. 14, no. 07. Scientific Research Publishing, Inc., pp. 1421–1439, 2023. doi: 10.4236/ce.2023.147090.

[23] Lu, S., Wang, M., Liang, S., Lin, J., & Wang, Z. (2020, September). Hardware accelerator for multi-head attention and position-wise feed-forward in the transformer. In 2020 IEEE 33rd International System-on-Chip Conference (SOCC) (pp. 84-89). IEEE.25. M. A. Rahman and E. Kumar Dey, “Datasets for aspect-based sentiment analysis in bangla and its baseline evaluation,” Data, vol. 3, no. 2, pp. 1-15.

[24] S. Chowdhury and W. Chowdhury, “Performing sentiment analysis in Bangla microblog posts,” 2014 International Conference on Informatics, Electronics &amp; Vision (ICIEV). IEEE, May 2014. doi: 10.1109/iciev.2014.6850712.

[25] M. H. Munna, M. R. I. Rifat, and A. S. M. Badrudduza, “Sentiment Analysis and Product Review Classification in E-commerce Platform,” 2020 23rd International Conference on Computer and Information Technology (ICCIT). IEEE, Dec. 19, 2020. doi: 10.1109/iccit51783.2020.9392710..

[26] Md. H. Alam, M.-M. Rahoman, and Md. A. K. Azad, “Sentiment analysis for Bangla sentences using convolutional neural network,” 2017 20th International Conference of Computer and Information Technology (ICCIT). IEEE, Dec. 2017. doi: 10.1109/iccitechn.2017.8281840.

[27] D. Sharma, M. Sabharwal, V. Goyal, and M. Vij, “Sentiment Analysis Techniques for Social Media Data: A Review,” First International Conference on Sustainable Technologies for Computational Intelligence. Springer Singapore, pp. 75–90, Nov. 02, 2019. doi: 10.1007/978-981-15-0029-9_7.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 12-19

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Fear Recognition Using Early Biologically Inspired Features Model

Elham Askari[1*]

[1.]Department of Computer Engineering, Fouman and Shaft Branch, Islamic Azad University, Fouman, Iran.

Received: 23 Aug 2022/ Revised: 04 Apr 2023/ Accepted: 01 May 2023

Abstract

Facial expressions determine the inner emotional states of people. Different emotional states such as anger, fear, happiness, etc. can be recognized on people's faces. One of the most important emotional states is the state of fear because it is used to diagnose many diseases such as panic syndrome, post-traumatic stress disorder, etc. The face is one of the biometrics that has been proposed to detect fear because it contains small features that increase the recognition rate. In this paper, a biological model inspired an early biological model is proposed to extract effective features for optimal fear detection. This model is inspired by the model of the brain and nervous system involved with the human brain, so it shows a similar function compare to brain. In this model, four computational layers were used. In the first layer, the input images will be pyramidal in six scales from large to small. Then the whole pyramid entered the next layer and Gabor filter was applied for each image and the results entered the next layer. In the third layer, a later reduction in feature extraction is performed. In the last layer, normalization will be done on the images. Finally, the outputs of the model are given to the svm classifier to perform the recognition operation. Experiments will be performed on JAFFE database images. In the experimental results, it can be seen that the proposed model shows better performance compared to other competing models such as BEL and Naive Bayes model with recognition accuracy, precision and recall of 99.33%, 99.71% and 99.5%, respectively

Keywords: Fear Recognition; Early Biological model; Support Vector Machine; Facial Expressions.

1- Introduction

Excitement is the general and short reaction of the body organism to an unexpected situation, which is accompanied by a pleasant emotional state. In terms of the root of the word, emotion means the factor that moves the organism. Emotion refers to positive and negative feelings that arise in a person in certain situations. The word emotion refers to feelings and does not mean behaviors. For example, we get upset when we are treated unfairly. Emotions such as anger, fear, sadness, hatred, surprise, jealousy, envy, shame, etc. are all examples of emotions, but defining all these states is very difficult. These states form important parts of emotional life because they manifest in important mental situations [1]. By using the characteristic of emotional states, psychological and physiological disorders can be diagnosed. In fact, the outward part of these disorders is called the manifestation of excitement.

Emotions are response patterns that consist of three behavioral, autonomous, and hormonal elements. In the behavioral element, there are muscle movements that are called in appropriate situations. The autonomic element facilitates behaviors and stores energy for movement, in which case the sympathetic nervous system increases and the parasympathetic decreases. As a result, the person's heart rate increases and the size of the blood vessels changes, and the blood circulation is diverted from the side of the digestive organs to the muscles. In the hormonal element, autonomic responses are enhanced. The hormones epinephrine and norepinephrine increase the blood flow to the muscles, which causes the food stored in the muscles to be converted into glucose. In addition, the adrenal glands secrete steroid hormones that make glucose available to the muscles [2].

The emotional state of fear is one of the most common mental disorders in every human being. Some fears are illogical, which causes a person's situations and activities to be disturbed in life. Some fears are dependent on the situation, that is, the person himself is aware of his fear, but he cannot control the fear, therefore, when facing that

✉ **Elham Askari**
askary.elham@gmail.com

situation or the factors he is afraid of, he experiences panic and anxiety [3]. The amygdala (almond) is responsible for the physiological reactions of fear and anger, which is located in the temporal part and detects and responds to threatening events. In the early 20th century, researchers identified the hypothalamus as a key structure in the nervous system, and the hypothalamus responds to emotional feelings in the brain [4]. Researchers have acknowledged that when a person feels fear and anger, signals are exchanged between the amygdala and the hypothalamus [5]. All kinds of emotional states are important in a person's daily life because they play an important role in the treatment of diseases.

The emotional state of fear has two states, conscious and unconscious, and many researches have tried to identify it optimally. In most researches, the recognition rate of fear is lower than other emotional states. Perhaps the reason is that some fears occur unconsciously, which causes the recognition rate to be low, such as when a person suddenly feels fear upon seeing a snake [6].

There are warning states in humans such as anger and fear. These feelings differ in different people (adults and children) according to environmental and social states. Functional magnetic resonance imaging (fMRI) studies have shown that the amygdala plays an important role in human fear [7]. Studies that have been reviewed so far have shown that the simultaneous activation of the amygdala and other parts of the brain such as the hypothalamus causes the feeling of fear in humans. To prove that there is a connection between the amygdala and the hypothalamus during fear, scientists conducted experiments among nine people. They placed electrodes in the amygdala and hypothalamus in these 9 people and showed them scenes from horror movies and observed that when a person feels fear, signals are exchanged between the amygdala and the hypothalamus. it will be As a result, at the time of fear, information is exchanged in the brain, amygdala and hypothalamus parts [8].

In their research, researchers from Datmouth College in New Hampshire have succeeded in identifying a region of the brain that is associated with the feeling of fear caused by uncertainty about the future. In this study, 61 students underwent an MRI scan after filling out a questionnaire related to their ability to tolerate possible negative events in the future. MRI images were analyzed and compared with future uncertainty tolerance scores. The author of this study emphasized that the results of our research show that there is a relationship between a person's ability to deal with this uncertainty and the volume of gray matter in a specific region of the brain [9]. The biological model in humans is related to the brain and in computers it is related to machine vision. One of the recently presented models is the biologically inspired model of BIM (Biologically Inspired Features Model). This model was designed in such a way that it can function like the human brain. The BIM model includes four computational layers S1, C1, S2, and C2. where S and C are complex cells in the visual cortex, respectively. The units in the S1 layer are easily matched to the cells of the visual cortex. These units combine the primary inputs using an ensemble of Gabor filters, each of which is the product of a Gaussian ellipse and a complex plane. Gabor filter works like the human eye. Therefore, this type of dense input leads to a heavy computational cost in S units, because each image will be interlaced with Gabor filters of different parameters [10]. C units are complex because they are themselves a pool of inputs obtained through a maximization operation. BIM is a forward process, which combines reactions S1, C1, S2 and C2. But due to the lack of a feedback step, it blindly selects features to indicate which features are important. Therefore, a large number of prototypes must be sampled to match features, and as a result, the computational cost of matching is very heavy [11]. Cereri and his colleagues tried to bridge the gap between computer and neuroscience perspectives with the standard model they presented for object recognition in the real world [12].

The biological model, due to problems such as heavy computational cost in S units, due to not having a feedback stage and uninformed selection of features, caused C units to become complicated. As a result, the researchers expanded the basic BIM model to include five layers. The image layer was added to improve the basic model and increase the features and maintain the amount of information space in the S2 feature [13].

The BEL model is an emotional model of the brain that corresponds to the human limbic system. In the BEL model, the network learns while training the parameters that are needed to reach the solution, to calculate this using the reward of a cost function that is in the form of reinforcement [14-16]. BEL model, in engineering systems, can increase the degree of freedom, control capacity, reliability and stability. Also, the performance of the BEL controller on various nonlinear systems has shown stable and high compatibility.

In 2017, researchers conducted a study in which different emotional states of people's images were tested. None of the participants had mental or nervous problems. For each identity, photographs of faces were selected in all major emotional states, including happiness, sadness, anger, fear, and disgust, plus a neutral state [17]. Each trial consisted of a face covered by a 6×8 grid of white tiles, with one tile revealed randomly and an additional tile revealed completely randomly every second. Participants were instructed to click the stop button below the image as soon as they were able to make a decision about the facial expression. After each decision, participants received feedback on whether they answered correctly. Examining the types of ambiguities in each case was analyzed. The values of all individual tiles were given as input to PCA and then the recognition process was performed. In this method, the accuracy of recognizing the emotional state of fear was 60% in men and 50% in women [17].

Sima et al presented a method based on convolutional neural network to show the effects of facial changes in people. They used a generative adversarial network to generate the main frame for identity recognition and expression of individual differences, and they also used CNN to extract emotion features [18].

Cambpell et al. tried to detect the fear state and face using GLM. They studied the act of detection on men and women and concluded that there is no difference in the detection of fear in the face in different genders [19].

Rinck et al investigated how face masks impair facial emotion recognition. They used facial expressions from the Radboud database and concluded in experiments that the mask interferes with the detection of disgust, fear, surprise, sadness and happiness [20].

Mellouk and et al. conducted a study on the recent works in the field of automatic recognition of emotions using faces through deep learning and by reviewing the recent works of researchers, they compared their works with each other[21]. Recently, emotion detection methods based on CNN, CNN-LSTM and 3D-CNN have been carried out, which have achieved an accuracy of more than 99%, but in fear detection.the accuracy is still not very high[22-24].

Humans and animals perform better than machines from a system point of view, and being able to simulate a system point of view that can recognize objects well has been one of the dreams of neuroscientists and computer engineers. There are different learning models for emotional state recognition, but in most models, the accuracy of fear recognition is lower than other models. In this article a biological model is used to increase the accuracy of fear identification. The biological model used in the proposed method is the EBIF model "Early Biologically Inspired Features". This model is an improvement of the basic biological BIM model. The reason for choosing this model is that because the nervous system involved with the human brain works very well, it is expected that any model inspired by the human brain model will also work well. Here, for the first time, a biological model has been used to extract the feature and recognize the emotional state of the face and then with the SVM, the emotion recognition is done. In this model, due to the use of the Gabor filter, which has good and multiple resolution properties in the space and frequency domain and is considered a powerful tool for texture analysis, it is expected to achieve good results of the recognition rate.

This article is organized in such a way that after the introduction, the proposed method is presented in the second part. In the third part, the results will be analyzed and in the fourth part, the conclusion will be expressed.

2- Proposed Model for Fear Recognition

One of the most important biometrics for recognition is the face, because this biometric contains small features that increase the recognition rate. There are different models to recognize the emotional state, but in most models, the accuracy of fear recognition is lower than other models, so this article deals with this issue, used the EBIF (Early Biologically Inspired Features) model for the recognition. In this article, the following model is used to improve fear recognition. The flowchart of the proposed method is shown in Figure (1).

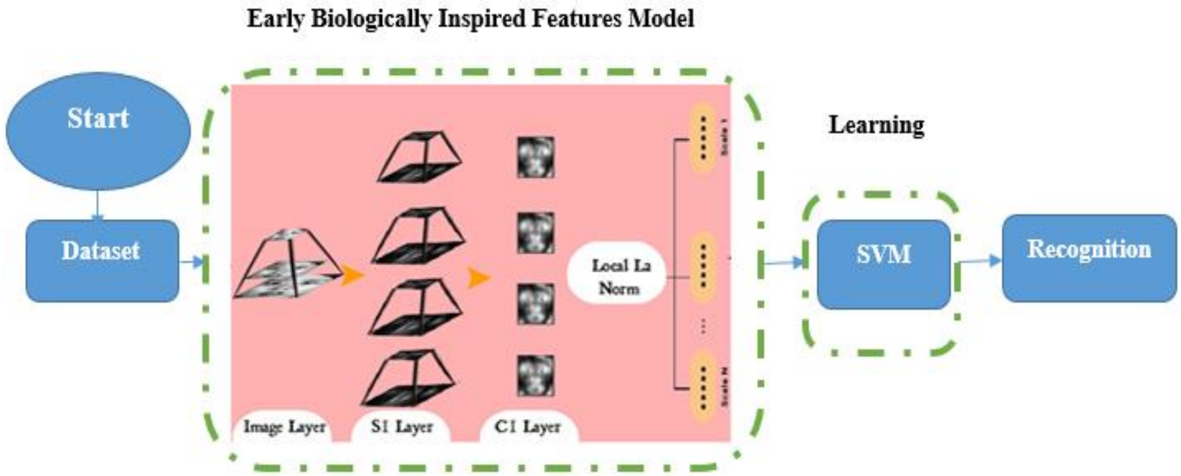


Fig. 1. Flowchart of the proposed method

In the proposed model, first the face images are called from the dataset, then they are scaled as a pyramid, and the images are entered into the S1 layer as a pyramid. Then the Gabor filter is applied to the pyramid images. The result entered in layer C1 is then reduced by Maxpooling in this layer so that two matrices of the same size are finally obtained because each layer is different from the higher layer in the pyramid. If both layers are selected in a row, the same matrices will be obtained and finally the average of the layers will be calculated. In the normalization stage, the brightness variance is obtained according to formula (5).

After extracting effective features by that model, classification will be done using SVM. Using this algorithm increases the efficiency and accuracy of the system. The evaluator in dividing the data is based on the k-fold method, where k=10 is considered. Below, the pseudo code and each step of the proposed model will be explained.

```
Inputs: Determine the training and test images
//EBIF model
while(stopping condition is not met) do
{Image Layer
S1 Layer
C1 Layer
Local L2 Norm}
end while
training SVM
Outputs: Determine the precision, accuracy and recall
```

2-1- EBIF Model

The improved EBIF model is the BIM model, which is used here to extract features to increase the recognition rate of the emotional state of fear. This model includes four layers: Image Layer, S1 Layer, C1 Layer, Local L2-norm Layer, the tasks of each of which are described below.

Image Layer: In the image layer, first the images are converted to gray scale and then they are obtained as a pyramid in N scale (in this research, N=6) from large to small. Each scale of the upper image is a factor of √2 smaller than the current image. The pyramids are scaled from K to K+1 so that (k=1,2,...,N-1). In fact, the images become a pyramid in six scales after graying. Database images enter the S1 layer when they are created as K to K+1 scales.

S1 Layer: Gabor filter bank is used in layer S_1. Gabor filters are used to extract features from facial regions. The most important reason for using Gabor filters is their resistance to rotation and scaling. In addition, it also counteracts photometric disorders such as brightness, image noise, etc. [25-29]. If the Gabor filters are properly and accurately adjusted, they have a very good performance in detecting the features of the texture and the edge of the texture. Another important feature of Gabor filters is their high common separation degree. This means that their response is completely local and adjustable both in the field of place and in the field of frequency. Gabor filter with M orientation filters all parts of pyramid images. The convolution kernel of the Gabor filter is the product of a complex exponential and Gaussian function. The formula used in the Gabor filter in the proposed model is given below.

$$G(x,y) = \exp\left(-\frac{X^2+\gamma^2Y^2}{2\sigma^2}\right)COS(\frac{2\pi}{\lambda}X) \qquad (1)$$

Where X=x cosΘ+y sinΘ and Y=-xsinΘ+ycosΘ. So that Θ controls the orientations of the filter. The filter bank contains M orientations, where M=9. The orientations are equally distributed in the range [0,π). The size of each filter is 5x5 and x and y are variable between 2 and -2. The parameters of aspect ratio, effective width and wavelength are set to 1, 3 and 5 respectively. Equation (2) shows the response of the Gabor filter to an image.

$$R = G \times I \qquad (2)$$

Gabor filter responses can localize the object and distinguish its structure with different orientations [29].

C1 Layer: From The C1 layer functions according to the complex cells of the visual cortex. In this part, the Gabor pyramid is divided equally according to a specific direction and the sub-pyramids are set. At the bottom, the size of the pyramids is 4x4 and at the top, the size is 3x3, which is in accordance with the basic BIM model. In this layer, dimension reduction is done by Max Pooling. In fact, a pyramid combination is used, and the mean and standard deviation are used instead of the highest response to display each sub-pyramid. The advantage of using the C1 layer is that each element in this layer is obtained by combining the response of the local Gabor filter which is based on edge detection and quantitative position and tolerable scale on the neighboring position and different orientations.

A)Dimension reduction by Pooling

The function of Pooling is to reduce the dimensions in depth. In this way, the spatial size (width and height) of the image is reduced in order to reduce the number of parameters and calculations inside the network. The Pooling layer operates independently on each depth slice of the input mass and spatially resizes it using the Max operation. The most common way to use this layer is to use this layer with 2 x 2 filters with step 2 (S=2), which reduces each depth cut in the input by removing two elements from the width and two elements from the height and causes Removing 75% of the values in it becomes a deep cut. Each Max operation here obtains the maximum between 4 numbers (a 2x2 region in the depth slice). In general, the Pooling layer receives a mass of size W1×H1×D1 as input. where W indicates width, H indicates height and D indicates its depth. Pooling requires two meta parameters: - The size of spatial extent (size (x and y filters) perceptual area) F - step size or S Then it produces an output mass with the size of 2 × H2 × D2, which:

$$W2 = \frac{W1 - F + 2P}{S} + 1 \qquad (3)$$

$$H2 = \frac{H1 - F + 2P}{S} + 1 \qquad (4)$$

The above relations show that both width and height are calculated equally symmetrically. Max Pooling also performs Noise Suppressant. Max Pooling works very well due to faster convergence and better generalization and selection of optimal features [30].

Local Normal Layer L2: In the L2 local normal layer, the variance of brightness, which is a very important feature for face recognition, is calculated. To have a powerful EBIF model against light changes, it must be normalized first. The feature dimensions of vector f are calculated from each location of layer C1 with L2-norm as follows.

$$\tilde{f} = \frac{f}{\sqrt{||f||^2 + \xi}} \qquad (5)$$

Here ξ is a constant. The vector features are normalized in different positions of the C1 layer of the subsets feature form for the response of scale k objects. All N-1s are subsets of multi-scale features that contain edge information and local normalized multi-orientation that are quantitatively position and scale-tolerant of the final EBIF feature response.

Illustrations or pictures: All halftone illustrations or pictures can be black and white and/or colored. Supply the best quality illustrations or pictures possible.

2-2- Support Vector Machine (SVM) Category

Support vector machine (SVM) is a classification algorithm that is considered one of the best classification techniques. The primary basis of the SVM classifier is the linear classification of data, in which, in the linear division of the data, an attempt is made to select a line that has a higher confidence margin. Various kernel functions can be used in svm, including exponential, polynomial and sigmoid kernels. The choice of non-linear kernels allows us to construct linear separators in the feature space if they are non-linear in the original space. They are also very good at solving computational problems that have many dimensions. In addition, it enables the use of infinite dimensions and is efficient in terms of time and memory [31]. In this article, the polynomial kernel that obtained the best result in svm is used, and its equation is given below.

$$k(x_i, x_j) = (1 + x_i^T x_j)^d \qquad (6)$$

where x is the training data and d is the degree of the function for the polynomial kernel.

3- Discussion and Results

The proposed model is simulated in the MATLAB 2021 software environment. The simulation part is divided into two parts, the first part is the simulation of the proposed model and training, and the second part is the test and comparison of the proposed model with Naive Bayes and BEL categories.

3-1- Dataset

In this article, JAFFE (Japanese) authentic dataset is used which contains 213, 256×256 pixel images of Japanese women's facial expressions. This dataset contains images of ten Japanese women who showed different emotions, including six emotional states: angry, disgust, happy, sad, fear, surprise, and a normal state. In this experiment, first, 200 images are selected from the JAFFE dataset, and then the features of the face are extracted from the images by the proposed model and the classification process is performed and it has been tried to distinguish fear from different states with high accuracy. Figure (2) shows an example of some used dataset images.

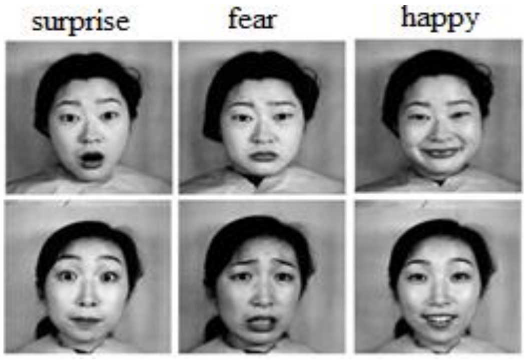


Fig. 2. An example of some dataset images

3-2- Obtained Results

The results of accuracy, precision and recall with SVM method are obtained using the following formula:

$$\text{Precision} = \frac{True\ Positive}{True\ Positive + False\ Positive} \quad (7)$$

$$\text{Accuracy} = \frac{True\ Positive + True\ Negative}{All\ test\ sample} \quad (8)$$

$$\text{Recall} \quad (9)$$

Where Fn is unrecalled related images, Tp recalled related images, Tn unrecalled unrelated images and Fp recalled unrelated images.
To train and validate the proposed model k-fold and k=10 have been used. In this type of partitioning, the data is divided into k-1 training sets and 1 testing set. In this type of validation, the data is divided into K subsets. From these K subsets, each time one is used for validation and another K-1 is used for training. This procedure is repeated K times and all data are used exactly once for training and once for validation. The figure below shows the ROC chart, based on the proposed model, Naive Bayes and BEL.
The figure below shows the ROC chart, based on the proposed model, Naive Bayes and BEL.

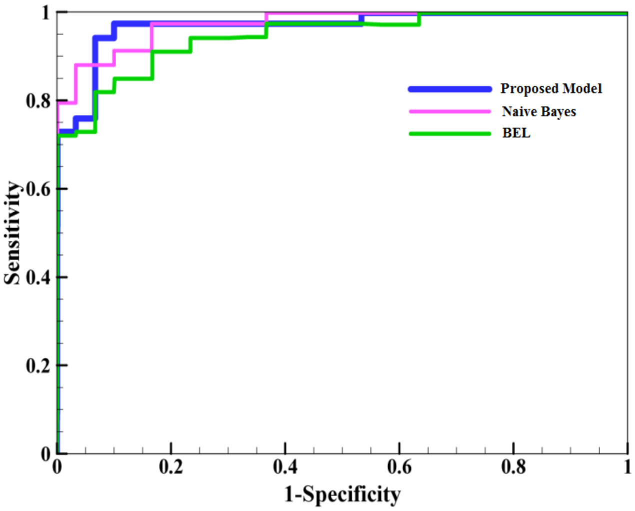


Fig. 3. ROC diagram for three methods

As shown in Figure (3), classification with the proposed model and based on SVM works better than other methods. Figure (4) shows the accuracy of the proposed method using the number of different images and EBIF features, compared to Naive Bayes and BEL methods with the same features.

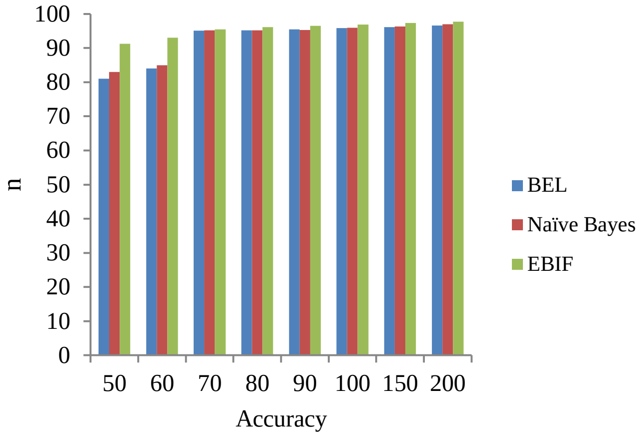


Fig.4. accuracy chart of the proposed method using the number of different images

As can be seen, the proposed model has shown better accuracy in each step. In Table (1), the proposed model was compared with the results of previous researches that used Naive Bayes and BEL classification using Gaussian distribution, mean, variance and PCA features, and the results are given below [11 and 32].

Table 1: comparing the accuracy of the proposed model with two competing models

model	Naive Bayes	BEL	EBIF
Accuracy	77.09	96.57	99.33
Precision	81.00	96.1	99.71
Recall	79.30	95.9	99.5

As can be seen, the proposed model is in the first place with 98.32% accuracy, the BEL model is in the second place with 96.57% accuracy, and the Naive Bayes model is in the third place with 77.09% accuracy. The accuracy of the proposed method when detecting anger, happiness and surprise has also been compared and its results are shown in Table (2).

Table 2: comparing the accuracy of the proposed model with two competing models in different states

State	Naive Bayes	BEL	EBIF
angry	59.27	98.2	99.1
happiness	81.16	92.16	99.2
surprise	68.15	96.5	99.03

As can be seen, the proposed model performs better than the competing models in other facial states. In anger mode, the proposed model has performed better with 99.1% accuracy compared to BEL and Naïve Bayes models, which are 98.25% and 59.27% accuracies, respectively. Also, in the state of happiness, the proposed model has performed better with an accuracy of 99.2% compared to the BEL and Naïve Bayes models, which are 92.16% and 81.16%, respectively. In addition, it can be seen that, in surprise state, the proposed model has performed better with 99.03% accuracy compared to BEL and Naïve Bayes models.

The precision of the proposed model when detecting anger, happiness and surprise has also been compared and the results are shown in Table (3).

Table 3: comparing the precision of the proposed model with two competing models in different states

State	Naive Bayes	BEL	EBIF
angry	57.3	97.1	98.7
happiness	83.7	94.2	99.1
surprise	69.04	95.7	98.2

As can be seen in table (3), the proposed model has shown better precision in all states. The recall of the proposed model when detecting anger, happiness and surprise are shown in Table (4).

Table 4: comparing the recall of the proposed model with two competing models in different states

State	Naive Bayes	BEL	EBIF
angry	54.1	94.9	98.1
happiness	81.2	91.1	98.9
surprise	65.3	94.2	98.4

As can be seen in table (4), the proposed model has shown better recall in all states.

The obtained results in the above tables show that the proposed model has performed better in all evaluation criteria compared to the other two methods.

4- Conclusions

In this article, the recognition of the emotional state of fear was discussed. The high accuracy of recognizing the emotional state of fear helps doctors in the treatment of many diseases, including schizophrenia, phobia, etc. Here, the early biological model was used for the first time to identify the emotional state of fear, which was observed to perform the recognition process with high accuracy. Four computational layers were used in the proposed model. In the first layer, the input images were pyramided in six scales from large to small. Then the whole pyramid was entered into the next layer and the Gabor filter was applied to each image and the result was given to the next layer. In the third layer, dimension reduction was done in feature extraction. In the last layer, normalization was done on the images and finally the results were entered into the SVM category because the use of this category increases efficiency.

The proposed method was applied on the JAFFE dataset to test the accuracy of the proposed method. The results of the proposed model showed that the accuracy of recognizing the emotional state of fear is 99.33%. Also, in order to evaluate, the proposed method was compared with two models, BEL and Naive Bayes, and by analyzing the information output, it was concluded that the proposed model has a higher identification accuracy and the BEL method performs better in the second place.

References

[1] H Ganji, “General Psychology”, Tehran:Savalan, Vol.350, 1386.(Persian)

[2] F Lotfi Kashani, SH Vaziri, “Child's Pathological Psychology”, Tehran:Arasbaran, Vol.344, 1395.(Persian)

[3] Karlson N, Kalat J, Bridola M, N Vatson, M Rosenzevig, “Physiological Psychology: An Introduction to Behavioral”, Cognitive, and Clinical Neuroscience, Tehran:Arasbaran, Vol.468, 1394.(Persian)

[4] J Ledoux, “The Emotional Brain, Fear, and the Amygdala”, Cellular and Molcular Neurobioilogy. Vol.23, 2003, pp.727-738.

[5] J Lin, J Zheng, “Modulating Amygdala–Hippocampal Network Communication: A Potential Therapy for Neuropsychiatric Disorders”, Neuropsychopharmacology. Vol.43, 2018, pp.218-219.

[6] A Fallah, “Fear of Nervous Attacks and Earthquakes”, Daneshmand, Vol.53(9), 1394, pp.56-60.(Persian)

[7] M Miyahara, T Harada, T Ruffman, N Sadato, T Iidaka, “Functional connectivity between amygdala -and facial regions involved in recognition of facial threat”, Soc Cogn Affect Neurosci, Vol.8(2), 2013, pp.181-189.

[8] J Lin, J zheng, “Modulating Amygdala–Hippocampal Network Communication: A Potential Therapy for Neuropsychiatric Disorders”, Neuropsychopharmacol, Vol.43, 2018, pp.218–219.

[9] M Kim, J Shin, J Taylor, A Mattek, S Chavez, P Whalen, “Intolerance of Uncertainty Predicts Increased Striatal Volume”, Emotion, Vol.12(6), 2017, pp.895-899.

[10] H Yongzhen, H Kaiqi, W Liangsheng, T Dacheng, T Tieniu, L Xuelong, “Enhanced Biologically Inspired Model”, Proceeding of IEEE Conference on Computer Vision and Pattern Recognition, 2008, Anchorage, AK, USA.

[11] Fyaghouti, S motamed, “Recognition of Facial Expression of Emotions Based on Brain Emotional Learning (BEL) Model”, Advances in Cognitive Science, Vol.20(4), 2019, pp.46-61.

[12] T Serre, L Wolf, T Poggio, “Object recognition with features inspired by visual cortex”, Proceeding of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2005, San Diego, CA, USA.

[13] J G Mutch, D Lowe, “Object Class Recognition and Localization Using Sparse Features with Limited Receptive Fields”, International Journal of Computer Vision, Vol.80(1), 2008, PP.45–57.

[14] J Morén J, C Balkenius, “A computational model of emotional learning in the amygdala”, From animals to animats, Vol.6, 2006, PP.115-124.

[15] J Morén J, “Emotion and learning- a computational model of the Amygdala [Ph.D. Dissertation]’, Lund, Sweden: Lund University; 2002.

[16] E Lotfi, “Mathematical modeling of emotional brain for classification problems”, Proceeding of IAM, 2013 January, Vol. 2(1), PP.60-71.

[17] M Wegrzyn, M Vogt, B Kireclioglu, J Schneider, J Kissler, “Mapping the emotional face. How individual face parts contribute to successful emotion recognition”, PLOS ONE, Vol.12(5), 2017.

[18] Y Sima, J Yi, A Chen, Z Jin, “Automatic expression recognition of face image sequence based on key-frame generation and differential emotion feature”, Applied Soft Computing, Vol.113, 2021.

[19] R Campbell, K Elgar, J Kuntsi, R Akers, J Terstegge, M Coleman, D Skuse, The classification of 'fear' from faces is associated with face recognition skill in women”, Neuropsychologia, Vol.40(6), 2002.

[20] M Rinck, M A. Primbs, I A. M. Verpaalen, G Bijlstra, “Face masks impair facial emotion recognition and induce specifc emotion confusions”, Cognitive Research: Principles and Implications, Vol. 83, 2022.

[21] W Mellouk, W Handouzi, “Facial emotion recognition using deep learning: review and insights” Proceeding of the 2nd international Workshop on the Future of Internet of Everything (FIoE), 2020 August 9-12, Leuven, Belgium.

[22] D H Kim, W J Baddar, J Jang, Y M Ro, “Multi-Objective Based Spatio-Temporal Feature Representation Learning Robust to Expression Intensity Variations for Facial Expression Recognition”, IEEE Trans. Affect. Comput, Vol. 10(2), 2019, pp.223 236.

[23] Z Yu, G Liu, Q Liu, J Deng, “Spatio-temporal convolutional features with nested LSTM for facial expression recognition”, Neurocomputing, Vol. 317, 2018, pp. 50 57.

[24] D Liang, H Liang, Z Yu, Y Zhang, “Deep convolutional BiLSTM fusion network for facial expression recognition”, Vis. Comput, Vol. 36(3), pp. 499 508.

[25] S Meshgini, A Aghagolzadeh, H Seyedarabi, “Face recognition using gabor-based direct linear discriminant analysis and support vector machine”, Computers & Electrical Engineering, Vol.39, 2013, pp.727–745.

[26] C Liu, H Wechsler, “Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition”, IEEE Transactions on Image Processing, Vol.11, 2002, pp.467-476.

[27] J K Kamarainen, V Kyrki, H Kalviainen, “Invariance properties of gabor filter-based features-overview and applications”, IEEE Transactions on Image Processing, Vol.15, 2006, pp.1088-1099.

[28] L Shen, L Bai, M Fairhurst, “Gabor wavelets and general discriminant analysis for face identification and verification”, Image and Vision Computing, Vol.25, 2006, pp.553-563.

[29] J G Daugman, “Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression”, IEEE Transactions on acoustics, speech, and signal processing, Vol.36(7), 1988, pp.1169-1179.

[30] Y-L Boureau, J Ponce, Y LeCun, “A theoretical analysis of feature pooling in visual recognition”, Proceedings of the 27th international conference on machine learning (ICML-10), 2010, Haifa, Israel.

[31] S M Tabatabaii, T M Nazeri, M Dastorani, “Archive of SID Performance comparison of GP, ANN, BCSD and SVM models for temperature simulation”, Journal of Meteorology and Atmospheric Sciences, Vol.1(1), 2018, pp.53-64. (Persian)

[32] N Sebe, M.S Lew, I Cohen, A Garg, T.S Huang, “Emotion recognition using a Cauchy Naive Bayes classifier”, Proceeding of 16th International Conference on Pattern Recognition, 2002 August.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 20-28

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Optimization of Query Processing in Versatile Database Using Ant Colony Algorithm

Hasan Asil[1*]

[1.]Department of Computer Engineering, Faculty of Electrical and Computer Engineering, Azarshahr Branch, Islamic Azad University, Azarshahr, Iran.

Received: 20 Jan 2023/ Revised: 04 Dec 2023/ Accepted: 21 Jan 2024

Abstract

Nowadays, with the advancement of database information technology, databases has led to large-scale distributed databases. According to this study, database management systems are improved and optimized so that they provide responses to customer questions with lower cost. Query processing in database management systems is one of the important topics that grabs attentions. Until now, many techniques have been implemented for query processing in database system. The purpose of these methods is to optimize query processing in the database. The main topics that is interested in query processing in the database makes run-time adjustments of processing or summarizing topics by using the new approaches. The aim of this research is to optimize processing in the database by using adaptive methods. Ant Colony Algorithm (ACO) is used for solving optimization problems. ACO relies on the created pheromone to select the optimal solution. In this article, in order to make adaptive hybrid query processing. The proposed algorithm is fundamentally divided into three parts: separator, replacement policy, and query similarity detector. In order to improve the optimization and frequent adaption and correct selection in queries, the Ant Colony Algorithm has been applied in this research. In this algorithm, based on Versatility (adaptability) scheduling, Queries sent to the database have been attempted be collected. The simulation results of this method demonstrate that reduce spending time in the database. According to the proposed algorithm, one of the advantages of this method is to identify frequent queries in high traffic times and minimize the time and the execution time. This optimization method reduces the system load during high traffic load times for adaptive query Processing and generally reduces the execution runtime and aiming to minimize cost. The rate of reduction of query cost in the database with this method is 2.7%. Due to the versatility of high-cost queries, this improvement is manifested in high traffic times. In the future Studies, by adapting new system development methods, distributed databases can be optimized.

Keywords: Database; Ant Colony Algorithm; Query Processing; Versatility; Optimization.

1- Introduction

Query processing is the technique of data transmissions in a database system. The efficiency of a database depends on the technique used by it to obtain and retrieve data. usually, the database must have the ability to respond to the queries of the users and provide information [1]. In fact, the main part of database-management system is query processing and optimizing it [1]. Several methods have been presented to optimize query processing in the database nowadays. Some of these methods have proposed proper solutions for optimizing the queries and running relational, textual, and Xml data [2].

The main reason for this need is the essential need to optimize queries. When optimization queries made with Selinger-Style (Application systems database) failed, the obtained outcomes (system chaos and creation of new algorithms) struck a blow for large companies' increase of research in the comparative features of their database products [3].

Ant Colony optimization algorithm was presented by Dorigo et al. for the first time for difficult issues of theoretical optimizing of the traveling salesman. So far, this algorithm has been used for optimization problems, such as traveling salesman, balanced scheduling in networks, various data mining techniques such as clustering, and so on [4].

The Ant Colony algorithm is inspired by studies and observations on Ant colonies [5,6]. These studies have

✉ **Hasan Asil**
h.asil@iauazar.ac.ir

shown that Ants are social insects that live in colonies and their behavior is more towards the survival of the Colony than towards the survival of a part of it. One of the most important and interesting behavior of Ants is their behavior to find food and especially how to find the shortest path between food sources and nest [7, 8].

This type of behavior of Ants has a kind of mass intelligence that has recently attracted the attention of scientists [9,10]. In the real world, Ants first randomly go back and forth to find food [11]. Then they return to the nest and leave a trail of pheromone. Such traces turn white after rain and are visible. When other Ants find this path, they sometimes stop roaming and follow it. Then, if they get food, they return home and leave another mark next to the previous one; And in other words, they strengthen the previous path [12,13]. The pheromone evaporates over time, which is useful in three ways [15,16]:

- Makes the path less attractive for subsequent Ants. Since an Ant in a long-time travel more and reinforces shorter paths, any path between home and food that is shorter (better) is reinforced more and the one that is farther is less.
- If the pheromone did not evaporate at all, the paths that were traveled multiple times would become so over-attractive that they would greatly limit random foraging.
- When the food at the end of an attractive path runs out, the trace remains.

Figure 1 shows these ways [17]. In this figure, for example, different ways of getting from the origin to the destination and back are shown. These paths have been improved over time and based on the routing Algorithm of Ants, and the short path in the three parts of the image has been selected in red color.

Therefore, when an Ant finds a short (good) path from home to food, the rest of the Ants will most likely follow the same path, and by continuously strengthening that path and evaporating other traces, eventually all Ants will follow the same path.

The purpose of the Ant Algorithm is to imitate this behavior by artificial Ants that are moving on the graph. The problem is to find the shortest path and the solution is these artificial Ants [18].

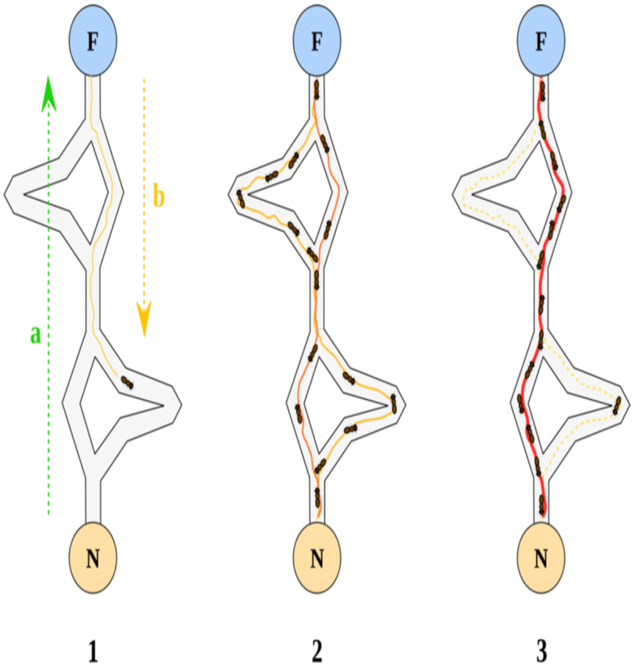


Fig. 1: Ant Colony ways

One of the applications of this algorithm is to optimize various problems. So that all kinds of Ant algorithms have been prepared to solve this problem. Because this numerical method has an advantage over analytical and genetic methods in cases where the graph constantly changes with time; and that it is an algorithm with repeatability; and therefore, with the passage of time, it can change the answer live; that this feature is used in computer network routing and caching. Urban transportation system is important [19].

Fig 2. Show the Ant Colony Algorithm and formula [20].

As shown in this algorithm. This method is a convergent method. In this method, the paths are selected by choosing random methods, but based on which paths are smaller, the amount of pheromone of the path is increased, and in the other part of the algorithm, the probability of choosing this path for other Ants is increased. With the passage of time and after the implementation of the algorithm, the convergence towards the optimal path has been done.

In this paper, we have tried to present a new method for optimization of query processing in database by using versatility methods and Ant Colony algorithm.

```
Step 1: [Initialization]
    t:=0; NC:=0;
    For each edge (I,j), initialize trail intensity to (0):=$T_{ij}(0)$:=$T_0$

Step 2: [String node]
    For each ant k:
        Place ant k on a randomly chosen city and store this information in $Tabu_k$

Step 3: [Build a tour for each ant]
    For i  from 1 to n:
        For k from 1 to m:
            Choose the next city I,j € $Tabu_k$, among the c candidate cities according
                $J = \{arg\, max\{[Ti(t)] \propto . [\mu]\beta\}$
        Where J is chosen according to the probability:
                $P_{ij}^{k}(t) = \frac{[Tij].[\mu]\beta}{\sum[Tij(t)\alpha.[\mu]\beta}$
        Store the chosen city in $Tabu_k$
        Local update of trail for chosen edge (I,j):
        $Tij = \rho.Tij(t) + (1 - \rho t).\Delta Tij \quad where\ \Delta Tij = T0$

Step 4: [Global update of trail]
    Compute length of tour, $L_k$, for each ant k
    Apply local improvement method for the tours of all ants k and recompute $L_k$
    For each edge (I,j) € Cycle*, update the trail according to:

Step 5: [Termination Conditions]
    Memorize the shortest tour found to this point
    IF ($NC<NC_{MAX}$ ) and (Not stagnation behavior)
    THEN empty all $Tabu_k$ and go step #2
    ELSE Stop
```

Fig. 2: Ant Colony Algorithm and formula

In this article, firstly, the previous work done regarding the optimization of processing in the database is examined, and then the proposed algorithm is explained in three parts, and then the proposed method is evaluated and examined with other parts, and finally, the results of the plan are also presented. The purpose of this research is to adapt database queries in order to reduce the execution time of queries in the database.

