



## المجلة الأردنية للحاسوب وتكنولوجيا المعلومات

ISSN 2415 - 1076 (Online)  
ISSN 2413 - 9351 (Print) العدد ٤ المجلد ٨ كانون الأول ٢٠٢٢

### عنوان البحث الصفحات

تصميم هوائي مُستوٍ على شكل F مقلوب PIFA ثلاثي النطاقات صغير الحجم، باستخدام الإبطال الجزئي للمستوى الأرضي وخوارزمية جينية ثنائية GA ليلى وكريم، أسمى خبّا، جمال أماديد، وسيدا إبنيايش	٣١٧ - ٣٠٨
نحو تطوير مُساعدٍ شخصيٍّ ذكيٍّ للعربية التونسية إيناس زربي، و لميا ه. بلغويث	٣٣٥ - ٣١٨
إطار للاندماج على مستوى السّمات لتصنيف صور الرّنين المغناطيسي للدماغ باستخدام التّعلّم العميق المراقب واستخلاص السّمات براشانثا س. ج.، و ه. ن. براكاش	٣٤٤ - ٣٣٦
تركيبية من نماذج التّعلّم العميق للتنبؤ بأسعار الأسهم في بورصتي AAPL و TSLA زهرة بزّادي، محمد لازار، أسامة محبوب، حليم بزّادي، و هشام عمارة	٣٥٦ - ٣٤٥
التنبؤ المبكر بسرطان عنق الرحم باستخدام تقنيات تعلّم الآلة محمد صبحي البطاح، مازن الزّيود، رائد الأزّايدة، مالك توبات، حنين الزّعي، و أريج غليات	٣٦٩ - ٣٥٧
COTA 2.0: مُصحّح أوتوماتيكي لنصوص وسائل التواصل الاجتماعي بالعربية التونسية أسما مكي، أينا س زربي، مريم إلوّز، و لميا ه. بلغويث	٣٨٧ - ٣٧٠

JJCIT

## Jordanian Journal of Computers and Information Technology

December 2022 VOLUME 08 NUMBER 04 ISSN 2415 - 1076 (Online)  
ISSN 2413 - 9351 (Print)

### PAGES PAPERS

308 - 317	A SEMI-DEFECTED GROUND PLANE AND A BINARY GENETIC ALGORITHM FOR DESIGNING A VERY COMPACT TRIPLE-BAND PIFA ANTENNA Layla Wakrim, Asma Khabba, Jamal Amadid and Saida Ibnyaich
318 - 335	TOWARD DEVELOPING AN INTELLIGENT PERSONAL ASSISTANT FOR TUNISIAN ARABIC Inès Zribi and Lamia H. Belguith
336 - 344	FEATURE LEVEL FUSION FRAMEWORK FOR BRAIN MR IMAGE CLASSIFICATION USING SUPERVISED DEEP LEARNING AND HAND CRAFTED FEATURES Prashantha S. J. and H. N. Prakash
345 - 356	COMBINATION OF DEEP-LEARNING MODELS TO FORECAST STOCK PRICE OF AAPL AND TSLA Zahra Berradi, Mohamed Lazaar, Oussama Mahboub, Halim Berradi and Hicham Omara
357 - 369	EARLY PREDICTION OF CERVICAL CANCER USING MACHINE LEARNING TECHNIQUES Mohammad S. Al-Batah, Mazen Alzyoud, Raed Alazaidah, Malek Toubat, Haneen Alzoubi and Areej Olaiyat
370 - 387	COTA 2.0: AN AUTOMATIC CORRECTOR OF TUNISIAN ARABIC SOCIAL MEDIA TEXTS Hadi Khosravi and Mohammad GhasemiGol

JJCIT

## Jordanian Journal of Computers and Information Technology (JJCIT)

The Jordanian Journal of Computers and Information Technology (JJCIT) is an international journal that publishes original, high-quality and cutting edge research papers on all aspects and technologies in ICT fields.

JJCIT is hosted and published by Princess Sumaya University for Technology (PSUT) and supported by the Scientific Research Support Fund in Jordan. Researchers have the right to read, print, distribute, search, download, copy or link to the full text of articles. JJCIT permits reproduction as long as the source is acknowledged.

### AIMS AND SCOPE

The JJCIT aims to publish the most current developments in the form of original articles as well as review articles in all areas of Telecommunications, Computer Engineering and Information Technology and make them available to researchers worldwide. The JJCIT focuses on topics including, but not limited to: Computer Engineering & Communication Networks, Computer Science & Information Systems and Information Technology and Applications.

### INDEXING

JJCIT is indexed in:



### EDITORIAL BOARD SUPPORT TEAM

#### LANGUAGE EDITOR

Haydar Al-Momani

#### EDITORIAL BOARD SECRETARY

Eyad Al-Kouz



All articles in this issue are open access articles distributed under the terms and conditions of the Creative Commons Attribution license (<http://creativecommons.org/licenses/by/4.0/>).

### JJCIT ADDRESS

**WEBSITE:** [www.jjcit.org](http://www.jjcit.org)

**EMAIL:** [jjcit@psut.edu.jo](mailto:jjcit@psut.edu.jo)

**ADDRESS:** Princess Sumaya University for Technology, Khalil Saket Street, Al-Jubaiha

**B.O. BOX:** 1438 Amman 11941 Jordan

**TELEPHONE:** +962-6-5359949

**FAX:** +962-6-7295534

## EDITORIAL BOARD

Wejdan Abu Elhaija ( <a href="#">EIC</a> )	Ahmad Hiasat ( <a href="#">Senior Editor</a> )	
Aboul Ella Hassanien	Adil Alpkoçak	Adnan Gutub
Adnan Shaout	Christian Boitet	Gian Carlo Cardarilli
Omer Rana	Mohammad Azzeh	Nijad Al-Najdawi
Hussein Al-Majali	Maen Hammad	Ayman Abu Baker
Ahmad Al-Taani	João L. M. P. Monteiro	Leonel Sousa
Omar Al-Jarrah		

## INTERNATIONAL ADVISORY BOARD

Ahmed Yassin Al-Dubai <a href="#">UK</a>	Albert Y. Zomaya <a href="#">AUSTRALIA</a>
Chip Hong Chang <a href="#">SINGAPORE</a>	Izzat Darwazeh <a href="#">UK</a>
Dia Abu Al Nadi <a href="#">JORDAN</a>	George Ghinea <a href="#">UK</a>
Hoda Abdel-Aty Zohdy <a href="#">USA</a>	Saleh Oqeili <a href="#">JORDAN</a>
João Barroso <a href="#">PORTUGAL</a>	Karem Sakallah <a href="#">USA</a>
Khaled Assaleh <a href="#">UAE</a>	Laurent-Stephane Didier <a href="#">FRANCE</a>
Lewis Mackenzies <a href="#">UK</a>	Zoubir Hamici <a href="#">JORDAN</a>
Korhan Cengiz <a href="#">TURKEY</a>	Marco Winzker <a href="#">GERMANY</a>
Marwan M. Krunz <a href="#">USA</a>	Mohammad Belal Al Zoubi <a href="#">JORDAN</a>
Michael Ullman <a href="#">USA</a>	Ali Shatnawi <a href="#">JORDAN</a>
Mohammed Benaissa <a href="#">UK</a>	Basel Mahafzah <a href="#">JORDAN</a>
Nadim Obaid <a href="#">JORDAN</a>	Nazim Madhavji <a href="#">CANADA</a>
Ahmad Al Shamali <a href="#">JORDAN</a>	Othman Khalifa <a href="#">MALAYSIA</a>
Shahrul Azman Mohd Noah <a href="#">MALAYSIA</a>	Shambhu J. Upadhyaya <a href="#">USA</a>

---

"Opinions or views expressed in papers published in this journal are those of the author(s) and do not necessarily reflect those of the Editorial Board, the host university or the policy of the Scientific Research Support Fund".