2- A Review of Previous Studies

Query-processing optimization is carried out with the aim of reducing resources, time, and so on. Optimization in the database can be done in three general categories. The three categories include server, query, and sessions [19]. In server method, hardware techniques are used to optimize. In the second method, using query process optimization, it is tried to optimize [21,22]. The third method is among the groups placed neither in the first nor in the second method.

In this paper, the third methods are used to optimize query processing in the database.

Among approaches to optimize query processing in database is using selection techniques [23]. In this method, the selected orders change. Some of the methods have used data classification for query processing [24]. Using caching mechanism to optimize query processing in the database is another method used for query processing in the database [25].

In this database query processing is used [26]. the results of the queries in this method are stored based on Xml model and when needed they are available [21]. Other versatility-based methods have been done in the database. These methods have been proposed in the past few years. In one of the methods, by versatility of the queries sent to the database, response time has reduced [22]. In another method, versatility method has been used to optimize the processing of queries distributed in the database [23]. In this method, using identification of the frequently-used queries in database and based on its implementation plan, it is attempted to optimize query processing. In this method, based on identification of similar and frequently-used queries, over time, it is attempted to optimize the query processing in database [27,28].

Ant Algorithm has various applications and it is used to optimize and solve various problems. Among these issues, can mention for using of this algorithm in improving route optimization. For example, this algorithm has been used in improving the balance of wireless sensor networks and in its routing based on the Internet of Things. In this issue, the purpose include to balance the traffic load and increase the speed of transferring packets in the network. So that the data packets reach the destination through paths with minimum density. As a result, one of the main methods to solve routing and load balancing problems is to use Ant-based algorithms. The aim of this research was to provide a suitable routing algorithm in order to shorten and improve the route in IoT-based systems [29].

In another research, Colony's Algorithm has been used to provide a nonlinear regression model for signal processing. In this issue, the goal was to provide a model and improve time [30].

One of the other finding for query optimizer processing in the database is DeepO. In this method, interactive optimization has been done in the PostgreSQL database with the help of machine learning which is presented in 2022 [31]. Another research in query processing optimization is a practical learner [32]. In this method, by creating an trial-and-error Agent, it can be used to reduce the number of transactions in the database with the memory-based learning method [33]. In another study, the query optimization process is redefined so that compiler optimizations come into high-speed transactions than previous data distributions, and compiler optimization employ as a driving force in query optimization. This new generation of query optimizers will be capable to optimize queries for significantly better performance than modern query optimizers [34]. Another effective way, with the applying of a margin generator, it has been added to provide a Matcher based on SQL customization to optimize descriptions. The purpose of this method is to preserve textual connections so that optimization can be done based on these connections. It has been presented due to its features and characteristics, including robustness, global optimization, Data parallelism involved the ability to act simultaneously and independently, and the ability to integrate [35]. In another research, to accelerate query operations and faster enrollment is Key performance indicators that has been added to the scalability of the distributed database. These indicators are aimed at improving the database query efficiency and speeding up the database query process and are provided with the help of the Ant Colony Algorithm [36].

In another study, the Clone Algorithm was used to generate test data for the database. In this research, search-based test data generation reformulates the test objectives as fitness functions, therefore, test data generation can be automated by meta-heuristic algorithms. Meta-heuristic algorithms search the domain of input variables in order to find input data that cover the objectives [37,38].

In another research, Ant Algorithm has been presented to provide a model for an intelligent virtual assistant. In this research, in order to increase reliability in education, an optimal model has been presented using the Ant method [39].

In this paper, we have tried to use the versatility of the above method for query processing in the database, but in this method, Ant Algorithm is used to enhance versatility.

3- The Proposed Algorithm

Today, the use of free and nature methods in solving complex problems are of the strategies used to solve complex problems [40,41].

In this study, we have tried to develop the previous methods ways to provide a new method to optimize query processing used in the database. Ant Colony Algorithm has been used to develop the method. The goal of the study is to examine the versatility of query processing in the database. In this paper, we try to identify the frequently-used queries sent to the database, so that by maintaining the implementation plans related to these queries and reducing implementation steps optimize query processing.

Utility software has databases sending queries to database according to user needs, based on which the needs of the users are met. This software usually sends queries to database according to which receives the responses and meets users' needs. In this software, the sent queries usually have the same structure, and these queries are iterated over time.

So far, various solutions have been proposed for query processing in the database. Some of these solutions are based on caching information and some are based on caching questions. The main reason for using query caching is related to the ٧ stages of query processing in the database. One of the most important points is the implementation of the question. For example, in stored procedures, the ready execution plan is used to provide the output and the execution time is reduced. In the proposed method, our main approach is high traffic times and identifying frequent questions by making them adaptable. Naturally, this method and maintenance of execution plans for subsequent executions can be effective, as in the proposed method, the identification Ant’s Algorithm is used.

In this paper, we have tried to present a model by using Ant Colony method, so that the database, over time [42,43], identifies the queries of the same type with high iteration and respond to them with specific implementation plans. In other words, the database matches the data to use ready implementation plans processes the queries at lower cost and faster.

There are several stages to process the queries in the database. With this method and with the help query implementation plan, one can reduce the cost of processing frequent queries in the database.

Various methods have been proposed for this type of versatility in the database [44, 45]. One of the four methods uses four parts to process the query in the database. These three parts include:

- Separator of commands
- Replacement policy
- Detector of the similar queries

By changing the method in replacement policy in this paper, we try to use Ant Colony Algorithm to optimize act query processing in the database.

The study uses the three components below to optimize query processing in the database.

3-1- Separator of Commands

The purpose of this part of the algorithm is to separate low-cost or non-optimizable commands. Commands such as Insert are among the commands that cannot be optimized and are added to the database by specific implementation plan, so these commands should be separated from optimizable ones.

For this method, the same method used in the previous work has been used [19,20]. This method has used comparison to separate query processing in database.

3-2- Replacement Policy

One of the most important parts of the versatility methods is determining how, when, and with what policy the placement is done. This means how database identifies the frequently used commands to maintain its implementation plan in the database.

For versatility, the system would listen to the queries submitted to the database and replace the implementation plans of the similar queries with the highest frequency sent to the database and wait for responses to the future queries (It should be noted that only the queries will be sent to this section that have been accepted by separator).

To identify the frequently-used queries, Ant Colony Algorithm is used.

Ant Colony optimization algorithm, which was presented by Dorigo et al. for the difficult problems of theoretical optimizing of the traveling salesman, is the important aspect of the behavior of Ants to find the shortest path between the nest and the food source [11].

Figure 3 shows how the random paths converge to the shortest path.

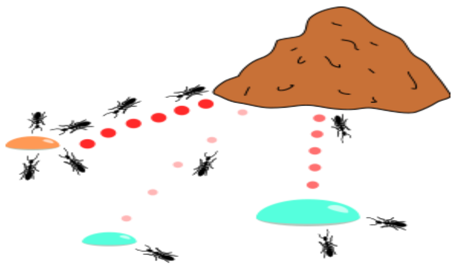


Fig. 3: Ant Colony Algorithm Procedure

One of the features of this algorithm is based on iteration and accidents that in case of increase in the number of iterations can help reach the response of solving problems [11].

In this method, for the queries, we develop the paths and based on the queries sent to the database, we enhanced the pheromones of the path. After a while, by identifying high pheromone queries and based on their frequency, one can identify frequently-used queries.

It is noted that due to the number of queries submitted to the database, in lack of exposure of pheromones after a certain period of time, paths (queries) are deleted.

Concerning the interval between two versatility actions, it should be stated that this interval will be calculated based on the frequency of pheromones of the queries made versatile in a dynamic way. This means that if the frequency of pheromones is high, the database will use these versatility queries for a long time for query processing in, and if pheromone frequency is low, it will work for a shorter time with this versatility. Not only, versatility operation is conducted on queries that are sent to the database in high-traffic time, but also stores the sent queries at versatility and high traffic time, and then the versatility is done on run queries in less time.

3-3- Detector of Similar Queries

This algorithm uses previous methods to detect similar queries [48,49]. The general approach in this part is that the queries submitted to relational data database are different based on parametric values. Thus, similarity detector can identified detecting excess similarity by sequence comparison. Obviously, implementation plan of similar queries is the same.

3-4- The Overall Approach of Algorithm

The Overall classification of algorithm approach is similar to previous studies [5]. In this algorithm, based on versatility scheduling is attempted to collect queries sent to the database. By versatility, exception queries are separated with the help of separator and are not stored.

Then with the help of any Colony Algorithm and replacement policy, frequently used queries are detected, and actions a taken to replace and maintain their implementation plans.

The task of the part related to the similarity of queries is detecting the similar queries in the database.

Figure 4 shows the general Algorithm of this procedure. In this algorithm, the way to implement the optimal algorithm is provided by three main procedures. In this algorithm, first, the commands of optimization ability are separated, then in the next section, the most common basic commands are selected, and in the last part, the replacement policy based on Ant Colony Algorithm.

```
Query Optimization Algorithm

1.  Begin
2.  Examine query by separator
3.  Produce query execution plan if query is one of
    exceptions
4.  If it's not an exception check its execution plan
    availability on the system by similarity
    recognizer
    4.1 If execution plan exists select it
    4.2 otherwise, send it in order to produce
    execution plan
5.  Executing plan for replying to query
6.  Check whether it is time to substitute or not?
6.1 If so, do the substitution.
7.  END
```

Fig. 4 The overall algorithm

It should be noted that in the case of implementation of the database, Database Management system can use the available implementation plan to implement similar queries. It is suggested that this part be added as a factor to the database.

4- Assessment of System

In operations research as explained, technique for solving computational problems is to optimize query processing in the database based on adapting queries (during high traffic times) by minimizing the processing time in the database.

This technique includes three parts: separating part, replacement policy, and query similarity detector, and the Ant Colony Algorithm has been applied to make it adaptable. Due to the approach of this algorithm with high-traffic times and identification of adaptive questions, the necessary cost to produce the query execution plan was reduced and as a result, the execution time was reduced. On the other hand, according to the adaptive mode commands based on the time of high traffic loading is also decreased cause of the cost reduction.

Currently, some methods are used for measuring performance of database system, the most primitive of the above-mentioned methods is implementation time in the system, which is the time required for implementation from the moment of sending until the response of the system. This time is calculated based on hours, minutes, and seconds [11].

To simulate this system, the proposed algorithm is classified and implemented in four classes. Then the results of implementation have been compared with the previous method using this method. In addition, we need DBMS and the intended data based on relation dependence, and we use SQL database and simulator SQLToolbelt to create data and identify the dependency of the tables. Moreover, in order to implement the algorithm and the intended comparisons, we will use VB.NET and API SQL functions. Code piece in Figure 5 shows the necessary implementation time of the query in milliseconds.

In this figure, the time of execution of questions has been compared with existing plans and without plans (creating a plan). As it is known, some time has been spent on creating the execution plan since the execution of the queries in the database.

After simulation of the query system, we obtained the following results:

- The cost of query implementation time as normal
- The cost of implementing the proposed algorithm
- The cost of versatile query as implementation plan

After obtaining these results, we added up the second and third costs and compared them and obtained the results of proposed method.

```
DECLARE @StartTime
datetime,@EndTime datetime
SELECT @StartTime=GETDATE

'query in database for sample
select * from tblKala

'query in database
SELECT @EndTime=GETDATE
SELECT
DATEDIFF(ms,@StartTime,@EndTime)
```

Fig. 5 Code piece to send implementation time

We added this factor on the software and compared its results with the normal case, and the results obtained as follows.

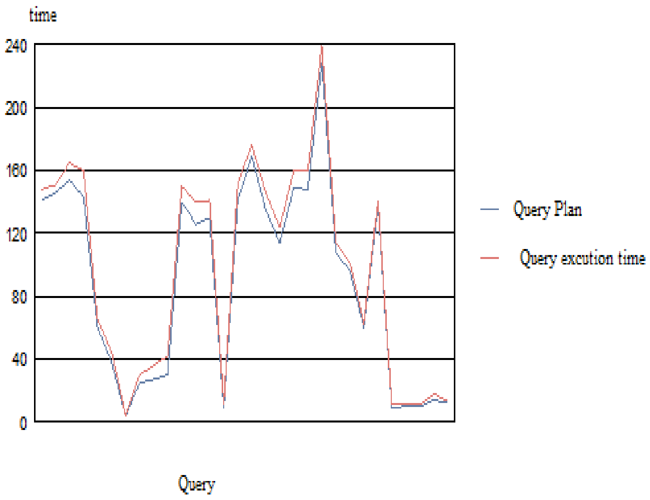


Figure 6: The reports of the response time per day for versatile queries (X: Questions sent toVersatile Database, Y: Sum of execution time)

Figure 6 shows the amount of time required to respond to queries and implementation plan.

Figure 7 shows the implementation in two modes along with plus the cost of versatility.

As shown in this figure for a sample question, in this algorithm, the difference between the start and end times of each algorithm's execution is the duration of the algorithm's execution, which is used in the evaluation.

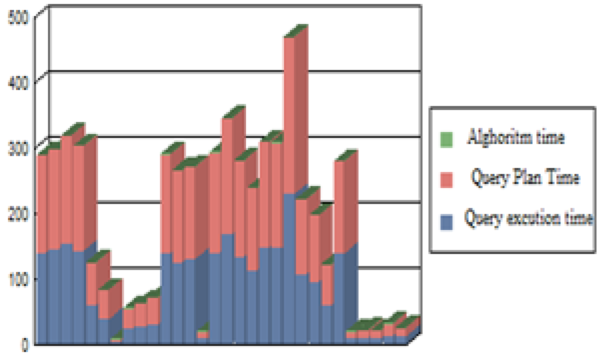


Fig. 7 The breakdown of implementation costs)X: Questions sent to Versatile Database in a day, Sum of Cost Time in a days)

Figure 7 represents the total time to respond to queries, reduction of response time to all versatile queries, as well as the cost required for the versatility of the queries. In this figure, the blue part is the execution time of the plan, the red part is the time of creating the implementation plan, and the green part is the policy duration of the proposed method. which has been tried over time to reduce the cost of creating plans for frequently used questions by maintaining plans for implementing frequently used questions.

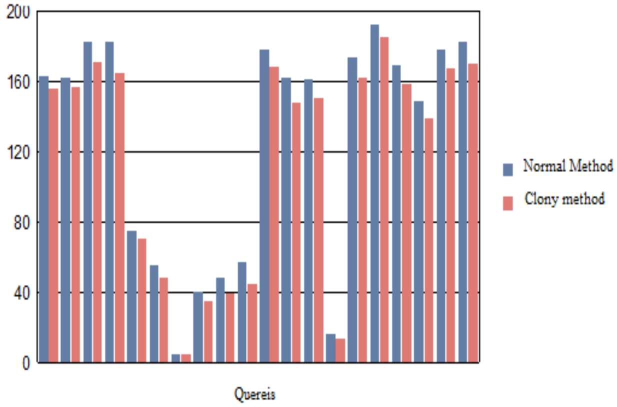


Fig. 8 The amount of reduction of the response time for versatile queries costs (X: Questions sent to Versatile Database in Days, Sum of execution time)

As is shown in this figure, the system has significantly reduced system response time to queries.

In this figure, it shows the duration of execution and reduction of the cost of execution of questions in normal mode and after applying the proposed method. In this algorithm, it shows the response time of two methods for the total number of questions sent to the database in a specified time.

5- Conclusion

Various methods have been proposed to optimize query processing in the database. These optimizations have been aimed at reducing processing time or reducing the use of resources. These methods try to make query processing versatile in the database.

This method for versatility of database is composed of three parts: separator, replacement policy, and query. To detect frequently-used queries in the database, Ant Colony algorithm is used. This method is based on the created pheromone to choose the optimal solution. The aim of this algorithm has been reducing implementation stages for frequently asked questions in the database.

The generality of this method is based on identifying frequent questions in the database. Then, by maintaining the query execution plan, it reduces the time in subsequent executions.

In fact, we get help from the Ant algorithm to identify repetitive questions and reduce the execution cost by caching the execution plan. At the same time, due to the fact that adaptability takes place at times with less load and based on the identified questions.

The results of this study demonstrate minimizing in implementation time of queries in 2.7%. Moreover, due to identifying the frequently-used queries in high traffic time and maintaining implementation plan, system traffic in times of high traffic load is reduced. In the future, the system can be made more optimized and be used for distributed database by developing the methods used. For example, machine learning methods can be used to identify execution patterns of plans.

Reference

[1] Saurabh gupta, Gopal Singh Tandel, Umashankar Pandey, "A Survey on Query Processing and Optimization in Relational Database Management System", International Journal of Latest Trends in Engineering and Technology, Vol. 5 Issue 1 January 2015

[2] Amol Deshpande, Zachary Ives, and Vijay Shankar Raman. Adaptive query processing. Foundations and Trends in Databases, 1(1), 2007. To appear. January 7-10, 2007, Asilomar, California, USA.

[3] Sybase, Performance and Tuning Series Query Processing and Abstract Plans, Sybase, Inc., One Sybase Drive, Dublin, CA 94568.

[4] Navid mohseni, mehdi mokhtarpor, hosein shirgah,"Application of Ant Colony Algorithm in data mining", National Conference on Emerging Trends in Computer Engineering and Data Recovery,2014

[5] Gupta, D.K.; Gupta, J.P.; Arora, Y.; Shankar, U., "Recursive Ant Colony optimization: a new technique for the estimation of function parameters from geophysical field data," Near Surface Geophysics , vol. 11, no. 3, pp.325-339

[6] Serap Ulusam Seçkiner, Yunus Eroğlu, Merve Emrullah, Türkay Dereli, Ant Colony optimization for continuous

functions by using novel pheromone updating, Applied Mathematics and Computation
Volume 219, Issue 9, 1 January 2013, Pages 4163-4175

[7] Xiao. M.Hu, J. ZHANG, and H. Chung, "An Intelligent Testing System Embedded with an Ant Colony Optimization Based Test Composition Method", IEEE Transactions on Systems, Man, and Cybernetics--Part C: Applications and Reviews, Vol. 39, No. 6, pp. 659-669, Dec 2009.

[8] B. Pfahring, "Multi-agent search for open scheduling: adapting the Ant-Q formalism," Technical report TR-96-09, 1996.

[9] A. Shmygelska, R. A. Hernández and H. H. Hoos, "An Ant Colony optimization algorithm for the 2D HP protein folding problem [permanent dead link]," Proceedings of the 3rd International Workshop on Ant Algorithms/ANTS 2002, Lecture Notes in Computer Science, vol.2463, pp.40-52, 2002.

[10] Gupta, D.K.; Arora, Y.; Singh, U.K.; Gupta, J.P., "Recursive Ant Colony Optimization for estimation of parameters of a function," Recent Advances in Information Technology (RAIT), 2012 1st International Conference on, vol., no., pp.448-454, 15–17 March 2012

[11] Zhang, Y. (2013). "A Rule-Based Model for Bankruptcy Prediction Based on an Improved Genetic Ant Colony Algorithm". Mathematical Problems in Engineering. 2013: 753251. doi:10.1155/2013/753251.

[12] Jevtić, A.; Melgar, I.; Andina, D. (2009). 2009 35th Annual Conference of IEEE Industrial Electronics. 35th Annual Conference of IEEE Industrial Electronics, 2009. IECON '09. pp. 3353–3358. doi:10.1109/IECON.2009.5415195. ISBN 978-1-4244-4648-3. S2CID 34664559.

[13] Warner, Lars; Vogel, Ute (2008). Optimization of energy supply networks using Ant Colony optimization (PDF). Environmental Informatics and Industrial Ecology — 22nd International Conference on Informatics for Environmental Protection. Aachen, Germany: Shaker Verlag. ISBN 978-3-8322-7313-2. Retrieved 2018-10-09.

[14] Zaidman, Daniel; Wolfson, Haim J. (2016-08-01). "PinaColada: peptide–inhibitor Ant Colony ad-hoc design algorithm". Bioinformatics. 32 (15): 2289–2296. doi:10.1093/bioinformatics/btw133. ISSN 1367-4803. PMID 27153578.

[15] J. ZHANG, H. Chung, W. L. Lo, and T. Huang, "Extended Ant Colony Optimization Algorithm for Power Electronic Circuit Design", IEEE Transactions on Power Electronic. Vol.24,No.1, pp.147-162, Jan 2009.

[16] L. Wang and Q. D. Wu, "Linear system parameters identification based on Ant system algorithm," Proceedings of the IEEE Conference on Control Applications, pp. 401-406, 2001.

[17] Martins, Jean P.; Fonseca, Carlos M.; Delbem, Alexandre C. B. (25 December 2014). "On the performance of linkage-tree genetic algorithms for the multidimensional knapsack problem". Neurocomputing. 146: 17–29. doi:10.1016/j.neucom.2014.04.069.

[18] L.M. Gambardella and M. Dorigo, "Solving Symmetric and Asymmetric TSPs by Ant Colonies", Proceedings of the IEEE Conference on Evolutionary Computation, ICEC96, Nagoya, Japan, May 20–22, pp. 622-627, 1996;

[19] Mohammad_Reza Feizi_Derakhshi, Hasan Asil, Amir Asil, “Proposing a New Method for Query Processing Adaption in DataBase, JOURNAL OF COMPUTING,NY,USA, VOLUME 2, ISSUE 1, JANUARY 2010, ISSN 2151-961

[20] Mohammad_Reza Feizi_Derakhshi, Hasan Asil, Amir Asil " Proposing a New Method for Query Processing Adaption in Data Base " WCSET 2009: World Congress on Science, Engineering and Technology Dubai, United Arab Emirates VOLUME 37, January 28-30, 2009 ISSN 2070-3740

[21] D. Martens, M. De Backer, R. Haesen, J. VAnthienen, M. Snoeck, B. Baesens, Classification with Ant Colony Optimization, IEEE Transactions on Evolutionary Computation, volume 11, number 5, pages 651—665, 2007.

[22] L.M. Gambardella and M. Dorigo, "Ant-Q: a reinforcement learning approach to the traveling salesman problem", Proceedings of ML-95, Twelfth International Conference on Machine Learning, A. Prieditis and S. Russell (Eds.), Morgan Kaufmann, pp. 252–260, 1995

[23] V. Donati, V. Darley, B. Ramachandran, "An Ant-Bidding Algorithm for Multistage Flowshop Scheduling Problem: Optimization and Phase Transitions", book chapter in Advances in Metaheuristics for Hard Optimization, Springer, ISBN 978-3-540-72959-4, pp.111-138, 2008.

[24] Warner, Lars; Vogel, Ute (2008). Optimization of energy supply networks using Ant Colony optimization (PDF). Environmental Informatics and Industrial Ecology — 22nd International Conference on Informatics for Environmental Protection. Aachen, Germany: Shaker Verlag. ISBN 978-3-8322-7313-2. Retrieved 2018-10-09.

[25] A. Deshpande and J. M. Hellerstein. Lifting the burden of history from adaptive query processing. In VLDB, 2004.

[26] Bingsheng He, Qiong Luo,“Cache-Oblivious Query Processing” Biennial Conference on nnovative Data Systems Research (CIDR)

[27] Wanhong Xu, “Xml Query Processing – SemAntic Cache System”, IJCSNS International Journal of Computer Science and Network Security, VOL.7 No.4, April 2007

[28] Elnaz zafarani , Mohammad_Reza Feizi_Derakhshi , Hasan Asil , Amir Asil “Presenting a New Method for Optimizing Join Queries Processing in Heterogeneous Distributed Databases

[29] Farhang Pedearan Moghadam, Hamid Maghsoudi, " Improved routing for load balancing in wireless sensor networks on the Internet of things, based on multiple Ant Colony algorithm", " Journal of Information Systems and Telecommunication (JIST) " Number 51, Volume 14

[30] Farid Ahmadi ,Mohammad Pourmahmood Aghababa , Hashem Kalbkhani, Nonlinear Regression Model Based on Fractional Bee Colony Algorithm for Loan Time Series, Journal of Information Systems and Telecommunication (JIST), 2022-04-21, Page: 141 – 1

[31] Luming Sun, Tao Ji, Cuiping Li, Hong ChenDeepO: A Learned Query Optimizer,SIGMOD '22: Proceedings of the 2022 International Conference on Management of DataJune 2022Pages 2421–2424https://doi.org/10.1145/3514221.3520167

[32] Ryan Marcus, Parimarjan Negi, Hongzi Mao, Nesime Tatbul, Mohammad Alizadeh, and Tim Kraska. 2021. Bao: Making Learned Query Optimization Practical. Proceedings of the 2021 International Conference on Management of Data (2021)

[33] Kristian F. D. Rietveld and Harry A. G. Wijshoff,Redefining The Query Optimization Process,IEEE TKDE 2022,arXiv:2203.01079v1,January 2022

[34] Yakov Kuzin, Anna Smirnova, Evgeniy Slobodkin, George Chernishev, Query Processing and Optimization for a Custom Retrieval Language,Proceedings of the First Workshop on Pattern-based Approaches to NLP in the Age of Deep Learning,October,2022

[35] Mohsin, S.A.; Younes, A.; Darwish, S.M. Dynamic Cost Ant Colony Algorithm to Optimize Query for Distributed Database Based on QuAntum-Inspired Approach. Symmetry 2021, 13, 70. https://doi.org/10.3390/sym13010070

[36] Zhiyong Ding,Study of Multi Ant Colony Genetic Algorithm in Query Optimization of Distributed Database, 2022 2nd International Conference on Artificial Intelligence and Advanced Manufacture (AIAM),202250,10.52547/jist.16015.10.38.141

[37] Mladineo, Marko; Veza, Ivica; Gjeldum, Nikola (2017). "Solving partner selection problem in cyber-physical production networks using the HUMANT algorithm". International Journal of Production Research. 55 (9): 2506–2521. doi:10.1080/00207543.2016.1234084. S2CID 114390939.

[38] Atieh Monemi Bidgoli, Hassan haghighi, Using Static Information of Programs to Partition the Input Domain in Search-based Test Data Generation, Journal of Information Systems and Telecommunication (JIST), 2021-01-13, Page: 219 – 229

[39] L. Bianchi, L.M. Gambardella et M.Dorigo, An Ant Colony optimization approach to the probabilistic traveling salesman problem, PPSN-VII, Seventh International Conference on Parallel Problem Solving from Nature, Lecture Notes in Computer Science, Springer Verlag, Berlin, Allemagne, 2002.

[40] Babu Kumar Ajay Vikram Singh Parul Agarwal, AI based Computational Trust Model for Intelligent Virtual AssistAnt, Journal of Information Systems and Telecommunication (JIST), Issue 32 Vol. 8 Autumn 2020, Page: 263 - 271 10.29252/jist.8.32.263,

[41] P. O'Neil and G. Graefe, "Multi-Table Joins Through Bitmapped Join Indices", ACM SIGMOD, 1995
", WKDD2010, Phuket, Thailand, 9-10 January, 2010.

[42] Rosa Karimi AdlEmail authorSeyed Mohammad Taghi Rouhani Rankoohi, "A new Ant Colony optimization-based algorithm for data allocation problem in distributed databases", Knowledge and Information Systems, September 2009, Volume 20, Issue 3, pp 349–373

[43] W. N. Chen and J. ZHANG "Ant Colony Optimization Approach to Grid Workflow Scheduling Problem with Various QoS Requirements", IEEE Transactions on Systems, Man, and Cybernetics--Part C: Applications and Reviews, Vol. 31, No. 1, pp.29-43, Jan 2009.

[44] Jevtić, A.; Quintanilla-Dominguez, J.; Cortina-Janich's, M.G.; Andina, D. (2009). Edge detection using Ant Colony search algorithm and multiscale contrast enhancement. IEEE International Conference on Systems, Man and Cybernetics, 2009. SMC 2009. pp. 2193–2198. doi:10.1109/ICSMC.2009.5345922. ISBN 978-1-4244-2793-2. S2CID 11654036.

[45] O. Okobiah, S. P. MohAnty, and E. Kougianos, "Ordinary Kriging Metamodel-Assisted Ant Colony Algorithm for Fast Analog Design Optimization Archived March 4, 2016, at the Wayback Machine", in Proceedings of the 13th IEEE International Symposium on Quality Electronic Design (ISQED), pp. 458--463, 2012.

[46] Gupta, D.K.; Arora, Y.; Singh, U.K.; Gupta, J.P., "Recursive Ant Colony Optimization for estimation of parameters of a function," Recent Advances in Information Technology (RAIT), 2012 1st International Conference on , vol., no., pp.448-454, 15–17 March 2012

[47] Muhammet Dursun Kaya, Hasan Asil, Dynamic Store Procedures in Database, German-Turkish Perspectives on IT and Innovation Management: Challenges and Approaches, 291-301,2018.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 29-40

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

TPALA: Two Phase Adaptive Algorithm based on Learning Automata for job scheduling in cloud Environment

Abolfazl Esfandi[1], Javad Akbari Torkestani[1*], Abbas Karimi[1], Faraneh Zarafshan[1]

[1.]Department of Computer Engineering, Arak Branch, Islamic Azad University, Arak, Iran.

Received: 30Aug 2022/ Revised: 04 Sep 2023/ Accepted: 21 Oct 2023

Abstract

Due to the completely random and dynamic nature of the cloud environment, as well as the high volume of jobs, one of the significant challenges in this environment is proper online job scheduling. Most of the algorithms are presented based on heuristic and meta-heuristic approaches, which result in their inability to adapt to the dynamic nature of resources and cloud conditions. In this paper, we present a distributed online algorithm with the use of two different learning automata for each scheduler to schedule the jobs optimally. In this algorithm, the placed workload on every virtual machine is proportional to its computational capacity and changes with time based on the cloud and submitted job conditions. In proposed algorithm, two separate phases and two different LA are used to schedule jobs and allocate each job to the appropriate VM, so that a two phase adaptive algorithm based on LA is presented called TPALA. To demonstrate the effectiveness of our method, several scenarios have been simulated by CloudSim, in which several main metrics such as makespan, success rate, average waiting time, and degree of imbalance will be checked plus their comparison with other existing algorithms. The results show that TPALA performs at least 4.5% better than the closest measured algorithm.

Keywords: Cloud Computing; Job scheduling; Learning Automata; Virtual Machine; CloudSim; Simulation.

1- Introduction

Cloud computing is a computer model that attempts to facilitate user access based on the type of demand they have from information and computing resources. This model tries to respond to the needs of users by reducing the need for human resources and costs as well as enhancing the speed of access to information [1]. By increasing the demand for running applications in the cloud, especially large and shared applications, Scheduling strategies for cloud requests have become very important. The key issue and challenge in cloud computing is ensuring the satisfaction of all users with cloud services. The scheduling plan consists of a scheduling algorithm that should plan the jobs and applications, so that users are satisfied and not harm the cloud manager at the same time[2]. Job scheduling is a key process in the IaaS layer that aims to execute logged requests to the system on the resources, in an efficient way by considering other features of the cloud environment. Job scheduling considers virtual machines as computing units to allocate heterogeneous physical resources for job execution. Each virtual machine is a unit containing computing and storage capabilities provided in the cloud[3]. Job scheduling in the cloud environment is NP-Hard due to its dynamic characteristics, heterogeneity, and varying workloads of users. In such a system, the scheduling algorithm must be done automatic and very quickly [4]. In cloud environments, according to the different needs of users, the workload of each user and as a result the computing resources required by them, which in our discussion are virtual machines, is different. Some users need more computing resources and others need fewer computing resources. In case of improper allocation of resources in any of the situations, the efficiency of the system will decrease significantly. Therefore, it is not possible to assign the same resources to all users. In addition, due to the dynamism of the cloud environment, the workload of users may change over time. Therefore, an efficient scheduling algorithm should dynamically allocate the most appropriate resource (virtual machine) to the jobs, according to the user's workload. Various algorithms have been presented for the job scheduling problem, but according to the review of the previous algorithms, it can be said that those algorithms

✉ **Javad Akbari Torkestani**
j-akbari@iau-arak.ac.ir

have not been able to fully adapt themselves to the dynamics of the cloud conditions and the diversity of resources [5].

The purpose of this paper is to propose an efficient algorithm for finding a near-optimal solution to the job scheduling problem in the cloud environment. Considering factors such as the fully dynamic environment of the cloud, the different capacity of virtual machines (VMs), the complexity and largeness of the requested jobs, the efficiency of cloud computing will be completely dependent on the existence of an effective scheduling algorithm.

The main difference between our proposed approach and previous methods that have used learning automata is in the way learning automata are employed and the type of learning automata used. Our proposed algorithm uses two different learning automata (LAs) for each scheduler to schedule the jobs optimally, because there are two main challenges for scheduling jobs generally. One challenge is choosing the proper job among the submitted jobs from users based on the priority and specific conditions of each job, and the other challenge is assigning that job to the most appropriate VM. Actually, according to these two challenges, our algorithm uses the two mentioned LA in two phases. The important contributions of this paper are as follows:

- Using two different learning automata (LAs) for each scheduler to optimally schedule the jobs
- Creating a list of ready-to-use virtual machines simultaneously with the first phase of the algorithm.
- Applying the variable learning automaton in the second phase increases the speed of convergence.
- Providing an online adaptive job scheduling method for cloud environment by using the two different LAs in two phases.
- Dynamically assigning workload to each virtual machine based on the VM's status and the submitted job's conditions.
- Using a hybrid set of jobs for simulations based on two factors: data volume and computational volume.
- Simulating the proposed algorithm using the CloudSim toolkit under different scenarios and comparing it to other scheduling algorithms.

Regarding the paper structure, Section 2 will first provide a comprehensive overview of the relevant and existing works in job scheduling concept in cloud. Then, Section 3 discusses the learning automata and its features. Section 4 describes our new proposed algorithm, and then, the implementation as well as experimental results of the simulation with the CloudSim toolkit and comparing it with other existing algorithms is shown in Section 5. Finally, Section 6 concludes the paper.

2- Related Works

From the birth of cloud computing to this day, there has always been extensive research into scheduling, and as a result, different methods and algorithms have been introduced. Some methods use heuristic algorithm such as Scheduling based on Min-Min [6] or Min-Max [7] Strategy. Researchers in [8] and [9] have investigated that in methods based on meta-heuristics optimization algorithms such as GA. papers [10-12] was presented its methods based on GA. In paper [13], the ant colony optimization algorithm (ACO) is used, in which several artificial ants generate distinct responses randomly based on the amount of pheromones. In [14], the ACO search method with GA was used for the job scheduling problem. In ref [15] an optimization strategy is proposed by using improved ACO algorithm for task scheduling in cloud. In paper [16] a metaheuristics method named GWOA is described that firstly proposed to overcome the early convergence problem and then create a balance between the local and the global search. A hybrid method combining the bee optimization and whale optimization model is proposed in the paper [17]. A fuzzy clustering method is used in [18] as a pre-processing operation to classify cloud resources; then directed graphs are used to schedule jobs to run on distinct clusters of hardware resources. In paper [19] was used fuzzy control theory for accomplish system accessibility between user requirements and users resources availability. In [20], the particle swarm optimization (PSO) has been used as an optimal answer searching method for the optimization problem and paper [21] was proposed hybrid task scheduling method by PSO and GA. Also paper [22] is presented a survey of scheduling algorithms based on PSO in cloud computing. In paper [23] a Hybrid Particle Swarm Optimization (HPSO) based scheduling was presented to optimal job scheduling in the cloud. HPSO against PSO was improved the performance of job scheduling issue by changing the various factors. paper [24] was proposed a hybrid job scheduling algorithm based on Fuzzy system and Modified PSO technique to improve load balancing and throughput in cloud environment. Also paper [25] by using integrated particle swarm algorithm and ant colony algorithm, proposed a efficiency algorithm for task scheduling. In paper [26] was used A Bidirectional Search Algorithm for Flow Scheduling in Cloud Data Centers. This approach was used for static scheduling and reduced makespan. In paper [27], a hybrid method named FUGE based on the fuzzy and GA was presented that reduce the execution time and costs. A combined method of the cellular automata and the bat algorithm (BatCL) was presented in paper [28] which aimed to reduce the cost and time of completion of tasks. Paper [29] proposes a hybrid approach for job scheduling in cloud computing, utilizing a combination of sparrow search algorithm and differential evolution optimization. A combination of the genetic algorithm and the gravitational emulation local search is presented in [30] for task scheduling in the cloud. In [31], a new method was proposed based on LA for energy-aware task scheduling to minimize energy consumption and task completion time in cloud environment. This paper presented a scheduling architecture by LA for optimal job assignment. The paper [32] described a new LA-based scheme in which load distribution was performed in such a way that the level of efficiency in different nodes was almost the same which also used paradigm of two-time scales to achieve its purpose. A LA Based algorithm for container cloud was presented in [33] which designed a task load monitoring framework for usage in real-time monitoring of resource and scheduling evaluation feedback, develop an intelligent scheduling technique, and enhance the performance.

Articles, [34-38], had a brief overview on existed scheduling algorithms.

In table 1, the related works have been categorized into four groups and their general advantages and disadvantages have been stated for each category.

Table 1: Job scheduling algorithms comparison

References	Category	Advantages	Disadvantages
[6, 7]	Heuristic Algorithms	•simplicity of the algorithm structure	• static scheduling • low compatibility with dynamic environments • single goal
[18, 19, 24, 27]	Fuzzy Theory	•making best decision based on inputs	• mostly using in combination • using for static environments
[13, 15-17, 20, 23, 29, 30]	Meta-heuristic Optimization	•searching optimal solution from all solution space •obtain better optimization effect	• complexity scheduling • long optimization time • getting caught in local optimal solutions
[31-33]	Automata Theory	•proper for dynamic environments •global optimization ability •suitable for large-scale job scheduling	• weak directivity in optimal solution searching • Easy to fall into the local optimum

3- Learning Automata theory

Learning automaton is one of the reinforcement learning techniques in artificial intelligence. Learning automata's learning ability in unknown environments is a useful technique for modeling, controlling, and solving many real problems in the distributed and decentralized environments[39]. The environment responds to the action taken in turn with a reinforcement signal. The action probability vector is updated based on the reinforcement signal back from the environment. The purpose of a LA is finding the optimal action from the action set so it was received minimized average penalty from the environment[40].

Figure 1 illustrates interactive between a learning automaton and its random environment that vector α(n) is an action vector, vector β(n) is a reinforcement signal and vector x(n) is called context vector. In nth iteration, an automaton receives x(n) from the environment. Depending on x(n), LA chooses one of its possible actions[41].

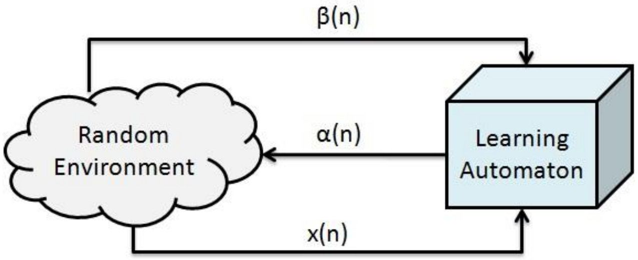


Fig. 1: interactive between a LA and its random environment

The environment can be mathematically modeled by quintuple $< \underline{X}, \underline{\alpha}, \underline{\beta}, \underline{F}, \underline{q} >$ where:

$\underline{X}$: is the set of context vectors

$\underline{\alpha}$: is the set of inputs

$\underline{\beta}$: is the set of values that can be taken by the reinforcement signal $\underline{F}$

$\underline{F}$: is the set of probability distributions

$\underline{q}$: is the probability distributions defined over $\underline{X}$ which is assumed to be unknown.

$\alpha(n)$ and $\beta(n)$ denote the input and output of environment at discrete time n ($n \geq 0$)[42].

The learning algorithm used to update the action probability based on a recurrence relation. Let $\alpha_i(n) \in \underline{\alpha}$ denotes the action selected by LA and P(n) denotes the probability vector defined over $\underline{\alpha}$ at instant n. Let a and b denote the reward and penalty parameters and r be shown the number of actions that can be taken by LA. At each instant n, if the selected action $\alpha_i(n)$ is rewarded by random environment, the action probability vector P(n) is modified by the linear learning algorithm given in Eq.1 and if the taken action is penalized, it is modified as given in Eq.2.

$$P_i(n+1) = P_i(n) + a[1 - P_i(n)]$$
$$P_j(n+1) = (1-a)P_j(n) \qquad \forall j; j \neq i \qquad (1)$$

$$P_i(n+1) = (1-b)P_i(n)$$
$$P_j(n+1) = \frac{b}{r-1} + (1-b)P_j(n) \qquad \forall j; j \neq i \qquad (2)$$

If a and b be equal, Eq.1 and Eq.2 are called a linear reward-penalty (L_{R-P}) algorithm, if a be more greater than b those equations are called linear reward-Ɛ penalty ($L_{R-\varepsilon P}$) and if b=0 they are named linear reward-inaction (L_{R-I})[41]. A variable action-set learning automaton (VLA) is a LA that the number of its actions maybe varies with time. If $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_r\}$ be action-set of VLA, $A = \{A_1, A_2, \dots, A_m\}$ is the set of action subsets and $A(n) \subseteq \alpha$ denote the subset of all the available actions for choose by the VLA, at instant n.

The special action subsets are chosen randomly by an external agency according to the probability distribution $\psi(n) = \{\psi_1(n), \psi_2(n), \dots, \psi_m(n)\}$ so that $\psi_i(n) = prob[A(n) = A_i | A_i \in \mathrm{A}, 1 \leq \mathrm{i} < 2^{\mathrm{r}}]$.