"ما ورد في هذه المجلة يعبر عن آراء الباحثين ولا يعكس بالضرورة آراء هيئة التحرير أو الجامعة أو سياسة صندوق دعم البحث العلمي والابتكار".

# A SEMI-DEFECTED GROUND PLANE AND A BINARY GENETIC ALGORITHM FOR DESIGNING A VERY COMPACT TRIPLE-BAND PIFA ANTENNA

Layla Wakrim<sup>1</sup>, Asma Khabba<sup>2</sup>, Jamal Amadid<sup>2</sup>, Saida Ibnyaich<sup>2</sup>

(Received: 19-May-2022, Revised: 6-Aug.-2022, Accepted: 18-Aug.-2022)

## ABSTRACT

We suggest in this study a very compact triple-band PIFA antenna for mobile and wireless applications by using the binary genetic algorithm and a semi-defected ground plane. This antenna with the dimensions of  $38 \times 40 \times 1.9 \text{ mm}^3$  is dedicated to LTE Band 11 (1427.9-1495.9 MHz), HIPERLAN/2 (5.15-5.35 GHz), WLAN (5.15-5.35 GHz) and 5G Sub-6GHz applications. To accomplish triple-band operation with acceptable performance, the genetic algorithm is used to dictate the form of the ground plane of the antenna. The simulation results showed that the developed PIFA antenna has optimal operation on three frequencies. The first resonance frequency is 1.32 GHz with a bandwidth ( $S_{11} < -10 \text{ dB}$ ) from 1.28 GHz to 1.38 GHz. The middle and higher bands are centred respectively at 3.12 GHz and 5.2 GHz, with a bandwidth from 3.05 to 3.17 GHz and from 4.93 to 5.44 GHz, respectively.

## KEYWORDS

PIFA antenna, Genetic algorithm, Optimization, Triple-band.

## 1. INTRODUCTION

Due to the rapid evolution of communication standards and devices operating in multiple frequency bands, the development of new antennas has become necessary to meet new demands. Planar antennas are a type of antenna that has all the desirable characteristics; one of them is the planar inverted-F antenna (PIFA). PIFAs are the best candidate to meet the requirements for mobile device antennas due to their tiny volume and low profile [1]. The biggest problem with PIFA antennas is the narrow bandwidth. Several techniques exist in the literature to improve the bandwidth of these antennas by modifying the geometries of the radiating plane [2], modifying the ground plane [3], adding parasitic elements [4] or using metamaterials [5].

The effect of the antenna's ground plane remains one of the most widely used techniques for obtaining different resonant frequencies [6]-[9]. Several research studies have investigated the effect of ground plane geometry on improving antenna performance. The impact of using defected ground structure DGS can increase the bandwidth from 30% to 119% without increasing the antenna volume as presented in [10]. In [11], the authors have shown that the use of a shaped ground plane allows for improving the gain without any effect on the impedance bandwidth and the radiation characteristics. In [12]-[13], the study presents a slotted antenna with a deformed ground plane. The proposed antenna's ground plane is deformed to enable multi-resonant wideband operation. The authors in [14] show that a PIFA antenna with a ground plane composed of multi-trip connection allowed to have ultra wideband. Other conceptions are exhibited by adding different slot shapes on the ground plane [15]-[21] in order to enhance the antenna bandwidth.

According to the previous analysis and still with the aim of improving the bandwidth of the PIFA antenna or introducing novel resonant frequencies, this study proposes the use of a semi-defected ground plane and the genetic algorithm in binary code [22]. The mobile's electronic components are normally attached to the antenna ground plane. The use of a fully defected ground plane makes the implementation of components impossible, while the use of a semi-defected ground plane is justified. The genetic algorithm remains among the most powerful algorithms to search for an optimal solution.

In this paper, we propose the use of this algorithm to determine the shape of the ground plane instead

- 
1. L. Wakrim is with Laboratory of Innovation in Management and Engineering for Business (LIMIE), Higher Institute of Engineering and Business (ISGA), Marrakech, Morocco. Email: Layla.wakrim@isga.ma
  2. A. Khabba, J. Amadid and S. Ibnyaich are with University Cadi Ayyad, FSSM Marrakech, Morocco. Emails: khabba.asma@gmail.com, jamal.amadid@edu.uca.ac.ma and s.ibnyaich@uca.ac.ma

of the traditional method based on the parametric study. It is used to indicate the location of the slots which must be placed on the part of the ground plane. With the proposed method, we can control the desired frequency ranges without modifying or increasing the antenna volume.

The purpose of this study is to present a new method of designing a multi-band PIFA antenna to overcome the problems of a voluminous antenna and narrow band with a simple deformation of the ground plane. The proposed antenna is a very compact triple-band antenna with a semi-defected ground plane for GPS, LTE band 11 and 5G applications. The antenna dimensions are  $38 \times 40 \times 1.9 \text{ mm}^3$  with an FR4 epoxy substrate of  $40 \times 100 \times 1.6 \text{ mm}^3$ . The antenna covers three bands, the first bandwidth is from 1.28 GHz to 1.38 GHz with the resonant frequency of 1.326 GHz ( $S_{11} = -17.12 \text{ dB}$ ), while the second bandwidth is from 3.05 GHz to 3.17 GHz with the resonant frequency of 3.12 GHz ( $S_{11} = -24.11 \text{ dB}$ ) and the third bandwidth is from 4.93 GHz to 5.44 GHz with the resonant frequency of 5.2 GHz ( $S_{11} = -30.88 \text{ dB}$ ).

The developed PIFA antenna exhibits good performance in terms of reflection coefficient, gain, efficiency and 2D radiation at the three resonant frequencies. This research is composed of five sections. The first section is an introduction. An overview of the genetic algorithm, methodology and antenna parameters is clarified in Section 2. The simulated results are introduced in Section 3. In Section 4, a comparative study is presented, while the conclusion is presented in Section 5.

## 2. DESIGN METHODOLOGY AND ANTENNA PARAMETERS

In this section, an overview of the genetic algorithm, the PIFA antenna theory, the design method based on DGP (Defected Ground Plane) and GA (Genetic Algorithm) and the antenna parameters and their optimal form is given.