Let $\hat{p}_i(n) = prob[\alpha(n) = \alpha_i | A(n), \alpha_i \in A(n)]$ be the probability of selecting action α_i if the action subset A(n) has already been selected and $\alpha_i \in A(n)$. That is defined as:

$$\hat{\mathrm{p}}_{\mathrm{i}}(\mathrm{n}) = \mathrm{p}_{\mathrm{i}}(\mathrm{n}) / \mathrm{K}(\mathrm{n}) \qquad (3)$$

Where $p_i(n) = prob[\alpha(n) = \alpha_i]$ and K(n) is the sum of the probabilities of the actions in subset A(k) which defined as:

$$\mathrm{K}(\mathrm{n}) = \sum\nolimits_{\alpha_{\mathrm{i}} \in \mathrm{A}(\mathrm{n})} \mathrm{p}_{\mathrm{i}}(\mathrm{n}) \qquad (4)$$

The function of a VLA is as follows: Before selecting an action from the selected subset, the probabilities of all the actions in the A(n) are calculated as defined in Eq.(3). Then VLA randomly selects one of actions of the A(n) based on the scaled action probability vector $\hat{p}_n$. Depending on the response received from the environment, the VLA updates its scaled action probability vector (only the available actions). Finally, the probability vector of the actions of the A(n) is rescaled as:

$$\mathrm{p}_{\mathrm{i}}(\mathrm{n}+1) = \hat{\mathrm{p}}_{\mathrm{i}}(\mathrm{n}+1) * \mathrm{K}(\mathrm{n}), \ \forall \alpha_{\mathrm{i}} \in \mathrm{A}(\mathrm{n}) \qquad (5)$$

for more details and Proof refer to [43].

4- Proposed Algorithm

As mentioned, various algorithms have been proposed for job scheduling in the cloud environment so far. However, most of them are not efficient enough due to the dynamic nature of the cloud environment, such as resource dynamics and network conditions. To represent our proposed method, First, we explain the cloud environment, the desired scheduling structure, and the general method of our algorithm, then we describe how each of the LA and the two mentioned phases work.

Let cloud contains several hosts that each of them has multiple virtual machines (VMs), so that the sum of all virtual machines in all Hosts is considered as m virtual machines. The intended resources are in the form of these virtual machines. We also consider k schedulers for scheduling submitted users jobs, each scheduler $\mathrm{s_i} \subseteq S$ is associated with one or more VMs and each virtual machine $VM_{\mathrm{j}} \subseteq VM$ can also be connected to one or more schedulers. Also, each virtual machine has set of processing element PE_j so that each processing element $\mathrm{PE_j(k)}$ has different processing powers. In this algorithm, two separate phases and two different LA are used to schedule jobs and allocate each job to the appropriate VM, so that a two phase adaptive algorithm based on LA is presented, which we call TPALA. In our algorithm, each scheduler $\mathrm{s_i} \subseteq S$ has two LA. The first LA (LA_i^{job}) has the task of selecting and receiving the submitted jobs by users with different workloads, and the second LA ($\mathrm{LA_i^{VM}}$) has the task of optimal allocating the jobs to proper VMs based on their computing capacity. To better understand the proposed algorithm, in the Table 2, the list of most symbols used and their meaning is brought.

Table 2: Symbols specifications

Symbols	**Description**
LA_i^{job}	First LA on scheduler $\mathrm{S_i}$ for selecting the jobs
$\mathrm{LA_i^{VM}}$	Second LA on scheduler Si for optimal allocating the job to proper VM
$\mathrm{RU_i}$	List of ready-to-use virtual machines on scheduler $\mathrm{S_i}$
$\mathrm{J_i^{sel}}$	Selected job by $\mathrm{LA_i^{job}}$ on scheduler $\mathrm{S_i}$
$\mathrm{VM_i^{opt}}$	Selected VM by $\mathrm{LA_i^{VM}}$ on scheduler $\mathrm{S_i}$
$\mathrm{\alpha_i^{job}}$	The action-set of learning automaton $\mathrm{LA_i^{job}}$
$\mathrm{\alpha_i^{job}(j)}$	Corresponding action of $\mathrm{LA_i^{job}}$ to user jth
$\mathrm{P_i^{job}}$	The action probability vector of $\mathrm{LA_i^{job}}$
$\mathrm{PE_j}$	Set of processing elements of $\mathrm{VM_j}$
$\mathrm{PE_j(k)}$	K$^{\mathrm{th}}$ processing elements of $\mathrm{VM_j}$

$T_i^{sel}(l)$	l^{th} task of job J_i^{sel}
α_i^{VM}	The action-set of learning automaton LA_i^{VM}
$\alpha_i^{VM}(j,k,l)$	Corresponding action of LA_i^{VM} to $PE_j(k)$ for allocating to task $T_i^{sel}(l)$.
$RC_i^{sel}(l)$	The required capacity for the task $T_i^{sel}(l)$
$AC_j(k)$	The available capacity of the processing unit $PE_j(k)$
$\overline{X}_i(AC)$	The average available capacities of all VM which have related to scheduler S_i

Now, according to what that has mentioned, we are going to describe TPALA algorithm step by step. As shown in the flowchart in Figure 3, in each stage, first, each VMs corresponding to the scheduler S_i, which has completed its previous work or is idle (has ready PE to work), generates a “request” signal for a new job, including its own characteristic, and send it for Si. In fact, in this way, a list of ready-to-use virtual machines is created, which we call RUi. Also, in RUi, the usable computing capacity of the VMs corresponding to the scheduler S_i is specified.

As it will be said in the explanations of the second phase, the existence of this list (RU_i) reduces the number of action-set used in the second phase and increases the rate and speed of convergence. After send request signal, VM waits for receive proper job or “retry” signal.

```
Input:
Output: list of ready-to-use VMs (RU_i)
For each $VM_j \subseteq$ VMs corresponding to $S_i$ do
    If $VM_j$.IsReady(PE) == True then
        $VM_j$.Send(“request”, $S_i$)
        $VM_j$.Wait(response, $S_i$)
    End If
End For
$S_i$.Collects(“request”)
$RU_i \leftarrow S_i$.Create(list of ready-to-use VMs)
Return $RU_i$
```

Fig. 2: Pseudo code of create RUi for scheduler S_i

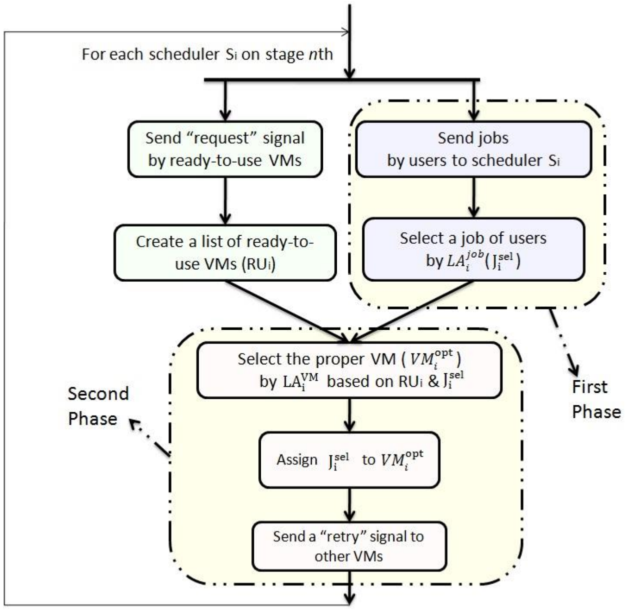

For each scheduler Si on stage nth
Send “request” signal by ready-to-use VMs
Send jobs by users to scheduler Si
Create a list of ready-to-use VMs (RUi)
Select a job of users by $LA_i^{job}(J_i^{sel})$
Select the proper VM (VM_i^{opt}) by LA_i^{VM} based on RUi & J_i^{sel}
Second Phase
First Phase
Assign J_i^{sel} to VM_i^{opt}
Send a “retry” signal to other VMs

Fig. 3: General Flowchart of TPALA algorithm

Simultaneously with this event, in the first phase, for each scheduler $s_i \subseteq S$ by using LA_i^{job} that its operation will be explained later (see pseudo code of Figure 4), one of the jobs sent by users to scheduler S_i is selected (J_i^{sel}). In the second phase of TPALA algorithm, for scheduler S_i, According to the selected job (LA_i^{job}) in the first phase and the list RU_i, by using LA_i^{VM} that its operation will be explained later (see pseudo code of Figure 5), based on the characteristics of the VMs and their computational capacity against the required computational capacity of selected job, J_i^{sel} is assigned to the most suitable VM. After selecting the proper VM (VM_i^{opt}), J_i^{sel} is assigned to VM_i^{opt} and a retry signal is sent to other VMs. These steps are repeated until all the users' jobs are serviced.

Now that the general structure of the TPALA algorithm has been explained, we will describe the details of the operation of each of the phases, that is, the operation of the two LAs related to each of schedulers.

As it was said, in the first phase for schedule S_i, learning automaton LA_i^{job} is used to select J_i^{sel}. Scheduler Si may receive jobs from several users. Let $U = \{U_1, U_2, \dots, U_R\}$ be the set of users that corresponded to scheduler S_i.

The action-set of learning automaton LA_i^{job} has R actions (one action for any user). That is defined as:

$$\alpha_i^{job} = \left\{\alpha_i^{job}(j) \middle| \forall j : j = 1,2,3,\dots,R\right\} \quad (6)$$

If action $\alpha_i^{job}(j)$ be selected, it is means that scheduler S_i allows user U_j to submits its job. Also the action probability vector of LA_i^{job} is defined as:

$$\mathrm{P}_{\mathrm{i}}^{\mathrm{job}} = \left\{\mathrm{p}_{\mathrm{i}}^{\mathrm{job}}(\mathrm{j}) \middle| \forall \mathrm{j} : \mathrm{j} = 1,2,3,\ldots,\mathrm{R}\right\} \qquad (7)$$

Since we have R actions, all Actions will have same probability $1/_{\mathrm{R}}$. Therefore all users in starting point have the same chance to send their jobs to cloud. If priority is considered for user jobs, instead of considering the equal probability for all actions, different values can be considered according to the priority of jobs and the initial probability of jobs with higher priority can be considered slightly higher. But in any case, the sum of the possibilities must be equal to one, that is:

$$\forall \mathrm{i} \in \{1,2,\ldots,\mathrm{k}\} \ : \ \textstyle\sum_{\mathrm{j}=1}^{\mathrm{R}} \mathrm{P}_{\mathrm{i}}^{\mathrm{job}}(\mathrm{j}) = 1 \qquad (8)$$

For scheduler $\mathrm{S_i}$ at any stage, $\mathrm{LA_i^{job}}$ as per its action probability vector select one of its actions randomly. Let action $\alpha_{\mathrm{i}}^{\mathrm{job}}(\mathrm{j})$ be selected, then scheduler Si checks user Uj that it has a ready job to send. If user Uj could be submit a job ($\mathrm{J_i^{sel}}$), selected action $\alpha_{\mathrm{i}}^{\mathrm{job}}(\mathrm{j})$ gets rewards and increases action $\mathrm{p_i^{job}}(\mathrm{j})$ by Eq.(1). Otherwise selected action $\alpha_{\mathrm{i}}^{\mathrm{job}}(\mathrm{j})$ gets penalty and decreases action $\mathrm{p_i^{job}}(\mathrm{j})$ by Eq.(2). Pseudo code of first phase is shown in Figure 4. In any case (reward/penalty) after updating the internal state of $\mathrm{LA_i^{job}}$, this stage will be end and start next stage and repeat same method. By applying of this method, effectively workload on users can be balanced.

```
Input: jobs queue
Output: a job form jobs queue of $S_i$ ($J_i^{sel}$)
$LA_i^{job}$.Initial()
$\alpha_i^{job}(j) \leftarrow LA_i^{job}$. Select-Action($P_i^{job}$)
If $U_j$.HasReady-job == True then
    update $LA_i^{job}$ by Eq.1 //reward
    $J_i^{sel} \leftarrow S_i$.Selected-job()
    Return $J_i^{sel}$
Else
    update $LA_i^{job}$ by Eq.2 //penalty
End If
```

Fig. 4: Pseudo code of one round of first phase of TPALA algorithm

After J_i^{sel} determined, second phase by learning automaton LA_i^{VM} can be run. In this phase LA_i^{VM} find a near optimal solution to allocate J_i^{sel} to proper VM based on their computational capacities.

Scheduler S_i has related to several VMs on cloud environment. Let $\{VM_1, VM_2, \ldots, VM_N\}$ be the set of VMs that corresponded to scheduler $\mathrm{S_i}$.

Every virtual machine $\mathrm{VM_j}$ have set of processing elements PE_j so that each processing element $\mathrm{PE_j(k)}$ has different processing powers. This set defined as:

$$PE_j = \left\{PE_j(k) \middle| \forall k : k = 1,2,3,\ldots,m\right\} \qquad (9)$$

Other hand, each job J_i^{sel} divided to several tasks $T_i^{sel}(l)$ ($\forall l \in \{1,2,..,H\}$). In this phase, purpose of learning automaton LA_i^{VM} is to find optimal scheduling method to allocate a processing element $PE_j(\mathrm{k})$ from $\mathrm{RU_i}$ to any task $T_i^{sel}(l)$. The used learning automaton LA_i^{VM} is a VLA and its action-set is defined as:

$$\alpha_i^{VM} = \left\{\alpha_i^{VM}(\mathrm{j,k,l}) \middle| \forall\, PE_j(k) \in PE_j\right\} \qquad (10)$$

If action $\alpha_i^{VM}(\mathrm{j,k,l})$ be selected, it is means that scheduler S_i chooses processing element $PE_j(\mathrm{k})$ to allocated to any task $T_i^{sel}(l)$. As mentioned, we use a VLA in the second phase. The reason is that due to the non-constant number of actions in this type of automata, the rate and speed of convergence increases and it has a more dynamic structure.

To bound the action-set of LA_i^{VM}, let $RC_i^{sel}(l)$ be the required capacity for the task $T_i^{sel}(l)$ and $AC_j(k)$ is the available capacity of the processing unit $PE_j(k)$, if $RC_i^{sel}(l)$ is less than or equal to $AC_j(k)$, then $PE_j(k)$ can be allocated to the $T_i^{sel}(l)$. Otherwise, there is no possibility of allocation. Therefore, the existence of the action $\alpha_i^{VM}(\mathrm{j,k,l})$ is not useful and it can be deleted from the action-set of LA_i^{VM} as explained in section 3.

Therefore for each task $T_i^{sel}(l)$, action-set Learning automaton LA_i^{VM} will be updated as:

$$\alpha_i^{VM} \leftarrow \alpha_i^{VM} - \left\{\alpha_i^{VM}(\mathrm{j,k,l}) \middle| RC_i^{sel}(l) \le AC_j(k)\right\} \qquad (11)$$

Let AC_j be the sum of available capacities of virtual machine VM_j that is defined as:

$$AC_j = \textstyle\sum_{k=1}^{N_j} AC_j(k) \qquad (12)$$

Where N_j is the number of process elements in VM_j. Also, let $\bar{X}_i(AC)$ be the average available capacities of all VM which have related to scheduler S_i. It's defined as:

$$\bar{X}_i(AC) = {\textstyle\sum_{j=1}^{N} AC_j}\Big/_{N} \qquad (13)$$

Learning automaton LA_i^{job} chooses one of actions as per its updated probability vector. Let action $\alpha_i^{VM}(\mathrm{j,k,l})$ be selected, then scheduler S_i checks to see if AC_j is greater than $\bar{X}_i(AC)$. If so, selected action $\alpha_i^{VM}(\mathrm{j,k,l})$ gets rewards and increases its choice probability by Eq.1. Otherwise selected action $\alpha_{\mathrm{i}}^{\mathrm{VM}}(\mathrm{j,k,l})$ gets penalty and decreases its choice probability by Eq.2. Pseudo code of second phase is shown in Figure 5.

```
Input: $J_i^{sel}$ And $RU_i$
Output: Allocate $J_i^{sel}$
$LA_i^{VM}$.Initial(RUi)
For each $\alpha_i^{VM}(j,k,l) \in$ action-set($LA_i^{VM}$) do
    If $RC_i^{sel}(l) \leq AC_j(k)$ then
        $LA_i^{VM}$.Remove($\alpha_i^{VM}(j,k,l)$)
    End If
End For
$\alpha_i^{VM}(j,k,l) \leftarrow LA_i^{VM}$. Select-Action($P_i^{VM}$)
If $AC_j > \overline{X}_i(AC)$ then
    update $LA_i^{VM}$ by Eq.1 //reward
Else
    update $LA_i^{VM}$ by Eq.2 //penalty
End If
$T_i^{sel}(l)$.Allocate( $PE_j(k)$ )
For each $VM_i \subseteq VM$ do
    $VM_i$.Send("retry")
End For
$LA_i^{VM}$.Restore-Actions()
```

Fig. 5: Pseudo code of one round of second phase of TPALA algorithm

In any case, scheduler S_i allocate process element $PE_j(k)$ to task $T_i^{sel}(l)$ and send a "retry" signal for any other VMs. Finally, at the finish of each allocation, all deleted actions must be added as explained in sec 3. By using this method, the submitted workloads will be distributed on different VMS based on its computational capacity and it is minimized the average running time of user's jobs.

5- Simulation and Experimental Results

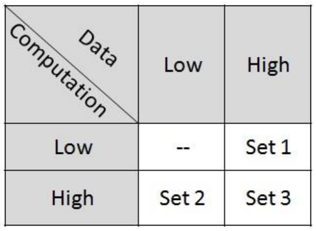
In this section, we describe CloudSim Toolkit, our simulation parameters and experimental results of comparing our proposed algorithm with three algorithms BatCL, FUGE, and HPSO on several evaluation metrics. The reason for choosing these algorithms to compare with our algorithm is that FUGE is a hybrid method based on fuzzy logic and genetic algorithms, HPSO is a metaheuristic algorithm, and BatCL is a cellular automata scheduling method. Therefore, these algorithms were selected from various algorithm categories.

5-1- CloudSim Toolkit

In this paper, the CloudSim Simulator was used to modeling and evaluation our proposed algorithm, because CloudSim simulator is top of simulator tools for cloud computing, that was developed in the Department of Software Engineering and Computer Science at the University of Melbourne [44]. This simulator is used in many industries and famous universities in the world to simulate cloud-based algorithms. Main limited of CloudSim is the lack of proper graphical user interface (GUI). CloudSim architecture has four layers which at now SimJava and GridSim layers are combined and it has changed to three layers architecture. In this version of CloudSim, uses SimJava as distinct event simulation engine that provides several services and process like: event and queuing processing, progression of cloud system elements e.g. hosts, datacenters, brokers and virtual machines[45].

5-2- Simulation Parameters

To simulate our proposed method and compare it with several others scheduling algorithms, we have used different structures and resources distributions in our cloud model which is implemented in the CloudSim simulator environment. We use five different scenarios with different numbers of jobs that in the smallest case we will have 100 input jobs and in the largest case we will have 500 input jobs. In addition, in each of these cases, for a more accurate comparison, jobs are based on two factors: data volume and computational volume. According to Figure 6, generated jobs are divided into three different categories: Set1: jobs with high data volume and low computational volume. Set2: jobs with low data volume and high computational volume. Set3: jobs which have a high data and computational volume.



Data Computation	Low	High
Low	--	Set 1
High	Set 2	Set 3

Fig. 6: Three different categories for generated jobs

It should be noted that there is a fourth case in which works have a low data volume and low computational volume, which we have not considered in our simulations because in general in small cases (whether in terms of number The jobs and in terms of data volume and computational volume) performance of most scheduling algorithms are the same and are not worth examining and cannot be used as an acceptable criterion. An example of a combination of these three sets of jobs used with their specifications is given in Table 3. For our simulation, we randomly generated the required number of cases in each case from this set of jobs.

Table 3: Typical jobs characteristics

Type	Job ID	Length (MI)	Num CPUs	File Size	Output Size
Set 1	0	5000	1	6000	500
Set 1	1	10000	1	8000	550
Set 1	2	20000	1	12000	650
Set 1	3	25000	1	14000	700
Set 2	5	60000	2	600	100
Set 2	5	65000	2	700	120
Set 2	6	75000	3	900	160
Set 2	7	80000	3	1000	180
Set 3	8	25000	1	4000	150
Set 3	9	30000	1	4500	160
Set 3	10	35000	2	5000	170
Set 3	11	45000	2	6000	190

In the simulation, three datacenters each with several hosts were used. The characteristics of datacenters and hosts are listed in Table 4 and Table 5 respectively. VMM and OS of all datacenters are Xen and Linux respectively. It is important to note that another very important class used in the CloudSim simulator is the Processing Element (PE), which is related to the hosts and this class represents the processing units or CPUs. This feature is expressed by the millions instructions per second (MIPS) factor and it’s listed in the hosts characteristics table.

Table 4: Datacenters characteristics

DC_ID	DC_Name	Architect	Cost per Memory	Cost per Storage
0	DataCenter_0	x64	0.05	0.001
1	DataCenter_1	x64	0.06	0.0015
2	DataCenter_2	x64	0.04	0.002

Where DS is datacenter and BW is bandwidth. Each host has 10 to 20 virtual machines that typical VMs characteristics were shown in table 6.

Table 5: Hosts characteristics

Host ID	DC_Name	MIPS	RAM (MB)	Storage (GB)	BW (Mbps)
0	DataCenter_0	6200	2048	500	500
1	DataCenter_0	7500	1024	1000	500
2	DataCenter_0	8000	4096	1000	500
3	DataCenter_1	4200	4096	500	500
4	DataCenter_1	5000	8192	1500	500
5	DataCenter_1	12100	8192	1500	500
6	DataCenter_2	7100	2048	1500	500
7	DataCenter_2	9495	2048	1500	500
8	DataCenter_2	8500	8192	1500	500
9	DataCenter_2	11900	2048	1500	500
10	DataCenter_2	12100	4096	1000	500

Table 6: Typical VMs characteristics

VM_ID	MIPS	Num CPUs	RAM (MB)	BW (Mbps)	Size
0	600	1	256	50	8000
1	1200	1	1024	50	15000
2	1000	2	512	50	10000
3	800	2	1024	50	20000
4	1200	2	512	50	12000

5-3- Experimental Results

To evaluate and compare the performance of scheduling algorithms, there are various metric that we have used here some of the main and important metric in this field to show the advantage of our proposed algorithm compared to the three algorithms: FUGE [27], HPSO [23] and BatCL[28] that we mentioned in Section 2. In the following, we first describe each of the metrics which used and then express the results of our simulation in the cloud environment using the CloudSim simulator. The four metrics used in this article are: makespan, success rate, average waiting time and degree of imbalance.

5-3-1- Makespan

Makespan is the most common measurement parameter of the optimization methods. This metric is defined as the maximum running time between submitted jobs. In other words, it indicates when the last job was completed. Minimizing this parameter indicates that things are not done in a long time. In our work, makespan is measured in milliseconds. The lower the value of this metric, it means that it has a better scheduling algorithm and the algorithm was able to process the jobs and deliver them to the users sooner. The value of this metric can be calculated based on the Eq.(14) [46]:

$$\text{Makespan} = \max\{\, C_i \;, i = 1,2,\dots,n\} \tag{14}$$

Where, n is number of jobs and C_i is completion time of job *i*th. In Figure 7 shows the average makespan of our algorithm and three other algorithms under different number of jobs. As the number of jobs increases, so does the makespan. When the number of jobs is low, the makespan of all scheduling algorithms is almost close to each other, and as the number of input jobs increases, the difference between the results increases and the TPALA algorithm will have more improvements, in which case the advantage of our proposed method becomes more apparent due to the use of LA. The results show that the TPALA improved the makespan by an average of 4.53% when compared with BatCA and respectively by an average of 10.88% and 19.62% when compared with FUGE and HPSO.

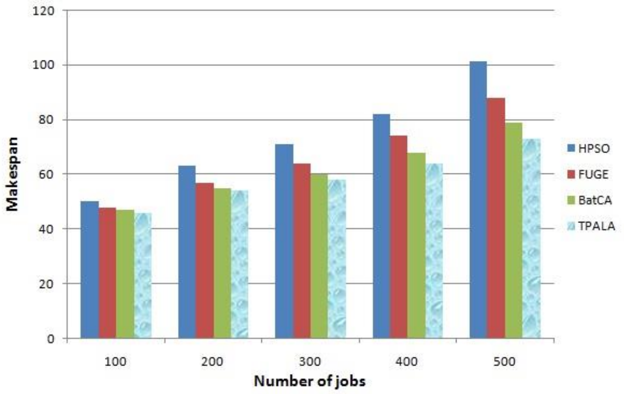


Fig. 7: makespan under number of jobs

5-3-2- Success Rate

Success rate is the fraction or percentage of success among a number of attempts therefore success rate in job scheduling is the ratio of number of successfully completed job to the all number of submitted jobs. So success rate is calculated based on the Eq.(15) [8]:

$$\text{success rate} = \frac{\text{successfully completed job}}{\text{All submitted job}} \tag{15}$$

The higher value of this metric shows the more successful the scheduling.

In Figure 8 shows the success rate under different numbers of jobs. As can be seen, with increasing the number of jobs, the success rate decreases. Especially in the case of 500 input jobs, there is a more significant reduction of this metric in all scheduling algorithms. The results show that the TPALA has been better performance in large number of jobs than other algorithms and HPSO had the worst performance, So that the success rate of TPALA was on average 20.59% better than HPSO.

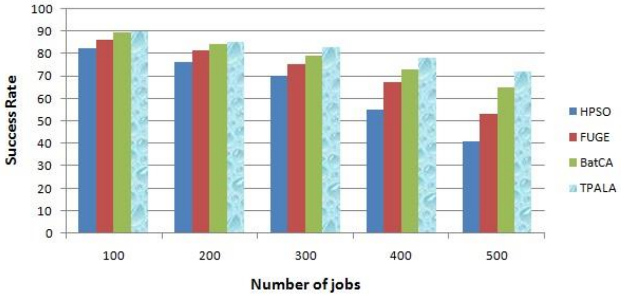


Fig. 8: success rate under number of jobs

5-3-3- Average Waiting time

The waiting time (WT_i) is the amount of time that a process waits in i^{th} run of scheduling algorithm for completion from its submission to completion. The average waiting time is the mean of waiting times in several run of scheduling. One of the goals of the scheduling algorithms is to reduce the waiting time. This metric is calculated based on the Eq.(16) [47]:

$$AWT = \frac{\sum_{i=1}^{n} WT_i}{n} \tag{16}$$

Where, n is number of jobs and WT_i is waiting time for process i^{th}. In Figure 9 shows the average waiting time under different numbers of jobs. As expected, it shows that in all scheduling algorithms, the average waiting time increases as the number of jobs increases. When the number of jobs is low, the average waiting time of all scheduling algorithms is acceptable and almost close to each other, But as the number of jobs increases, time differences in different algorithms become apparent. The results show that the TPALA has least average waiting time compare to other algorithms and BatCA performance is better than FUGE. The average waiting time of TPALA was on average 5.22% better than BatCA and BatCA was on average 7.25% better than FUGE.

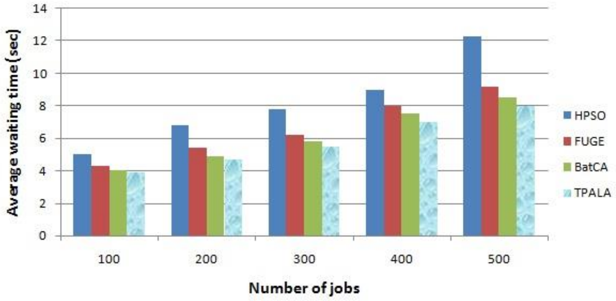


Fig. 9: average waiting time under number of jobs

5-3-4- Degree of Imbalance

Degree of imbalance is the metric for measuring the imbalance among virtual machines. It is a measure that is inversely related to the load balance of the system. If value of this metric was been lower, it shows that the distributed job among the virtual machines is more balanced. Degree of imbalance is computed based on the Eq.(17) [5]:

$$Degree\_Imbalance = \frac{J_{max} - J_{min}}{J_{avg}} \qquad (17)$$

Where, J_{max}, J_{min} and J_{avg} respectively show the maximum, minimum and average J_i between all virtual machines. Also for calculating of J_i, the Eq.(18) is used.

$$J_i = \frac{Length\_jobs}{Num\_PE \times PE\_MIPS} \qquad (18)$$

Where, *Length_jobs* is the total length of jobs which sent to the VM_i, *Num_PE* shows the number of PE and PE_MIPS is the capability of corresponding PE. In Figure 10 shows the average waiting time under different number of jobs. As can be seen, with increasing the number of jobs, the degree of imbalance increases. From the results shown in this figure, it can be seen that the proposed algorithm performed better than other algorithms. The degree of imbalance of TPALA and BatCA was somewhat closer to each other and is clearly better than the HPSO and FUGE, so that the degree of imbalance of TPALA was on average 4.13% better than BatCA algorithm but was on average 22.29% better than HPSO.

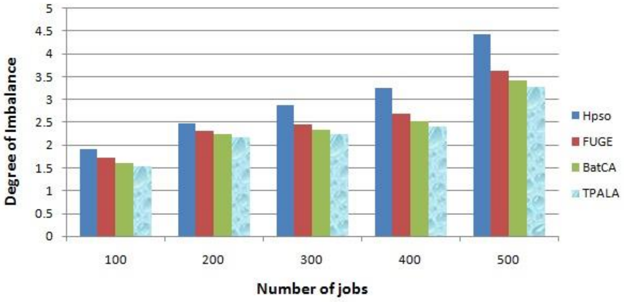


Fig. 10: degree of imbalance under number of jobs

6- Conclusions

In this paper, a new online algorithm based on LA for job scheduling in cloud environment, called TPALA was presented. Our proposed algorithm uses two different LAs for each scheduler to schedule jobs, as there are generally two main challenges in job scheduling. The first challenge is selecting the appropriate job from the submitted jobs based on their priority and specific conditions, while the second challenge is assigning the selected job to the most suitable virtual machine. To address these challenges, our algorithm employs two LAs in separate phases. In first phase a fixed action-set learning automaton was used and in second phase a variable action-set learning automaton was used. To prove the performance of the proposed method, several simulation case based on different scenarios have been simulated by CloudSim toolkit, in which several metric in job scheduling such as: makespan, success rate, average waiting time and degree of imbalance and compared to the three algorithms FUGE, HPSO and BatCL. In contrast to most job scheduling algorithms that use a single job type, we considered a combination of jobs based on two factors in our simulations: data volume and computational volume. In future work, we plan to explore learning automata-based methods for multi-objective job scheduling in cloud computing.

References

[1] B. Varghese, and R. Buyya, “Next generation cloud computing: New trends and research directions,” Future Generation Computer Systems, vol. 79, pp. 849-861, 2018.

[2] N. Moganarangan, R. Babukarthik, S. Bhuvaneswari et al., “A novel algorithm for reducing energy-consumption in cloud computing environment: Web service computing approach,” Journal of King Saud University-Computer and Information Sciences, vol. 28, no. 1, pp. 55-67, 2016.

[3] A. Ghaffari, and A. Mahdavi, “Embedding Virtual Machines in Cloud Computing Based on Big Bang–Big Crunch Algorithm,” Journal of Information Systems and Telecommunication (JIST), vol. 28, no. 7, pp. 305-315, 2020.

[4] L. F. Bittencourt, A. Goldman, E. R. Madeira et al., “Scheduling in distributed systems: A cloud computing perspective,” Computer science review, vol. 30, pp. 31-54, 2018.

[5] N. Mansouri, and M. M. Javidi, “Cost-based job scheduling strategy in cloud computing environments,” Distributed and Parallel Databases, pp. 1-36, 2019.

[6] U. Bhoi, and P. N. Ramanuj, “Enhanced max-min task scheduling algorithm in cloud computing,” International Journal of Application or Innovation in Engineering and Management (IJAIEM), vol. 2, no. 4, pp. 259-264, 2013.

[7] Y. Mao, X. Chen, and X. Li, "Max–min task scheduling algorithm for load balance in cloud computing." pp. 457-465, 2014.

[8] S. A. Hamad, and F. A. Omara, “Genetic-based task scheduling algorithm in cloud computing environment,” International Journal of Advanced Computer Science and Applications, vol. 7, no. 4, pp. 550-556, 2016.

[9] A. Kaleeswaran, V. Ramasamy, and P. Vivekanandan, “Dynamic scheduling of data using genetic algorithm in cloud computing,” International Journal of Advances in Engineering & Technology, vol. 5, no. 2, pp. 327, 2013.

[10] H. Aziza, and S. Krichen, “Bi-objective decision support system for task-scheduling based on genetic algorithm in cloud computing,” Computing, vol. 100, no. 2, pp. 65-91, 2018.

[11] B. Keshanchi, A. Souri, and N. J. Navimipour, “An improved genetic algorithm for task scheduling in the cloud environments using the priority queues: formal verification,

simulation, and statistical testing," Journal of Systems and Software, vol. 124, pp. 1-21, 2017.

[12] H. Y. Shishido, J. C. Estrella, C. F. M. Toledo et al., "Genetic-based algorithms applied to a workflow scheduling algorithm with security and deadline constraints in clouds," Computers & Electrical Engineering, vol. 69, pp. 378-394, 2018.

[13] M. A. Tawfeek, A. El-Sisi, A. E. Keshk et al., "Cloud task scheduling based on ant colony optimization." pp. 64-69, 2013.

[14] C. Z. a. P. W. C. Liu, "A Task Scheduling Algorithm Based on Genetic Algorithm and Ant Colony Optimization in Cloud Computing," in 13th International Symposium on Distributed Computing and Applications to Business, Engineering and Science, Xi'an, China, 2014, pp. 68-72.

[15] X. Wei, "Task scheduling optimization strategy using improved ant colony optimization algorithm in cloud computing," Journal of Ambient Intelligence and Humanized Computing, 2020/10/21, 2020.

[16] F. Hemasian-Etefagh, and F. Safi-Esfahani, "Dynamic scheduling applying new population grouping of whales meta-heuristic in cloud computing," The Journal of Supercomputing, vol. 75, no. 10, pp. 6386-6450, 2019.

[17] N. Manikandan, N. Gobalakrishnan, and K. Pradeep, "Bee optimization based random double adaptive whale optimization model for task scheduling in cloud computing environment," Computer Communications, vol. 187, pp. 35-44, 2022/04/01/, 2022.

[18] Z. Liu, W. Qu, W. Liu et al., "Resource preprocessing and optimal task scheduling in cloud computing environments," Concurrency and Computation: Practice and Experience, vol. 27, no. 13, pp. 3461-3482, 2015.

[19] V. Priya, and C. N. K. Babu, "Moving average fuzzy resource scheduling for virtualized cloud data services," Computer Standards & Interfaces, vol. 50, pp. 251-257, 2017.

[20] S. Zhan, and H. Huo, "Improved PSO-based task scheduling algorithm in cloud computing," Journal of Information & Computational Science, 2012.

[21] A. Kamalinia, and A. Ghaffari, "Hybrid Task Scheduling Method for Cloud Computing by Genetic and PSO Algorithms," Journal of Information Systems and Telecommunication (JIST), vol. 16, no. 4, pp. 1-10, 2016.

[22] M. Masdari, F. Salehi, M. Jalali et al., "A survey of PSO-based scheduling algorithms in cloud computing," Journal of Network and Systems Management, vol. 25, no. 1, pp. 122-158, 2017.

[23] G. Babu, and K. Krishnasamy, "Task scheduling algorithm based on Hybrid Particle Swarm Optimization in cloud computing environment," Journal of Theoretical and Applied Information Technology, vol. 55, pp. 33-38, 2013.

[24] N. Mansouri, B. M. H. Zade, and M. M. Javidi, "Hybrid task scheduling strategy for cloud computing by modified particle swarm optimization and fuzzy theory," Computers & Industrial Engineering, vol. 130, pp. 597-633, 2019.

[25] X. Chen, and D. Long, "Task scheduling of cloud computing using integrated particle swarm algorithm and ant colony algorithm," Cluster Computing, vol. 22, no. 2, pp. 2761-2769, 2019/03/01, 2019.

[26] H. Naseri, S. Azizi, and A. Abdollahpouri, "BSFS: A Bidirectional Search Algorithm for Flow Scheduling in Cloud Data Centers," Journal of Information Systems and Telecommunication (JIST), vol. 27, no. 7, pp. 175-183, 2020.

[27] M. Shojafar, S. Javanmardi, S. Abolfazli et al., "FUGE: A joint meta-heuristic approach to cloud job scheduling algorithm using fuzzy theory and a genetic method," Cluster Computing, vol. 18, no. 2, pp. 829-844, 2015.

[28] Y. Shi, L. Luo, and H. Guang, "Research on Scheduling of Cloud Manufacturing Resources Based on Bat Algorithm and Cellular Automata." pp. 174-177, 2019.

[29] M. I. Khaleel, "Efficient job scheduling paradigm based on hybrid sparrow search algorithm and differential evolution optimization for heterogeneous cloud computing platforms," Internet of Things, vol. 22, pp. 100697, 2023/07/01/, 2023.

[30] H. G. S. Phani Praveen, Negar Shahabi, Fatemeh Izanloo, "A Hybrid Gravitational Emulation Local Search-Based Algorithm for Task Scheduling in Cloud Computing," Mathematical Problems in Engineering, 2023.

[31] S. Sahoo, B. Sahoo, and A. K. Turuk, "An Energy-Efficient Scheduling Framework for Cloud Using Learning Automata." pp. 1-5, 2018.

[32] A. Yazidi, I. Hassan, H. L. Hammer et al., "Achieving Fair Load Balancing by Invoking a Learning Automata-Based Two-Time-Scale Separation Paradigm," IEEE Transactions on Neural Networks and Learning Systems, vol. 32, no. 8, pp. 3444-3457, 2021.

[33] L. Zhu, K. Huang, Y. Hu et al., "A Self-Adapting Task Scheduling Algorithm for Container Cloud Using Learning Automata," IEEE Access, vol. 9, pp. 81236-81252, 2021.

[34] S. A. Murad, A. J. M. Muzahid, Z. R. M. Azmi et al., "A review on job scheduling technique in cloud computing and priority rule based intelligent framework," Journal of King Saud University - Computer and Information Sciences, vol. 34, no. 6, Part A, pp. 2309-2331, 2022/06/01/, 2022.

[35] M. Masdari, and M. Zangakani, "Efficient task and workflow scheduling in inter-cloud environments: challenges and opportunities," The Journal of Supercomputing, vol. 76, no. 1, pp. 499-535, 2020.

[36] A. Arunarani, D. Manjula, and V. Sugumaran, "Task scheduling techniques in cloud computing: A literature survey," Future Generation Computer Systems, vol. 91, pp. 407-415, 2019.

[37] M. A. Rodriguez, and R. Buyya, "A taxonomy and survey on scheduling algorithms for scientific workflows in IaaS cloud computing environments," Concurrency and Computation: Practice and Experience, vol. 29, no. 8, pp. e4041, 2017.

[38] F. Wu, Q. Wu, and Y. Tan, "Workflow scheduling in cloud: a survey," The Journal of Supercomputing, vol. 71, no. 9, pp. 3373-3418, 2015.

[39] J. Kazemi Kordestani, M. Razapoor Mirsaleh, A. Rezvanian et al., "An Introduction to Learning Automata and Optimization," Advances in Learning Automata and Intelligent Optimization, J. Kazemi Kordestani, M. R. Mirsaleh, A. Rezvanian et al., eds., pp. 1-50, Cham: Springer International Publishing, 2021.

[40] K. S. Narendra, and M. A. Thathachar, "Learning automata-a survey," IEEE Transactions on systems, man, and cybernetics, no. 4, pp. 323-334, 1974.

[41] K. S. Narendra, and M. A. Thathachar, Learning automata: an introduction: Courier corporation, 2012.

[42] A. Rezvanian, A. M. Saghiri, S. M. Vahidipour et al., Recent advances in learning automata: Springer, 2018.

[43] M. A. L. Thathachar, and B. R. Harita, “Learning automata with changing number of actions,” IEEE Transactions on Systems, Man, and Cybernetics, vol. 17, no. 6, pp. 1095-1100, 1987.

[44] R. N. Calheiros, R. Ranjan, C. A. De Rose et al., “Cloudsim: A novel framework for modeling and simulation of cloud computing infrastructures and services,” arXiv preprint arXiv:0903.2525, 2009.

[45] R. N. Calheiros, R. Ranjan, A. Beloglazov et al., “CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms,” Software: Practice and experience, vol. 41, no. 1, pp. 23-50, 2011.

[46] R. D. Lakshmi, and N. Srinivasu, “A dynamic approach to task scheduling in cloud computing using genetic algorithm,” Journal of Theoretical & Applied Information Technology, vol. 85, no. 2, 2016.

[47] J. Blazewicz, K. H. Ecker, E. Pesch et al., Scheduling Computer and Manufacturing Processes: Springer Science & Business Media, 2013.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 41-48

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Image Fake News Detection using Efficient NetB0 Model

Yasameen Khalid Zamil[1,2] ,Nasrollah Moghaddam Charkari[1*]

[1.]Department of Electrical and Computer Engineering, Tarbiat Modares University of Tehran, Tehran, Iran,
[2.]DirectorateYouth and Sport of Babylon, Iraq.

Received: 1 Feb 2023/ Revised: 10 May 2023/ Accepted: 07 Jul 2023

Abstract

Today, social networks have become a prominent source of news, significantly altering the way people obtain news from traditional media sources to social media. Alternatively, social media platforms have been plagued by unauthenticated and fake news in recent years. However, the rise of fake news on these platforms has become a challenging issue. Fake news dissemination, especially through visual content, poses a significant threat as people tend to share information in image format. Consequently, detecting and combating fake news has become crucial in the realm of social media. In this paper, we propose an approach to address the detection of fake image news. Our method incorporates the error level analysis (ELA) technique and the explicit convolutional neural network of the EfficientNet model. By converting the original image into an ELA image, it is possible to effectively highlight any manipulations or discrepancies within the image. The ELA image is further processed by the EfficientNet model, which captures distinctive features used to detect fake image news. Visual features extracted from the model are passed through a dense layer and a sigmoid function to predict the image type. To evaluate the efficacy of the proposed method, we conducted experiments using the CASIA 2.0 dataset, a widely adopted benchmark dataset for fake image detection. The experimental results demonstrate an accuracy rate of 96.11% for the CASIA dataset. The results outperform in terms of accuracy and computational efficiency, with a 6% increase in accuracy and a 5.2% improvement in the F-score compared with other similar methods.

Keywords: Fake News; EfficientNet; Fake Image; Social Media; Error Level Analysis.

1- Introduction

In the recent decade, It has greatly facilitated the rise of social media, the common way for people to get in touch with each other and share information. Social media has many positive characteristics, like ease of use, low cost, and 7x24 information accesses. Unfortunately, fake news has greatly increased on social platforms. The increasing rate of fake news has become one of the troubling issues since it can mislead people. Online fake news tends to be diverse and intrusive regarding topics, platforms, and styles. Fake news could have many negative impacts on individuals, business, and society. So, it is crucial to introduce and launch a system that could detect, explore and interpret fake news on social media. It is challenging to come up with a definition for “fake news” that could be accepted in general. According to Stanford University, fake news is “news articles that are intentionally and verifiably false and could mislead readers.” According to online Wikipedia, “a type of yellow journalism or propaganda that consists of deliberate misinformation or hoaxes spread through traditional print and broadcast media or online social media fake news[1],[2]. Social media is among the widely accepted platforms globally. Its characteristics are ease-of-use, rapid rate, and low cost, making it the most welcoming online platform for information sharing and social interaction [2],[3]. Today, more than two-thirds of American adults can access online news outlets. This increasing rate has made the Internet an ideal channel for fake news dissemination. Social media is the primary media for the propagation of fake news and, consequently, has been one of the prominent studied areas by industries and universities.

There are different social media platforms with distinctive features. The most popular ones are Facebook and Twitter, that have been found as the primary sources of fake news diffusion [4]. The significant difference between the two is that each post on Twitter, a microblogging site, is limited to 380 characters, while on Facebook, the limit is nearly 60,000 [5],[6].

The study in [7] indicates that 54% of users in industrial countries are worried about “what is real or fake” on social

✉ **Nasrollah Moghaddam Charkari**
Moghadam@modares.ac.ir

platforms. This concern has become more significant after the 2016 U.S. presidential election [8],[9] due to the influence of social media on political polarization and conflicts among the political parties during the campaign period [10]. Another instance is the surge in online fake news following the lookdown measures to curtail the spread of COVID-19 disease. A study recently reported a 25% increase in social media users following the global lockdown. According to a UNESCO report, "during this coronavirus pandemic. Hence, the WHO described all misinformation related to COVID-19 is often referred to as an "infodemic," which they defined as an overabundance of offline and online information[10],[3], [12]. To curtail this menace, many fact-checking online systems such as FactCheck.org have recently come up to verify political news; however, the practicability of these systems is restricted due to the numerous" types and formats of fake news that facilitate its dissemination on the social network [7], [13]

Fake images in the news play an outstanding part. Fake images are often used to provoke public anger and gather public opinion. When it shares in serious repercussions such as mass killings and religious conflicts, it has an even more devastating impact. Various software tools usually modify fake images. Since, they might severely affect people's thoughts. Adobe's state of the content survey revealed that engagement for posts with images is three times more destructive than posts with text only. As a result, the fake images inside fake news has been increased in social media in recent years. So, developing solutions to discover fake images and text content on social media platforms is a crucial task [3]. Moreover, online social data is time-sensible, meaning it appears in a real-time type and represents current events and issues. There is an urgent necessity for early detection approaches of fake news from the huge number of news articles published daily[14].

In this article, we propose an approach for fake image detection. The main advantages of the proposed method are as follow;

- Computational time: Using efficientNetB0 model helps us to learn image features with fewer parameters compared to other deep learning approaches. Consequently, it leads to find fake images in lower computational time, which is a crucial task in this area.
- Proper feature extraction: Additionally, converting the original image into an error-level analysis (ELA) image enables the model to capture the manipulated features that further lead to effectively detect fake images.
- Improved Efficiency: The results of the experiments on popular datasets indicate that the proposed approach outperforms the current state-of-the-art methods regarding precision, recall, and accuracy rates.

The rest of this paper is structured as follows: Section 2 presents some of the interesting related work on fake news detection. In section 3, we discuss the methodology. Section 4 present the dataset and experimental results. Finally, the conclusion and feature work are discussed in section 5.

2- Related Works

Social media has evolved into a crucial source of information and an integral aspect of our daily life. The majority of information on social media is in the form of photographs. Meanwhile, phony news events have been increasingly distributed on social media, leading to user confusion. The existing news verification techniques rely on features collected from the text content of tweets, whereas image features are frequently overlooked for verification of news. Fake news detection on photos has been the subject of few studies. The absence of training data is one of the drawbacks of using visual-based features. Building a human-labeled a fake news dataset is time-consuming and labor-intensive. As a result, creating a fake news dataset with images or videos to train is considerably a complex task. The following are the most recent studies on images in the field of fake news identification [15], [16], [17]

Dinesh Kumar Vishwakarma et al. [15]proposed an image-based fake news detection method. The method comprises four core components: "image text extraction, entity extractor, web processing, and processing unit." Initially, an algorithm was employed to extract the text region, and then the text was recovered from photos using optical character recognition (OCR). This way, results are fetched and further classified as reliable or unreliable connections. The high classification rate for this method is 85%. The dataset included the Google image/ Kaggle / Onion dataset. Zhiwei Jin et al. [18]proposed a method to detect fake images based on visual and statistical image features; the gain ratio method was used to remove redundant features. This procedure selects 11 elements from 42 features. Four classification models, SVM, LR, KStar, and RF, have been employed to train the method. The dataset comprises 50,287 tweets and 25,953 images of fake and actual news events on SinaWeibo. The highest accuracy rate was 83.6% using the Random Forest classifier.

Francesco et al. [19] proposed a fake image detection method that relies on GAN-based image to image translation; this method relies on the modern approaches taken from the image forensic. CNN has been used to train data. An accuracy rate of 89.03% was reported using 36302 image dataset.

D. Mangal [16] presented a Multi-Domain Visual Neural Network (MVNN) model for the detection of fake news; the model is comprised of "a frequency domain sub-network, a pixel domain sub-network, and a fusion sub-network.” The fusion sub-network fuses the obtained feature vectors from the pixel and frequency domain sub-network through a fully connected layer; SoftMax activation is utilized to project the vector into either fake-news images or actual news. The Weibo dataset has been used in the experiments, and the accuracy rate reached to 84.6%.

Singh et al. [20] proposed an image-based fake news detection method. CNN with an attention mechanism is employed to detect fake images over the social network. The model uses high-pass filters to the kernel weights of the NN initialization. The two-dataset from Twitter and

The approach described in [18], used a statistical and machine learning methods, suffers from efficiently fake news detection due to the challenge of identifying manipulated features in handcrafted image features. This limitation results in poor model generalization. As described in [22], the forensic methods have been employed to extract image features. However, these features are specific to particular manipulated features, whereas image fake news contents may contain multiple manipulated features. So, these features could not be ideal for effectively detecting image fake news. Previous studies [21] on image fake news detection have utilized other models, which effectively identify fake images based on general features but necessitate a significant number of training data.

CASIA 2.0 have been employed. The observed Accuracy rates were 83.2% and 94.7%, respectively.

Xue et al.[21]built a model called “Multi-Vision Fusion Neural Network” to detect pictures in fake news. To extract the image features from an image in pixel domain, the visual modal module is utilized. Meanwhile, the ELA is employed for feature extraction at the frequency domain. The input image features are extracted in the semantic detection phase using the pre-trained ResNet50. The physical features module extracts the physical part to recognize fake news images. All elements in the visual model are connected and passed to PCA to reduce the number of features. Then, the physical features module is combined with the visual feature one in an ensemble module for fake image detection.

To identify the final fake news image, XGBoost has been used. The datasets used in this approach are (D1) and (D2), while the accuracy rates reach to 93.41% and 88.53%, respectively.

However, to address these models' limitations, a new approach to image fake news detection has been proposed that uses the EfficientNetB0 model in this paper. The use of EfficientNetB0 results in higher accuracy with fewer parameters and shorter execution times, making it more efficient and faster option than other models.

3- Proposed Method

We propose a method to deal with the problem of image fake news identification in this paper.

The overall framework of the proposed method is displayed in fig1 and algorithm1.

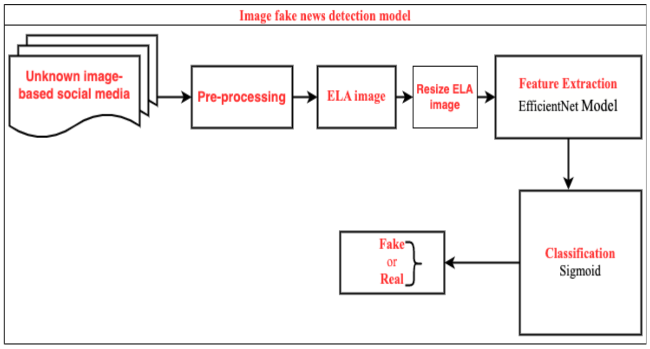


Fig. 1. The overall framework of the proposed method.

```
1. Input: (1) $\boldsymbol{I_R}$ : Regular Image
2. $\boldsymbol{L_I}$:   Image Label (Real/Fake)

3. Output:  Prediction Image as Fake or Real

4. Begin
5. $\boldsymbol{I_{ELA}} \leftarrow$ Convert Regular Image To Error Level Analysis Image ($\boldsymbol{I_R}$)
6. $\boldsymbol{I_S} \leftarrow$ Perform Error Level Analysis Image Resizer Method ($\boldsymbol{I_{ELA}}$)
7. $\boldsymbol{F}_{\boldsymbol{Ef}\text{ficient }\boldsymbol{N}\text{et}\boldsymbol{B0}} \leftarrow$ Extract Image Features Using EfficientNetB0 Framework ($\boldsymbol{I_S}$)
8. $\boldsymbol{F_I} \leftarrow$ Process Image Features Using Dense Layers Framework $\left(\boldsymbol{F}_{\boldsymbol{Ef}\text{ficient }\boldsymbol{N}\text{et}\boldsymbol{B0}}\right)$
9. $\boldsymbol{L_I} \leftarrow$ Regular Image Label PredictionMethod$(F_I)$
10. Optimize Image Fake Detection Results ($\boldsymbol{L_I}$)
11. end
```

Algorithm1.The steps of the proposed Image Fake News Detection

3-1- Pre-Processing Stage

The main preprocessing operations are:

- The tiff image is removed. This is lossless image.
- Convert all the images into the RGB color space.
- Resizing the image: in this step, all the ELA images convert to (128*128 pixels).

3-2- Error Level Analysis Method

After the preprocessing stage, all the images are converted into ELA. ELA is a forensics technique. Created by [23]to draw attention to the areas where a picture has been compressed. It takes the feature of the lossy compression method of manipulated images to identify whether the image is tampered or not. Briefly, ELA is done as follows;

$$ELA_{im} \ = \left|org_{img} - \ rq_{img}\right| \tag{1}$$

Where $\mathrm{org}_{\mathrm{img}}$ the original image and $\mathrm{rq}_{\mathrm{img}}$ reduce quality image.

The difference between the two images is known as the error levels related to the original pixels. The error level indicates a number of changes that are directly connected with the compression loss. If the variation is minimal, the pixel has attained its local minimum for error at the specified error rate. However, if there is substantial alteration, the pixels are not in their local minimum and may be extraneous[24],[25],[26]. In our proposed method, the ELA algorithm is employed; accordingly, the re-saved version is compressed at a quality of 95%. Furthermore, the absolute variance between the quality and original images is found. The dissimilarities among images indicate the error levels associated with the initial pixels. To improve the performance of the model, we fine-tune the brightness of the images; a scale parameter has been used to fine-tune the results; the value of this parameter has been calculated as follow;

$$sca = 0.255|mix\ pixel \tag{2}$$

$$ela_{enh} = ELA_{im}\ .sca \tag{3}$$

where mix pixel is the maximum image pixel in ELA_{im}. Figure 2 shows the original image after converting into ELA image. Image a:

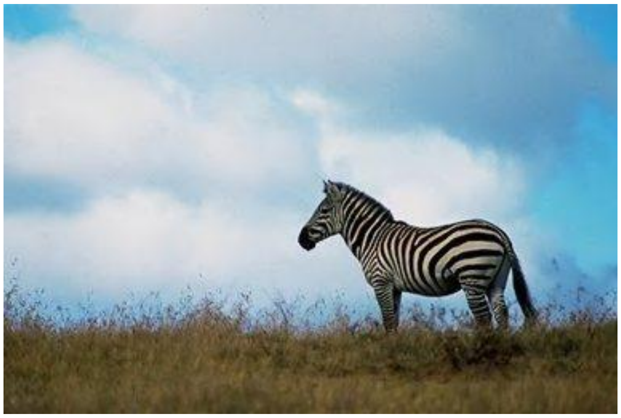


Image b:

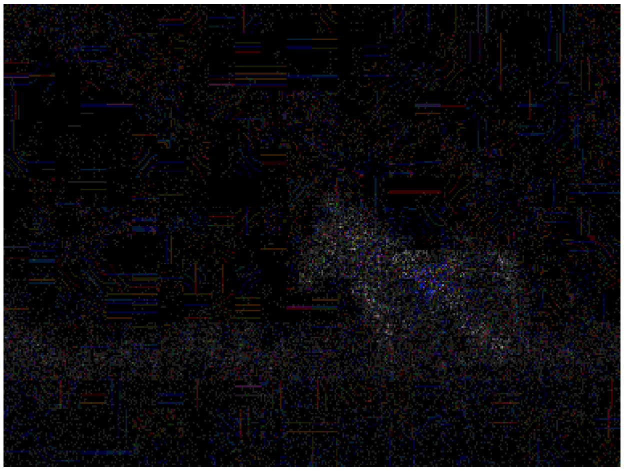


Fig. 2. a) The original image from Casia dataset and b) the image upon converting into ELA image.

3-3- Image Feature Learning

After the ELA images are produced, feature extraction is undertaken to establish feature vectors. In this paper, We have employed the EfficientNetB0 model belongs to the family EfficientNet from B0 to B7 in this study [27]; Accordingly, each variant of EfficientNet introduces different parameters as well as computational complexity time. Comparing them, EfficientNetB0 finds the lowest number of parameters while EfficientNetB7 needs the highest number of parameters for training.

Since one of the urgent necessities in fake news detection on social media is early and fast detection, we opted for EfficientNetB0, which uses fewest parameters. This allows for faster execution time and reduces computational resources without sacrificing the quality of the results. EfficientNetB0 has been designed using a novel scaling method that optimizes the model's depth, width, and resolution for the given computational resources. This approach has resulted in a model that can achieve high accuracy rate with fewer parameters, making it more efficient and faster than other deep learning models like ResNet, Inception, and VGG.

For feature extraction and added a global_average_pooling2d layer to reduce the overall number of parameters and hence limit overfitting. The EfficientNetB0 is a trained model on the ImageNet Dataset. To increase the model's efficiency, we re-trained the EfficientNetB0 model on our dataset, and the first layers from the EfficientNetB0 model pass to the global_average_pooling2d layer to reduce overfitting. The feature vector length (FS_{im}) is 1280 for the EfficientNetB0 model. Let

$$\mathrm{FS}_{im} = \emptyset_{f}(W_{\mathrm{i}}\boldsymbol{F}_{\boldsymbol{EN}\mathrm{et}\boldsymbol{B0}}) \tag{4}$$

Where $\emptyset$ represents the activation function, Wi is the weight for each layer in EfficientNetB0, $\boldsymbol{F_{EN}}\mathrm{et}\boldsymbol{_{B0}}$ is the output from the layer.

3-4- News Classification

The final stage in our proposal predicts the image into two classes as fake or real. For image feature extraction, we use the EfficientNetB0 model. Accordingly, the sigmoid function is used to ascertain whether an image is authentic or fake. The Relu (Rectifier Linear Units) is used in the dense layer as the activation function. We perform the predictor for image fake news by the sigmoid function as:

$$\tilde{p}_i = \mathrm{FE}(\mathrm{FS}_{im\ ,}\emptyset_f). \qquad (5)$$

Where $\emptyset_f$ indicates the parameters set of the sigmoid function, and FE is the mapping function.

Adam's optimizer has been utilized to optimize learning. Binary Cross Entropy is applied to calculate the loss function. $\tilde{p}_i = [\tilde{p}_1, \tilde{p}_2]$, hence $\tilde{p}_1$ denotes the probability of the given image as actual (0). $\tilde{p}_2$ indicates the likelihood of the image being fake (1).

4- Experimental Results & Analysis

We show the experiments conducted to assess the effectiveness of the proposed model. This section details the dataset, outcomes, and comparison with other related methods.

Fine-Tuning Given that social media has become a fundamental aspect of human daily life, detecting fake news on these platforms has become a crucial issue. Those methods used to spread fake news have evolved from text to images and even videos. In this study, we proposed a method to detect fake images using the EfficientNetB0 model, a member of the CNN family that is trained on the ImageNet dataset. In general, images play a critical role in news verification. In this regard, we have investigated on images to enhance fake news detection performance. The ELA method is employed with the EfficientNetB0 model. Furthermore, a global_average_pooling2d layer is added to reduce the number of parameters and to prevent overfitting. The EfficientNetB0 model has also been trained on our dataset, and weights were set for the EfficientNetB0 during the training process. We validate the effectiveness of feature learning on one of popular dataset, the CASAI. The proposed method achieves a validation accuracy rate of 96.11%. The model is designed to calculate the probability of the posts in the form of the entered image being real or fake. The results outperformed state-of-the-art methods on CASAI dataset, with a rate of approximately 6% in accuracy and 5.2% in the case of F-score rates.

In future, we intend to expand our method to social media datasets by extracting text from images and studying its impact on fake news detection.

4-1- Parameters

Transfer learning includes fine-tuning. We adjust our model that has previously undergone training on the ImageNet dataset. As mentioned, the images are initially converted into ELA images and further resized to 128*128 pixels. The EfficientNetB0 models have been used with pre-trained ImageNet weights (just the part CNN feature extraction, without prediction layers) by the EfficientNetB0 model and re-train parts of the network on our dataset. ImageNet dataset was frozen so that the weights of the ImageNet would not be affected by re-training on our dataset. After training the network and adjusting the parameters, we unfreeze the entire network. The last four layers (the top layer) related to the classification process in CNN from the network are removed and replaced with the proposed classification and activation function layers.

After re-training the network and extracting the features, we added a GlobalAveragePooling2D Layer with a dropout layer to eliminate the repetition in the features resulting from the re-training process and overfitting.

To make it fit with our classes, we have added two dense layers with a sigmoid function to predict whether the class type is fake or real.

The learning rate $1^{e\ -\ 6}$ is set to warm up the FC. When applying fine-tuning, we allow the warm-up stage to train for 10 epochs based on our dataset. We will proceed to measure our network performance on the testing set after the warm-up phase. Table 1 describes the parameters used to fine-tuning model.

Table 1. Hyper parameter settings for EfficientNetB0

Hyperparameter	**Values**
Optimizer	Adam
Learning rate	10^{-6}
No. of dense layers	2
Dropout	0.5
Batch Size	32
Epochs	10
Total parameters	4,049,571
Trainable Parameters	4,007,548
Non- Trainable Parameters	42,023

4-2- Experimental Setup

The model was produced using a machine on Colab, employing the Keras library and the Google TensorFlow frame. To choose optimal hyperparameters, we have studied different batch sizes and dropout probabilities. The best results were achieved by utilizing the Adam optimizer with a learning rate of 10^{-4}; a batch size of 32, and training for 20 epochs. The hyperparameter values in this study are shown in table 2. Each experiment has been carried out randomly. Accordingly, the CASIA dataset is split into 80% as training and 20% as validation. The final findings were obtained when the ultimate level of accuracy was attained.

Table 2 . Hyperparameter values in the proposed model

Hyperparameter	Values
Optimizer	Adam
Learning rate	10^{-4}
Dropout	0.5
Batch Size	32
Epochs	20
Total params	168,129
Trainable Params	168,129
Non- Trainable Parameters	0

4-3- CASIA 2.0 Dataset

There are 12,616 images in the CASIA 2.0 dataset, where 5124 of them are manipulated, and the remaining 7492 images are legitimate. Copy-move and image-splicing techniques are used to manipulate the images. While performing tampering to the image, cropping and resizing are also done [28]. The number of CASIA images is shown in table 3.

Table 3 The statistics of CASIA V.2 dataset

Image type	Image size
Authentic – image	7200
Tamper – image	5123

4-4- Experimental Results

As mentioned in the previous section, the EfficientNetB0 model has been employed to learn the essential features, represented by the fewest number of parameters with high efficiency compared to the EfficientNetB1 model to EfficientNetB7. We have conducted our experiments on the CASIA dataset that contains images. The highest accuracy rate has been found as 96.11% in value. Four assessment measures have been used to evaluate the experimental results. Those are F1-score, Accuracy, Recall, and precision. Accuracy indicates how well the model classifies the images as real or fake.

The F-score measures the consistent mean of Precision and Recall; the performance of the proposed model, as shown in **Fig 3**, is found by four measures, Accuracy, Precision, Recall, and F-Score denoted as follows;

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \qquad (6)$$

$$Precision = \frac{TP}{TP + FP} \qquad (7)$$

$$Recal = \frac{TP}{TP+FN} \qquad (8)$$

$$F - score = \frac{2 * (Precision\ * Recall)}{(Precision\ + Recall)} \qquad (9)$$

Where,
$\text{True Positive}(\text{TP}) = \text{Correctly classified.}$
$\text{False Positive}(\text{FP}) = \text{Incorrectly classified.}$
$\text{True Negative}(\text{TN}) = \text{Correctly Rejected.}$
$\text{False Negative}(\text{FN}) = \text{Incorrectly Rejected}$

AUC represented a level of separability. It indicates the model's capability to distinguish between classes. Although AUC has not been taken into comparison with other methods, it is significant for checking in classification problems; Fig 4 describes the confusion matrix of the proposed method on the CASIA data set. Fig 5 represents an ROC graph used to evaluate an imbalanced dataset, which is essential in binary classification. Table 4 displays a comparison between our proposed approach and other baseline methods. Refer to Table 4, [20]used the high pass filter with CNN to detect fake images. [29]Used the VGG19 model to detect fake images. MVNN[30] employed physical and semantic visual features to find fake news. In [31] utilized a CNN to extract features that help in the identification of fake news. As shown in Table 4, the proposed model can efficiently capture the modified characteristics in the fake image.

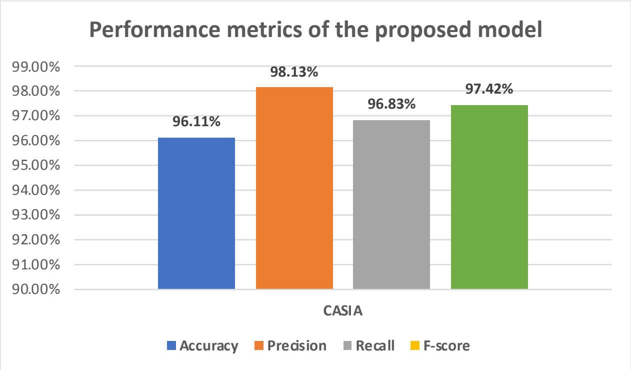


Fig. 3. The results of the proposed model's performance

Table 3 . Comparison of different models with our model

Method	Year	Accuracy	F-score
CNN [20]	2021	94.7%	95%
VGG19 [29]	2019	74.07%	79.11%
MVNN[30]	2019	89.12	94.53
CNN [31]	2017	74%	74.4%
Our proposed EfficientNetB0 model	2023	96.11%	97.42 %

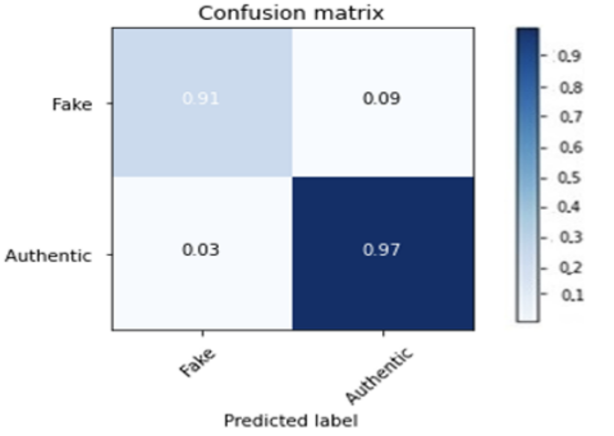


Fig. 4. Confusion matrix of the proposed method on CASIA 2.0 dataset

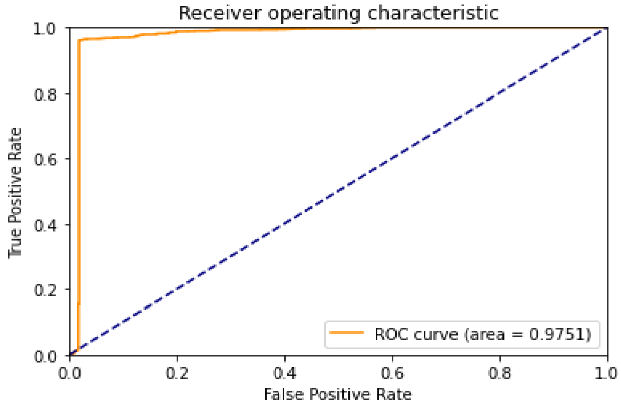


Fig. 5. AUROC curve on the CASIA 2.0 dataset

5- Conclusion

Given that social media has become a fundamental aspect of human daily life, detecting fake news on these platforms has become a crucial issue. Those methods used to spread fake news have evolved from text to images and even videos. In this study, we proposed a method to detect fake images using the EfficientNetB0 model, a member of the CNN family that is trained on the ImageNet dataset. In general, images play a critical role in news verification. In this regard, we have investigated on images to enhance fake news detection performance. The ELA method is employed with the EfficientNetB0 model. Furthermore, a global_average_pooling2d layer is added to reduce the number of parameters and to prevent overfitting. The EfficientNetB0 model has also been trained on our dataset, and weights were set for the EfficientNetB0 during the training process. We have validated the effectiveness of feature learning on one of popular dataset, the CASAI. The proposed method achieves a validation accuracy rate of 96.11%. The model is designed to calculate the probability of the posts in the form of the entered image being real or fake. The results outperformed state-of-the-art methods on CASAI dataset, with a rate of 6% in accuracy and 5.2% in the case of F-score rates.

In future, we intend to extend our method to social media datasets by extracting text from images and studying its impact on fake news detection. Furthermore, introducing an explanatory model is another further direction of our research.

References

[1] M. Celliers and M. Hattingh, “A Systematic Review on Fake News Themes Reported in Literature,” in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Springer, 2020, pp. 223–234. doi: 10.1007/978-3-030-45002-1_19.

[2] X. Zhang and A. A. Ghorbani, “An overview of online fake news: Characterization, detection, and discussion,” *Inf Process Manag*, vol. 57, no. 2, p. 102025, 2020, doi: 10.1016/j.ipm.2019.03.004.

[3] W. S. Paka, R. Bansal, A. Kaushik, S. Sengupta, and T. Chakraborty, “Cross-SEAN: A cross-stitch semi-supervised neural attention model for COVID-19 fake news detection,” *Appl Soft Comput*, vol. 107, p. 107393, 2021, doi: 10.1016/j.asoc.2021.107393.

[4] Y. Wang et al., “Weak supervision for fake news detection via reinforcement learning,” AAAI 2020 - 34th AAAI Conference on Artificial Intelligence, no. December 2019, pp. 516–523, 2020, doi: 10.1609/aaai.v34i01.5389.

[5] N. Guimarães, Á. Figueira, and L. Torgo, “An organized review of key factors for fake news detection,” pp. 1–10, 2021, [Online]. Available: http://arxiv.org/abs/2102.13433

[6] S. Preston, A. Anderson, D. J. Robertson, M. P. Shephard, and N. Huhe, “Detecting fake news on Facebook: The role of emotional intelligence,” PLoS One, vol. 16, no. 3 March, pp. 1–13, 2021, doi: 10.1371/journal.pone.0246757.

[7] P. Meel and D. K. Vishwakarma, “Fake news, rumor, information pollution in social media and web: A contemporary survey of state-of-the-arts, challenges and opportunities,” Expert Syst Appl, vol. 153, p. 112986, 2020, doi: 10.1016/j.eswa.2019.112986.

[8] H. Allcott and M. Gentzkow, “Social media and fake news in the 2016 election,” Journal of Economic Perspectives, vol. 31, no. 2, pp. 211–236, 2017, doi: 10.1257/jep.31.2.211.

[9] S. Raza and C. Ding, “Fake news detection based on news content and social contexts: a transformer-based approach,” Int J Data Sci Anal, vol. 13, no. 4, pp. 335–362, May 2022, doi: 10.1007/s41060-021-00302-z.

[10] X. Zhang and A. A. Ghorbani, “An overview of online fake news: Characterization, detection, and discussion,” Inf

Process Manag, vol. 57, no. 2, p. 102025, 2020, doi: 10.1016/j.ipm.2019.03.004.

[11] N. Hoy and T. Koulouri, “A Systematic Review on the Detection of Fake News Articles,” Oct. 2021, [Online]. Available: http://arxiv.org/abs/2110.11240

[12] J. Jing, H. Wu, J. Sun, X. Fang, and H. Zhang, “Multimodal fake news detection via progressive fusion networks,” Inf Process Manag, vol. 60, no. 1, Jan. 2023, doi: 10.1016/j.ipm.2022.103120.

[13] A. Biswas, D. Bhattacharya, K. Anil Kumar, and A. Professor, “DeepFake Detection using 3D-Xception Net with Discrete Fourier Transformation,” Journal of Information Systems and Telecommunication (JIST) 3, no. 35. 2021. 161.

[14] S. Hangloo and B. Arora, “Combating multimodal fake news on social media: methods, datasets, and future perspective,” Multimed Syst, vol. 28, no. 6, pp. 2391–2422, Dec. 2022, doi: 10.1007/s00530-022-00966-y.

[15] D. K. Vishwakarma, D. Varshney, and A. Yadav, “Detection and veracity analysis of fake news via scrapping and authenticating the web search,” Cogn Syst Res, vol. 58, pp. 217–229, Dec. 2019, doi: 10.1016/j.cogsys.2019.07.004.

[16] D. Mangal and Di. K. Sharma, “Fake News Detection with Integration of Embedded Text Cues and Image Features,” ICRITO 2020 - IEEE 8th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions), pp. 68–72, 2020, doi: 10.1109/ICRITO48877.2020.9197817.

[17] A. Mahmoodzadeh, “Human Activity Recognition based on Deep Belief Network Classifier and Combination of Local and Global Features,” .” J. Inf. Syst. Telecommun 9, 2021.

[18] Z. Jin, J. Cao, Y. Zhang, J. Zhou, and Q. Tian, “Novel Visual and Statistical Image Features for Microblogs News Verification,” IEEE Trans Multimedia, vol. 19, no. 3, pp. 598–608, 2017, doi: 10.1109/TMM.2016.2617078.

[19] F. Marra, D. Gragnaniello, D. Cozzolino, and L. Verdoliva, “Detection of GAN-Generated Fake Images over Social Networks,” Proceedings - IEEE 1st Conference on Multimedia Information Processing and Retrieval, MIPR 2018, pp. 384–389, 2018, doi: 10.1109/MIPR.2018.00084.

[20] B. Singh and D. K. Sharma, “SiteForge: Detecting and localizing forged images on microblogging platforms using deep convolutional neural network,” Comput Ind Eng, vol. 162, no. October, 2021, doi: 10.1016/j.cie.2021.107733.

[21] J. Xue, Y. Wang, S. Xu, L. Shi, L. Wei, and H. Song, MVFNN: Multi-Vision Fusion Neural Network for Fake News Picture Detection, vol. 1300, no. 2018. Springer International Publishing, 2020. doi: 10.1007/978-3-030-63426-1_12.

[22] C. Boididou et al., “Verifying multimedia use at MediaEval 2015,” CEUR Workshop Proc, vol. 1436, no. September, 2015.

[23] I. B. K. Sudiatmika, F. Rahman, Trisno, and Suyoto, “Image forgery detection using error level analysis and deep learning,” Telkomnika (Telecommunication Computing Electronics and Control), vol. 17, no. 2, pp. 653–659, 2019, doi: 10.12928/TELKOMNIKA.V17I2.8976.

[24] N. Krawetz, “A Picture’s Worth... Digital Image Analysis and Forensics,” 2007. [Online]. Available: www.hackerfactor.com

[25] Paganini and Pierluigi, "Photo forensics: Detect Photoshop manipulation with error level analysis." Chief Information Security Officer at Bit4Id, 2013.

[26] H. Farid, “Exposing Digital Forgeries from JPEG Ghosts,” IEEE transactions on information forensics and security 154-160, 4.1 .2009.

[27] M. Tan and Q. V. Le, “EfficientNet: Rethinking model scaling for convolutional neural networks,” 36th International Conference on Machine Learning, ICML 2019, vol. 2019-June, pp. 10691–10700, 2019.

[28] W. and T. T. Jing Dong, “CASIA IMAGE TAMPERING DETECTION EVALUATION DATABASE Jing Dong , Wei Wang and Tieniu Tan Institute of Automation , Chinese Academy of Sciences,” pp. 422–426, 2013.

[29] D. Khattar, M. Gupta, J. S. Goud, and V. Varma, “MvaE: Multimodal variational autoencoder for fake news detection,” The Web Conference 2019 - Proceedings of the World Wide Web Conference, WWW 2019, pp. 2915–2921, 2019, doi: 10.1145/3308558.3313552.

[30] P. Qi, J. Cao, T. Yang, J. Guo, and J. Li, “Exploiting multi-domain visual information for fake news detection,” Proceedings - IEEE International Conference on Data Mining, ICDM, vol. 2019-Novem, no. Icdm, pp. 518–527, 2019, doi: 10.1109/ICDM.2019.00062.

[31] F. Yu, Q. Liu, S. Wu, L. Wang, and T. Tan, “A Convolutional Approach for Misinformation Identification,” 2017. [Online]. Available: http://www.npr.org/2016/11/08/500686320/did-social-media-

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 49-61
http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

An Analysis of the Signal-to-Interference Ratio in UAV-based Telecommunication Networks

Hamid Jafaripour [1*], Mohammad Fathi [1]

[1.]Department of Electrical Engineering, University of Kurdistan, Sanandaj, Iran.

Received: 05 Dec 2022/ Revised: 29 Apr 2023/ Accepted: 08 May 2023

Abstract

One of the most important issues in wireless telecommunication systems is to study coverage efficiency in urban environments. Coverage efficiency means improving the signal-to-interference ratio (SIR) by providing a maximum telecommunication coverage and establishing high-quality communication for users. In this paper, we use unmanned aerial vehicle (UAVs) as air base stations (BS) to investigate and improve the issue of maximizing coverage with minimal interference. First, we calculate the optimal height of the UAVs for the coverage radius of 400, 450, 500, 550, and 600 meters. Then, using simulation, we calculate and examine the value and status of SIR in UAVs with omnidirectional and directional antenna modes in symmetric and asymmetric altitude conditions, with and without considering the height of the UAVs. The best SIR is the UAV system with a directional antenna in asymmetric altitude conditions where the SIR range varies from 4.44db (the minimum coverage) to 52.11dB (maximum coverage). The worst SIR is the UAV system with an omnidirectional antenna in symmetrical height conditions without considering the height of the UAV. We estimate the range of SIR changes for different coverage ranges between 1.39 and 28dB. Factors affecting the SIR values from the most effective to the least, respectively, are coverage range and the antenna type, symmetrical and asymmetric height, and finally, considering or not considering the height of the UAV.

Keywords: Base Station; Telecommunication Coverage; UAV; Signal-to-Interference Ratio.

1- Introduction

Mobile terrestrial communication systems are used with the help BS for mobile phone users, which has many problems such as high installation, maintenance and commissioning costs, lack of coverage of all points, especially obstacles and buildings, and lack of line of sight (LoS) communications [1]. UAVs are new tools, and their range of performance has expanded to include the next generation of telecommunications UAVs. UAVs offer better service as an alternative to ground-based stations because of the LoS connection between users and UAV stations. Due to UAVs' limitations, it is not practical to use them continuously and sustainably, and this should provide researchers with an ideal solution for maximum coverage. UAVs provide a cost-effective wireless connection for devices without infrastructure coverage. Compared to terrestrial communications or high-altitude-based operating systems (satellite communications), wireless systems with low-altitude UAVs are generally faster and configurable with more flexibility. They also have better communication channels due to direct vision communication [2].

With the continuous development of UAV in the communications and other related technologies, the UAV industry has developed rapidly in recent years. The 5G network also predicts significant growth in data traffic scale, number of terminal connections, high reliability, low latency and high transmission speed. With the advent of the 5G, this technology is expected to significantly increase the capacity and new applications for large-scale connections. The 5G wireless system is still terrestrial and has the same coverage complexity as other terrestrial networks. However, in the event of the destruction of terrestrial infrastructure due to sudden disasters, these land cover networks will be in place. They are damaged or even out of reach. The space communication network complements the terrestrial communication network. This can not only provide extensive communication coverage for people and vehicles at sea, remote rural areas and the air, but also provide timely network connections in the event of a ground network failure [3]. UAVs equipped

✉ **Hamid Jafaripour**
Hamid.jafaripour@uok.ac.ir

with radio receivers can meet the requirements of an air communication platform as a mobile base station or as an aerial relay. Due to their flexible deployment, UAVs can be used in multilayer mobile networks with the help of UAVs to provide on-demand communication services in disaster areas and increase the performance, capacity and reliability of existing terrestrial mobile networks. However, several challenges such as optimal 3D placement, flight endurance time, energy constraints, and interference management may hinder the widespread use of UAV communications [2]. In UAV communications, an air base station is primarily a low-altitude platform that can cover ground as UAV-based small cells (USCs). The size of USCs varies according to altitude, position, transmission power, type of UAVs and environmental characteristics. Given this, the optimal placement of UAVs for USC cap performance analysis has attracted much research interest. For example, in [4-5], the UAV deployment problem is considered to increase the coverage of a single USC. Analysis probability for signal-to-noise ratio (SNR) thresholds were analyzed. In [8] they analyzed the problem of optimization for UAV placement to increase the number of users covered by different quality of service (QoS). However, this has been done for single-UAV networks. When multiple UAVs are available, [9-10] exploit the deployment of multiple UAVs to reduce the number of air BSs and expand coverage for ground users. In addition, most works optimize the horizontal coordinates of the UAVs for a fixed UAV height above ground level [11] or, while having a fixed horizontal position [12-13] optimize the UAV height. These studies analyzed the UAV location using an optimization framework in a non-interference environment. The ratio of the area covered and the area given is usually a measure of the UAV network coverage capability [14]. In addition, by adding influential factors in the mission area, it makes the ability to cover more specific and accurate.

About the environment and reduction of communications and flight overhead in order to avoid severe consequences, it has been pointed out that under the influence of UAV cooperation, the UAV cover mission can be performed well. However, Multi-UAV is rarely studied because it is difficult to describe a collaborative relationship. In the paper, the coverage area and signal to interference experienced by users at a UAV-enabled network is investigated against different configurations of antenna and UAV heights. Most common configurations in this network including antenna type in terms of omnidirectional and directional, antenna height, and UAV locations are examined numerically for their impacts on the network performance.

The paper is organized as follows. Research done in the area is given in Section 2 as literature review. The adopted network model of UAVs and the employed channel modes are presented in Section 3. Achieved results and detailed discussions are presented in Section 4, and the paper is concluded in Section 5.

2- Literature Review

Research has been done to investigate the problem of UAV coverage. In [15], by designing a paradigm that considers the energy consumption and communication power of the UAV, authors examined the relationship between the low-consumption UAV and a ground terminal and also defined the energy efficiency of the UAV communication. In [16], the probability function of the coverage for the ground user was extracted from a given UAV, which consists of increasing the antenna and altitude. In terms of UAV communications, various tasks have been proposed to describe the interference created by UAVs. A multi-dimensional multi-UAV deployment approach has been proposed to meet the QS requirements for different types of user distribution in the presence of common channel interference by maximizing the minimum achievable throughput for all ground users [17]. Investigates the interference characteristics of UAVs equipped with directional antennas in three-dimensional space in done in [18]. A cooperative beam forming technique is proposed for the BS to reduce the strong interference caused by common ground channel transmissions to the UAV in [19]. The problem of reducing interference and resource allocation in a wireless communication system, with two UAVs and two ground nodes has been investigated in [20]. Investigated the maximization of coverage in the presence of co-channel interference (CCI) generated by several UAVs in a specific target area in done in [21]. Authors in [22] investigate the effects of interference in urban environments for four-engine UAVs based on inter-carrier interference (ICI) and inter-symbol interference (ISI), which arise from multi-route scenarios. Minimizes signal-to-interference-plus-noise ratios (SINR) among all UAVs by jointly optimizing channel and power allocation strategy under severe resource availability is done in [23]. Authors studies the interaction of two UAV-enabled users for wireless powered communication networks in [24].

Authors in [25] propose a three-dimensional coordination model for interference management through the formation of the multicellular beam in multi-antenna UAV networks. The work in [26] offers an adaptive interference cancellation (IC) approach in which each BS can decrypt terrestrial user messages by adaptive switching between IC modes. Authors in [27] explains the main concept of high-power microwave (HPM) pulse interference and examines the possibility of electromagnetic interference against UAVs. Increased variability in cut-off probability and SNR in a multi-UAV network in the presence of interlink UAVs and cellular BSs has been investigated in [28].

Authors in [29] examines the key challenges of UAV-based radio scanner measurements to evaluate 5G aerial emissions to manage interference in non-public networks. Authors in [6] proposes a scheme for non-orthogonal multiple access (NOMA) in UAV communication systems in the presence of granted and un-granted users. Fast machine learning is presented for 5G beam selection for unmanned aerial vehicle applications in [7].

3- Mathematical Model

Small cells. Therefore, the main challenge to be addressed is maximizing UAV coverage in the presence of interference. It is evident from the Channel model that if there is no coordination between the UAVs, we will face interference problems and interference. If the distance between adjacent UAVs is too large, parts of the urban environment will not be covered. Also, if this distance reduces, it will lead to overlap of different areas. This causes severe interference of the channels (especially in the case that all sides use the same frequency). In UAV communications, co-frequency interference occurs when multiple UAVs share the same frequency sources simultaneously in separate space locations.