### 2.1 PIFA Antenna Theory

The inverted-F antenna is evolved from a quarter-wavelength monopole antenna. It is basically a modification of the inverted-F antenna (IFA) which is consisting of a short vertical monopole wire. To increase the bandwidth of the IFA, a modification is made by replacing the wires with a horizontal plate and a vertical short circuit plate to obtain a PIFA antenna [28].

The conventional PIFA is constituted by a top patch, a shorting plate and a feeding plate. The top patch is mounted above the ground plane, which is connected also to the shorting plate and the feeding plate at proper positions. They have the same length as the distance between the top patch and the ground plane. The standard design formula for a PIFA antenna is [29]:

$$f_r = \frac{C}{4(L + h)}$$

Where  $f_r$  is the resonant frequency of the main mode,  $C$  is the speed of light in the free space;  $h$  and  $L$  are the height and the length of the radiating plate, respectively.

### 2.2 Overview of Genetic Algorithm

In the field of antenna design, the application of metaheuristic algorithms is critical. The genetic algorithm is one of these algorithms (GA). GA is regarded as a resilient and stochastic search approach based on the concepts and principles of natural evolution and selection. This algorithm's strength stems from its capacity to use prior solutions' historical information structures in an optimization process of making future solution structures operate better.

The most important parameters of genetic algorithms which must be chosen carefully are [23]-[24], [30]:

- Population number: It represents the number of chromosomes that are considered in a generation. The bigger this number is, the more important is the calculation time, but the solution becomes better.
- Generation numbers: The genetic algorithm can evolve for the maximum number of generations or iterations before stopping.
- Selection: It is based on the evaluation of criteria to choose the best individuals in a population that will reproduce. Roulette selection and tournament selection are the two most used techniques.

-Crossover: For an optimization issue, the crossover is an exchange of sub-strings signifying chromosomes. It could be a one-point crossover or a two-point crossover.

-Mutation: Mutation is the process of changing the state of a bit (gene) in the chromosomes to produce a population variation.

The optimization process can be summarized in four steps; Step 1, A random population will be generated based on the number chosen for this population. In Step 2, it is required to assess the fitness function. The fitness function is a function that can be used to determine whether a solution is excellent or bad for the problem under consideration. If the stopping criteria are satisfied, then stop, else go to Step 3. In Step 3, after evaluating the fitness function, the GA selects the individuals who will reproduce in the new population and removes the others. In Step 4, after the selection phase and to vary the population, the GA uses genetic operators, such as mutation and crossover, by selecting the number of this population. This process is repeated until the optimal solution is found.

GAs are defined by their ability to adapt to difficult problems, their ease of implementation and the fact that they can be used to find one or more variables. On the other hand, the GA has a slow convergence and a low risk of having a premature convergence, because the adjustment of these parameters is delicate.

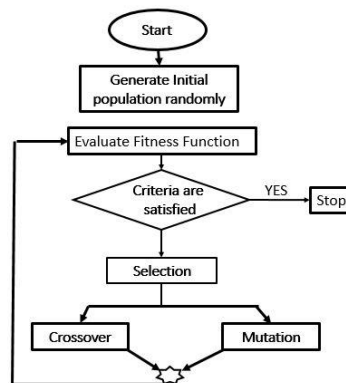


Figure 1. Genetic algorithm flowchart.

### 2.3 Design Methodology

According to the literature, the ground plane has a remarkable impact on the performance variation of a planar antenna (PIFA). This sub-section proposes the combination of a defected ground plane and a binary genetic algorithm for the design of a new PIFA antenna structure with multi-band option without increasing its volume. The base antenna to be improved is a single-band antenna. The use of a subdivided ground plane instead of a continuous plane is intended to keep just the cells that will give the desired performance. These cells will be determined using the genetic algorithm in binary code while evaluating fitness functions. This method's major objective is to create a new antenna design from the conventional PIFA antenna.

Figure 2 shows the technique proposed in this study to optimize the performance of a PIFA antenna. The first step is to divide the ground plane of the base antenna into 260 small cells of the size of  $2 \times 2 \text{ mm}^2$ , according to the X-axis (Figure 2 a). The adjacent cells and the parallel cells are separated by a width of 1 mm called overlap to ensure physique support of the antenna and to find a better solution for conductive cells. The second step is the use of the binary genetic algorithm (BGA). The choice of the algorithm is justified by the fact that only two cases exist; existing cells and non-existing cells; The conductivity qualities of each cell are determined by the usage of "0" or "1" in the chromosome. A gene with the value "1" defines the existing cell, while a gene with "0," often known as a slot, defines the non-existent cell. Figure 2b shows how the binary genetic algorithm was used to create a design that could produce the desired and anticipated antenna performances. The non-existing cells shape the slots, while the existing cells create the new ground-plane shape.

The parameters of the genetic algorithm used in this study to obtain the final solution are 40 individuals in the population, while the single-point crossover approach is utilized with a 100% probability of

crossover. Considering tournament selection type and mutation probability of 0.8% the generation number is set to 100 iterations or generations.

The genetic algorithm has two options; either it minimizes a function or it maximizes it. With the function  $F$  described below whose value is negative because  $S_{11}$  is less than zero, the genetic algorithm will minimize it. If we want to maximize it, we must use the  $F$  function in absolute value. However, in the desired bands, we wanted to obtain a reflection coefficient of less than -10 dB. To discover the optimum design, the  $F$  function proposed in this study is represented as:

$$F = \frac{\sum_{i=1}^N S_{11}(f(i))}{N}$$

In this equation, the variable  $f(i)$  represents the sampling frequency and  $N$  represents the number of samples. The fitness function  $F$  is defined as having an impedance bandwidth of less than -10 in the desired range of frequencies. The minimum value of  $F$  is used as the iteration's end condition to determine when the algorithm should be stopped.

The parameters of the genetic algorithm used for this obtained solution are:

Population size: 40, Crossover: Two-point, Mutation: 100%, Selection: tournament selection and Number of generations: 50.

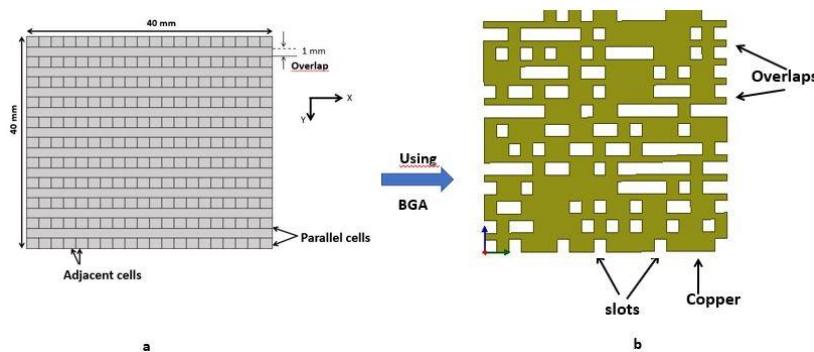


Figure 2. Ground plane: a. Conventional ground plane, b. Obtained ground plane by BGA.