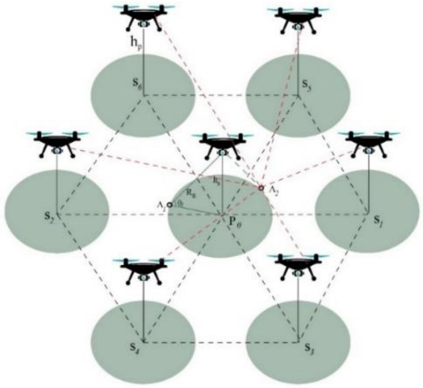


Fig. 1. Primary and secondary UAVs placement and user interference at the primary UAV's cell border

In this paper, we have used a multi-UAV synchronized network to reduce co-frequency channel interference in UAV communications. We assume a general multi-UAV model for M air BSs. The initial UAV is stationary and located above the center of the specified target surface. This UAV acts as a reference node to adjust the separation distance while the secondary UAVs are in proportion to the location of the primary UAV. After optimization, we place the secondary UAVs in a fixed position to fix their coverage area on the ground. Primary and secondary UAVs are located at $\boldsymbol{h_p}$ and $\boldsymbol{h_s}$ altitudes, respectively. According to Figure 1, we consider seven UAVs in the target area (square) with a side length of L= 2000 meters in a two-dimensional Cartesian system. We can use this model for any number of UAVs, but in practice a large number of UAVs complicates the calculations.

In this case, P_0 is the image center of the primary UAV and S_1 ,…, S_{M-1} are the coordinates of the secondary UAV. In addition, we can deploy coordinated multi-UAV networks on the basis of a regular convex polygon design to meet the required coverage at the target level with the required number of UAVs. Among all the possible possibilities of interference, we examine the worst type of interference, i.e., interference at the cell boundary of the primary UAV by the secondary UAVs shown in Figure 1. We focus on the use of quasi-stationary UAVs. The position of the UAVs remains unchanged for a specified period of time. For such settings, it is important to determine the coordinates of the UAVs to avoid collisions between them. To control interference, we must create spatial separation between the UAVs. Therefore, in the deployment strategy, we assume that the initial UAV is in the position P0 = {0,0}. When M = 7, we plot the coordinates of the secondary UAVs in a hexagonal pattern as in Figure 1[21].

$$S_x = \begin{cases} S_1, & ((D_{min}+D),O) \\ S_2, & (-(D_{min}+D),O) \\ S_3, & (\frac{1}{2}(D_{min}+D),-\frac{\sqrt{3}}{2}(D_{min}+D)) \\ S_4, & (-\frac{1}{2}(D_{min}+D),-\frac{\sqrt{3}}{2}(D_{min}+D)) \\ S_5, & (\frac{1}{2}(D_{min}+D),\frac{\sqrt{3}}{2}(D_{min}+D)) \\ S_6, & (-\frac{1}{2}(D_{min}+D),\frac{\sqrt{3}}{2}(D_{min}+D)) \end{cases} \tag{1}$$

here D_{min} is the minimum separation distance to avoid collisions between the UAVs and to ensure minimum coverage performance for all participating UAVs in the presence of interference. In this case, D is the only variable that controls the performance of the coating surface between a target surfaces.

3-1- Channel Model

In this model, the main communication components as shown in Figure 2 include ground BS, BS UAV, and ground users. We have divided the channels according to the type of connection of the main units. Generally, there are four types of channels: A2G channels, ground-to-air (G2A) channels, air-to-air (A2A) channels, and ground-to-ground channels. In this article, we examine only the A2G channel (UAV to the user).

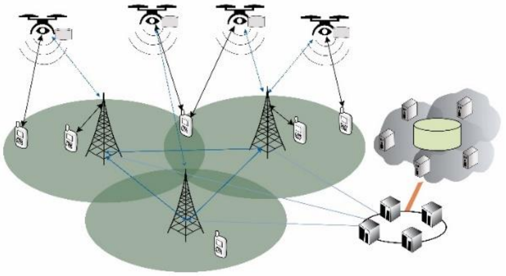


Fig. 2. The main components of the communication network

In general, the UAV propagation channel to the user is usually modeled separately with the possibility of LoS and NLoS occurring. NLoS links have more path loss than LoS links due to the effect of shadow and signal reflection from obstacles. The probability that UAV(i) communicates LoS with user(j) is calculated as follows [10],[23]:

$$P_{(LoS,UAV,User)}=\frac{1}{1+\alpha(\exp(-\beta(\theta_{User}-\alpha))} \qquad (2)$$

where α and β are fixed values that depend on the environment (dense and non-dense regions). θ_{User} is the angle between the UAV(i) and the user(j). $\theta_{User} = \frac{180}{\pi\, tanh\,(h/r)}$, h is the altitude of the UAV, and r is the ground distance between the UAV(i) and the user(j). We calculate this distance as below:

$$r=\sqrt{(X_{User}-X_{UAV})^2+(Y_{User}-Y_{UAV})^2} \qquad (3)$$

here X_{UAV} and Y_{UAV} are the position and coordinates of the UAV; X_{User} and Y_{User} are the position and coordinates of the ground users. In the urban environment, UAVs are located without intermediaries and with the shortest distance from the users in the coverage area. Due to the dynamics of the UAVs, the barriers do not prevent LoS from communicating with ground users. As a result, the chances of establishing an NLoS are very little. Given the above, we will skip the NLoS computing in the urban environment. Therefore, the total path loss is equal to the path loss in LoS mode and is calculated based on the following equation:

$$L_{Total}=L_{LoS}=\eta_{Los}\;(4\pi\frac{f}{c})^2\;R^2 \qquad (4)$$

where f is he carrier frequency, c is the speed of light, η_{Los} (in db) is the loss related to the LoS connection of the environment, and R is the direct distance between the UAV(i) and the user(j)which is calculated according to the following equation:

$$R=\sqrt{\mathrm{r}^2+h^2} \qquad (5)$$

3-2- UAV Height Calculations

The deployment of primary and secondary UAVs is divided into two categories, symmetric and asymmetric, according to Figure 3. In symmetrical mode, the altitude and transmission power of all UAVs (primary and secondary UAVs) are the same.

However, for asymmetric cases, the primary UAV is placed at the desired height and the secondary UAVs can be placed at a height higher or lower than the height of the primary UAV. In this paper, we will calculate the effects of symmetric and asymmetric heights on the degree of interference in the urban environment in different conditions. By introducing different coverage areas, in addition to the minimum interference, we have also examined the maximum coverage. We have introduced D as the variable parameter to determine the coverage range of the UAVs, D_{min} as the minimum separation distance to avoid collisions between the UAVs. According to Figure 1, the different values of D and D_{min} can be calculated based on the following equation:

$$3D+2D_{min}=2000\text{m} \qquad (6)$$

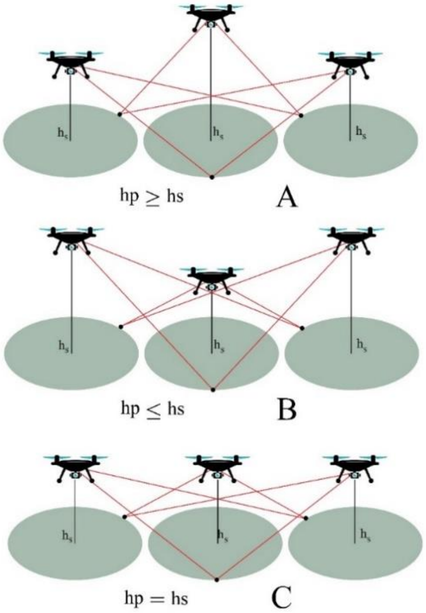


Fig. 3. a and b: The UAVs' heights in asymmetric conditions.
c: The UAVs' heights in symmetrical conditions

If we set the value D = D_{min}, we will have the minimum coverage and maximum security between the UAVs, in which case the value of D is equal to 400m. We calculate the optimal values of D and D_{min} and put them in Table 1. The appropriate height of the UAV to cover area D (based on the UAV antenna angle), which can cover the area from an angle of 0 to 60 degrees, is calculated from the following equation:

$$h = r * \tan 30 = 200 * \left(\frac{\sqrt{3}}{2}\right) = 115\ m \qquad (7)$$

At D= 400m we have the minimum coverage and D= 600m the maximum coverage. Our goal is to maximize coverage area with minimum interference and the number of UAVs.

Table 1. D and D_{min} values and optimal height and distance of the cell border user from the UAV

D	400	450	500	550	600
$\boldsymbol{D_{min}}$	400	325	250	175	100
Optimal Height	115	130	145	159	173
User Distance	230	450	288	318	346

4- Results and Discussion

In this paper, the effects of antenna type (all-purpose antenna and directional antenna) and the conditions of the height of the primary and secondary UAVs relative to each other (symmetrical and asymmetric) and with and without considering the height of the UAVs on the SIR are investigated. Regardless of the height of the UAV, the user distance of the primary UAV cell border to the center point of the secondary UAV cell is calculated as the ground distance and the effect of the height of the secondary UAV is not counted, but in the case of altitude, the height of the UAVs is also calculated.

4-1- Analyzing the SIR of Directional Antennas with Symmetrical Height, without Considering the UAVs' Heights

We consider the user at the cell boundary of the primary UAV as shown in Figure 1, which receives telecommunication services by UAV S1. We have not first applied the effects of UAV height on user interaction in the equation. Assuming that the user of the primary UAV cell border is receiving fixed frequency telecommunication services from the primary UAV and UAVs S2 to S7 are sending data to users inside their cell with the same operating frequency.

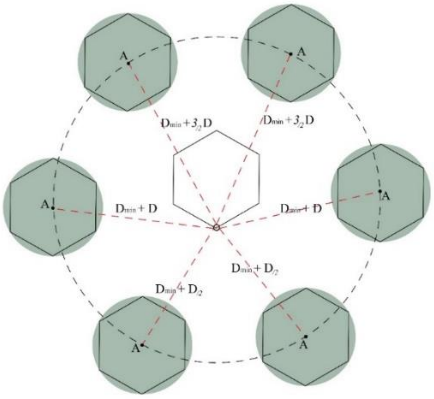


Fig. 4. The effect of secondary cell interference on the user at the primary cell border

In this situation, S2 to S7 UAVs affect the user performance at the S1 UAV cell boundary and interfere. The ratio of the received signal power (S) to the interference power received from the environment is called SIR. SIR affects QoS and determines the bit error rate. The SNR is calculated from the following equation:

$$\text{SIR} = \frac{S}{I} = \frac{KP_{tp}d^{-\gamma}}{\sum_{n=1}^{N} KP_{tn}{d_n}^{-\gamma}} \qquad (8)$$

where K is a constant value and γ is a value between 2 and 4. P_{tp} is the transmitting power of the primary UAV, d is the distance between the UAV and the user, P_{tn} is the transmitting power of each of the secondary UAVs and d_n is the distance of each of the secondary UAVs relative to the user of the primary UAV cell boundary.

The simplified interference of the secondary UAVs relative to the user at the cell boundary of the primary UAV as shown in Figure 4 is as follows:

$$\text{SIR} = \qquad (9)$$

$$= \frac{\left(\frac{D}{2}\right)^{-4}}{2\left(D_{min}+\frac{D}{2}\right)^{-4} + 2\left(D_{min}+\frac{3D}{2}\right)^{-4} + 2(D_{min}+D)^{-4}}$$

$$= \frac{1}{2(\frac{2D_{min}}{D}+1)^{-4} + 2(\frac{2D_{min}}{D}+3)^{-4} + 2(\frac{2D_{min}}{D}+2)^{-4}}$$

In Figure 5, for the primary UAV we consider D= 400m and for the secondary UAV, all default values of D (450, 500, 550 and 600m) to calculate the optimal SIR value. In this case, the size of the D_{min} distance between the UAVs is the maximum value, so the SIR value at D = 400m has the maximum possible value. But the main problem of the coverage range D = 400m is the minimum telecommunication coverage, which has the most non-coverage of areas. As D increases, D_{min} decreases and SIR decreases.

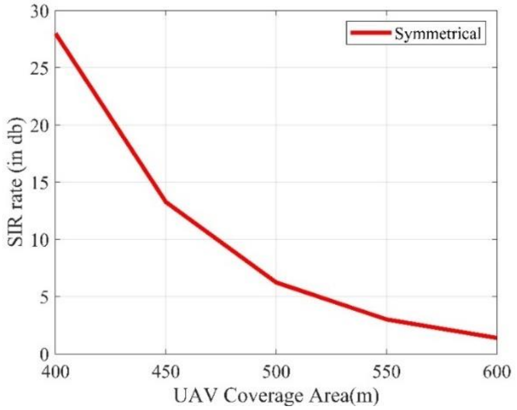


Fig. 5. Estimating SIR based on D in terms of symmetric altitude of UAVs

The coverage level ratio determines the overall performance of the coverage level in the desired multi-UAV system. On the other hand, in particular, the ratio of the total effective surface covered by the primary UAV and the secondary UAV to the target surface is defined. This ratio can be calculated from the following equation[21]:

$$A_C(D) = \frac{2}{L^2}[\int_0^{R_P(D)} \int_{\emptyset_P}^{\emptyset_P=\pi} Rd\,Rd + (M-1) \quad * \quad \int_0^{R_S(D)} \int_{\emptyset_S}^{\emptyset_S=\emptyset_{max}} Rd\,Rd\emptyset \qquad (10)$$

Where L2 is the target surface area desired in the system model. $\emptyset_{max}$ is the coverage of secondary UAVs and is calculated as follows:

$$\emptyset_{max} = \pi - \arccos\left\{\frac{D_{min} + D}{R_S}\right\} \qquad (11)$$

Through equations 10 and 11, we calculated the coverage rate in the environment with symmetrical height (Table 2). Table 2. The coverage rate based on the function of D and D_{min}in UAV symmetric altitude ($h_s = h_p$)

Table 2. The coverage rate based on the function of D and D_{min}in UAV symmetric altitude ($h_s = h_p$)

D	D_{min}	h_p	h_s	***Coverage Rate***	***Coverage percentage***	***The Surface without Service***
400	400	1250	1250	879646	21.99	3120354
450	325	1400	1400	1113302	27.83	2886698
500	250	1570	1570	1374447	34.36	2625553
550	175	1727	1727	1663080	41.57	2336920
600	100	1880	1880	1979203	49.48	2020797

According to Table 2and Figure 6, we found the appropriate coverage area and the desired SIR value, then we designed and implemented the UAV network based on it.

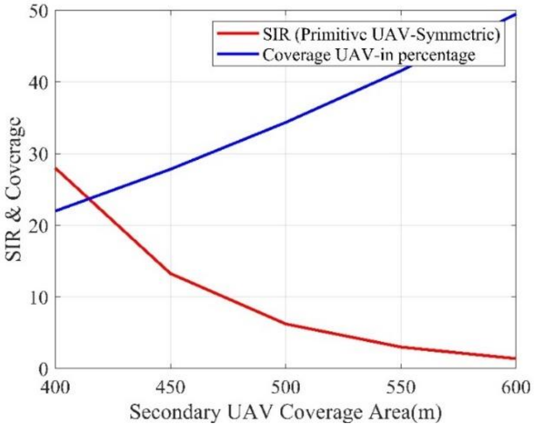


Fig. 6. Estimating the SIR and coverage area based on D in UAVs' symmetrical altitude conditions

4-2- Analyzing the SIR of Directional Antennas with Symmetrical Height Considering the UAVs' Heights

In this case, according to Figure 7, we have included the height of the UAV in the intercellular interference problem, and the new conditions for calculating the SIR has changed as follows:

$$SIR = \frac{(\frac{D}{2})^{-4}}{2(L_1)^{-4} + 2(L_2)^{-4} + 2(L_3)^{-4}} \qquad (12)$$

$$L_1 = \sqrt{(D_{min} + D)^2 + (h_s)^2} \qquad (13)$$

$$L_2 = \sqrt{(D_{min} + \frac{D}{2})^2 + (h_s)^2} \qquad (14)$$

$$L_3 = \sqrt{(D_{min} + \frac{3D}{2})^2 + (h_s)^2} \qquad (15)$$

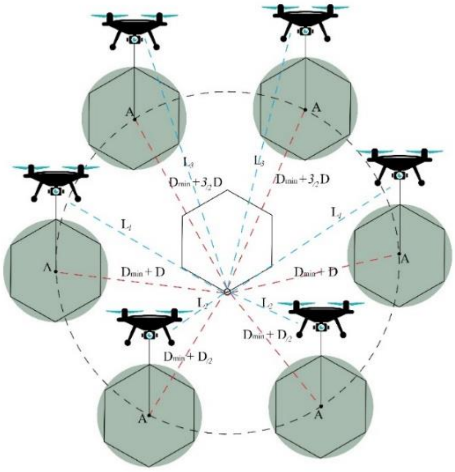


Fig. 7. The Effect of secondary cell interference on the user at the primary cell border considering the height of the UAV

Figure 8 shows the SIR status in symmetric altitude conditions, considering the height of the UAV. This case acts like a situation when the height of the UAV is not considered, and in areas with high coverage and fewer cell separation distances, system status and user coverage will be problematic.

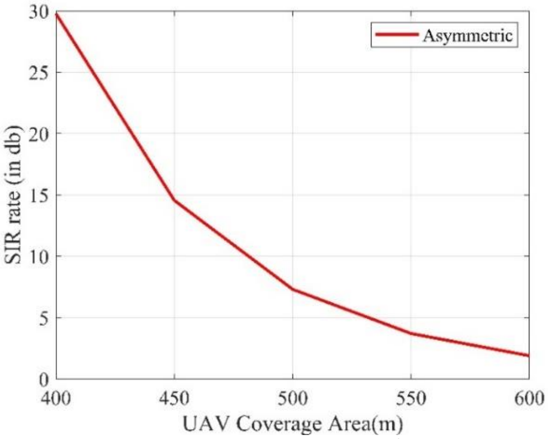


Fig. 8. Estimating the SIR based on D in terms of symmetric heights of UAVs considering the UAVs heights

The effects of secondary UAV signals on user performance at the primary cell border in both conditions with and without considering the height of the UAV are shown in Figure 9. The blue color indicates SIR considering the height of the UAV and the red color indicates SIR regardless of the height of the UAV. The overall condition of the SIR is better in terms of UAV altitude, but in both conditions, there is extensive coverage of the poor quality of customer service at the cell border. The rates of area coverage in conditions regardless of the UAV and considering the UAVs are equal.

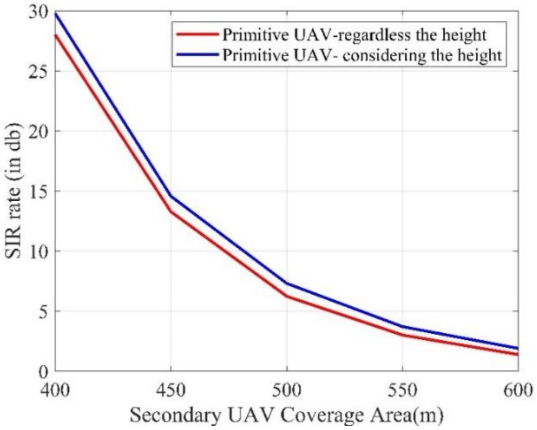


Fig. 9. SIR at symmetrical altitude conditions with and without UAV altitude

4-3- Analyzing the SIR of Omnidirectional Antennas with Asymmetric Height without Considering the UAVs' Heights

When UAVs fly in different classes with asymmetric heights in the urban environment, the goal is to reduce interference and transmit power to have high quality communication. The effect of height on the interference of the urban environment is very obvious. UAVs can be located at the desired and optimal height and continue to provide services in a situation where the number of active users in a cell to receive services is less than its coverage capacity. However, if for humanitarian reasons such as marches, reopening of shopping malls, etc., the number of active users is more than the UAV coverage capacity, based on the amount of demand, other devices help the UAV to get maximum user coverage. The difference is that the height of the UAVs is lower than the non-crowded conditions and their coverage area is also reduced. In this case, the initial UAVs are first placed at the desired height and optimal h_p.

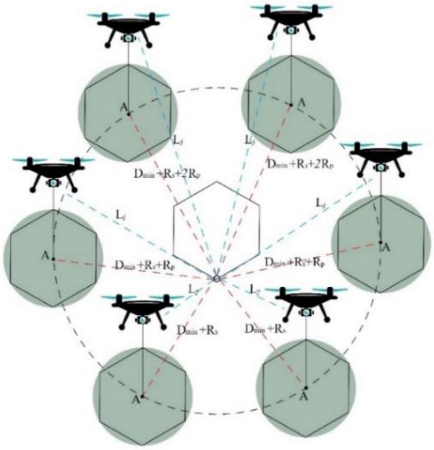


Fig. 10. The effect of secondary cell interference on the user present at the primary cell boundary considering the height of the UAV

The secondary UAVs are then placed at the optimum altitude of h_s relative to the primary UAVs. Therefore, the first goal is to place the initial UAVs at the desired h_p altitude to achieve maximum user coverage in the urban environment. Optimal UAV height selection and proper antenna angle selection minimize interference, reduce transmission power, and increase SIR. For example, if the initial UAV is to cover the D = 500m range, consider other D values (400, 450, 550, and 600m). In this case, in the values of D = 400/450m, the height of the primary UAV is higher than the secondary UAV, in other words: $h_p \succ h_s$. At values D = 550/600m the height of the primary UAV is less than the secondary UAV ($h_p \prec h_s$).

The grond distance of the center of the secondary UAVs relative t the user at the primary cell boundary (Figure 10) is calculaed from the following equation:

$$SIR_{DP,DS} = \frac{(R_p)^{-4}}{2(D_1)^{-4} + 2(D_2)^{-4} + 2(D_3)^{-4}} \quad (16)$$

$$D_1 = D_{min} + R_s + R_p \quad (17)$$

$$D_2 = D_{min} + R_s \quad (18)$$

$$D_3 = D_{min} + R_s + 2R_p \quad (19)$$

$$D_{min} = 1000 - 2R_s - R_p \quad (20)$$

According to Figure 11, the larger is the coverage area of the initial UAV, the lower is the SIR of the network. The red lines indicate the primary cell with D = 600m, which has the lowest SIR value. This range is even lower than the conditions when the primary UAV is D = 400,450,500m and the secondary UAVs are at D = 600m and causes many problems for users' performance in this range.

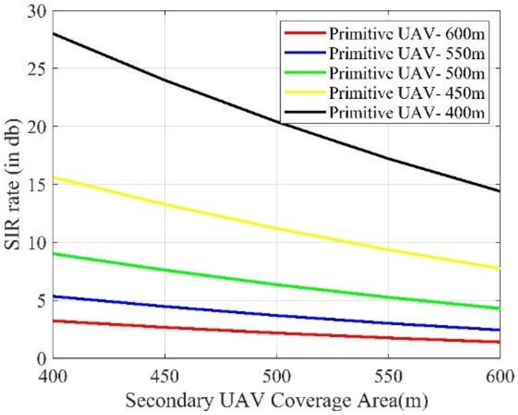


Fig. 11. Analyzing the symmetric interference regardless of the UAVs' heights

For the initial UAV's cover values D = 500, 550,600m, the SIR value is lower than the acceptable threshold for providing communication services to cell border users and the communication link either is not established or has poor quality communication links. In situations where the primary UAV has a value of D = 450m, the coverage ranges of the secondary UAVs provide an acceptable threshold of D = 400, 450m. For D = 500, 550, 600m it is not possible to provide an acceptable minimum threshold, but the condition of the UAV network will be much better than the initial UAV modes D = 500, 550, 600m. If the primary UAV has a coverage range of D = 400m, it will provide a quality link for all D values as areas covered by the secondary UAV.

The UAV's coverage rate in different conditions of primary and secondary UAV deployment can be calculated through Equations 12 and 13. According to Figure 12, the highest value of coverage and the lowest value of SIR are provided by the initial UAV with D=600m and the lowest area coverage is related to D=400m, which provides the best quality of service for users. To get the optimal value, you have to choose the middle ground between different values of D.

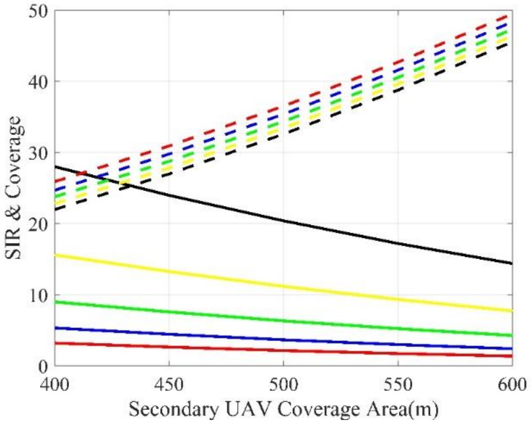


Fig. 12. Estimated SIR (red, blue, green, yellow, and black lines, respectively, for initial UAV coverage with values of 400, 450, 500, 550, and 600 meters) and the coverage area based on D (continuous lines) in the condition of asymmetric height of UAVs.

4-4- Analyzing the SIR of Omnidirectional Antennas with Asymmetric Height without Considering the UAVs' Heights

The interference is calculated by considering the height of the secondary UAVs to the user at the primary cell boundary based on the following equations:

$$L_1 = \sqrt{(D_{min} + R_s + R_p)^2 + (h_s)^2} \quad (21)$$

$$L_2 = \sqrt{(D_{min} + R_s)^2 + (h_s)^2} \quad (22)$$

$$L_3 = \sqrt{(D_{min} + R_s + 2R_p)^2 + (h_s)^2} \quad (23)$$

$$SIR_{Dh,Ds} = \frac{(R_p)^{-4}}{2(L_1)^{-4} + 2(L_2)^{-4} + 2(L_3)^{-4}} \quad (24)$$

As shown in Figure 13, there are many functional similarities between asymmetric systems with and without UAV height.

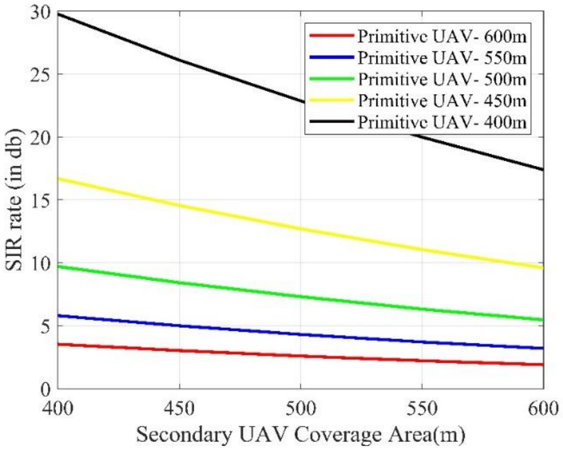


Fig. 13. Analyzing the asymmetric SIR considering the UAV's height

As we move away from D= 400m and move towards D = 600m, the coverage range gradually increases and the SIR gradually decreases, which can be explored between different values of D to find an area with a suitable coverage range and a desirable SIR.

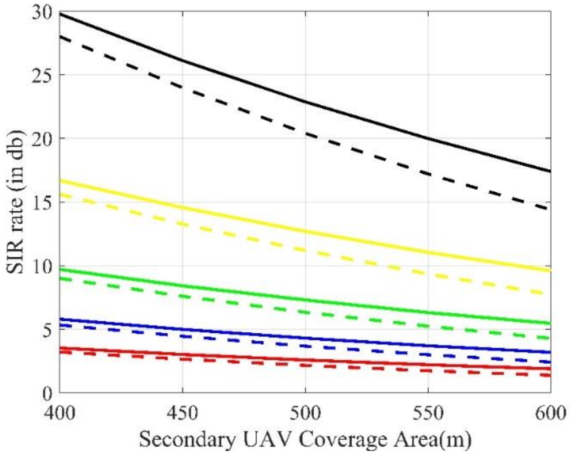


Fig. 14. Comparison of asymmetric SIR considering the height of the UAV (red, blue, green, yellow and black lines respectively for the initial UAV coverage with values of 400, 450, 500, 550 and 600 meters) and without the height of the UAV (continuous lines).

Figure 14 compares SIRs between asymmetric systems with and without UAV heights, which are similar in performance. SIR performance in an asymmetric system considering the height of the UAV at all D values is better than the asymmetric system without considering the height of the UAV. In the primary UAV area with D= 450m and secondary D = 550m, which did not receive the acceptable minimum SIR threshold in the system without considering the UAV height, in the system, considering the UAV height, received the desired threshold. Also, in the primary UAV with D = 500m and the secondary D = 400m, it is close to the desired threshold and has an acceptable value compared to the system, regardless of the height of the UAV. Other cases are equal in terms of SIR threshold and area coverage.

4-5- Analyzing the SIR of Directional Antennas with Symmetrical Height without Considering the UAVs' heights

In omnidirectional antennas, the wave propagation angle is only 360 degrees, while in directional antennas, the angle can be selected between 0 and 360 degrees. Normally, the angles of 60- or 120-degree directional antennas are selected, which have already been studied in terms of system and efficiency and have received appropriate results.

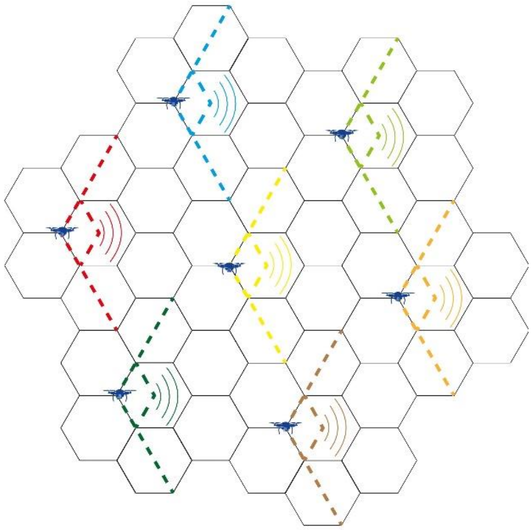


Fig. 15. Using a unidirectional antenna mounted on a UAV in a two-dimensional space of an urban environment

The directional antenna can reduce interference and blur and improve the quality of communication by concentrating the transmitted energy in one direction. We have installed antennas with 120-degree angles on the UAVs. The number of UAVs that affect the user at the cell boundary of the primary UAV if all-round antennas are used is at least 6 UAVs.

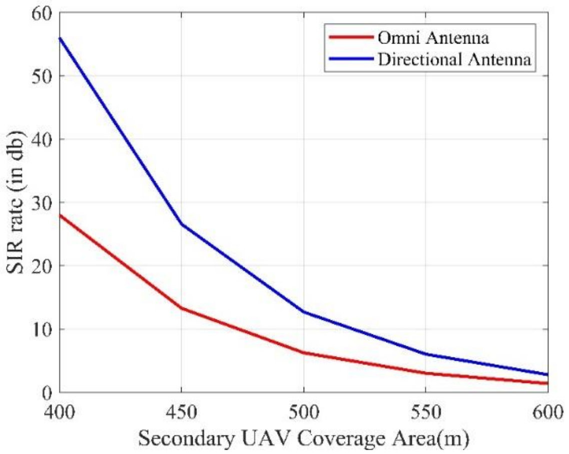


Fig.16. The comparison of SIR with and without directional antenna in symmetrical UAVs conditions without applying UAVs' heights effects

We have installed antennas with 120-degree angles on the UAVs. The number of UAVs that affect the user at the cell boundary of the primary UAV if all-round antennas are used is at least 6 UAVs. By using one-way antennas according to Figure 15 and considering the position of the UAVs relative to the cells, the interference is reduced to 3 UAVs and we have eliminated 50% of the network interference with this action. In one sentence, the use of a one-way antenna with an angle of 120 degrees maximizes interference in the urban environment and the power consumption for data transmission to a minimum and the quality of communication. The amount of SIR in the direction of using the directional antenna is calculated from the following equation:

$$SIR = \frac{S}{I} = \frac{1}{(\frac{2D_{min}}{D}+1)^{-4} + (\frac{2D_{min}}{D}+3)^{-4} + (\frac{2D_{min}}{D}+2)^{-4}} \qquad (25)$$

SIR function using directional antenna has the same function as omnidirectional antenna and has the maximum SIR value at D = 400m and at any distance from D = 400m and moving towards D = 600m, the SIR value gradually decreases (Figure 16). By using directional antenna in all D values, SIR has a better situation than all-purpose antenna conditions and has improved network condition and service quality.

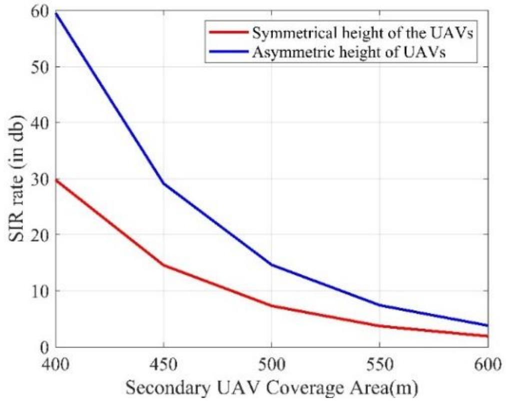


Fig.17. Comparing the SIR with and without directional antenna in symmetrical UAV conditions with UAV altitude effects

4-6- Analyzing the Directional Antenna's SIR with Symmetrical Height, Considering the UAVs' Heights

The SIR values in directional antenna conditions with symmetrical height, considering the UAVs' heights are calculated based on the following equation:

$$SIR = \frac{(\frac{D}{2})^{-4}}{(L_1)^{-4} + (L_2)^{-4} + (L_3)^{-4}} \qquad (26)$$

In the above equation, L1, L2 and L3 can be calculated from equations 12, 13 and 14, respectively.

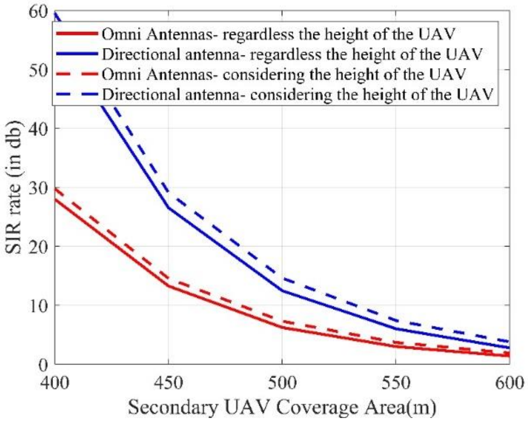


Fig. 18. The comparison of SIR with and without directional antenna in symmetrical UAVs conditions with UAV altitude effects

Figure 18 shows an overview of the SIR situation in terms of the use of symmetric UAVs, which was the lowest absolute SIR to symmetric UAVs with all-round antennas, regardless of the height of the UAVs.

4-7- Analyzing the SIR in Directional Antennas with Asymmetric Height without Considering the UAVs' Heights

The SIR values in directional antenna conditions with asymmetric height, without considering the UAVs' heights are calculated based on the following equation:

$$SIR = \frac{(\frac{D}{2})^{-4}}{(L_1)^{-4} + (L_2)^{-4} + (L_3)^{-4}} \qquad (27)$$

The values of L_1 , L_2 and L_3 can be calculated from equations 21, 22 and 23, respectively. In Figure 19, the smaller the radius of coverage of the original UAV, the smaller the coverage range. But the SIR will be at its highest, and vice versa .

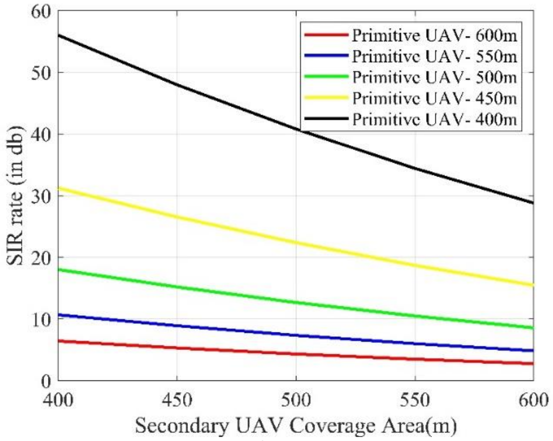


Fig. 19. Analyzing the SIR of asymmetric UAVs using directional antenna without applying UAV Altitude effects

The most important factor that determines the amount of SIR in an asymmetric network is the amount of coverage area D of the primary UAV. Selecting different values of D changes the amount of SIR on a large scale and creates a large jump in the dimensions of SIR, which is a significant difference between SIR with values of 400 and 600m.

To access the desired SIR, we must first calculate and select the optimal range of the initial UAV, which has the greatest impact on link quality. Then we select the optimal value of the area covered by the secondary UAV, which is the range of SIR improvement in a specific and limited range and has a much less effect than the coverage range of the primary UAV. To better understanding of the performance of directional antenna and omnidirectional antenna at asymmetric altitude conditions between the primary and secondary UAVs, we must compare the performance between them, which is illustrated in Figure 20. In general, the SIR performance of a directional antenna is significantly better than an omnidirectional antenna, providing better performance and a better communication link for all D values.

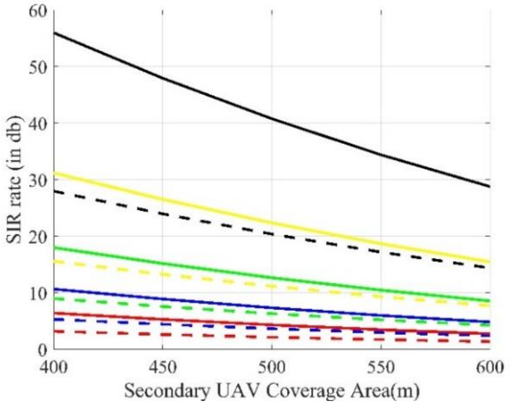


Fig.20. Analyzing the SIR in symmetric UAVs using directional antennas (red, blue, green, yellow and black lines respectively for the initial UAV coverage with values of 400, 450, 500, 550 and 600 meters) and omni-directional antennas (continuous lines) without applying the effects of UAVs' heights.

4-8- Analyzing the SIR of Directional Antennas with Asymmetric Height, without Considering the UAVs' Heights

The SIR values in directional antenna conditions with asymmetric height, without considering the UAVs' heights are calculated according to the following equation:

$$SIR_{DH,DP} = \frac{(R_p)^{-4}}{(L_1)^{-4} + (L_2)^{-4} + (L_3)^{-4}} \qquad (28)$$

According to Figure 21, we conclude that the effects of selecting the coverage range D for the initial UAV had the greatest effect on the SIR. The larger the coverage range of the initial UAV, the higher the SIR value and the better the link quality. The red lines indicate the primary cell with D = 600m, which has the lowest SIR value.

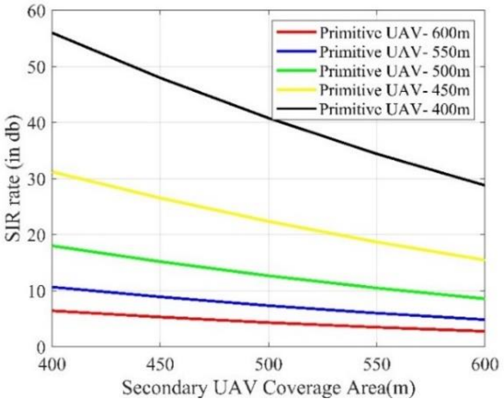


Fig. 21. Investigating the SIR of asymmetric UAVs using a directional antenna by applying the effects of the UAV height

The SIR value is lower in the condition that the coverage range of the primary UAV is D = 600m, even compared to the condition when the primary UAV is D =400, 450 and 500m and the secondary UAV is D=600m, and causes many problems for users in this range. In the coverage range of the initial UAV with D = 600m, the SIR value is below the acceptable threshold for providing communication services to cell border users and the communication link is not established or the communication is of poor quality. In case the primary UAV has a value of D = 500m, it has provided an acceptable threshold for the coverage ranges of the secondary UAVs D = 400/450m and for D = 500,550 and 600m it has not been able to provide an acceptable minimum threshold. For other D points for the primary UAV an acceptable threshold is provided for all coverage areas D for the secondary UAV. The best case in this situation is the primary UAV with D = 500m and the secondary UAV with D = 600m.

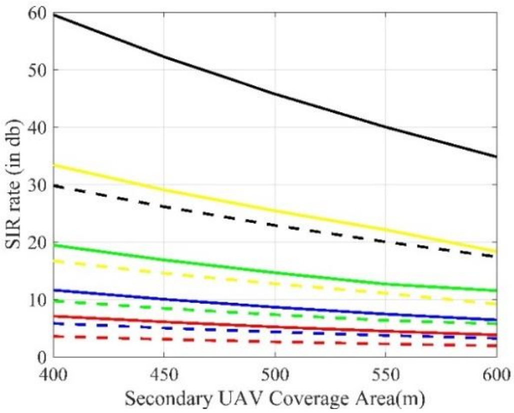


Fig. 22. Analyzing the SIR in asymmetric UAVs using directional antennas (red, blue, green, yellow and black lines respectively for the initial UAV coverage with values of 400, 450, 500, 550 and 600 meters) and omni-directional antennas (continuous lines) with applying the effects of UAVs' heights.

Figure 22 examines the SIR of UAVs with omnidirectional and directional antennas, considering the heights of the UAVs. In general, we notice that the network's SIR in the directional antenna is better than that of the omnidirectional antenna.

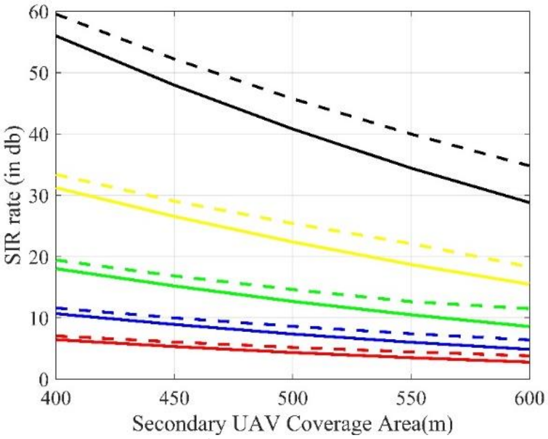


Fig. 23. The comparison of SIR using directional antenna in symmetric UAV conditions with (red, blue, green, yellow and black lines respectively for the initial UAV coverage with values of 400, 450, 500, 550 and 600 meters)/without (continuous lines) applying the effects of UAVs' height.

Figure 23 compares the SIR between asymmetric systems using directional antennas with and without considering the heights of the UAVs, which are very similar in terms of performance. Finally, we have put the optimal SIR values in the use of omnidirectional and pointed antennas, with and without considering the height and symmetric and asymmetric height in Table 3.

Table 3. Optimal SIR Values for Different D Conditions

Secondary Optimal D	***Primary Optimal D***	***UAVs' Height Calculation***	***UAVs' Height***	***Antenna Type***
450	450	Without height	Sym	Omni
450	450	With height	Sym	Omni
550 600	500 450	Without height	Asym	Omni
600 550	400 450	With height	Asym	Omni
500	500	Without height	Sym	direct
500	500	With height	Sym	direct
550 600	500 450	Without height	Asym	direct
600 450	500 550	With height		direct

*Sym= Symmetric and Asym= Asymmetric
Omni= Omnidirectional and direct=directional

5- Conclusions

In this paper, the coverage area and signal to interference experienced by users at a UAV-enabled network were investigated against different configurations of antenna and UAV heights. The optimal options have been achieved using extensive numerical computations. In overall, three factors affect the improvement of the coverage area and SIR. The first and the most effective factor is choosing the type of antenna. The directional antenna has a better SIR in all coverage areas than the omnidirectional antenna. The second factor on SIR is the symmetric and asymmetric height of UAVs. After calculations, we found that asymmetrical systems achieved a better SIR than symmetrical systems. The third factor with the least impact on the amount of interference is the height of the drone, which has a very small impact on the system performance. For more comparison, we also estimated the situation without considering the height of the drone, which results in no system performance improvement. Finally, the best SIR condition related to the system is the directional antenna, asymmetric height, and considering the height of the UAVs.

References

[1] Indu and R. Singh, “Trajectory planning and optimization for UAV communication: A review,” J. Discret. Math. Sci. Cryptogr., vol. 23, no. 2, pp. 475–483, 2020, doi: 10.1080/09720529.2020.1728901.

[2]Y. Zeng, R. Zhang, and T. J. Lim, “Wireless communications with unmanned aerial vehicles: Opportunities and challenges,” IEEE Commun. Mag., vol. 54, no. 5, pp. 36–42, 2016, doi: 10.1109/MCOM.2016.7470933.

[3] J. Ding, H. Mei, I. Chih-Lin, H. Zhang, and W. Liu, “Frontier progress of unmanned aerial vehicles optical wireless technologies,” Sensors (Switzerland), vol. 20, no. 19, pp. 1–35, 2020, doi: 10.3390/s20195476.

[4] A. Al-Hourani, S. Kandeepan, and S. Lardner, “Optimal LAP altitude for maximum coverage,” IEEE Wirel. Commun. Lett., vol. 3, no. 6, pp. 569–572, 2014, doi: 10.1109/LWC.2014.2342736.

[5] A. Al-Hourani, S. Chandrasekharan, G. Kaandorp, W. Glenn, A. Jamalipour, and S. Kandeepan, “Coverage and rate analysis of aerial base stations [Letter],” IEEE Trans. Aerosp. Electron. Syst., vol. 52, no. 6, pp. 3077–3081, 2016, doi: 10.1109/TAES.2016.160356.

[6] S. H. Mostafavi Amjad, V. Solouk, and H. Kalbkhani, "Energy-Efficient User Pairing and Power Allocation for Granted Uplink-NOMA in UAV Communication Systems", JIST, Vol. 10, No. 4, 2022, 312-323. doi: 10.52547/jist.27369.10.40.312

[7] W. Shafik, S. M. Matinkhah, and M. Ghasemzadeh, "A Fast Machine Learning for 5G Beam Selection for Unmanned Aerial Vehicle Applications", JIST, Vol. 7, No. 4, 2019, 262-277. doi: 10.7508/jist.2019.04.003

[8] M. Alzenad, A. El-Keyi, and H. Yanikomeroglu, “3-D Placement of an Unmanned Aerial Vehicle Base Station for

Maximum Coverage of Users with Different QoS Requirements," IEEE Wirel. Commun. Lett., vol. 7, no. 1, pp. 38–41, 2018, doi: 10.1109/LWC.2017.2752161.

[9] J. Lyu, Y. Zeng, R. Zhang, and T. J. Lim, "Placement Optimization of UAV-Mounted Mobile Base Stations," IEEE Commun. Lett., vol. 21, no. 3, pp. 604–607, 2017, doi: 10.1109/LCOMM.2016.2633248.

[10] R. I. Bor-Yaliniz, A. El-Keyi, and H. Yanikomeroglu, "Efficient 3-D placement of an aerial base station in next generation cellular networks," 2016 IEEE Int. Conf. Commun. ICC 2016, 2016, doi: 10.1109/ICC.2016.7510820.

[11] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Mobile internet of things: Can UAVs provide an energy-efficient mobile architecture?," 2016 IEEE Glob. Commun. Conf. GLOBECOM 2016 - Proc., 2016, doi: 10.1109/GLOCOM.2016.7841993.

[12] S. Kumar, S. Suman, and S. De, "Backhaul and delay-aware placement of UAV-enabled base station," INFOCOM 2018 - IEEE Conf. Comput. Commun. Work., pp. 634–639, 2018, doi: 10.1109/INFCOMW.2018.8406910.

[13] M. Gruber, "Role of altitude when exploring optimal placement of UAV access points," IEEE Wirel. Commun. Netw. Conf. WCNC, vol. 2016-Septe, no. Wcnc, 2016, doi: 10.1109/WCNC.2016.7565073.

[14] Y. Chen, H. Zhang, and M. Xu, "The coverage problem in UAV network: A survey," 5th Int. Conf. Comput. Commun. Netw. Technol. ICCCNT 2014, no. 3, pp. 3–7, 2014, doi: 10.1109/ICCCNT.2014.6963085.

[15]Y. Zeng and R. Zhang, "Energy-Efficient UAV Communication with Trajectory Optimization," IEEE Trans. Wirel. Commun., vol. 16, no. 6, 2017, doi: 10.1109/TWC.2017.2688328.

[16] M. Mozaffari, W. Saad, M. Bennis, and M. Debbah, "Efficient Deployment of Multiple Unmanned Aerial Vehicles for Optimal Wireless Coverage," IEEE Commun. Lett., vol. 20, no. 8, pp. 1647–1650, 2016, doi: 10.1109/LCOMM.2016.2578312.

[17] I. Valiulahi and C. Masouros, "Multi-UAV Deployment for Throughput Maximization in the Presence of Co-Channel Interference," IEEE Internet Things J., vol. 8, no. 5, pp. 3605–3618, 2021, doi: 10.1109/JIOT.2020.3023010.

[18] E. Chu, J. M. Kim, and B. C. Jung, "Interference Analysis of Directional UAV Networks: A Stochastic Geometry Approach," Int. Conf. Ubiquitous Futur. Networks, ICUFN, vol. 2019-July, pp. 9–12, 2019, doi: 10.1109/ICUFN.2019.8806095.

[19] W. Mei and R. Zhang, "Cooperative Downlink Interference Transmission and Cancellation for Cellular-Connected UAV: A Divide-and-Conquer Approach," IEEE Trans. Commun., vol. 68, no. 2, pp. 1297–1311, 2020, doi: 10.1109/TCOMM.2019.2955953.

[20] W. Lu, P. Si, G. Huang, H. Peng, S. Hu, and Y. Gao, "Interference Reducing and Resource Allocation in UAV-Powered Wireless Communication System," 2020 Int. Wirel. Commun. Mob. Comput. IWCMC 2020, pp. 220–224, 2020, doi: 10.1109/IWCMC48107.2020.9148329.

[21] A. A. Khuwaja, G. Zheng, Y. Chen, and W. Feng, "Optimum Deployment of Multiple UAVs for Coverage Area Maximization in the Presence of Co-Channel Interference," IEEE Access, vol. 7, pp. 85203–85212, 2019, doi: 10.1109/ACCESS.2019.2924720.

[22] M. Jacovic, O. Bshara, and K. R. Dandekar, "Waveform Design of UAV Data Links in Urban Environments for Interference Mitigation," IEEE Veh. Technol. Conf., vol. 2018-Augus, pp. 1–5, 2018, doi: 10.1109/VTCFall.2018.8690581.

[23] L. Zhou, X. Chen, M. Hong, S. Jin, and Q. Shi, "Efficient Resource Allocation for Multi-UAV Communication against Adjacent and Co-Channel Interference," IEEE Trans. Veh. Technol., vol. 70, no. 10, pp. 10222–10235, 2021, doi: 10.1109/TVT.2021.3104279.

[24] L. Xie, J. Xu, and Y. Zeng, "Common Throughput Maximization for UAV-Enabled Interference Channel with Wireless Powered Communications," IEEE Trans. Commun., vol. 68, no. 5, pp. 3197–3212, 2020, doi: 10.1109/TCOMM.2020.2971488.

[25] W. Tang, H. Zhang, Y. He, and M. Zhou, "Performance Analysis of Multi-Antenna UAV Networks with 3D Interference Coordination," IEEE Trans. Wirel. Commun., 2021, doi: 10.1109/TWC.2021.3137347.

[26] P. Li, L. Xie, J. Yao, and J. Xu, "Cellular-Connected UAV with Adaptive Air-to-Ground Interference Cancellation and Trajectory Optimization," IEEE Commun. Lett., no. April, pp. 1–1, 2022, doi: 10.1109/lcomm.2022.3164905.

[27] C. Gao, Z. Xue, W. Li, and W. Ren, "The influence of electromagnetic interference of HPM on UAV," 2021 Int. Conf. Microw. Millim. Wave Technol. ICMMT 2021 - Proc., 2021, doi: 10.1109/ICMMT52847.2021.9617977.

[28] T. Z. H. Ernest, A. S. Madhukumar, R. P. Sirigina, and A. K. Krishna, "Impact of Cellular Interference on Uplink UAV Communications," IEEE Veh. Technol. Conf., vol. 2020-May, 2020, doi: 10.1109/VTC2020-Spring48590.2020.9128682.

[29] J. Urama et al., "UAV-Aided Interference Assessment for Private 5G NR Deployments: Challenges and Solutions," IEEE Commun. Mag., vol. 58, no. 8, pp. 89–95, 2020, doi: 10.1109/MCOM.001.00042.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 62-71

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Proposing an FCM-MCOA Clustering Approach Stacked with Convolutional Neural Networks for Analysis of Customers in Insurance Company

Meisam Yadollahzadeh Tabari[1*], Motahare Ghavidel[1], Mehdi Golsorkhtabaramiri[1]

[1.]Department of Computer Engineering, Islamic Azad University, Babol Branch, Babol, Iran.

Received: 06 March 2023/ Revised: 20 Sep 2023/ Accepted: 11 Nov 2023

Abstract

To create a customer-based marketing strategy, it is necessary to perform a proper analysis of customer data so that customers can be separated from each other or predict their future behavior. The datasets related to customers in any business usually are high-dimensional with too many instances and include both supervised and unsupervised ones. For this reason, companies today are trying to satisfy their customers as much as possible. This issue requires careful consideration of customers from several aspects. Data mining algorithms are one of the practical methods in businesses to find the required knowledge from customer’s both demographic and behavioral. This paper presents a hybrid clustering algorithm using the Fuzzy C-Means (FCM) method and the Modified Cuckoo Optimization Algorithm (MCOA). Since customer data analysis has a key role in ensuring a company's profitability, The Insurance Company (TIC) dataset is utilized for the experiments and performance evaluation. We compare the convergence of the proposed FCM-MCOA approach with some conventional optimization methods, such as Genetic Algorithm (GA) and Invasive Weed Optimization (IWO). Moreover, we suggest a customer classifier using the Convolutional Neural Networks (CNNs). Simulation results reveal that the FCM-MCOA converges faster than conventional clustering methods. In addition, the results indicate that the accuracy of the CNN-based classifier is more than 98%. CNN-based classifier converges after some couples of iterations, which shows a fast convergence in comparison with the conventional classifiers, such as Decision Tree (DT), Support Vector Machine (SVM), K-Nearest Neighborhood (KNN), and Naive Bayes (NB) classifiers.

Keywords: Customer Clustering; Fuzzy C-Means; Cuckoo Optimization; Convolutional Neural Networks.

1- Introduction

Customers are one of the prominent parts of the commercial exchanges, and no business can succeed without having satisfied and faithful customers. This issue requires the study of customers from various aspects. To create a customer-based marketing strategy, several techniques, such as clustering and scoring, can be utilized for data analysis. Data mining algorithms offer several practical methods for businesses to extract the expected data.

This paper presents some solutions for clustering and classification of the customers. The Insurance Company (TIC) dataset is utilized which first introduced in Computational Intelligence and Learning (COIL) Challenge 2000 [1]. This challenge is a common data mining problem to predict the potential customers using the training and test datasets of customers. Identify potential purchasers is a powerful approach to advertising and market a product. If the company had a precise data of their clients, they can send fewer advertising emails and some expenses can be reduced in this way. For example, an insurance company often wants to know which customers are willing to buy a particular product, such as caravan insurance policy.

Several solutions have been recommended for this challenge based on data mining and computational intelligence. We generally tend to remove some features that are not effective. On the other hand, we tend to preserve client records, since removing clients may eliminate some important client groups. To handle this tradeoff, Fuzzy C-Means (FCM) clustering is a popular technique for selecting effective features and clients [2]. Using the k-means approach for clustering is another well-

✉ **Meisam Yadollahzadeh Tabari**
m_tabari@baboliau.ac.ir

known technique that uses an iterative approach to minimizing the sum of squared errors [3]. The purpose of FCM clustering is to group the data to pick out the groups with considerable numbers of potential purchasers. To identify the best centroid, the FCM algorithm employs the membership function. However, the number of features is generally large, and clustering is computationally difficult in such a condition. Moreover, the data are inherently sparse, and these sporadic data can be problematic for the clustering mechanisms. In summary, the clustering approaches shoulld combat with two main problems: the large number of clusters and imbalanced data. The former problem can be solved by merging similar clusters through an optimization process [4]. A popular and simple technique for reducinge features is selecting one feature each time. When the dimensions of the features are reduced, the basic form of the FCM clustering may lead to acceptable results [3].

The latter problem is that the distribution of the features is unbalanced, i.e., there are some classes with numerous members and the rest of classes have a few members. Existing classification techniques tend to find the classes with more members. There are several data-level approaches in the literature [5], [6], such as sampling techniques, that modify the distribution of the train data. These techniques have this power that are not dependent on the classifier types utilized. For example, the under-sampling technique ignores some data samples and provides a subgroup of the initial data. In addition to these data-level aproaches, imbalence data problem can effectively be solved by modifying well-known algorithms, such as the FCM algorithm

Customers are one of the prominent parts of the commercial exchanges, and no business can succeed without having satisfied and faithful customers. This issue requires the study of customers from various aspects. To create a customer-based marketing strategy, several techniques, such as clustering and scoring, can be utilized for data analysis. Data mining algorithms offer several practical methods for businesses to extract the expected data.

This paper presents some solutions for clustering and classification of the customers. The Insurance Company (TIC) dataset is utilized which first introduced in Computational Intelligence and Learning (COIL) Challenge 2000 [1]. This challenge is a common data mining problem to predict the potential customers using the training and test datasets of customers. Identify potential purchasers is a powerful approach to advertising and market a product. If the company had a precise data of their clients, they can send fewer advertising emails and some expenses can be reduced in this way. For example, an insurance company often wants to know which customers are willing to buy a particular product, such as caravan insurance policy.

Several solutions have been recommended for this challenge based on data mining and computational intelligence. We generally tend to remove some features that are not effective. On the other hand, we tend to preserve client records, since removing clients may eliminate some important client groups. To handle this tradeoff, Fuzzy C-Means (FCM) clustering is a popular technique for selecting effective features and clients [2].

Using the k-means approach for clustering is another well-known technique that uses an iterative approach to minimizing the sum of squared errors [3]. The purpose of FCM clustering is to group the data to pick out the groups with considerable numbers of potential purchasers. To identify the best centroid, the FCM algorithm employs the membership function. However, the number of features is generally large, and clustering is computationally difficult in such a condition. Moreover, the data are inherently sparse, and these sporadic data can be problematic for the clustering mechanisms. In summary, the clustering approaches shoulld combat with two main problems: the large number of clusters and imbalanced data. The former problem can be solved by merging similar clusters through an optimization process [4]. A popular and simple technique for reducinge features is selecting one feature each time. When the dimensions of the features are reduced, the basic form of the FCM clustering may lead to acceptable results [3].

The latter problem is that the distribution of the features is unbalanced, i.e., there are some classes with numerous members and the rest of classes have a few members. Existing classification techniques tend to find the classes with more members. There are several data-level approaches in the literature [5], [6], such as sampling techniques, that modify the distribution of the train data. These techniques have this power that are not dependent on the classifier types utilized. For example, the under-sampling technique ignores some data samples and provides a subgroup of the initial data. In addition to these data-level aproaches, imbalence data problem can effectively be solved by modifying well-known algorithms, such as the FCM algorithm.

2- Related Works

In [4], the authors suggested a new method for handling the CoIL Challenge 2000, where the goal of the clustering is to find a scoring table and select more important features. They perform FCM clustering for each feature and compute a response density, as a measure of the predictive capability for each cluster. The response density can be defined as the ratio of the number of caravan policy purchasers to the total number of people in the cluster. In other words, calculating the response density yields a criterion that permits us to order the clusters based on their predictive effectiveness. We compute a score for each client adopting their membership values and response densities. The clustering procedure can be executed many times, and the selection of useful features can be completed using these scores [7], [8].

In optimization-based clustering, the clustering objective is considered as the minimum sum of the square error and the optimization procedure is used in the related algorithm to solve the clustering objective. In this regard, most of the algorithms have utilized centroid sets as the solutions to generate optimal cluster centroids. In a fuzzy technique, a customer can belong to different clusters simultaneously, and the results highly depend on the initial cluster centers. Improper selection of centers causes that the algorithm fall into a local solution and this degrades the performance of the FCM clustering method [9], [10]. Several techniques, such as Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO) have been recommended to combat with this issue [11]. The authors in [27] have introduced a hybrid data clustering approach using GA and K-means algorithms. Their results demonstrate that this hybrid approach can improve the clustering performance. In [12-14], some hybrid optimization algorithms are suggested to obtain a better clustering performance. In order to achieve a faster convergence, in [15] and [16], the authors have suggested some hybrid methods based on the K-means and FCM clustering.

Cuckoo search algorithm is a powerful tool which can be utilized in many optimization problems [17], [18]. Cuckoo Optimization Algorithm is inspired by the behavior of a bird family called Cuckoo. Smart egg laying and breeding procedure of cuckoos is the essential of this algorithm. Instead of establishing their own nest, female cuckoos offspring in nests of host birds, and if these eggs are not destroyed by the host birds, they can grow into adult birds. Mature cuckoos immigrate to new habitats and this immigration lead to discover more food and better conditions for reproduction. Several groups are created in other locations due to the immigration, and the community with outstanding conditions is preferred as the target for other cuckoos. From an algorithmic point of view, the best environment is the global optimum of the objective function. We can form our groups, identify the best group, and find most proper target for immigration. Since the mature cuckoos are dispersed in an environment, understanding that which cuckoo is a member of which of the groups is a problematic task. The grouping is often accomplished with one of the well-known clustering techniques and their objective value is calculated. After some couples of iterations, all population immigrate to the best habitat and algorithm converges. The cuckoo search algorithm can work better than many other optimization techniques in solving common benchmark problems [19], [20].

2-1- Contributions

Innovative aspects of this work are summarized below:

- To improve the results of the clustering, a new hybrid clustering method based on FCM and Modified Cuckoo Optimization Algorithm (MCOA) is presented. The results show that FCM-MCOA converges faster than some existing methods, such as and Invasive Weed Optimization (IWO) and GA methods. Moreover, the final value of the cost function is less than the aforementioned conventional methods.
- Our ultimate goal is to identify potential caravan insurance purchasers. In order to predict who would be interested in purchasing a caravan policy, a classifier is also designed based on the principles of the Convolutional Neural Networks (CNNs). Simulation results show that our classifier is more accurate than some conventional classifiers, such as SVM and NB classifiers. We also showed that increasing the number of out-put neurons in the FC layers can improve the performance of the classifier.

The rest of this paper is organized as follows. The suggested FCM-MCOA approach and CNN-based classifier are represented in Section 2. Simulation results are described in Section 3. Finally, conclusions are represented in Section 4.

3- The Proposed Method

In this section, we first introduce the hybrid FCM-MCOA approach. Afterward, we define the designed CNN-based classifier in order to predict probable purchasers of a special type of insurance service.

3-1- Subheadings

In this paper, the MCOA approach is considered as a stochastic search technique to find new cluster centers. The cuckoo search procedure is to avoid the solutions from being captured into the local points and finding the global solutions. Another advantage of the MCOA is that a few parameters should be adjusted in this method. The proposed hybrid FCM-MCOA can be summarized as follows:

1. Some initial habitats are randomly allocated to the cuckoos.
2. Some eggs are assigned to each cuckoo. The minimum and maximum number of eggs should be defined.
3. For each cuckoo, an Egg Laying Radius (ELR) should be defined:

$$\mathrm{ELR} = \alpha \times \frac{\text{Number of current cuckoo's eggs}}{\text{Total number of eggs}} \times (\mathrm{var_h} - \mathrm{var_l}) \tag{1}$$

where α is an integer, var_h and var_i are the upper and lower bounds of the decision variables, respectively.

4. Each cuckoo lays its eggs within its ELR.
5. A number of eggs are detected and destroyed by the host bird.
6. Cuckoo eggs turn into young birds.
7. The habitat of the young birds is assessed. The habitat is actually a cost function of the problem.
8. The number of live cuckoos should be limited. The N_{max} indicates the maximum number of live cuckoos.
9. Cuckoos should immigrate to a better habitat in the search area. Therefore, we should first group the cuckoos. In this paper, grouping is done utilizing FCM method. Calculating the cluster centers by considering the membership values of each data sample as well as calculating the membership values are performed using the FCM. The objective function for data samples is defined as [10], [11].

$$J_m(U,V) = \sum_{i=1}^{c}\sum_{k=1}^{n} u_{ik}^{m} D_{ik}(x_k, v_i) \tag{2}$$

In where n is the number of samples, $m > 1$ is the "fuzzifier", c is the number of clusters, U is the membership matrix, V is the cluster centers set, and $D_{ik}(x_k, v_i)$ represents squared distance between the k^{th} data sample and i^{th} cluster center. The membership values and cluster centers are given by

$$u_{ik} = \frac{D_{ik}(x_k, v_i)^{\frac{1}{1-m}}}{\sum_{j=1}^{c} D_{jk}(x_k, v_j)^{\frac{1}{1-m}}} \tag{3}$$

$$v_i = \frac{\sum_{j=1}^{n} (u_{ij})^m x_j}{\sum_{j=1}^{n} (u_{ij})^m} \tag{4}$$

1. The average cost function of each group is calculated and the best group is identified. This group of cuckoos then migrate to the best habitat.
2. In the MCOA, the ELR should be reduced in the next iteration. It can be done by reducing the α in Eq. (1).
3. If the stop condition of the algorithm is fulfilled, the optimization process ends. Otherwise, it starts again from the step 2.

Calculating the centers of clusters is often a difficult task. The optimization algorithm based on the FCM-MCOA is adopted to effectively calculate the centers of the clusters. For this purpose, the following cost function is minimized by the algorithm.

$$J = \sum_{j=1}^{k}\sum_{i=1}^{n} \left\| x_i^{(j)} - c_j \right\|^2 \tag{5}$$

where $\left\| x_i^{(j)} - c_j \right\|$ is a measure of the distance between the data points $x_i^{(j)}$ and the center of the cluster c_j .The parameter J defines the distance of n data points from their corresponding cluster centers. The flowchart of the proposed clustering technique is presented in Figure 1.

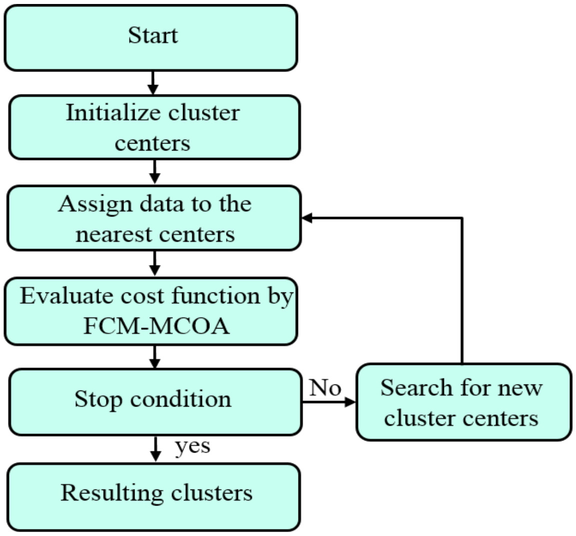

Start
Initialize cluster centers
Assign data to the nearest centers
Evaluate cost function by FCM-MCOA
Stop condition
No
Search for new cluster centers
yes
Resulting clusters

Figure 1 Flowchart of the proposed FCM-MCOA approach.

3-2- Proposed CNN-based Classifier

Several classification techniques are designed to solve the COIL challenge problem reducing the imbalanced data issue [25]. A strong classifier should properly identify two classes of the caravan policyholders and the rest of people (i.e., 0 and 1 classes).

CNNs have many applications in pattern recognition [21]-[23], and digital image processing [28]. CNN-based classifiers may have different architectures in different applications. After the input layer, there are one or more Fully Connected (FC) layers, such as the standard feed-forward neural networks. These layers are stacked to construct a deep model. The basic architecture of the CNN-based classifiers includes some other layers, such as convolutional filters and batch normalization layers. Finally, the last FC layer out-puts the class label. In addition to this basic architecture, several architecture variations have been suggested for modern applications. CNNs represent the input data in the form of multi-dimensional arrays, and each layer connects to the next layer by neurons. The effect of each neuron is different from other neurons. A typical CNN-based classifier includes the parts described below:

- Input layer: Consider the l^{th} layer of the network, whose inputs form an order 3 tensor as $x^l \epsilon R^{M^l \times N^l \times P^l}$. A function converts the input x^l to an out-put y. Note that the out-put of the l^{th} layer is the input to the layer $l+1$. In other words, y and x^{l+1} actually refer to one object. We assume that the out-put is of size $M^{l+1} \times N^{l+1} \times P^{l+1}$. Thus, an out-put element is indexed by a triplet $(i^{l+1}; j^{l+1}; k^{l+1})$, where $0 \leq i^{l+1} < M^{l+1}$, $0 \leq j^{l+1} < N^{l+1}$, and $0 \leq k^{l+1} < P^{l+1}$.
- ReLU layer: This layer does not change the size of the input, and x^l and y have the same size. It means that the ReLU layer can be considered as a separate section for each element of the input, i.e.

$$y_{i,j,k} = max\{0, x^l_{i,j,k}\} \tag{6}$$

where, $0 \leq i < M^l = M^{l+1}$, $0 \leq j < N^l = N^{l+1}$, $0 \leq k < P^l = P^{l+1}$.

- Convolutional layer: This layer extracts the features and learns them from the input data. Each neuron in the convolutional layer has a receptive field, and each receptive field is connected to the other neurons in the previous layer. Multiple convolution kernels are often used in a convolutional layer. Assume that D kernels are used, and each kernel is of spatial span $M \times N$. We show all these kernels by $\boldsymbol{h}$. The convolution process can be expressed as

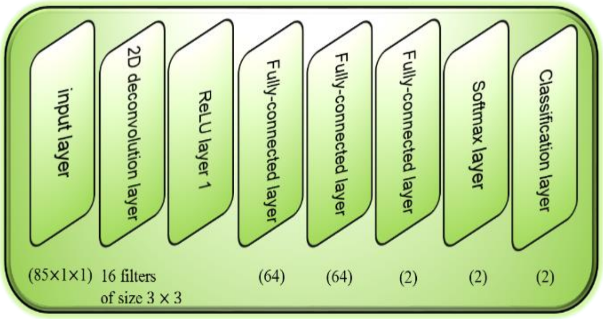
$$k = \sum_{i=0}^{M} \sum_{j=0}^{N} \sum_{k^l=0}^{P^l} h_{i,j,k^l,k} \times x^l_{i^{l+1}+i, j^{l+1}+j, k^l}, \tag{7}$$

where $x^l_{i^{l+1}+i, j^{l+1}+j, k^l}$ refers to an element of the x^l that indexed by the triplet $(i^{l+1}+i, j^{l+1}+j, k^l)$.

- Fully Connected (FC) Layer: The FC layer enhances the stability using several non-linear functions. Fully interconnected layers can follow these layers to extract and analyze the features, and perform the function of high-level information. Assume that the input of the l^{th} layer is of size $M^l \times N^l \times P^l$. Adopting convolution kernels with size $M^l \times N^l \times P^l$, using D kernels form an order 4 tensor of size $M^l \times N^l \times P^l \times D$, where the out-put is $\boldsymbol{y} \epsilon R^D$. To calculate the elements of the out-put , all elements of x^l should be used.
- Training: In order to gain the desired out-put, Deep Learning (DL) models typically utilize a learning algorithm, such as the back propagation algorithm, to adjust their parameters This algorithm controls the

network parameters by minimizing an objective function.

- Figure 2 illustrates the architecture of the suggested CNN-based classifier. The 2-D convolutional layer creates a layer with 16 filters of size 3 × 3. During the training process, the size of the padding can be calculated such that the out-put has the same size as the input. We consider two successive FC layers of size 64. The size of the final FC layer is 2, same as the number of out-put classes.



.

Figure 2 The proposed architecture of the CNN-based classifier

4- Simulation Result

In order to evaluate the performance of the proposed clustering and classification methods, some simulations are presented. All algorithms are implemented in MATLAB R2020a version, and tested on Intel(R) core TM i8-7000k CPU @ 5 GHz with 64 GB internal RAM.

4-1- The Insurance Company (TIC) Dataset

In this study, The Insurance Company (TIC) dataset is utilized for simulations [24]. TIC dataset comprises 86 features, product holders, and analytical information. The training dataset comprises more than 5000 reports about clients, including the information on having the caravan policy. The test dataset contains of 4000 clients. For the prediction task, the ultimate objective is to detect the clients that may purchase a caravan policy. The identified purchasers can be pulled out, and the other people receive the advertising emails. We organize the train and test data as the following:

- Train data: This dataset is used to train our prediction models, and comprises the information of 5822 clients. Each client record comprises 86 features, consisting of demographic data (features1-43), and purchasers of an insurance policy (features 44-86). All clients in the same cities have similar demographic features. Feature 86 includes the information about the holders of the caravan insurance policy, and their phone number. In other words, feature 86 can be utilized as the target variable.
- Test data: This dataset includes the information about 4000 clients. This dataset is adopted for prediction task and only the insurance company managers know that whether the clients have caravan insurance policy or not. It has the same form as train data, only the target variable is missing.
- Target values: Target values are used for the performance evaluation.

The suggested clustering and classification methods are tested on some informative features. The meaning of the utilized features and their values are listed in Table 1. The following illustrates the simulation results for data clustering and classification tasks.

Table 1: Margin specifications

Number	Name	Description	Domain
1	MOSTYPE	Customer subtype	1-41
2	MAANTUHI	Number of houses	1-10
5	MOSHFOOD	Customer main type	1-10
16	MOPLHOOG	High level education	-
68	APERSAUT	Number of car policies	-
82	APLEZIER	Num. of boat policies	-
86	CARAVAN	Target variable	0-1

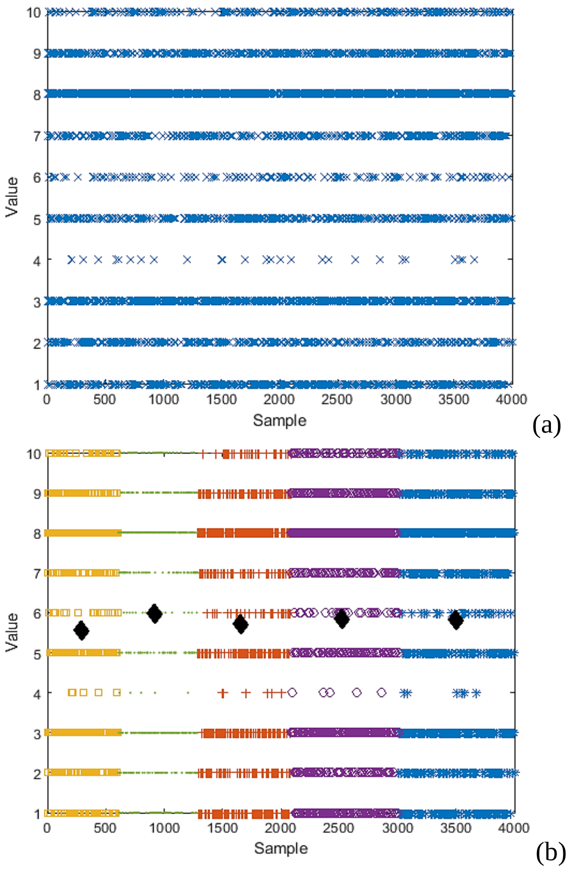
4-2- Clustering Results

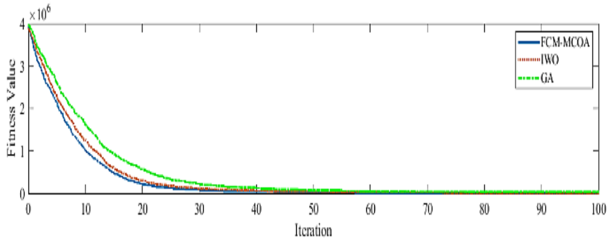
The utilized parameters for FCM-MCOA are listed in Table 2. The results for 5 clusters of different features of the TIC dataset are shown in Figures 3 to 6. The selected features are MOSHOOF, MOPLHOOG, APERSAUT, and APLEZIER. Clusters are marked with different colors in each shape.

Figure 3 (a) displays the initial distribution of cuckoos/people in the environment. Figure 3 (b) shows 5 best habitat for 5 selected clusters. The centers of the clusters are marked with diamond in the figures. For example, for the first cluster of the MOSHOOF feature, the best habitat is (293.4, 5.6). Figure 3 (C) provides a comparison between the convergence curve of the FCM-MCOA, GA, and IWO methods. The initial population, mutation rate, and selection rate of GA are set to 100, 0.2, and 0.5, respectively. The convergence curves of IWO [26] and GA [27] reach to their best point after about 40 iterations, while the FCM-MCOA reaches to its best point only after 30 iterations. From the cost minimization curves, it can be concluded that the FCM-MCOA offers a great convergence. These advantages come from this fact that the cuckoo search process solves the local optimum points problem and converges rapidly to its global solution.

Table 2: The parameters of the FCM-MCOA

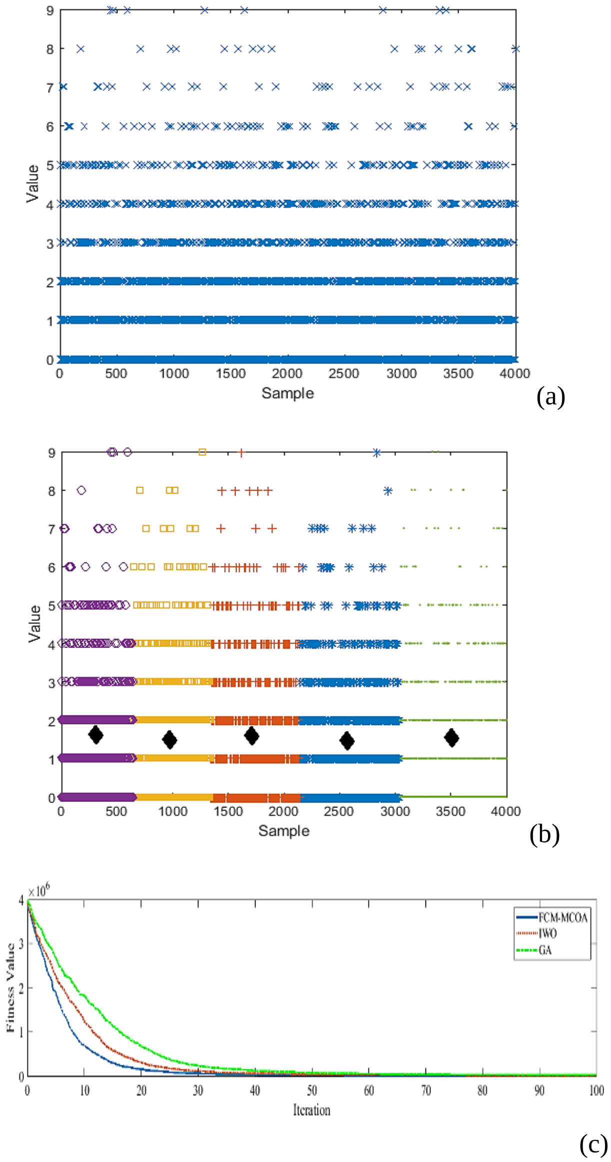
Parameter	Values
population Initial	10
Minimum number of eggs	5
Maximum number of eggs	10
Maximum number of iterations	100
Number of cluster centers reported by FCM	3
Lambda variable	2
N max	80
ELR	5
Population variance for process termination	10-15





(c)

Figure 3 a) MOSHOOF feature, b) Clustering results, c) The cost functions of the FCM-MCOA, IWO and GA after 100 iterations.



(c)

Figure 4 a) MOPLHOOG feature, b) Clustering results, c) The cost functions of the FCM-MCOA, IWO and GA after 100 iterations.

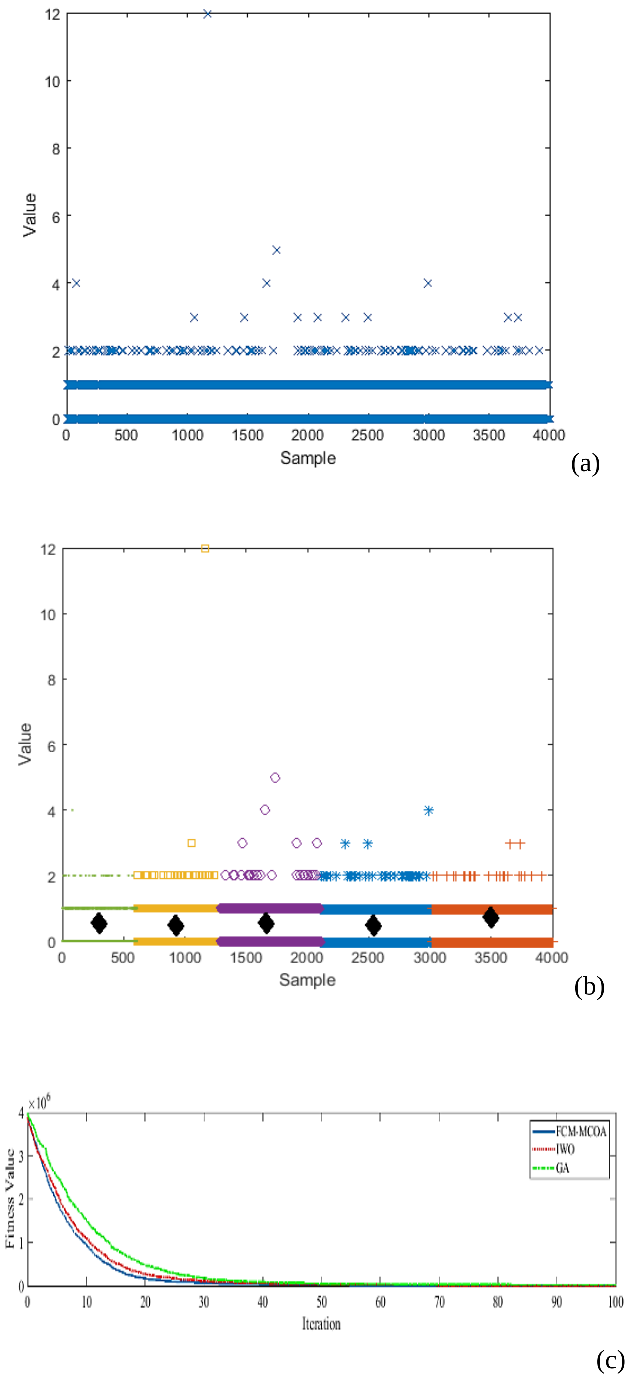


Figure 5 a) APERSAUT feature, b) Clustering results, c) The cost functions of the FCM-MCOA, IWO and GA after 100 iterations.

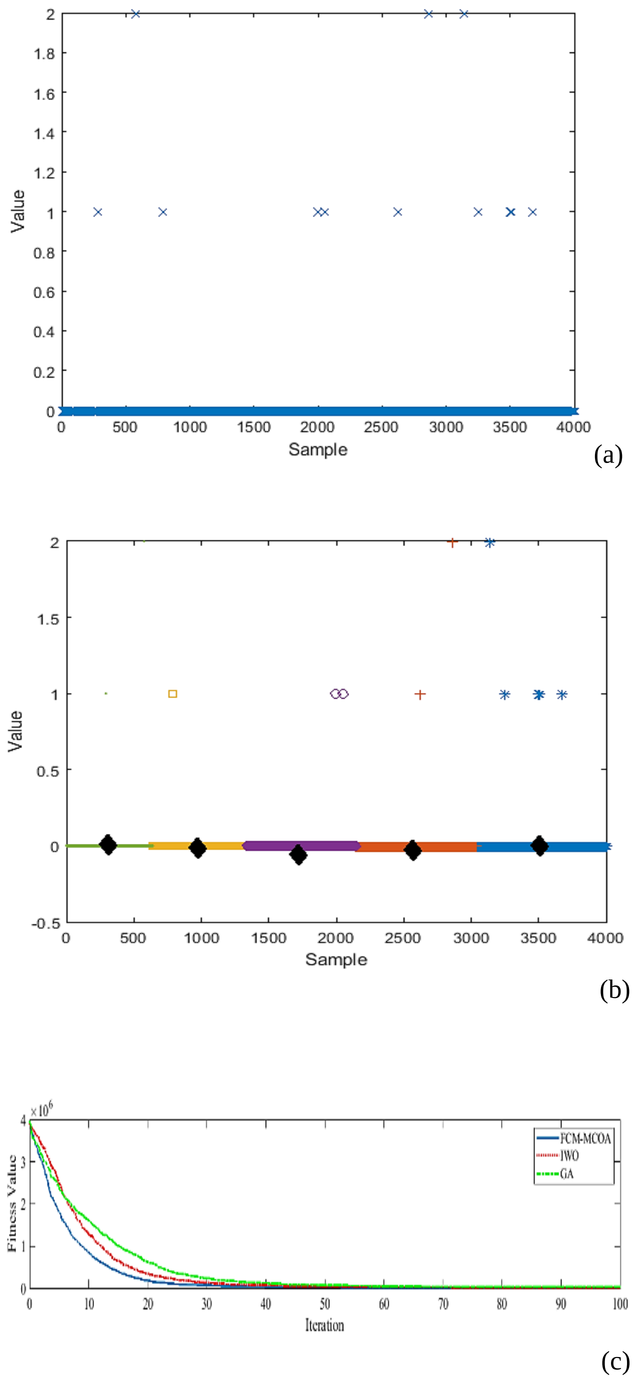


Figure 6 a) APLEZIER feature, b) Clustering results, c) The cost functions of the FCM-MCOA, IWO and GA after 100 iterations.