The reflection coefficient of the conventional antenna (Fig 2a) and the proposed antenna (Figure 2b) are shown in Figure 3. The conventional antenna is a single-band antenna with a resonance frequency of 3.56 GHz and a reflection coefficient of -28.29 dB at this frequency. It has a bandwidth of -10 dB at 130 MHz. The proposed antenna has three resonant frequencies; the main frequency of the conventional antenna with two other frequencies that have appeared at 1.32 GHz and 5.2 GHz, the reason why two new bandwidths appear. The use of a slotted ground plane not only improves the main bandwidth but also makes new resonances appear and thus new operating bands appear. This miniature antenna is designed for mobile-phone applications to cover the following frequency bands: LTE Band 11 (1427.9-1495.9 MHz), HIPERLAN/2 (5.15-5.35 GHz), WLAN (5.15-5.35 GHz) and 5G Sub-6GHz applications. Table 1 summarizes the evolution of the bandwidth from the conventional antenna to the proposed antenna.

Table 1. Bandwidth of the antenna without/with the slots.

Characteristics	Conventiounnal antenna	Proposed antenna
Frequency (GHz)	3.56	1.32
		3.12
		5.2
Bandwidth (GHz)	0.13 (3.5 to 3.63)	0.17 (1.28 to 1.45)
		0.12 (3.05 to 3.17)
		0.61 (4.83 to 5.44)
S11 (dB)	-28.29	-17.12
		-24.11
		-30.88

Results across Table 1 show that the single-band antenna features have been improved to have a triple-band antenna. The first band at the 1.53 GHz resonance frequency is enhanced from 0 to 20%. The

second band that has been improved at the 5.45 GHz resonance band; its value was 0 and changed to 30 %. The main band is improved by 35%. The objective of this technique is to increase the bandwidth of the PIFA antenna at low and high frequencies  $f_1$  and  $f_3$  to reach triple frequency bands.

During this study, we have used two principal software. The first one is a programming platform to use the genetic algorithm and the second is an electromagnetic simulation software which is used to model and design the antenna.

To implement the genetic algorithm in electromagnetic software, we checked the electromagnetic software from the programming platform. This technique is compiled by the programming platform and VbScript of the electromagnetic software. The idea is to introduce some parameters of design into the electromagnetic software from the programming platform and use the electromagnetic software for simulation, exporting the reflection coefficient from the electromagnetic software using the programming platform and drawing data by the programming platform. The flowchart boxes contain the operations to be performed, while the software package that operates. Figure 4 shows a flowchart of the programming platform- electromagnetic software configuration.

In the programming platform, we create the antenna design and the VBScript file through an interface that allows us to pass the communication between electromagnetic software and the programming platform. Then, the electromagnetic software is triggered. In electromagnetic software, the antenna will be modelled and simulated and finally, the simulated results will be imported from the programming platform to plot them. The programming platform can directly call the electromagnetic software to calculate the reflection coefficient (S11) in dB.

Figure 5 presents the implementation of the genetic-algorithm optimization with the electromagnetic software. As indicated in Figure 5, the optimization procedure is carried out. The programming platform and the electromagnetic software are two software applications that are linked. The electromagnetic software gives the modeling of the PIFA antenna in terms of its characteristics, while the programming platform contains the genetic algorithm and allows the calculation of the fitness value. This function is the physical link with the GA to obtain the optimal solution.

By comparing the design of the PIFA-antenna ground plane to the evaluation of a fitness function, the design of the PIFA-antenna ground plane is justified. The method is finished if the fitness meets the parameters. Otherwise, a GA technique is used to create new structures. To justify their performances, those new structures are utilized in the following generation of electromagnetic-software analysis. Figure 2 shows the optimized ground plane obtained by the proposed method in retrieving the objectives.

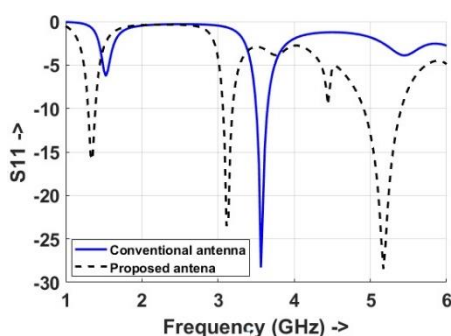


Figure 3. Reflection coefficients of the conventional and proposed antennas.

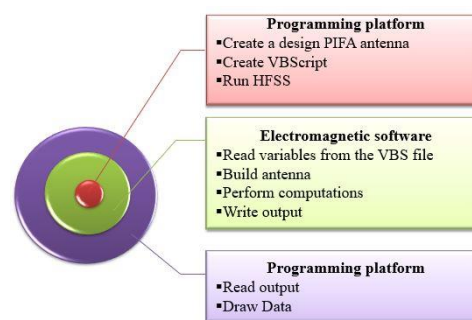


Figure 4. Interfacing the programming platform and electromagnetic software for PIFA-antenna design.

## 2.4 Antenna Parameters

The proposed antenna was developed from the conventional PIFA antenna by modifying its ground plane. The dielectric type FR4-epoxy is placed on the ground plane with a relative permittivity value of 4.4 and a thickness  $t_1$  of 0.5 mm. The radiating plate is composed of a continuous plane of dimensions  $L_p \times W_p \times 0.2 \text{ mm}^3$ . The overall dimensions of the proposed antenna is  $W_p \times L_p \times h$ . The dimensions of the ground plane were  $W_g \times L_g \times t_1$ . The distance between the radiating plate and the ground plane represents the height  $h$  of the antenna filled with air. The dimensions of the feeding and shorting plates were  $W_1 \times L_1$  and  $W_2 \times L_2$ , respectively.



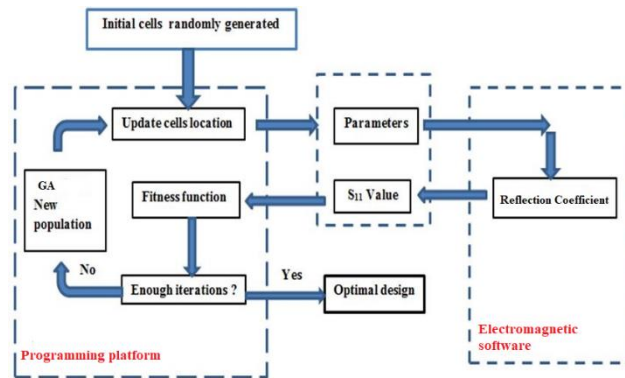


Figure 5. Implementation of the genetic-algorithm optimization with electromagnetic software.

Figure 6 exhibits the configuration of the proposed PIFA antenna. Figure 6a presents the geometry of the antenna and its parameters. Table 2 summarizes the parameters and their corresponding coordinates. Figure 6b exposes the geometry of the ground plane obtained by the genetic algorithm. The ground plane contains 134 slots of size  $2 \times 2 \text{ mm}^2$  which are distributed on its surface, being reduced to 53.54% by adding slots using the genetic algorithm. This technique results in seven forms of slots that have the same width of 2 mm, but with different lengths.

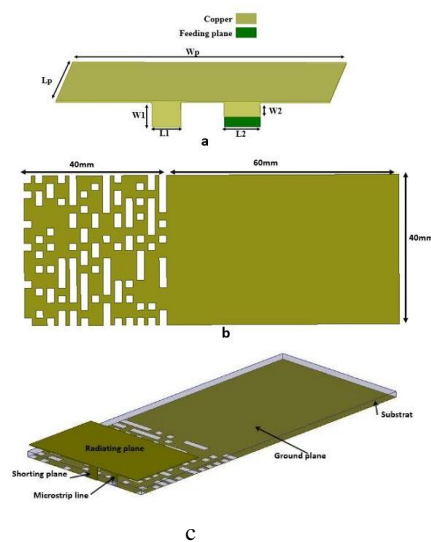


Table 2. Antenna parameters.

Parameters	Dimensions in mm
$L_p$	38
$W_p$	24
$L_g$	40
$W_g$	100
$h$	1.9
$W_1$	3.5
$L_1$	3
$W_2$	1.9
$L_2$	2
$t_1$	1.6

Figure 6. Configuration of the proposed PIFA antenna: (a) Antenna, (b) Semi-defected ground plane and (c) Global view of the proposed antenna.

### 3. RESULTS AND DISCUSSION

In this section, we studied the proposed-antenna performances in terms of reflection coefficient  $S_{11}$ , gain, efficiency, radiation pattern and current distribution.

#### 3.1 Reflection Coefficient

We used another electromagnetic software to verify the reflection-coefficient result that was generated by the first electromagnetic software. The reflection coefficients from the first and second electromagnetic software are compared in Figure 7. From this figure, we see that the two simulated  $S_{11}$  variations are correlated and represent a good agreement between the results given by the two electromagnetics software.

Based on the analysis of the results, the proposed antenna is qualified to operate in three bands ( $S_{11} < -10 \text{ dB}$ ), such as LTE Band 11, HIPERLAN/2, WLAN and 5G Sub-6GHz applications. The antenna covers three bands; the first bandwidth is from 1.28 GHz to 1.45 GHz with the resonant frequency of 1.326 GHz, while the second bandwidth is from 3.05 GHz to 3.17 GHz with the resonant frequency of

3.12 GHz and the third bandwidth is from 4.83 GHz to 5.44 GHz with the resonant frequency of 5.2 GHz.

The slots on the ground plane are responsible for attaining the new resonant frequencies at 1.32 GHz and 5.2 GHz, because of using a semi-defected ground plane. The slots on the ground plane are formed as part of the GA optimization technique to get the desired bands. They provide several resonant current paths, allowing the antenna to resonate at many frequencies. The simulation results demonstrate the importance of the optimization method adopted.

### 3.2 Gain and Efficiency

Figure 8 displays the gain- and efficiency-simulation results for the operating bands. The gain varies in growth: the lower frequency gain is 2.5 dB, the middle-frequency gain is 4.5 dB and the higher-frequency gain is 6 dB. We can notice that the suggested antenna has a good gain across all bands. At 1.32 GHz, 3.12 GHz and 5.2 GHz, respectively, the antenna's radiation efficiency displays respectable values in all three bands: 96%, 90% and 95%.

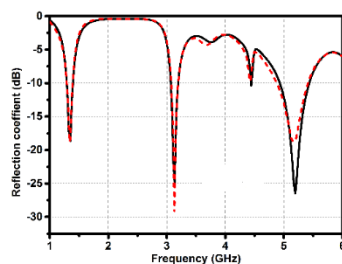


Figure 7. Reflection coefficients of the PIFA antenna extracted from different electromagnetic software.

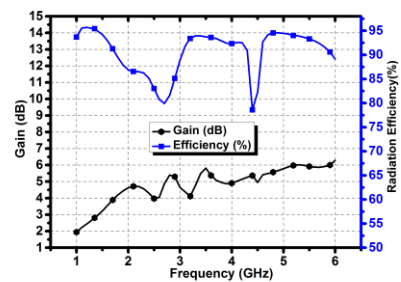


Figure 8. Gain and efficiency of the suggested antenna.

### 3.3 Radiation Pattern

The 2D radiation pattern determined in the far-field region for  $\Phi = 0^\circ$  and  $\Phi = 90^\circ$  at the three resonant frequencies 1.32 GHz, 3.12 GHz and 5.2 GHz was simulated. We present in Figure 9 the simulated 2D radiation pattern. We can see that the proposed PIFA antenna offers an approximately omnidirectional radiation pattern.

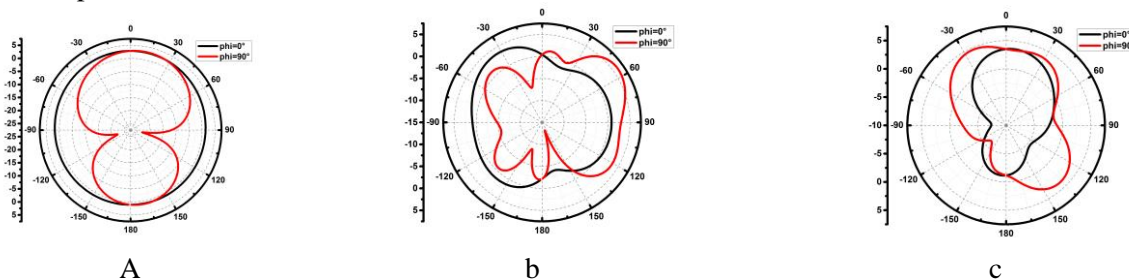


Figure 9. 2D Radiation pattern: (a) at 1.32 GHz, (b) at 3.12 GHz and (c) at 5.2 GHz.

### 3.4 Current Distributions

Figure 10 displays the simulated surface current distributions at the three resonances of the PIFA-antenna. The radiating plate and ground plane's current patterns at 1.32 GHz, 3.12 GHz and 5.2 GHz are shown in this figure. We can observe that the current is distributed on the ground plane and the radiating plane and concentrated on the shorting plane at the resonant frequencies.