In Table 3, The values of the cluster centers obtained from the FCM-MCOA approach are reported. The final cost function values, for the FCM-MCOA, IWO [26], and GA [27] methods are also presented in Table 4 . In all scenarios, FCM-MCOA not only has faster convergence but also its final cost function values are considerably less than the above-mentioned conventional clustering methods.

Table 3: The cluster centers for some features.

Feature	Cluster 1	Cluster 2	Cluster 3	Cluster 4	Cluster 5
MOSHOOF	(293.4, 5.6)	(920.5, 6)	(1653.4, 5.7)	(2522.6, 5.8)	(3493.4, 5.8)
MOPLHOOG	(3493.4, 5.8)	(3493.4, 5.8)	(3493.4, 5.8)	(3493.4, 5.8)	(3493.4, 5.8)
APERSAUT	(295.6,0.55)	(928.4,0.49)	(1667.6,0.53)	(2538.4,0.48)	(3499.6,0.71)
APLEZIER	(311.4,0.0046)	(970.6,-1.018)	(1715.4,-.062)	(2568.5,-.027)	(3511.5,0.0019)

Table 4: The performance of the FCM-MCOA, IWO, and GA.

Feature	Final cost function value of FCM-MCOA	Final cost function value of IWO	Final cost function value of GA
MOSHOOF	2635	10917	31365
MOPLHOOG	1798	10712	31072
APERSAUT	2428	9722	21761
APLEZIER	1783	26923	11015

4-3- Classification Results

We implement the suggested CNN-based classifier to find out who is interested in purchasing caravan policy. Pre-defined functions and objects of the MATLAB Neural Network Toolbox are employed to create the classifiers and define training options, such as learning rate, number of convolutional filters, and the number of neurons in the FC layers. To analyze the performance of the CNN-based classifier, the specificity and sensitivity values are computed. Sensitivity and specificity are used as the criteria that show the accuracy of the two classes, i.e., zero and one class, respectively. Due to imbalance data, a classifier with large sensitivity and specificity values achieves an appropriate score. In order to explore the effect of increasing the number of out-put neurons in the FC layers on the performance of the CNN-based classifier, we changed the number of neurons, except for the last FC layer. The obtained results are reported in Figure 7. The results show that the maximum accuracy for this dataset is 98.1%. The behavior of CNN-based classifier is compared with some popualr classifiers, such as the Naive Bayes (NB) and SVM classifiers [25].

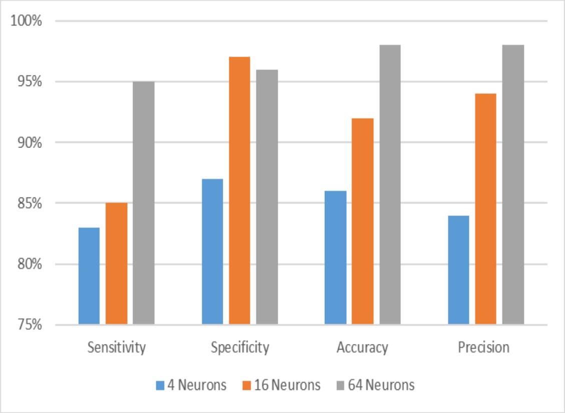


Figure 7 The effect of increasing the number of out-put neurons in the FC layers on the performance of CNN-based classifier.

Figure 8 provides a comparison between the accuracy rate of the CNN-based classifier and some conventional classifiers. The chart show that the proposed method leads to a better performance compared to the conventional methods in terms of the accuracy rate.

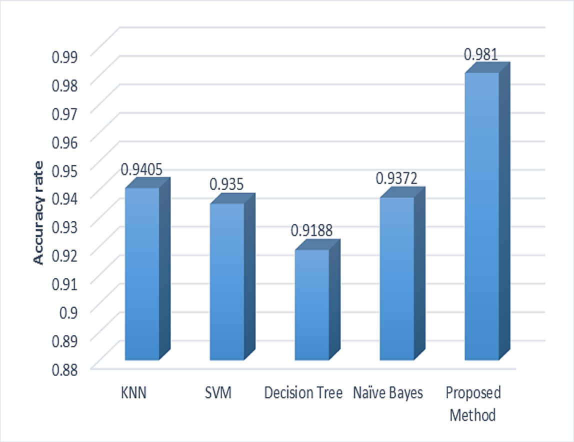


Figure 8 The accuracy rate of the CNN-based classifier compared to some conventional classifiers.

5- Conclusions

This study brings forward a hybrid FCM-MCOA approach in which the grouping task of cuckoos was performed by the FCM technique. A CNN-based classifier is also adopted for customer classification task, and the effect of the out-put neurons in the FC layers is evaluated. To investigate the effectiveness of our methods, we did some simulations, and the results showed that the FCM-MCOA scheme converges faster than conventional clustering methods. The results also showed that the CNN-based classifier is able to predict who wants to use the caravan insurance policy, providing an accuracy above 98%. It can be concluded that the CNN-based classifier outperforms the conventional classifiers, such as NB and SVM, in terms of the accuracy rate and convergence speed.

References

[1] A. Voulodimos, N. Doulamis, A. Doulami, and E. Protopapadakis, “Deep learning for computer vision: A brief review”, Computational intelligence and neuroscience, 2018.

[2] M. Jahangiri, and S. Ghavami, “Hybrid fuzzy c-means clustering algorithm and multilayer perceptron for increasing the estimate accuracy of the geochemical element concentration case study: eastern zone of porphyry copper deposit of Sonajil”, Iranian Journal of Geology, Vol. 48, No. 48, pp. 0, 2019.

[3] M. K. Pakhira, “A fast k-means algorithm using cluster shifting to produce compact and separate clusters”, Int J Eng, Vol. 28, No. 1, pp. 35-43, 2015.

[4] M. Setnes, and U. Kaymak, “Extended fuzzy c-means with volume prototypes and cluster merging”, In Proceedings of the 6th European Conference on Intelligent Techniques and Soft Computing (EUFIT’98), 1998, pp. 1360-1364.

[5] G. S. Budhi, R. Chiong, and Z. Wang, “Resampling imbalanced data to detect fake reviews using machine learning classifiers and textual-based features”, Multimedia Tools and Applications, Vol. 80, No. 9, pp. 13079-13097, 2021.

[6] S. Cateni, V. Colla, A.Vignali, and M.Vannucci, “Data Pre-processing for Efficient Design of Machine Learning-Based Models to be Applied in the Steel Sector”, In Impact and Opportunities of Artificial Intelligence Techniques in the Steel Industry: Ongoing Applications, Perspectives and Future Trends, pp. 13-27, 2021.

[7] Z. Abtahi, R. Sahraeian, and D. Rahmani, “A Stochastic Model for Prioritized Outpatient Scheduling in a Radiology Center”, International Journal of Engineering Transactions A: Basics, Vol. 33, No. 4, 2020.

[8] J. MacQueen, “Classification and analysis of multivariate observations”, In 5th Berkeley Symp. Math. Statist. Probability, pp. 281-297, 1967.

[9] T. Abukhalil, M. Patil, and T. Sobh, “A comprehensive survey on decentralized modular swarm robotic systems and deployment environments”, International Journal of Engineering (IJE), Vol. 7, No. 2, pp. 44, 2013.

[10] C. Li, L. Liu, X. Sun, J. Zhao, and J. Yin,” Image segmentation based on fuzzy clustering with cellular automata and features weighting”, EURASIP Journal on Image and Video Processing, Nom. 1, pp. 1-11, 2019.

[11] T. M. Silva filho, B. A. Pimentel, R. M. Souza, and A. L. Oliveira, “Hybrid methods for fuzzy clustering based on fuzzy c-means and improved particle swarm optimization”, Expert Systems with Applications, Vol. 42, No. 17-18, pp. 6315-6328, 2015.

[12] S. Das, A. Abraham, and A. Konar, “Automatic clustering using an improved differential evolution algorithm”, IEEE Transactions on systems, man, and cybernetics-part A: Systems and Humans, Vol. 38, No. 1, pp. 218-237, 2007.

[13] S. Paterlini and T. Krink, “Differential evolution and particle swarm optimization in partitional clustering”, Computational statistics & data analysis, Vol. 50, No. 5, pp. 1220-1247, 2006.

[14] T. Niknam, M. Nayeripour, and B. B. Firouzi, “Application of a new hybrid optimization algorithm on cluster analysis”, In Proceedings of world academy of science, engineering and technology, Vol. 36, pp. 599, 2008.

[15] K. Krishna, and M. N. Murty, “Genetic K-means algorithm”, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), Vol. 29, No. 3, pp. 433-439, 1999.

[16] H. Izakian, A. Abraham, and V. Snášel, “Fuzzy clustering using hybrid fuzzy c-means and fuzzy particle swarm optimization”, In 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), pp. 1690-1694, 2009.

[17] X. S. Yang, and S. Deb, “Cuckoo search: recent advances and applications”, Neural Computing and applications, Vol. 24, No. 1, pp. 169-174, 2014.

[18] X. S.Yang, and S. Deb, “Engineering optimisation by cuckoo search” , International Journal of Mathematical Modelling and Numerical Optimisation, Vol. 1, No. 4, pp. 330-343, 2010.

[19] R. Rajabioun, “Cuckoo optimization algorithm”, Applied soft computing, Vol. 11, No. 8, pp. 5508-5518, 2011.

[20] H. Kahramanli, “A modified cuckoo optimization algorithm for engineering optimization”, International Journal of Future Computer and Communication, Vol. 1, No. 2, pp. 199, 2012.

[21] M. Momeny, M. Agha Sarram, A.M. Latif, and R. Sheikhpour, “Improving the Architecture of Convolutional Neural Network for Classification of Images Corrupted by Impulse Noise”, Nashriyyah-i Muhandisi-i Barq va Muhandisi-i Kampyutar-i Iran, Vol. 76, No. 4, pp. 267, 2020.

[22] M. Rohanian, M. Salehi, A. Darzi, and V. Ranjbar, “Convolutional Neural Networks for Sentiment Analysis in Persian Social Media”, arXiv preprint arXiv:2002.06233, 2020.

[23] M. Mobini, G. Kaddoum, and M. Herceg, “Design of a SIMO deep learning-based chaos shift keying (DLCSK) communication system”, Sensors, Vol. 22, No. 1, pp. 333, 2022.

[24] D. Madurasinghe, and G. K. Venayagamoorthy, “LVQ neural network for online identification of power system network branch events”, In 2020 Clemson University Power Systems Conference, 2020, pp. 1-7.

[25] B. Zadrozny, and C. Elkan, “Transforming classifier scores into accurate multiclass probability estimates”, In Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining, 2002, pp. 694-699.

[26] A. R. Mehrabian, and C. Lucas, “A novel numerical optimization algorithm inspired from weed colonization”, Ecological informatics, Vol. 1, No. 4, pp. 355-366, 2006.

[27] K. J. Kim, and H. Ahn, “Using a clustering genetic algorithm to support customer segmentation for personalized recommender systems”, In International Conference on AI, Simulation, and Planning in High Autonomy Systems, 2004, pp. 409-415.

[28] C. Mouton, J. C. Myburgh, and M. H. Davel, “Stride and translation invariance in CNNs.”, In Southern African Conference for Artificial Intelligence Research, 2021, pp. 267-281.

Journal of Information Systems and Telecommunication
Vol.12, No.1, January-March 2024, 72-89

http://jist.acecr.org
ISSN 2322-1437 / EISSN:2345-2773

Persian Ezafe Recognition Using Neural Approaches

Habibollah Asghari[1*], Heshaam Faili[2]

[1.] ICT Research Institute, ACECR, Tehran, Iran,
[2.] Department of ECE, School of Engineering, University of Tehran, Tehran, Iran.

Received: 18 Apr 2023/ Revised: 24 Feb 2024/ Accepted: 13 Mar 2024

Abstract

Persian Ezafe Recognition aims to automatically identify the occurrences of Ezafe (short vowel /e/) which should be pronounced but usually is not orthographically represented. This task is similar to the task of diacritization and vowel restoration in Arabic. Ezafe recognition can be used in spelling disambiguation in Text to Speech Systems (TTS) and various other language processing tasks such as syntactic parsing and semantic role labeling.

In this paper, we propose two neural approaches for the automatic recognition of Ezafe markers in Persian texts. We have tackled the Ezafe recognition task by using a Neural Sequence Labeling method and a Neural Machine Translation (NMT) approach as well. Some syntactic features are proposed to be exploited in the neural models. We have used various combinations of lexical features such as word forms, Part of Speech Tags, and ending letter of the words to be applied to the models. These features were statistically derived using a large annotated Persian text corpus and were optimized by a forward selection method.

In order to evaluate the performance of our approaches, we examined nine baseline models including state-of-the-art approaches for recognition of Ezafe markers in Persian text. Our experiments on Persian Ezafe recognition based on neural approaches employing some optimized features into the models show that they can drastically improve the results of the baselines. They can also achieve better results than the Conditional Random Field method as the best-performing baseline. On the other hand, although the results of the NMT approach show a better performance compared to other baseline approaches, it cannot achieve better performance than the Neural Sequence Labeling method. The best achieved F1-measure based on neural sequence labeling is 96.29%

Keywords: Persian Ezafe Recognition; Vowel Restoration; Diacritization; Neural Sequence Labeling.

1- Introduction

In the Persian language, Ezafe construction is an unstressed short vowel /-e/ (or /-ye/ after vowels) which is used to link two words in some contexts. The function of Ezafe in Persian is to link elements of an NP or PP in (a) possessive constructions, (b) modification of the noun, (c) connecting some of the preposition types to the following NP elements, and (d) Proper Names linking first name to last name [1].

Although Ezafe is an important part of Persian phonology and morphology, it does not have a specific orthographical representation and it is not usually written. However, it should be pronounced as a short vowel /e/. Ezafe appears between two words indicating some relationships between the words to show the occurrence of a genitive case. Sometimes, its presence is explicitly marked by the diacritic “Kasre” in order to facilitate the correct pronunciation. Common uses of the Persian Ezafe are pronominal possession, possessive suffixes, adjective-nouns, and title-family names. We will discuss the various cases of Ezafe connecting two words in more detail in Section 3.

In most cases, despite the lack of orthographical representation, the location of Ezafe can be identified by human readers. However, very often, automatic recognition of Ezafe markers is a challenging problem. One of the challenges in Persian language processing is to determine how this important unwritten short vowel should be recognized.

The most important application of Ezafe identification is in the context of text to speech (TTS) systems as a text to phoneme tool [2]. Other applications include Ezafe recognition for identifying the dependency of a word in a

✉ **Habibollah Asghari**
habib.asghari@ictrc.ac.ir

Noun Phrase [3], or back-and-forth transliteration from the Perso-Arabic writing system and Latin-based scripts [4]. It has also been shown that adding information regarding Ezafe markers can greatly improve dependency parsing and shallow parsing as well [5].

Some tagging algorithms in computational linguistics can be applied to accomplish the task of Ezafe recognition. Various rule-based and statistical approaches for recognition of Ezafe markers have been investigated including HMM POS tagger [3], Maximum Entropy (MaxEnt) POS tagger, Conditional Random Fields (CRF) tagger [6], phrase-based Statistical Machine Translation [6], and Genetic Algorithms [7]. NLP techniques such as Probabilistic Context-Free Grammars (PCFGs) are also good tools for finding NPs and searching Ezafe construction inside the constituent [8].

In this research, we would like to investigate various approaches to automatically recognize the location of Ezafe construction in the Persian language. Our main focus is on neural approaches, including neural sequence labeling and neural machine translation (NMT) as well. In modeling the Ezafe recognition task as a translation problem, the input to the system is a Persian text without Ezafe markers, and the output of the algorithm is the same text marked with Ezafe tags. To evaluate the performance of our proposed methods, we conducted various baseline experiments and previous approaches and compared them with our approach. To the best of our knowledge, no work has been done so far to investigate neural approaches to the problem of Persian Ezafe recognition.

In a short view, the contributions of this paper are as follows:

- The use of a Neural sequence labeling approach that is state-of-the-art in Ezafe recognition.
- Using a Neural Machine Translation (NMT) approach.
- Incorporating versatile syntactic and lexical features of the Persian language in the Ezafe recognition task and embedding them in our models as a whole.
- The use of a large corpus for training the system, so the results are considerably reliable.

In addition, we investigated a Factored–based SMT model (FB-SMT) as an extension to the statistical machine translation model that was previously developed by [6] for recognizing Ezafe in Persian. To this end, we applied various features to enhance the functionality of SMT in Ezafe recognition.

The paper is organized as follows. In the next section, a clear definition of the problem is presented and the characteristics of the Persian language are introduced. In Section 3, we will give a precise definition of Ezafe and its role in the Persian language. Section 4 provides an overview of previous works in Ezafe recognition. Our approach will be described in Section 5, in which we will explain the proposed neural models and also the features that are incorporated into them. Experiments and results will be provided in Section 6 including experimental setup, evaluation measures, the baselines, and implementation of our proposed method. Discussion and analysis of the results will be explained in Section 7. Conclusion and recommendations for future works will be discussed in the final section.

2- An Overview of Persian Language

Persian is a rich morphology language that belongs to Arabic script-based languages. This category of languages includes Kurdish, Urdu, Arabic, Pashtu, and Persian [9]. They all have common scripting and somehow similar writing systems. This language family has some common properties and features such as the absence of capitalization, right-to-left direction, encoding issues in the computer environment, lack of clear word boundaries in multi-token words, and a high degree of ambiguity due to non-representation of short vowels in the writing system [9]. Note that Ezafe recognition and homograph disambiguation problems mostly deal with the last two mentioned features.

It should be noted that despite the mentioned commonalities, these languages do not belong to a single language family. Although Persian and Arabic have almost the same scripts and share some common characteristics, Persian belongs to the Indo-European language family, while Arabic is a Semitic language, belongs to the Afro-Asiatic family of languages, completely different in lexical and syntactic features [10]. For example, Urdu and Arabic have grammatical gender determiners, while Persian does not have a gender marker. There are also some word order differences, for example, while Arabic has predominantly SVO order, Persian and Urdu languages follow SOV order [11, 12]. Persian is the official language of three countries including Iran, Tajikistan, and Afghanistan.

3- Ezafe Definition

In the Persian language, the elements within a noun phrase are linked by an enclitic particle called Ezafe. This morpheme is usually an unwritten vowel, but it could also have an orthographic realization. In most cases, this relation can be translated as a genitive structure. An example of this construction is as follows:

- */ketâb -e mæn/*
- */book-EZ I/*
- *my book*

Note that in the ordinary Persian writing system, the short vowel /-e/ is not written, and this causes ambiguities in pronunciation. It should be mentioned that the Ezafe is not typically written when it follows a consonant or glide (i.e., 'ye'), but it is overtly written when it follows a vowel (i.e., /a/, /u/, /i/). It is also sometimes overtly written following the final (silent) 'he' in current writing, where the Ezafe can be written as a separate 'ye'. Ezafe is a property of the Arabic script used in Iran and Afghanistan, while it is written overtly in Tajiki Persian which employs the Cyrillic script.

While reading text, native speakers can generally vocalize each word based on their familiarity with the lexicon and the context of the text. However, it is hard to automatically recognize Ezafe in ordinary text because of considerable ambiguity. In recognizing Ezafe, the computer program should take into account morphological, syntactic, semantic, and discourse views [13].

There is also Ezafe construction in other languages; for example Ezafe construction in Urdu, which borrowed the construction from Persian [14].

Historically, the origin of enclitic Ezafe was in a demonstrative-relative morpheme in old Iran [15]. In Persian, it can be related to a demonstrative /hya/, which links the head noun to adjectival modifiers in possessor NP in Old Persian [16]. In the evolution of the Persian language, /hya/ changed to /–i/ in Middle Persian and progressively lost its demonstrative value to end up as a simple linker [16]. Contrary to Persian, Kurdish and Zazaki still have a so-called “Demonstrative Ezafe”, different from the affixal Ezafe, which functions as a demonstrative pronoun heading nominal phrases.

Ezafe can appear in noun phrases, adjective phrases, and some prepositional phrases linking head and modifiers. It should be stated that Ezafe is limited to specific POS tags such as N, ADJ, P, NUM, DET, ADV, and PRO respectively. So for example, it cannot appear on 'yek' in the above two examples, and it would rarely appear on a pronoun, which limits the permutations [17].

3-1- Ezafe iteration

Ezafe can be iterated within NPs, occurring as many times as there are modifiers [16]. In the following example, fourteen words are related to each other by iterated Ezafe markers:

- */Lozum-e tæqyir-e zæmân-e bærgozâri-ye dour-e moqqædæmâti-ye mosâbeqât-e futbâl-e jâm-e jæhâni-ye sâl-e 2010-e âfriqâ-ye jonubi/*
- /need-EZ change-EZ time-EZ hold-EZ round-EZ preliminary-EZ competition-EZ soccer-EZ cup-EZ world-EZ year-EZ 2010-EZ Africa-EZ south/
- /The need to change the time of holding the preliminary round of South Africa’s year 2010 soccer World Cup competition /

As can be shown in the above sentence, the chain of Ezafe markers can iterate within a phrase linking several elements together. As a result, there is no limitation in the number of words in a connected chain of Ezafe markers.

3-2- Ezafe Domain definition

We refer to all elements that are consecutively linked by the Ezafe marker as "Ezafe domain“. Ezafe domain is a specific phrase domain comprised of all the words that relate to each other using Ezafe. Determination of the Ezafe domain is equal to determining the Ezafe location between words [18, 3]. So the task of Ezafe identification is to correctly and accurately define the Ezafe domain boundaries.

Assume that /w1 w2 w3 ... wn/ is a Persian sentence. In this sentence, the sequence wi ... wi+j is an Ezafe domain if all of the words in the sequence have Ezafe as /w1 w2 … wi-e wi+1 -e wi+2 -e ... wi+j-1 -e …wn /

3-3- Problems with Ezafe Marking

There are some problems with automatically recognizing Ezafe markers in Persian text. It is because in recognizing Ezafe, we should consider morphological, syntactic, semantic, and discourse views [13]. The following examples show the importance of each of the mentioned views:

The example below needs morphological analysis to find the location of Ezafe marker (represented as -ye in the example):

- */hæme-ye osærâ-ye jæng âzâd ʃodænd/*
- */all-EZ captives-EZ war were freed/*
- *All of the captives of the war were freed*

As mentioned before, when the last letter of the word is a vowel, then the Ezafe marker changes to /-ye/ instead of /e/.

Sometimes we need syntactic analysis to locate the Ezafe marker in the sentence [56]. In this example, we need to analyze the status of the verb in the sentence to find the exact position of the Ezafe marker:

- */yek mærd-e dɒːneʃmænd râ didæm/* (I saw a scientist man)
- */yek mærd dɒːneʃmænd râ did/* (A man saw a scientist)

In the example above, both sentences are written in the exact same way, but should be pronounced differently. In the first sentence, a pronoun drop has occurred; this can be derived by analyzing the verb ‘didam’. In the second sentence, two meanings can be deducted based on the location of the Ezafe marker. If an Ezafe marker exists on ‘mærd’, then a pronoun drop for the subject ‘he’ has occurred, and ‘mærd’ is an object. On the other hand, if no Ezafe marker exists, then ‘mærd’ is a subject.

The example below needs semantic analysis. Notice that the pronoun has been dropped from the second sentence and so, this creates an ambiguity in finding the subject of the sentence which results in difficulty in correctly locating the Ezafe marker:

- */æhmæd kelid-e otâq râ âværd/*
- */Ahmad the key-EZ room brought/*
- */Ahmad brought the key to the room/*
- */kelid-e otâq râ âværd/*
- */the key-EZ room brought/*
- */he/she brought the key to the room/*

In the above example, 'ahmad', the subject of the sentence has been omitted. The object 'kelid' can be analyzed mistakenly as the subject if it's not linked to the next element and the pronoun drop is not caught.

In the following example, in order to find the location of Ezafe marker, we need a discourse analysis:

- */Dâneʃâmuzân xâneh-ye xod râ peidâ kærdænd/*
- */Students home-EZ their found/*
- */The students found their home/*

However, to find the syntactic structure of the sentence, a discourse analysis is required. First of all, we need to recognize the role of the first word. In other words, we should discover whether it is the object or the subject of the sentence. By analyzing just the current sentence, we cannot find the correct position of Ezafe tags, so we should know what happened in the previous sentences. This example also shows the ambiguity caused by the combination of the Ezafe not being written and the SOV order in the Persian clause, since the object directly follows the subject and the boundary between the two is difficult to identify. Note that the origin of the above-mentioned problems in Ezafe recognition is mainly because of pronoun drop in the Persian language.

3-4- Challenges of Ezafe marking in computational linguistics

There are some challenges that we encounter in Persian language processing [59]. One of the problems in Persian language processing is related to long-distance dependencies which increase the difficulty of correctly identifying the Ezafe marker. This phenomenon complicates Ezafe recognition for humans as well; one would need to read the entire sentence before recognizing the place of Ezafe [57]. The example below shows this case (Note that /e/ in the examples stands for Ezafe marker):

- */ʔu xâne-ye særshâr æz æfsus væ ænduh-e xod râ tærk kærd/*
- */he home-EZ full of regret and sorrow-EZ him left/*
- *He left his house which was full of sorrow and regret.*

In the above example, note that the genitive case /xâne/ and /xod / are not next to each other, and so recognizing the presence of Ezafe in /xâne/ can be achieved by determining this long-distance dependency.

Another problem is to determine boundaries in multi-token words. In some cases, when the parts of a multi-token word are separated by a space delimiter, it is hard to recognize the Ezafe marker in the sentence.

- */ʔânhâ bâ naxost vazir molâghât kardand/*
- */They with prime minister met/*
- *They met prime minister.*

In the above example, /naxost vazir/ is a multi-token word, and so /naxost/ do not need Ezafe marker.

The third challenge arises by pronoun drop due to the morphology of the Persian language. Persian is a null-subject or pro-drop language. So personal pronouns such as 'I', 'he', and 'she' are optional and can be omitted from the sentence. As can be seen in the following example, the subject can be removed from the sentence, so make it difficult to correctly recognize Ezafe in the sentence:

- */mæn âb-e khonak râ nu:ʃidam/=/âb-e khonak râ nu:ʃidam /*
- */I water-EZ cold drank/ = /water-EZ cold drank /*
- *I drank the cold water*

In the above example, a pronoun resolution is required in order to correctly find the location of the Ezafe marker.

Another challenging issue is the homograph ambiguity as a result of dropping short vowels in writing. This problem is also the origin of the main challenges we encounter in Ezafe recognition. So, Ezafe recognition can be expressed as a kind of homograph disambiguation task. The difference here is that homograph disambiguation generally deals with all of the diacritics that can be attached to the letters inside a word, but in Ezafe recognition, we are only concerned with the ending letter of the word.

Finally, another main problem in Ezafe recognition is to detect word/phase boundaries especially when we encounter multi-token words. In Persian language, affixes and words having multi tokens can be written in three kinds of writing formats; completely separated by a space delimiter, separated by a Zero Width Non-Joiner (ZWNJ) letter, or can be attached to its main word. In the first case, the computer determines them as two separate words, while in the latter two cases, the borders of words can be correctly recognized. Most of the time, Persian writers do not obey the Persian Academy rules and the writer is free to choose one of them at will. The problem of determining word boundaries makes it difficult to recognize Ezafe markers.

4- Related Work

In this section, we will explain the previous researches in recognizing Ezafe in the Persian language. The problem of determining short vowels in other languages such as Arabic and French is also discussed.

4-1- Ezafe Recognition in Persian

There have been some efforts to recognize Ezafe in the Persian language. As a first attempt to recognize Ezafe in Persian text, [18] used POS tags and also semantic labels (such as place, time, ordinal numbers ...) to obtain a statistical view of Ezafe markers. The most frequent combinations were extracted based on a 10 million-word corpus. In research accomplished by [8], the researchers focused on noun phrases. In NPs, Ezafe can relate between the head and its modifiers. Hence, by parsing the sentences and finding phrase borders, the location of the Ezafe marker in the sentence could be found. The sentences were analyzed using a Probabilistic Context Free Grammar (PCFG) to derive phrase borders. Then based on the extracted parse tree, the head and modifiers in each phrase could be determined. In the last phase, a rule-based approach was also applied to increase the accuracy of Ezafe marker labeling. There were also other attempts to effectively recognize Ezafe marker in Persian text, such as [19] based on fuzzy sets. Also, researchers in [3] developed a system based on the Hidden Markov Model to correctly identify Ezafe markers. In [20] they approached the problem using syntactic analysis. There are also some implementations using neural networks [21]. Another research for recognizing the position of Ezafe construction in Persian text has used a combined framework based on rule-based models and genetic algorithms [7]. Genetic algorithms provide a search strategy to learn general Ezafe patterns, while the rule-based model handles special cases and exceptions to general patterns. The results of this study show that the proposed algorithm outperformed classical HMM-based methods. As a last related work, researchers in [6] used three approaches named Maximum Entropy (MaxEnt) POS tagger, a Conditional Random Fields (CRF) tagger, and a phrase-based Statistical Machine Translation (PB-SMT) method. The latter approach is closely related to our approach. The difference is that we have used the FB-SMT model instead of a simple phrase-based SMT, incorporating many well-defined selected features into the model.

As a result, the Ezafe tagging problem can be classified into three categories, each of them can use algorithms at the character and/or word level. In the following subsections, we will explain them in more detail.

Ezafe recognition using Rule-based tagging

The most straightforward way to tag words with an Ezafe marker is to use some lexical or grammatical rules so as to find potential words having an Ezafe marker. This method needs human intervention by handcrafted rules. As an example, we can define some linguistic rules such as follows:

- IF the current word is NOUN and the next word is ADJ THEN (Tag the current word with an Ezafe marker)
- IF the current word has the ending letter /ی / and the next word is ADJ THEN (Tag the current word with Ezafe marker)

Rule extraction can also be done by examining the confusion matrix. It can greatly help in evaluating false positive (FP) and false negative (FN) cases, and then derive some rules for correcting the misclassified cases of statistical methods [6].

Some of the above-mentioned related works have used a rule-based approach as a post-processing step to increase the accuracy of Ezafe recognition. In [8] the researchers used a rule-based method in the last phase of their work. Moreover, [6] applied high precision and low recall rule sets to decrease FP and FN and increase the total accuracy. They have used five Persian-specific features. Another research in [7] used a combined framework based on rule-based models and genetic algorithms.

Ezafe recognition as a sequence tagging problem

Part of Speech (POS) tagging is an effective way for automatically assigning grammatical tags to words of the sentence. There are powerful statistical POS tagger algorithms and methods. In the case of Ezafe recognition, instead of simple POS tags, an extended POS tag set is used comprised of Part of Speech tags plus Ezafe marker, which is called POSE tag that can be constructed by adding Ezafe markers to original first-level POS tags.

Among previous works, researchers in [3] has used a sequence tagging approach based on Hidden Markov Model for recognizing POSE tags. Another attempt based on sequence tagging was also done by [6] using the Conditional Random Fields (CRF) POSE tagger and also the Maximum Entropy (MaxEnt) POSE tagger. Moreover, there are some related works in word segmentation based on sequence tagging that can also be used for the task of Ezafe recognition such as the works accomplished by [22], [23], and [24] as well.

Ezafe recognition as a translation problem

Ezafe recognition problem can be considered as a translation problem; the original training text without Ezafe marker can be used as a text in the source language and the tagged text can be used as a text in the target language. In research accomplished by [6], they have used a phrase-based SMT model. Our work in this paper is also based on machine translation with a different approach.

Ezafe recognition using transformers

In a research by Doostmohammadi et al., they have exploited Transformer-based, BERT, and XLMRoBERTa methods, and achieved the best results, with respect to the previous works [58]. In another research accomplished by Ansari et al., they tackled the problem of Ezafe recognition using ParsBert transformer. They have also compared their proposed method with the XLMRoBERTa and BERT multilingual models [60].

4-2- Diacritization and Vowel Restoration in Arabic

The recognition of Ezafe could be compared with prediction tasks in vowelization of Arabic. Since Persian has borrowed many words from Arabic, so Ezafe recognition in some ways is similar to diacritization in Arabic. As mentioned before, there are some similarities in Arabic-script-based languages. For example, in the Arabic language there is an ambiguity resulting from the absence of short vowel representations in contemporary Arabic texts [25]. Arabic readers infer the appropriate diacritics based on linguistic knowledge and the context. However, in the case of text-to-speech or automatic translation systems, Arabic letters need to be diacritized. Otherwise, the system will not be able to know which word to select. Like Persian, vowel restoration of Arabic text is also a homograph disambiguation process. The restoration of diacritics in written Arabic is an important processing step for several natural language processing applications [26]. In Arabic, there are eight diacritics as shown in Table 1. A diacritic may be placed above or below a letter hinting how the letter should be pronounced. In other words, they are written above or below the consonants they follow. The first three diacritics represent the Arabic short vowels. In our case in Persian Ezafe recognition, the third one in the table is important.

Table 1: Diacritics in Arabic Language

No.	Diacritic Shape	Diacritic Name	Location of diacritic	Pronunciation
1.	◌َ	*Fathah*	کَ	/Ka/
2.	◌ُ	*Dhammah*	کُ	/Ku/
3.	◌ِ	*Kasrah*	کِ	/Ke/
4.	◌ً	*Tanweenfathah*	کً	/Kan/
5.	◌ٌ	*Tanweendhammah*	کٌ	/Kun/
6.	◌ٍ	*Tanweenkasrah*	کٍ	/Kin/
7.	◌ّ	*Shaddah (Double consonant marker)*	کّ	/KK/
8.	o	*Sukoon*	کo	/K/

As an example of Arabic diacritization, [27] proposes an approach based on HMM to solve the problem of automatic generation of diacritical marks of the Arabic text. The system should be trained based on a specific topic, e. g. sports, weather, local news, international news, business, economics, religion, etc. For testing purposes, they have used a fully diacritized transcript of the Holy Quran. The recognition rate was about 95.9%.

There is also another research done based on the effect of Arabic language diacritization on Statistical Machine Translation. It is shown that this method outperforms the previous methods [28]. In another research for diacritization of Arabic text using morphological tagging, they used lexical resources [29]. They have stated that Out Of Vocabulary (OOV) words such as foreign words and names can affect the system. Diacritization using deep neural approaches has also been investigated in Arabic (Belinkov and Glass 2015, Abandah, et al., 2015) [30, 31]. They have used recurrent neural networks for the task of vowel restoration. Another research introduces an approach for Arabic diacritization utilizing Bidirectional Encoder representations from Transformers (BERT) models [61]. The performance of the model was assessed using various error metrics, including Diacritic Error Rate (DER) and Word Error Rate (WER).

A literature review of the previous works in automatic diacritics restoration in Arabic has been accomplished by Lapointe, et al, [62].

4-3- Other Languages

There are also some closely related problems in other languages. One of them would be the liaison in French grammar. Liaison is a grammatical circumstance in which a usually silent consonant at the end of a word is pronounced at the beginning of the word that follows it. Part of the reason that French pronunciation and aural comprehension are somehow difficult is due to liaisons [32].

In the Caspian languages Gilaki and Mazandarani, and in neighboring languages like Taleshi, nominals are near mirror inverses of Persian; a wide range of noun complements occur pre-nominally, and link to N via “reverse Ezafe” particle that can be shown by -REZ [33]. The following example shows this phenomenon:

- */surx-e Gul/*
- */red-REZ flower/*
- */red flower/*

Moreover, by comparing Chinese to this family of Iranian languages, which show rich variation in nominal structure, it can be shown that Chinese /de/ has the essential properties of a reverse Ezafe particle, as exemplified by the Caspian languages Gilaki and Mazandarani [33].

5- Our Approach

In this paper we deal with neural models to tackle the problem of Ezafe recognition. In the first approach, two models of neural sequence taggers are employed for the task of Ezafe recognition. In the second approach, a neural machine translation approach is used. Neural machine translation has shown its performance in machine translation problems. We compare the performance of our two approaches with that of the baselines.

Recognition of Ezafe construction heavily depends on the surrounding context. By incorporating good features, they can effectively present the context into the model, and so enhance the recognition rate of Ezafe markers in the text.

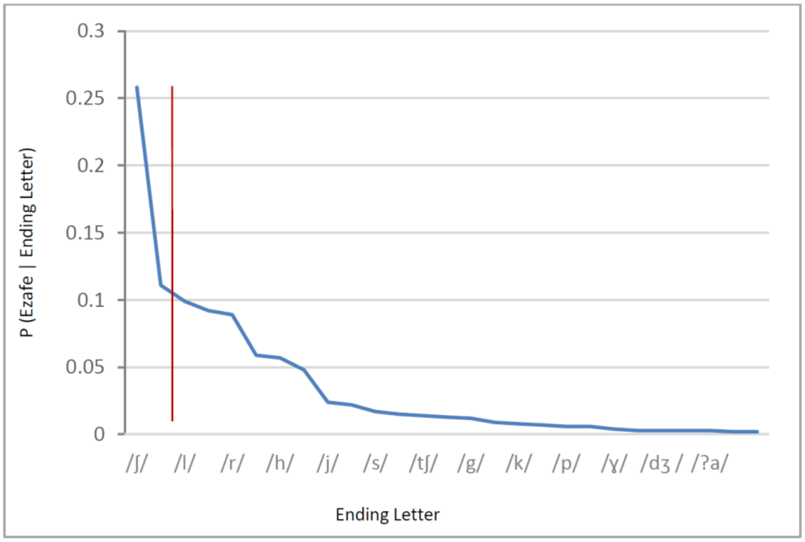


Fig. 1: Conditional probability of Ezafe appearance vs. ending letters

One of the advantages of our approach is to construct an integrated model that includes all the features as a whole, whereas, in the previous research, a two-stage task comprised of a statistical model followed by a rule-based post-processing step is employed. In our approach, all of the features (including the Persian-specific features) are included into one integrated model. As a result, the optimization of the features is more flexible in our model.

In the following subsections, we first describe the syntax-aware features that we have used in our model. Then, in the two subsequent sections, we present the two above-mentioned neural approaches.

5-1- Features for Ezafe Recognition

In this paper, we have applied some well-behaved syntax-aware features and examined that if they can improve the performance of Persian Ezafe recognition. Since Persian Ezafe Recognition is a Syntax-oriented task, so we selected the best combination of lexical and grammatical Persian linguistic features for incorporating them into our model.

In the training phase, some features are used into the system along with a large amount of annotated text data. It should be noted that the selection of the features is a very important task and should be done precisely. The features that we have selected for the recognition of Ezafe markers are as follows:

- POS tag of the current word: The existence of Ezafe tags greatly depends on the POS tag of the current word.
- POS tag of the next word: The POS tag of the next word is also of great importance because the Ezafe marker occurs between the current and the next word.
- P(Ezafe|word) with different values for each word: This feature is a maximum likelihood estimate by calculating all the words having Ezafe marker in the training corpus.
- P(Ezafe|POS) with different values for each POS tag: This feature is a maximum likelihood estimate by calculating all POS tags having Ezafe marker in the training corpus.
- POS tag of the previous adjacent word: Our experiments have shown that Part of Speech tags of the previous word can be used as an effective feature in Ezafe recognition.
- POS tags of the two next adjacent words: The experiments have shown that Part of Speech tags of

the second next word can be used as a good feature in Ezafe recognition.

The effect of the ending letter of the target word was also examined as a feature in the model. In our experiments, it was shown that the ending letter of the target word is of great importance in Ezafe recognition. So we tried various features that can be constructed based on ending letters. The Persian language has 32 letters, and they may all appear as the ending letters of a word. Moreover, some diacritics could appear at the end of some words and should be taken into account as ending characters. As a result, 46 ending characters (comprised of 32 ending letters plus 14 types of diacritics) exist in our corpus. To test the effectiveness of the ending letters feature, we calculated *P(Ezafe|EndingLetter)*, and then selected the most important ones as shown in Fig. 1. The graph shows the likelihood of the Ezafe marker with respect to the ending letters. As shown in the figure, some letters make a high probability on *P(Ezafe|EndingLetter)*.

By considering the breakpoint in the curve, the nine letters with the highest conditional probability were selected, which are */ʃ/, /m/, /l/, /d/, /r/, /t/, /h/, /n/, /j/* respectively. We call these highly important letters as golden ending letters.

As a result, three sets of features were constructed based on ending letters as follows:

- Ending letter of the current word (one hot representation of length 32): By assigning 32 different features for each word, this feature can take binary values for each letter.
- Ending letter of the current word (showed by one byte of binary coded format): By assigning one feature for each word, this feature can take 32 different values for each letter.
- Golden Ending letters: The high probable ending letters of the words that can take Ezafe markers can also be used as a feature. In this paper we have called them as golden letters and they are used as binary features (9 factors with binary values).

In order to find the best feature set, various combinations of these features were examined. Table 2 demonstrates the combination of features that have been investigated to be applied to our models. An approach would be to use Forward Selection procedure to obtain the best variables and then incorporate them into the models. In selecting the best set of features, we applied the Forward Selection method; taking into account what variables are eligible to be added to the set of features. This method is often used to provide an initial screening of the candidate variables when a large group of variables exists. As a result, the Ending letter (non-binary feature) and Golden Ending letters (binary features) were removed from the final feature set.

As an example, feature set 3 comprised of four features including likelihood of the word to take the Ezafe marker, the POS tag of the target word, the likelihood of the POS to take the Ezafe marker, and the ending letter (binary factors) were used as factors into the system.

5-2- Ezafe Marking by Neural Sequence Labeling

With the advances in deep learning, neural sequence labeling models have achieved state-of-the-art for many tasks [34, 23, and 35]. Features are extracted automatically through network structures including long short-term memory (LSTM) [36], and convolution neural network (CNN) [37] with distributed word representations. Similar to discrete models, a CRF layer is used in many state-of-the-art neural sequence labeling models for capturing label dependencies ([38, 35]).

In this research, we investigate two neural sequence taggers for our Ezafe marking problem. The first sequence tagger shares the same encoder as the encoder in our NMT approach but does not need a decoder since each input is synced with an output.

The architecture of the second neural sequence labeling includes an encoder and a CRF layer at the end for considering the dependencies between tags. The encoder itself is comprised of two layers of bi-directional LSTM cells. For the two above-mentioned models, the input of this encoder at each time is a concatenation of a word and its features.

To investigate the performance of our neural sequence labeling approaches, we compared it with a Conditional Random Field (CRF) model proposed by [6].

5-3- Ezafe Marking Using NMT approach

Our second approach is to employ the Neural Machine Translation model to tackle the problem of Ezafe recognition. Neural machine translation is an emerging approach to machine translation that has been proposed by [39], [40] and [41]. Unlike the traditional phrase-based translation model such as [42] which consists of many small sub-components that are tuned separately, NMT attempts to build and train a single, large neural network that reads a sentence and results in a correct translation at the output. Most of the proposed neural machine translation models belong to a family of encoder–decoders [40]; [41] with an encoder and a decoder for each language, or employ a language-specific encoder applied to each sentence whose outputs are then compared [43]. An encoder neural network reads and encodes a source sentence into a fixed-length vector. In the next step, a decoder outputs a translation from the encoded vector. The whole encoder-decoder system which consists of the encoder and the decoder for a language pair is jointly trained to maximize the probability of a correct translation given a source sentence. The most common approach for encoder and decoder is to use an RNN, but it should be noted that other architectures such as a hybrid of an RNN and a de-convolutional neural network can be used. [44].

Table 2 – Selection of features

Feature set	Feature set 1	Feature set 2	Feature set 3	Feature set 4	Feature set 5	Feature set 6
# of features	2	3	49	50	51	52
Word conditional probability P(Ezafe|word)	■	■	■	■	■	■
POS	■	■	■	■	■	■
POS conditional probability P(Ezafe|POS)		■	■	■	■	■
Ending letter (binary factors)			■	■	■	■
Next POS				■	■	■
Previous POS					■	■
Second next POS						■

Our NMT model is comprised of an encoder and a decoder, each one contains two layers of LSTM cells. Moreover, the attention model used in the decoder is global attention. Furthermore, we used a random vector as the embedding of the Ezafe marker and constructed the word embedding of the word+Ezafe marker by summation of the embedding of the word and the embedding of the Ezafe marker.

To investigate the performance of our approach, we run two baselines to compare them with neural machine translation. The first baseline is the research done by [6], which proposed a phrase-based SMT approach. For the second baseline, we developed a Factored-based SMT (FB-SMT) approach.

6- Experiments and Results

In order to evaluate the method, as the first step an experimental setup should be provided including preparation of training and test corpus. Then we should select effective evaluation measures to accurately evaluate the results. In the next step, we examine six baseline experiments. In baseline experiments 7 and 8 we investigate two competitor approaches that were investigated by [6]. Our approach will be described in experiments 1 and 2 in which the factored-based and neural machine translation models will be investigated. In the next subsections, we will describe the experimental setup and the experiments in detail.

A. Experimental Setup

For investigating the performance of our algorithms, an evaluation framework is required. The framework is comprised of an evaluation corpus along with evaluation measures. In the following subsections, we will thoroughly describe these elements.

Evaluation Corpus:

In this research, we have used the Bijankhan corpus that is gathered from daily news and common texts ([45, 46]. This corpus contains about 10 million tagged words and covers 4300 different subjects. The words in the corpus have been marked by a tag set containing 550 tags based on a hierarchical order, with more fine-grained POS tags like 'noun-plural-subj'. About 23% of words in the corpus have Ezafe marker tags [1]. The corpus is freely available on the Web for research purposes[1].

Fig. 2 shows the number of Ezafe markers in a sentence (normalized by total words), versus sentence length in Bijankhan corpus. This plot shows that for short sentences, the number of Ezafe markers in a sentence increases with sentence length, whereas in long sentences the average number of Ezafe markers approximately remains constant and does not increase.

In order to unify the character encoding for the next steps, a preprocessing step was applied to the corpus [47]. Furthermore, since determining the border of sentences in our research is of great importance, so the punctuation marks with different character encodings in the corpus were also mapped into standard UTF-8 encodings.

Evaluation Measures:

Ezafe recognition is indeed a binary classification problem. Precision and Recall are the ordinary measures in this type of classification, which are the measures of exactness and completeness respectively. As a combined measure that assesses precision/recall tradeoff, we have used F-measure (harmonic mean), a parameterized E-measure that equally weights precision and recall.

[1] https://dbrg.ut.ac.ir/span-design/

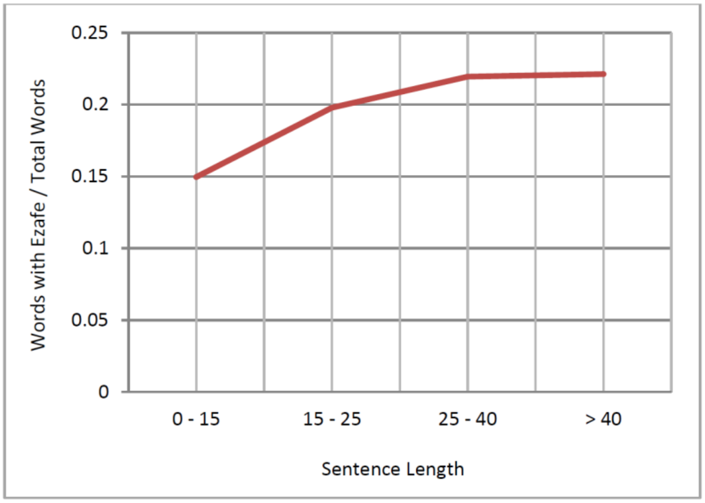


Fig. 2: Percentage of words having Ezafe marker versus Sentence length in the corpus

In our experiments, we have also calculated F0.5 to apply more weight to Precision. This measure can show us how we can deal with the Ezafe tagging problem if we need to emphasize more on Precision rather than Recall. The reason behind this selection is that, in the task of Ezafe recognition, precision is of higher importance with respect to recall. This general aim at high precision has been verified by [48] for evaluating a grammar checker system, and also was in line with Bernth's observations on end-user valuations, in which satisfaction was specified as high precision, i.e. few false recalls, even at a remarkable loss of recall [49]. In Bernth's experiment, even though users expect a proofing tool to find as many errors as possible, they prefer easing up on this expectation if the proportion of correct error flagging is relatively high.

Another measure that can be used in this binary classification problem is the Mathews Correlation Coefficient (MCC). This measure indicates the quality of the classifier for binary class problems especially when two classes are of very different sizes and so there is a class imbalance problem [50]:

$$Mcc = \frac{(TP * TN) - (FP * FN)}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \qquad (1)$$

where TP, TN, FP, and FN are the true positive rate, true negative rate, false positive rate, and false negative rate, respectively. The Matthews correlation coefficient is often used as a measure of the quality of 2-class binary classifications. It is generally regarded as a balanced measure that can be used even if the classes are of very different sizes. The MCC is in essence a correlation coefficient between the observed and predicted binary classifications. MCC ranges from -1 to 1, where -1 corresponds to inverse classification, zero corresponds to average classification performance and +1 represents perfect classification.

We have also considered two other measures that can be useful in the calculation of accuracy:

$$ezafe\_presence\_accuracy = \frac{TP}{TP + FN} \qquad (2)$$

$$ezafe\_absence\_accuracy = \frac{TN}{TN + FP} \qquad (3)$$

Note that Eq. 2 is the same as the Recall equation. The total average can be calculated using a weighted average of the two above-mentioned equations. In calculating the total weighted average, the weighting factor for Eq. 2 is the percentage of the words with the Ezafe marker in the test corpus which is 18%, while the weighting factor for Eq. 3 is the percentage of the words without the Ezafe marker in test corpus which is 82%. As a result, the weighted accuracy is presented as the final score. This is the calculation that has previously been done by [6], and so the results can be compared with each other.

B. Baseline Experiments

There should always be a simple baseline besides examinations to assess the efficiency of new approaches. This could be random assignment of Ezafe to words that can carry Ezafe markers, assignments based on the frequency of words having Ezafe markers in the training set, etc. So, at the first step in the experiments, we investigated nine types of taggers as baselines. The first six baseline experiments are classic taggers that are based on conditional probabilities of words and/or POS tags.

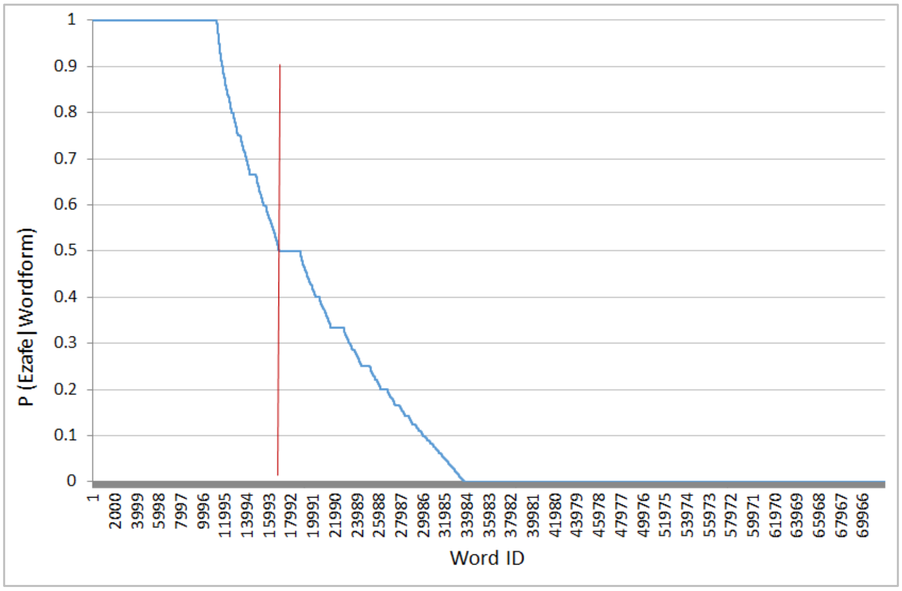


Fig. 3: Conditional probability of Ezafe appearance vs. word ID

Experiments 7 and 8 are Conditional Random Fields (CRF) and monotone SMT approaches that were previously examined by [6]. Moreover, another baseline experiment (experiment 9) was performed based on the Factored-based Machine Translation model. Our approaches are presented in experiments 11 and 12 that incorporate neural sequence tagger and neural machine translation, respectively.

In the following subsections, these approaches will be explained in more detail.

Baseline1: Using Word form

The lexical unit of a word has a great effect on the recognition of Ezafe tags. So, in the first experiment for deriving a baseline, we calculated the conditional probability *P(Ezafe|wordform)*. We counted the number of times each word with the Ezafe marker appeared in the training set. Then, all words were sorted based on their probability of taking the Ezafe marker. In this stage, we need to set a threshold in such a way that all of the words below that point can take the Ezafe marker.

The reason for selecting a threshold of 15% in the graph is as follows: The formula for the Naïve Bayes classifier can be described as:

$$R_1 = \{f(x|w_1)P(w_1)\ > f(x|w_2)P(w_2)\} \rightarrow\ R_1 = \{f(x|w_1)\ > f(x|w_2)\} \qquad (4)$$

$$R_2 = \{f(x|w_2)P(w_2)\ > f(x|w_1)P(w_1)\} \rightarrow\ R_2 = \{f(x|w_2)\ > f(x|w_1)\} \qquad (5)$$

In the above equation, w1 and w2 are class one and class two respectively, and x is the observation which to be classified in one of these two classes. R1 and R2 are the domain space for the mentioned classes.

In our Ezafe recognition problem, there are two classes of words with the Ezafe marker and words without the Ezafe marker. Based on these two classes, we can calculate the conditional probability of a word form with Ezafe marker:

$$P(Ezafe|word\ form) \propto P(word\ form|Ezafe)P(Ezafe) = \frac{\#word\ forms\ with\ Ezafe}{\#\ word\ with\ Ezafe} \times \frac{\#\ word\ with\ Ezafe}{\#\ Total\ words\ in\ corpus} = \frac{\#word\ forms\ with\ Ezafe}{\#\ \ Total\ words\ in\ corpus} \qquad (6)$$

Since we deal with a binary classification problem, we have just two classes in which their sum of probability should become 1.

$$P(Ezafe|word\ form) + P(\sim Ezafe|wordform) = 1 \qquad (7)$$

So, for assigning a word form to the Ezafe marker class, it is just enough that:

$$P(Ezafe|word\ form) = \frac{\#\ word\ forms\ with\ Ezafe}{\#\ \ total\ words\ in\ corpus} > 0.5 \qquad (8)$$

By investigating a line of probability 0.5 as shown in Fig. 3, we can see that selecting a threshold of 15% of word forms is a good selection for threshold. So, we selected the top 15% of the most probable words in training data that occur with Ezafe marker which *P(Ezafe|wordform)≥Threshold=0.5*, and then used them to mark the same words in the test corpus. Words encountered in the test set that appear in the top 15% list of the training set were tagged with Ezafe marker. The results are shown in Table 3 named as the “ Baseline with word form” approach, which shows a precision of 90.7% and an F-measure value equals to 46.17.

Baseline 2: Incorporating Wordform+POS tags

Part of Speech tags can demonstrate more detailed features of a word, so they can be used to better recognize the presence of Ezafe tags in Persian Text. It has been proven that POS tags are good features for recognizing Ezafe markers. It is because the presence of Ezafe in a word greatly depends on the grammatical features of the word. So, part of speech information can help us to improve Ezafe recognition algorithms.

In this experiment, we calculated the conditional probability *P(Ezafe|wordform, POS tag)*. The results of this approach have been indicated in Table 3 as the ‘word form + POS tags’ approach, and show a better performance with respect to the previous experiment. The result shows a precision of 93.43% and the F-measure value equals to 54.48.

Baseline 3: Using POS tags with FPS

In the third experiment, the role of POS tags in predicting the Ezafe marker was examined. We calculated the conditional probability *P(Ezafe|POS tag)*. Then we tagged all the words in the test corpus based on a Fitness Proportionate Selection (FPS). In FPS (also known as roulette wheel selection), individuals are given a probability of being selected that is directly proportionate to their fitness [51].

Fig. 4 shows the probability of POS tags having Ezafe marker. The accuracy and other measures of this experiment are shown in Table 3 as the ‘POS tags with FPS’ approach. The result shows a precision of 34.88%, while recall is 40.63%, so the F-measure value equals to 37.54.

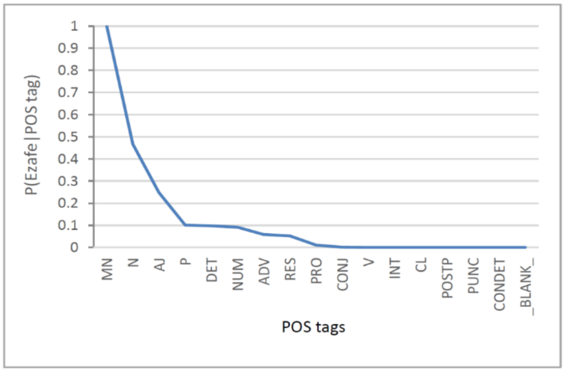


Fig. 4: Conditional probability of Ezafe appearance vs. POS tags

Baseline 4: Word form with FPS

In this experiment, we examined the role of all word forms in predicting the Ezafe marker. At first, we calculated the conditional probability of *P(Ezafe|wordform)*. In the next step, all the words in the test corpus were tagged based on a Fitness Proportionate Selection method. The results of the accuracy of this approach are shown in Table 3 as the ‘word form with FPS’ approach. The result shows a precision of 66.62%, recall of 66.64% and the F-measure value equals to 66.63.

Baseline 5: word form + POS tag with FPS

In this experiment, the conditional probability *P(Ezafe|*wordform, *POStag)* was derived. Then we tagged all the words in the test corpus based on the value of this conditional probability on a Fitness Proportionate Selection basis. The results of this approach are shown in Table 3 marked as the ‘wordform+POS tag with FPS’ approach. In this experiment, we achieved a precision and recall of 68.02% and the F-measure value equals to 68.02 as shown in Table 3.

Baseline 6: Binary Classifier

In this experiment, we developed a simple binary classifier that used a window around a word to predict the existence of Ezafe marker. This is a basic baseline system for tasks like WSD. At first, we selected a window size of 3 and the features of the classifier were as follows:

- Current word
- Ending letter of the current word
- POS tag of the current word
- POS tag of previous adjacent word
- POS tag of next adjacent word

All the words in the corpus were classified by this classifier with a 10-fold cross-validation approach and the results show a precision of 84.60%, recall of 94.10%, and F-measure value equals to 89.10. In the next step, we examined the effect of window size on the performance of the classifier. We selected a window size of 5 which means we considered the POS tag of two words before and two words after the current word as well as the current word itself, the ending letter of the current word, and the POS tag of the current word. The result shows a precision of 85.13%, recall of 94.20%, and F-measure value equals to 89.44 which is a little better than the results of window size 3. The results of these two approaches are marked in Table 3 as ‘Window Size 3’ and ‘Window Size 5’.

Baseline 7: Conditional Random Field Model

The next experiment was examined based on Conditional Random Field (CRF) which is a framework for building probabilistic models to segment and label sequence data.

Table 3: Baseline approaches

Baseline Experiments	Features	Precision	Recall	Accuracy	MCC	F_1	$F_{0.5}$
Baseline #1	Word form	90.7	30.96	87.57	48.50	46.17	65.44
Baseline #2	Word + POS tags	93.43	38.44	88.92	54.27	54.48	72.64
Baseline #3	POS tags with FPS	34.88	40.63	78.58	23.11	37.54	35.89
Baseline #4	Word form with FPS	66.62	66.64	86.07	66.63	66.63	66.62
Baseline #5	Word form +POS with FPS	68.02	68.02	86.82	68.02	68.02	68.02
Baseline #6	Binary classifier Window size 3	84.60	94.10	94.8	85.9	89.1	86.3
	Binary classifier Window size 5	85.13	94.20	95.0	86.4	89.44	86.8
Baseline #7	CRF (Asghari et al, 2014)	94.81	96.64	98.0	94.4	95.72	95.15
	CRF (with feature set 6)	95.05	**96.85**	**98.16**	**94.76**	**95.94**	95.4
Baseline #8	PB-SMT (Asghari et al, 2014)	82.42	75.91	86.82	77.81	79.03	81.03
Baseline #9	FB-SMT (with feature set 6)	**95.94**	94.47	97.22	94.32	95.20	**95.64**

CRFs are a type of discriminative undirected probabilistic graphical models.

The definition of CRF on observations X and random variables Y would be as follows: Let $G = (V, E)$ be a graph such that $Y = (Y_v)_{v \in V}$, so that Y is indexed by the vertices of G. Then *(X,Y)* is a conditional random field when the random variable Y_v, conditioned on X, obey the Markov property with respect to the graph:

$$p(Y_v|X, Y_w, w \neq v) = p(Y_v|X, Y_w, w \sim v)$$

where $w \sim v$ means that w and v are neighbors in G.

The experiment was performed based on 10-fold cross-validation. According to the previous research by [6], we set the window size to 5 for achieving the best performance. Based on this experiment, we achieved a precision of 94.81%, recall of 96.64% and F-measure value equals to 95.72.

We also investigated the CRF method of [6] with feature set 6. A precision of 95.05%, recall of 96.85%, and F-measure value equals to 95.94 was achieved. The difference is that instead of post processing Persian-specific features, we used all the features as a whole into the model and the results show better performance with respect to the [6].

Baseline 8: Monotone SMT approach

The Ezafe recognition problem can be considered as a translation problem. The original training text without the Ezafe marker can be used as the source language, and the tagged text can be mentioned as the target language. So, we can use these parallel corpora as training data into a machine translation system. In the testing phase, the text without Ezafe markers would be converted into text with Ezafe markers.

In this experiment, we re-examined the approach of [6] on our corpus to compare it with our new approach. So, a phrase-based Statistical Machine Translation (PB-SMT) algorithm was incorporated with a distortion limit set to zero. Table 3 shows the result of the simple monotone translation approach with precision rates of 82.42%, recall as 75.91, and F-measure of 79.03.

Baseline 9: Factor-based SMT approach

In this experiment, we have exploited Factor-based SMT model. The Statistical Machine Translation system only considers the surface word forms of sentences and does not include the linguistic knowledge of the languages. So, its performance is poor for dissimilar language pairs when compared to similar language pairs. The factored model was introduced as an extension of phrase-based SMT to reduce the problems of the inability to handle linguistic description beyond surface forms [52]. In a factored model, the system no longer translates words. Instead, each word is represented by a vector of factors that can contain the surface form, but also the lemma, word class, morphological characteristics, or any other information relevant to translation. Factored models can employ various types of additional information to improve translation quality between language pairs. A word in the FB-SMT framework is not a simple token; instead, it is a vector of factors representing different levels of annotation [52]. As in phrase-based translation, the main source of data for training factored models is a parallel corpus.

Table 4: Comparison of various approaches to Persian Ezafe recognition

No	Approaches	Precision	Recall	Accuracy	MCC	F1	F0.5
1	Neural sequence labeling (with feature set 6)	95.74	**96.85**	**98.27**	**95.16**	**96.29**	**95.96**
2	Neural Sequence Labeling – BILSTM+CRF (with feature set 6)	**95.78**	96.47	98.25	95.00	96.13	95.92
3	NMT (with feature set 6)	95.41	95.91	98.04	94.40	95.66	95.51

While phrase-based translation models usually memorize local translation literally and make independent assumptions between phrases which makes the model not to be in sentence level, FB-SMT models provide better generalization and richer structures [53]. In this experiment, we have examined the FB-SMT model as an approach to Ezafe recognition, and various features were used as factors into the model.

The result of FB-SMT is shown in Table 3. We selected feature set 6 of Table 2 since it has resulted in the best performance with respect to the other feature sets. The result shows a precision of 95.94% and an F-measure value equal to 95.20 with a total accuracy of 97.22%.

C. Experiments with Neural Approaches

Experiment 1: Neural Sequence Labeling

In this experiment, we have examined two neural sequence tagging models to be employed into our Ezafe recognition problem. For the first tagger, we utilized the default sequence tagger of OpenNMT Torch which is an open-source initiative for neural machine translation and sequence modeling [54]. For the second tagger, we investigated a more advanced model that incorporates a Bi-LSTM-based encoder into a CRF layer for capturing label dependencies and implemented it by the use of the OpenNMT-tf. The input of this encoder at each time is a concatenation of a word and its features. We used the Bijankhan corpus for training the models and performed a 5-fold cross-validation method to get significant results.

The results of the Neural Sequence Labeling approach have been depicted in Table 4. According to the table, in the first model, we have reached improved results with respect to baseline approaches with precision rates of 95.74%, recall at 96.85% and F1 equals to 96.29. The Neural Sequence Tagger method shows improvements in F1 and F0.5 measure in comparison with the CRF approach by 1.003 and 0.56 respectively.

Furthermore, in the second model, we investigated a BILSTM-CRF model with the same feature set. It indicates lower rates in comparison to the first one with respect to recall (96.47%) and F-measure (96.13), but its precision improves the first experiment by 95.78%.

Experiment 2: Neural Machine Translation

In this experiment, we investigated the NMT model for the Ezafe recognition task. Our NMT model includes an encoder and a decoder, each one contains two layers of LSTM cells. For this experiment, we used OpenNMT and provided it with pre-trained word embeddings. We used a random vector as the embedding vector of the Ezafe marker and constructed the word embedding of the Word+Ezafe marker by the summation of the embedding of the word and the embedding of the Ezafe marker. The input of this encoder at each time is a concatenation of a word and its features. We used Bijankhan corpus with the feature set 6 as their features for the input of this tagger, with a 5-fold cross-validation for getting better results.

To investigate the performance of our approach, we run two baseline models for comparing them with neural machine translation. The first baseline is a PB-SMT model proposed by [6] and for the second one, we developed a Factored-based SMT (FB-SMT) approach as presented in Table 3.

Table 4 shows the result of the Neural Machine Translation approach with incorporation of all the features in feature set 6 which results in precision equals to 95.41%, recall equals to 95.91% and F-measure reaches 95.66.

7- Discussion

We investigated eleven experiments to evaluate the performance of different approaches to Ezafe marking. The first nine experiments depicted in Table 3 were the baselines, in which CRF and PB-SMT (experiments 7 and 8) are the two competitor approaches that previously investigated by [6]. By the use of feature set 6, the CRF approach can perform better results than that of [6]. This baseline experiment shows the effectiveness of the syntactic features used in our investigations. Moreover, the FB-SMT approach achieved the best results when dealing with $F_{0.5}$ and precision as well.

Our approaches have been presented in experiments 1 and 2, in which two Neural Sequence Labeling methods and an

NMT model were examined. It is worth mentioning that for a fair comparison, we implemented the models with the same features as of the FB-SMT and CRF baseline models. Since CRF is the best approach of previous work investigated by [6], we selected it as the best baseline for comparing to our approaches. The results show that the neural Sequence Labeling approach can perform better than CRF method in F1 and $F_{0.5}$ by 0.35 and 0.56 respectively, but the recall rate of the two approaches is the same.

Table 5: Flase Positive and False Negative matrix of CRF and Neural Sequence Labeling approach

Recognition Algorithm	FN	FP
CRF	71159	109787
Neural Sequence Labeling	70010	95651

By investigating the behavior of the neural models, we have studied the False Positive and False Negative rates versus CRF as the best baseline. Table 5 shows the number of false positive and false negative cases in CRF and Neural Sequence Labeling as the two competitors. As can be seen in the table, there is not a big difference between the numbers of FNs in the table; we can say that the FN rates of the two methods are approximately the same. So we focus on the FP rate in more detail to evaluate the differences. By investigating the various POS tags in FP cases, the most important differences between the two methods belong to N-N and N-Adj POSE tags. So we conclude that the Neural Sequence Labeling approach can achieve better performance when encountering N-N and N-Adj cases.

Table 6: Some examples that shows the performance of Neural Sequence Labeling over CRF

No	Example	True Case	Neural Sequence Labeling	CRF
1	آنها استفاده از روش تک کتابی را در تدریس (N) ترک (N) کرده اند. ?ânhâ Estefâdeh æz ræveʃe tæk ketâbi râ dær tædris-EZ tærk kærdeh ænd.	N-N	TN	FP
2	معلم در ترغیب کودکان مدرسه (N) مسئولیت (N) سنگینی در ایجاد رغبت مطالعه دارد Moællem dær tærqibe kudækâne mædrese-EZ mæsouliate sængini dær ijâde ræqbæte motâlele dâræd.	N-N	TN	FP
3	ما مدعی هستیم که الجزیره یک بنگاه خبری ست، نه سازمانی (N) ایدئولوژیک (ADJ) Mâ modæie hæstim ke æljæzireh yek bongâhe khæbæri æst næ sâzmâni-EZ ideologic	N-ADJ	TN	FP
4	کارشناسان مسائل رسانه‌ای این نقش را به مراتب (N) مهمتر (ADJ) و تآثیرگذارتر در جنگ خلیج فارس می‌دانند. Kârʃenâsâne mæsâlele ræsâneiy in næqʃ râ be mærâteb-EZ mohemtær væ tasir gozârtær dær jænge khælije fârs midânænd.	N-ADJ	TN	FP

Some examples that show the advantage of Neural Sequence Labeling with respect to CRF approach are depicted in Table 6. The examples show that Neural Sequence Labeling usually performs well in the case of N-ADJ or N-N.

8- Conclusion

In this study, some experiments on Persian Ezafe recognition were conducted to test the impact of neural approaches in automatic Ezafe recognition. The baseline experiments were designed based on a combination of features such as word forms, POS tags, and ending letter of each word to obtain a baseline for comparing to our new approaches. The results partially confirmed the claim that there is poor accuracy by using simple baseline approaches, while CRF approach and FB-SMT performed well among all of the baselines.

The first contribution of this study is to use neural sequence labeling models, in which we exploited some lexical and grammatical Persian-specific features as factors into the model. At first, we used a Neural Sequence Labeling model. The results show a better performance with respect to the baseline experiments. We also investigated a BILSTM+CRF sequence labeling model which although shows a better precision rate, it cannot perform better than the first model in the case of F1 measure. The reason might be that we have just two labels and there is not any special dependency between the labels (with or without Ezafe marker).

The second contribution was based on Neural Machine Translation along with feature set 6 as features into the model. The performance of the NMT approach outperforms other MT approaches in the baselines by F1. But still FB-SMT model has better performance in the case of Precision and F0.5. In comparing NMT to Neural Sequence labeling models, both Neural Sequence Labeling models outperform the NMT approach.

As a result, adding various Persian-specific features to the Neural Sequence labeling algorithm resulted in a significant improvement in precision and recall with respect to CRF approach. Moreover, our approach outperforms the CRF method in the case of F0.5 measure in which the precision is more important than recall.

A research line that can be proposed for future studies is tackling the problem of Ezafe recognition as a spell-checking problem. Suppose the words that should take Ezafe markers in the original text are misspelled words. So, the problem of Ezafe recognition can be defined as a spell-checking problem; finding the words that are incorrectly written without Ezafe tags, and correcting them to the words with Ezafe marker. The output of the system can be regarded as the corrected text. One more suggestion for future work is to implement a rule-based approach by incorporating error-driven Transformation Based Learning or TBL [55], in which a sequence of rules is accepted to be applied to the corpus that leads to the most improvement in error reduction in the text. In TBL, the rules are learned iteratively and must be applied in an iterative fashion for retagging. So we may require a rule-ordering mechanism; Rules become increasingly specific as we go down the sequence. More specific rules cover just a few cases. In TBL, we should also set a stopping criterion; learning is stopped when we reach an error rate lower than a predefined threshold. The advantage of TBL is that, unlike statistical methods, allows making more sense of rules and their actions.

Another research line that can be proposed for future studies is exploiting word embeddings to solve the problem of Ezafe recognition.

Acknowledgments

We gratefully acknowledge the help and support provided by the members of the Natural Language Processing Lab, University of Tehran. The authors also wish to express their gratitude to Mr. Amin Mansouri, Behzad Mirzaabaee, Nima Hemmati, and Arezou Hatefi for their valuable assistance. Without their help, this work would have not been possible.

References

[1] Bijankhan, M., Sheykhzadegan, J., Bahrani, M., & Ghayoomi, M. (2011). Lessons from building a Persian written corpus: Peykare. Language resources and evaluation, 45(2), 143-164.

[2] Bahaadini, S., Sameti, H., & Khorram, S. (2011, September). Implementation and evaluation of statistical parametric speech synthesis methods for the Persian language. In Machine Learning for Signal Processing (MLSP), 2011 IEEE International Workshop on (pp. 1-6). IEEE.

[3] Oskouipour, N. (2011). Converting Text to phoneme stream with the ability to recognizing ezafe marker and homographs applied to Persian speech synthesis. Msc. Thesis, Sharif University of Technology, Iran.

[4] Maleki, J., Yaesoubi, M., & Ahrenberg, L. (2009, July). Applying Finite State Morphology to Conversion Between Roman and Perso-Arabic Writing Systems.Proceedings of the 2009 conference on Finite-State Methods and Natural Language Processing: Post-proceedings of the 7th International Workshop FSMNLP 2008 (pp. 215-223).

[5] Nourian, A, Rasooli, M. S., Imany, M., and Faili, H., (2015) On the Importance of Ezafe Construction in Persian Parsing, The 53rd Annual Meeting of the Association for Computational Linguistics (ACL) and the 7h International Joint Conference on Natural Language Processing (IJCNLP), Beijing, China, July 2015., Volume 2: Short Papers: 877.

[6] Asghari, H., Maleki, J., & Faili, H. (2014). A Probabilistic Approach to Persian Ezafe Recognition. 14th Conference of the European Chapter of the Association for Computational Linguistics(EACL 2014), 138. 26–30 April 2014, Gothenburg, Sweden.

[7] Noferesti, S., and Shamsfard, M., (2014). A Hybrid Algorithm for Recognizing the Position of Ezafe Constructions in Persian Texts.International Journal of Artificial Intelligence and Interactive Multimedia (IJIMAI) 2(6): 17-25 (2014).

[8] Isapour, S., Homayounpour, M. M., and Bijankhan, M. (2007). Identification of ezafe location in Persian language with Probabilistic Context Free Grammar, 13th Computer Association Conference, Kish Island, Iran.

[9] Farghaly, A., (2004). Computer Processing of Arabic Script-based Languages: Current State and Future Directions. Workshop on Computational Approaches to Arabic Script-based Languages, COLING 2004, University of Geneva, Geneva, Switzerland, August 28, 2004.

[10] Asgari, E., & Mofrad, M. R. (2016). Comparing Fifty Natural Languages and Twelve Genetic Languages Using Word Embedding Language Divergence (WELD) as a Quantitative Measure of Language Distance. arXiv preprint arXiv:1604.08561.

[11] Marno, H., Langus, A., Omidbeigi, M., Asaadi, S., Seyed-Allaei, S., & Nespor, M. (2015). A new perspective on word order preferences: the availability of a lexicon triggers the use of SVO word order. Frontiers in Psychology, 6.

[12] Moghaddam, M. D. (2001). Word order typology of Iranian languages. The Journal of Humanities of the Islamic Republic of Iran.–2001 (Spring), 8(2), 17-23.

[13] Parsafar P. (2010). Syntax, Morphology, and Semantics of Ezafe. Iranian Studies [serial online]. December 2010; 43 (5): 637-666. Available in Academic Search Complete, Ipswich, MA.

[14] Bögel, T., Butt, M., and Sulger, S., (2008). Urdu ezafe and the morphology-syntax interface. Proceedings of LFG08 (2008). CSLI Publications Stanford.

[15] Estaji, A., and Jahangiri, N. (2006). The origin of kasre ezafe in Persian language. Journal of Persian language and literature, Vol. 47, pp. 69-82, Isfahan University, Iran.

[16] Samvelian, P. (2007). The Ezafe as a head-marking inflectional affix: Evidence from Persian and Kurmanji Kurdish. Aspects of Iranian Linguistics: Papers in Honor of Mohammad Reza Bateni, 339-361.

[17] Megerdoomian, K. (2000). A computational analysis of the Persian noun phrase. Memoranda in Computer and Cognitive Science MCCS-00-321, Computing Research Lab, New Mexico State University.

[18] Bijankhan, M. (2005). A feasibility study on Ezafe Domain Analysis based on pattern matching method. Published by Research Institute on Culture, Art, and Communication, Tehran, Iran.

[19] Zahedi, M. (1998). Design and Implementation of an Intelligent Program for Recognizing Short Vowels in Persian Text. Msc. Thesis, University of Tehran, Iran.

[20] Mavvaji, V., and Eslami, M., (2012). Converting Persian Text to Phoneme Stream Based on a Syntactic Analyser. The first international conference on Persian text and speech, September 5,6, 2012, Semnan, Iran.

[21] Razi, B., and Eshqi, M., (2012). Design of a POS tagger for Persian speech based on Neural Networks, 20th Conference on Electrical Engineering, 15-17 May 2012, Tehran, Iran.

[22] Chen, X., Qiu, X., Zhu, C., Liu, P., & Huang, X. (2015). Long short-term memory neural networks for Chinese word segmentation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 1197-1206).

[23] Ma, X., & Hovy, E. (2016). End-to-end Sequence Labeling via Bi-directional LSTM-CNNs-CRF. In Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 1064-1074)

[24] Cuong, N. V., Ye, N., Lee, W. S., & Chieu, H. L. (2014). Conditional random field with high-order dependencies for sequence labeling and segmentation. The Journal of Machine Learning Research, 15(1), 981-1009.

[25] Farghaly, A., & Shaalan, K. (2009). Arabic natural language processing: Challenges and solutions. ACM Transactions on Asian Language Information Processing (TALIP), 8(4), 14.

[26] Elshafei, M., Al-Muhtaseb, H., & Al-Ghamdi, M. (2006 a). Machine Generation of Arabic Diacritical Marks. MLMTA, 2006, 128-133.

[27] Elshafei, M., Al-Muhtaseb, H., & Al-Ghamdi, M. (2006 b). Statistical methods for automatic diacritization of Arabic text. In The Saudi 18th National Computer Conference. Riyadh (Vol. 18, pp. 301-306).

[28] Diab, M., Ghoneim, M., & Habash, N. (2007, September). Arabic diacritization in the context of statistical machine translation. In Proceedings of MT-Summit.

[29] Habash, N., & Rambow, O. (2007, April). Arabic diacritization through full morphological tagging. In Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics (NAACL 2007); Companion Volume, pp. 53-56, Association for Computational Linguistics.

[30] Belinkov, Y., & Glass, J. (2015). Arabic diacritization with recurrent neural networks. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 2281-2285).

[31] Abandah, G. A., Graves, A., Al-Shagoor, B., Arabiyat, A., Jamour, F., & Al-Taee, M. (2015). Automatic diacritization of Arabic text using recurrent neural networks. International Journal on Document Analysis and Recognition (IJDAR), 18(2), 183-197.

[32] De Mareüil, P. B., Adda-Decker, M., & Gendner, V. (2003). Liaisons in French: a corpus-based study using morpho-syntactic information. In Proc. of the 15th International Congress of Phonetic Sciences.

[33] Larson, R.K. (2009), Chinese as a reverse Ezafe language. Yuyanxue Luncong, Journal of Linguistics, 39: 30-85. Peking University.

[34] Ling, W., Dyer, C., Black, A. W., Trancoso, I., Fermandez, R., Amir, S., ... & Luis, T. (2015). Finding Function in Form: Compositional Character Models for Open Vocabulary Word Representation. In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (pp. 1520-1530).

[35] Peters, M., Ammar, W., Bhagavatula, C., & Power, R. (2017). Semi-supervised sequence tagging with bidirectional language models. In Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (Vol. 1, pp. 1756-1765).

[36] Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735-1780.

[37] LeCun, Y., Boser, B., Denker, J. S., Henderson, D., Howard, R. E., Hubbard, W., & Jackel, L. D. (1989). Backpropagation applied to handwritten zip code recognition. Neural Computation, 1(4), 541-551.

[38] Lample, G., Ballesteros, M., Subramanian, S., Kawakami, K., & Dyer, C. (2016). Neural Architectures for Named Entity Recognition. In Proceedings of NAACL-HLT (pp. 260-270).

[39] Koehn, P., Och, F. J., and Marcu, D. (2003). Statistical phrase-based translation. In Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology - Volume 1, NAACL '03, pages 48–54, Stroudsburg, PA, USA. Association for Computational Linguistics.

[40] Kalchbrenner, N. and Blunsom, P. (2013). Recurrent continuous translation models. In Proceedings of the ACL Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 1700–1709. Association for Computational Linguistics.

[41] Sutskever, I., Vinyals, O., and Le, Q. (2014). Sequence to sequence learning with neural networks. In Advances in Neural Information Processing Systems (NIPS 2014).

[42] Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014a). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[43] Hermann, K. M., & Blunsom, P. (2014). Multilingual models for compositional distributed semantics. arXiv preprint arXiv:1404.4641.

[44] Bahdanau, D., Cho, K., & Bengio, Y. (2014). Neural machine translation by jointly learning to align and translate. arXiv preprint arXiv:1409.0473.

[45] Bijankhan, M. (2004). The Role of the Corpus in Writing a Grammar: An Introduction to a Software, Iranian Journal of Linguistics, Vol. 19, No. 2, fall and winter 2004.

[46] Amiri, H., Hojjat, H., & Oroumchian, F. (2007). Investigation on a feasible corpus for Persian POS tagging. In Proceedings of the 12th International CSI Computer Conference (CSICC), 2007.

[47] Mohtaj, Salar, Behnam Roshanfekr, Atefeh Zafarian, Habibollah Asghari, (2018), Parsivar: A Language Processing Toolkit for Persian, 11th edition of the Language Resources and Evaluation Conference (LREC 2018), 7-12 May 2018, Miyazaki (Japan).

[48] Arppe, A. (2000). Developing a grammar checker for Swedish. In Proceedings of NODALIDA (Vol. 99, pp. 13-27).

[49] Bernth, A. (1997, March). EasyEnglish: a tool for improving document quality. In Proceedings of the fifth conference on applied natural language processing (pp. 159-165). Association for Computational Linguistics.

[50] Powers, D.M.W., (2011). Evaluation: from Precision, Recall, and F-measure to ROC, Informedness, Markedness, and Correlation. Journal of Machine Learning Technologies, 2(1), 37-63.

[51] Bäck, T. (1996). Evolutionary algorithms in theory and practice. Oxford University Press.

[52] Koehn, P., & Hoang, H. (2007, June). Factored Translation Models. Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, In EMNLP-CoNLL 2007, pp. 868–876, Prague, June 2007. Association for Computational Linguistics.

[53] Feng, Y., Cohn, T., & Du, X. (2014). Factored Markov translation with robust modeling. In Proceedings of the Eighteenth Conference on Computational Natural Language Learning (pp. 151-159).

[54] Klein, G., Kim, Y., Deng, Y., Senellart, J., & Rush, A. (2017). OpenNMT: Open-Source Toolkit for Neural Machine Translation. Proceedings of ACL 2017, System Demonstrations, 67-72.

[55] Brill, E. (1995). Transformation-based error-driven learning and natural language processing: A case study in part-of-speech tagging. Computational Linguistics, 21(4), 543-565.

[56] Kahnemuyipour, Arsalan (2003). Syntactic categories and Persian stress. Natural Language & Linguistic Theory 21.2: 333-379.

[57] Ghomeshi, J. (1996). Projection and Inflection: A Study of Persian Phrase Structure. Ph.D. Thesis, Graduate Department of Linguistics, University of Toronto.

[58] Doostmohammadi, E., Nassajian, M., & Rahimi, A. (2020, November). Persian Ezafe Recognition Using Transformers and Its Role in Part-Of-Speech Tagging. In Findings of the Association for Computational Linguistics: EMNLP 2020 (pp. 961-971).

[59] Larson, R., & Samiian, V. (2020). The Ezafe construction revisited. Advances in Iranian linguistics, 351, 173.

[60] Ansari, A., Ebrahimian, Z., Toosi, R., & Akhaee, M. A. (2023, May). Persian Ezafeh Recognition using Transformer-Based Models. In 2023 9th International Conference on Web Research (ICWR) (pp. 283-288). IEEE.

[61] Kharsa, R., Elnagar, A., & Yagi, S. (2024). BERT-Based Arabic Diacritization: A state-of-the-art approach for improving text accuracy and pronunciation. Expert Systems with Applications, p. 123416.

[62] Lapointe, M., Kadim, A., & Dliou, A. (2023, November). Literature Review of Automatic Restoration of Arabic Diacritics. In 2023 IEEE International Conference on Advances in Data-Driven Analytics And Intelligent Systems (ADACIS) (pp. 1-5). IEEE.