## 4. COMPARATIVE STUDY

In the literature, the impact of the slots on the ground plane of the PIFA antenna has been researched by numerous researchers. In [25], the totally defected ground plane is used to minimize the PIFA antenna size and to convert the single-band antenna to a dual-band antenna with an important bandwidth. The authors have proposed two types of metaheuristic algorithms: the first one is the binary genetic algorithm and the second one is particle-swarm optimization to have a multi-band antenna. The ground plane was

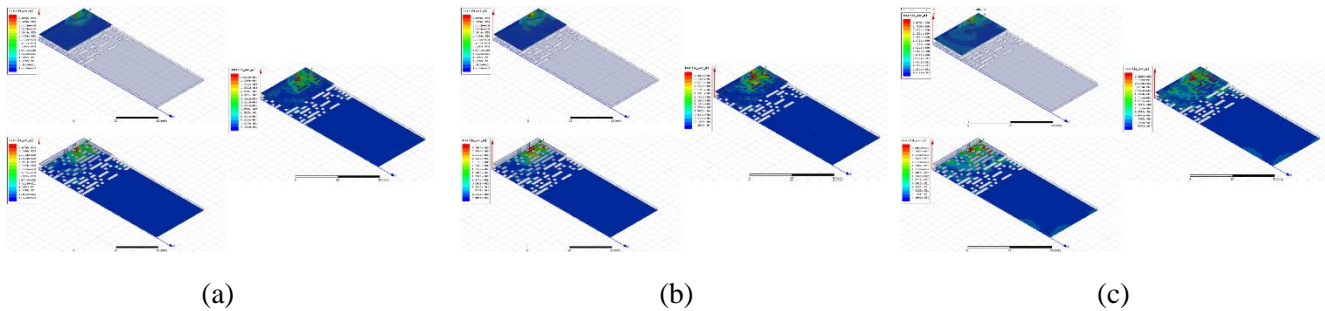


Figure 10. Current distribution of the radiating and ground planes, (a) at 1.32 GHz, (b) at 3.12 GHz, (c) at 5.2 GHz.

divided into 56 cells with the same dimensions of  $5 \times 5 \text{ mm}^2$ . Both algorithms were used to determine which of these cells will be removed to construct the slots and the cells will be kept forming the ground-plane shape. The simulation results were successful in finding two dual-band antennas with good performance in terms of bandwidth, reflection coefficient and radiation pattern.

In [26], the authors used a ground plane with multiple slots on the ground plane combined with PIFA antenna. In this case, the ground plane has two effects; the first one is to adjust the lower frequency and the second effect is to react as a parasitic element at the higher frequency. The antenna is a dual PIFA antenna for GSM and Bluetooth applications. In [27], the authors considered the use of a partial ground plane for the PIFA antenna to operate at 2.3 GHz. The motivation behind that research is to enhance the bandwidth to be useful for several applications. In this work, the semi-defected ground plane is used for PIFA antenna. The motivation behind this research is to achieve a triple band without modifying the PIFA antenna or increasing its volume. The simulation results present a very compact PIFA antenna compared to the works illustrated in Table 3.

Table 3. PIFA performances comparison with published papers in literature.

References	Used method	Antenna dimensions (mm <sup>3</sup> )	Frequency	Gain & Efficiency	Type
Proposed work	Semi-defected ground plane	38×40×1.9	1.32 3.12 5.2	2.5 dB (96%) 4.5 dB (90%) 6 dB (95%)	Triple-band PIFA
[25]	Totally defected ground plane	24×38×3	3.5 5.78	- -	Dual-band PIFA
[26]	Slots on the ground plane+ slots on radiating plane	40×15×6	0.9 1.8	- -	Double-band PIFA
[27]	Partial ground plane	20×10×4	2.3	4.98 dB	Mono-band PIFA

## 5. CONCLUSION

A very compact tripe-band PIFA antenna with semi-defected ground plane suitable for 5G applications is proposed in this paper. This antenna has dimensions of  $38 \times 40 \times 1.9 \text{ mm}^3$  with FR4-epoxy substrate. By using semi-defected ground plane and the genetic algorithm, the size is reduced by about 40%. This method presents a simple way that does not need changing the PIFA antenna's design or increasing its volume to have multi-band operation. This antenna can cover LTE Band 11 (1427.9-1495.9 MHz), HIPERLAN/2 (5.15-5.35 GHz), WLAN (5.15-5.35 GHz) and 5G Sub-6GHz applications. This antenna gives a maximum gain value of 2.5 dB, 4.5 dB and 6 dB and a radiation efficiency of 96%, 90% and 95%, at the resonant frequencies. Good radiation characteristics are shown in the operating frequency ranges.

## REFERENCES

- [1] K. L. Virga and Y. Rahmat-Samii, "Low-profile Enhanced-bandwidth PIFA Antennas for Wireless Communications Packaging," *IEEE Transactions on Microwave Theory and Techniques*, vol. 45, no. 10,

- pp. 1879-1888, Oct. 1997, doi: 10.1109/22.641786.
- [2] J.-Y. Sze and K.-L. Wong, "Slotted Rectangular Microstrip Antenna for Bandwidth Enhancement," *IEEE Transactions on Antennas and Propagation*, vol. 48, no. 8, pp. 1149-1152, Aug. 2000.
  - [3] K. K. So and K. W. Leung, "Bandwidth Enhancement and Frequency Tuning of the Dielectric Resonator Antenna Using a Parasitic Slot in the Ground Plane," *IEEE Transactions on Antennas and Propagation*, vol. 53, no. 12, pp. 4169-4172, Dec. 2005.
  - [4] S. T. Fan, Y. Z. Yin, B. Lee, W. Hu and X. Yang, "Bandwidth Enhancement of a Printed Slot Antenna with a Pair of Parasitic Patches," *IEEE Antennas and Wireless Propagation Letters*, vol. 11, pp. 1230-1233, DOI: 10.1109/LAWP.2012.2224311, 2012.
  - [5] A. D. Tadesse, O. P. Acharya and S. Sahu, "Application of Metamaterials for Performance Enhancement of Planar Antennas: A Review," *International Journal of RF and Microwave Computer-aided Engineering*, vol. 30, no. 5, p. e22154, 2020.
  - [6] M. F. Abedin and M. Ali, "Modifying the Ground Plane and Its Effect on Planar Inverted-F Antennas (PIFAs) for Mobile Phone Handsets," *IEEE Antennas and Wireless Propagation Letters*, vol.2, no. 1, pp. 226-229, 2003.
  - [7] R. Hossa, A. Byndas and M. E. Bialkowski, "Improvement of Compact Terminal Antenna Performance by Incorporating Open-end Slots in Ground Plane," *IEEE Microwave and Wireless Components Letters*, vol.14, no. 6, pp. 283-285, 2004.
  - [8] C. Picher, J. Anguera, A. Cabedo et al., "Multiband Handset Antenna Using Slots on the Ground Plane: Considerations to Facilitate the Integration of the Feeding Transmission Line," *Progress in Electromagnetics Research C*, vol. 7, pp. 95-109, DOI: 10.2528/PIERC09030605, 2009.
  - [9] L.-J. Xu, Y.-X. Guo and W. Wu, "Dual-band Implantable Antenna with Open-end Slots on Ground Plane," *IEEE Antennas and Wireless Propagation Letters*, vol. 11, pp. 1564-1567, 2012.
  - [10] H. F. Abutarboush, W. Li and A. Shamim, "Flexible-screen Printed Antenna with Enhanced Bandwidth by Employing Defected Ground Structure," *IEEE Antennas and Wireless Propagation Letters*, vol. 19, no. 10, pp. 1803-1807, Oct. 2020.
  - [11] L. Ji, P. Qin and Y. J. Guo, "Wideband Fabry-Perot Cavity Antenna with a Shaped Ground Plane," *IEEE Access*, vol. 6, pp. 2291-2297, DOI: 10.1109/ACCESS.2017.2782749, 2018.
  - [12] S. Das, P. Chowdhury, A. Biswas, P. P. Sarkar and S. K. Chowdhury, "Analysis of a Miniaturized Multiresonant Wideband Slotted Microstrip Antenna with Modified Ground Plane," *IEEE Antennas and Wireless Propagation Letters*, vol. 14, pp. 60-63, DOI: 10.1109/LAWP.2014.2354474, 2015.
  - [13] S. Ibnyaich, S. Chabaa, L. Wakrim et al., "A Pentagonal Shaped Microstrip Planar Antenna with Defected Ground Structure for Ultrawideband Applications," *Wireless Personal Communications*, vol. 124, no. 1, pp. 499-515, 2022.
  - [14] H.-Y. Li, C.-C. Lin, T.-K. Lin and C.-Y. Huang, "Low-profile Folded-coupling Planar Inverted-F Antenna for 2.4/5GHz WLAN Communications," *International Journal of Antennas and Propagation*, vol. 2014, Article ID 182927, 2014.
  - [15] I. J. G. Zuazola and J. C. Batchelor, "Compact Multiband PIFA Type Antenna," *Electronics Letters*, vol. 45, no. 15, pp. 768-769, 2009.
  - [16] F. Wang, Z. Du, Q. Wang et al., "Enhanced-bandwidth PIFA with T-shaped Ground Plane," *Electronics Letters*, vol. 40, no. 23, pp. 1504-1505, 2004.
  - [17] A. Kunwar, A. K. Gautam and K. Rambabu, "Design of a Compact U-shaped Slot Triple Band Antenna for WLAN/WiMAX Applications," *AEU-International Journal of Electronics and Communications*, vol. 71, p. 82-88, DOI: 10.1016/j.aeu.2016.10.013, 2017.
  - [18] R. Azim, M. T. Islam, N. Misran et al., "Planar UWB Antenna with Multi-slotted Ground Plane," *Microwave and Optical Technology Letters*, vol. 53, no. 5, pp. 966-968, 2011.
  - [19] X. Zhang and A. Zhao, "Enhanced-bandwidth PIFA Antenna with a Slot on the Ground Plane," *PIERS Proceedings*, pp.1268-1272, Beijing, China, 2009.
  - [20] S. Jeon, H. Choi, H. Kim et al., "Hybrid Planar Inverted-F Antenna with a T-shaped Slot on the Ground Plane," *ETRI J*, vol. 31, no. 5, pp. 616-618, 2009.
  - [21] G. Singh and J. Kaur, "Design of a Compact Superstrate-loaded Slotted Implantable Antenna for ISM Band Applications," *Sādhanā*, vol. 46, no. 3, pp. 1-10, 2021
  - [22] L. Wakrim, S. Ibnyaich and M. M. Hassani, "Optimization by Genetic Algorithm of PIFA Antenna Parameters for Wi-Fi Application," *Proc. of the IEEE International Conference on Multimedia Computing and Systems (ICMCS)*, DOI: 10.1109/ICMCS.2014.6911223, Marrakesh, April 2014.
  - [23] J. W. Jayasingh, J. Anguera and D. N. Uduwawala, "A Simple Design of Multi-band Microstrip Patch Antennas Robust to Fabrication Tolerances for GSM, UMTS, LTE and BLUETOOTH Applications by Using Genetic Algorithm Optimization," *Progress in Electromagnetics Research M*, vol. 27, pp. 255-269, DOI: 10.2528/PIERM12102705, 2012.
  - [24] S. A. Salama and A. Z. Abdalla, "Study of Smart Antenna Characteristics Using Genetic Algorithms (GA) Applications," *Proc. of the 21<sup>st</sup> IEEE Telecommunications Forum Telfor (TELFOR)*, DOI: 10.1109/TELFOR.2013.6716223, Belgrade, Serbia, 2014.

"A Semi-defected Ground Plane and a Binary Genetic Algorithm for Designing a Very Compact Triple-band PIFA Antenna", L. Wakrim, A. Khabba, J. Amadid and S. Ibnyaich.

- [25] L. Wakrim, S. Ibnyaich and M. M. Hassani, "The Study of the Ground Plane Effect on a Multiband PIFA Antenna by Using Genetic Algorithm and Particle Swarm Optimization," Journal of Microwaves, Optoelectronics and Electromagnetic Applications, vol. 15, no. 4, pp. 293-308, 2016
- [26] A. Cabedo, J. Anguera, C. Picher et al., "Multiband Handset Antenna Combining a PIFA, Slots and Ground Plane Modes," IEEE Trans. on Antennas and Propagation, vol. 57, no. 9, pp. 2526-2533, 2009.
- [27] G. Viswanadh Raviteja and V. Rajya Lakshmi, "Gain and Bandwidth Investigations of a Novel PIFA Antenna Employing Partial Ground at 2.3 GHz for WiMax/WiFi/WLAN Applications," Microwave and Optical Technology Letters, vol. 61, no. 7, pp. 1841-1844, 2019.
- [28] C. H. See, H. I. Hraga, R. A. Abd-Alhameed, N. J. McEwan, J. M. Noras and P. S. Excell, "A Low-profile Ultra-wideband Modified Planar Inverted-F Antenna," IEEE Transactions on Antennas and Propagation, vol. 61, no. 1, pp. 100-108, DOI: 10.1109/TAP.2012.2216494, 2013.
- [29] N. Kumar and G. Saini, "A Novel Low profile Planar Inverted-F Antenna (PIFA) for Mobile Handsets," International Journal of Scientific and Research Publications, vol. 3, no. 3, pp. 1-4, 2013.
- [30] W. A. Awan, A. Zaidi, M. Hussain et al., "The Design of a Wideband Antenna with Notching Characteristics for Small devices Using a Genetic Algorithm," Mathematics, vol. 9, no. 17, p. 2113, 2021.

### ملخص البحث:

في هذه الورقة، نقترح هوائياً مُدمجاً ثلاثي النطاقات على شكل F مقلوب (PIFA) للتطبيقات المتنقلة واللاسلكية باستخدام خوارزمية جينية ثنائية مع الإبطال الجزئي للمستوى الأرضي. تبلغ أبعاد الهوائي المقترح (38\*40\*1.9) مم، وهو مخصص للتطبيقات المتعلقة بالنطاقات الترددية LTE11 و HIPERLAN/2 و WLAN، والنطاق الفرعي للجيل الخامس (6 جيجاهيرتز). ولضمان الحصول على هوائي ثلاثي النطاقات يعمل بأداء مقبول، استُخدمت الخوارزمية الجينية لتحديد شكل المستوى الأرضي للهوائي. وأثبتت النتائج أن الهوائي المقترح مثالي من حيث الأداء عند ثلاثة ترددات 1.32 و 3.12 و 5.2 جيجاهيرتز، على الترتيب، وبمقارنة التصميم المقترح مع تصاميم سابقة لهوائيات مماثلة، أثبتت المقارنة نجاعة التصميم المقترح.

# TOWARD DEVELOPING AN INTELLIGENT PERSONAL ASSISTANT FOR TUNISIAN ARABIC

Inès Zribi<sup>1</sup> and Lamia Hadrach Belguith<sup>2</sup>

(Received: 6-Jun.-2022, Revised: 9-Aug.-2022, Accepted: 27-Aug.-2022)

## ABSTRACT

*Intelligent systems powered by artificial intelligence techniques have been massively proposed to help humans in performing various tasks. The intelligent personal assistant (IPA) is one of these smart systems. In this paper, we present an attempt to create an IPA that interacts with users via Tunisian Arabic (TA) (the colloquial form used in Tunisia). We propose and explore a simple-to-implement method for building the principal components of a TA IPA. We apply deep-learning techniques: CNN [1], RNN encoder-decoder [2] and end-to-end approaches for creating IPA speech components (speech recognition and speech synthesis). In addition, we explore the availability and free-dialog platform for understanding and generating the suitable response in TA for a request. For this proposal, we create and use TA transcripts for generating the corresponding models. Evaluation results are acceptable for the first attempt.*

## KEYWORDS

*Intelligent personal assistant, Tunisian Arabic, Speech recognition, Natural-language understanding, Dialog management, Response generation, Speech synthesis.*

## 1. INTRODUCTION

Technological progress has made a large number of advanced application technologies. Among them, we cite smart systems, including spoken-dialog systems and especially the Intelligent Personal Assistant (IPA). As illustrated by [3], IPA is a speech-compatible software that can be found on a specialized device (e.g. Amazon Echo, Google Dot), a mobile device or a computer. It assists the user by answering questions in natural language, giving suggestions, DOing tasks, etc. Nowadays, IPAs are becoming essential in human lives and have a powerful effect on our everyday lives. They are able to replace humans in some ordinary cases that are repetitive in nature and can be easily automated, including providing flight information, sport results, weather forecasts, share prices, booking hotels, renting cars, etc. [4]. IPAs are designed to accept spoken dialog, which is a natural mode of communication, or typed input in a natural language [5]. Some of them give responses to queries by voice and/or text messages. The architecture of most of them is based principally on five principal modules [6]: speech-recognition module (SR), natural-language understanding module (NLU), dialog-management module (DM), natural-language generation module (NLG) and finally, speech-synthesis module (SS). The quality of an assistant is based principally on the quality of each component. Figure 1 presents the basic architecture of voice IPA inspired from [6].

Apple's Siri, Amazon's Alexa, Google Assistant and Cortana from Microsoft are the most popular and used IPAs developed to help users do some usual and simple-to-complex tasks. They are now a signature feature of some smartphones and tablets. There is also a set of free and open-source assistants, such as Mycroft Core<sup>1</sup>, Open Jarvis<sup>2</sup>, etc. With the development of deep-learning techniques, many researchers have developed specialized IPAs. Some of them have an object to build a social relation with the user [7]. The IPA determines user's goals and preferences, so that it can recommend conferences to attend and people to meet. Some other IPAs have a goal to check a patient's health indicator [8], support human operators to empower operators in industry environments etc. [9]-[10]. Despite the continuous development of this type of technology, IPAs cover a limited set of languages. They differ from one another. Due to the variety and differences of language dialects, each dialect needs a distinct linguistic model [11]. Therefore, only some dialectal forms are also considered by some IPAs. English, French and Chinese are the most commonly treated languages by the majority of IPAs. However, their

<sup>1</sup> <https://mycroft-ai.gitbook.io/docs/mycroft-technologies/mycroft-core>

<sup>2</sup> <https://openjarvis.com/>

1. I. Zribi is with MIRACL Laboratory, Sfax University, Sfax, Tunisia. Email: [ineszribi@gmail.com](mailto:ineszribi@gmail.com)

2. L. H. Belguith is with Faculty of Economics and Manag. of Sfax, Sfax Uni., Tunisia. Email: [lamia.belguith@fsegs.usf.tn](mailto:lamia.belguith@fsegs.usf.tn)

performances deteriorate when Arabic is the used language. Table 1 presents the languages treated by the four IPAs: Cortana, Siri, Alexa and Google IPAs. We remark that Modern Standard Arabic (MSA) is considered by some IPAs, while the colloquial form is neglected. Dialectal Arabic (DA) is mainly spoken and used in daily communication. It is used in chat, utilities, radio, phone conversations and so on. As a rule, Arabs are unable to speak the standard form of their language on a day-to-day basis. Therefore, they interact with IPAs using a foreign language (e.g. English for Anglophone persons or French for Francophone persons). So, it is important to develop a spoken-dialog system able to understand the DA.

In this paper, we investigate the possibility to build an Intelligent Personal Assistant for dialectal Arabic, especially, Tunisian Arabic (TA). We explore a simple-to-implement method for building TA IPA components with the availability and free resources (corpus, GPU, APIs, etc.). We apply deep-learning techniques: CNN [1], RNN encoder-decoder [2] and end-to-end approaches for creating IPA speech components (SR and SS). In addition, we use the available and free-dialog platform for understanding and generating the suitable response in TA for a request. For this proposal, we build about 5 hours of TA speech corpora composed of IPA requests. To the best of our knowledge, our work is the first attempt to build IPA-system components for TA. Indeed, no work has been done to building speech synthesis and language understanding and generation for TA. Furthermore, only TA transcripts are used for generating the different models.

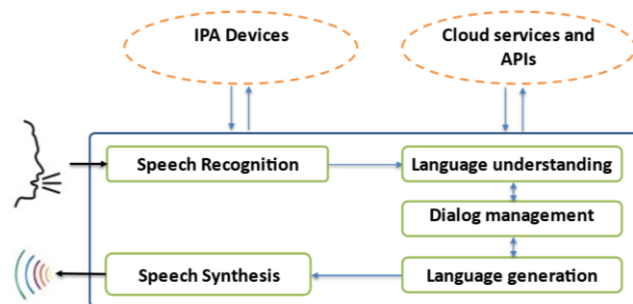


Figure 1. Basic architecture of voice IPA.

This paper has five main sections. Section 2 describes our motivations and some challenges in building an IPA for TA. Section 3 presents an overview of previous work that studied the building dialog systems and the speech components of an IPA. In Section 4, we present our proposed method and the evaluation results. Finally, in Section 5, we present the conclusion and expose our future directions.

## 2. BACKGROUND

### 2.1 Motivations for Developing IPA for Tunisian Arabic

#### 2.1.1 Tunisian Arabic

In this paper, we intend to develop components for an IPA that understands and speaks Tunisian Arabic (TA); a dialect of the North African (i.e., the Maghrib) dialects spoken in Tunisia by approximately twelve million people [12]. Although TA is mainly spoken, it is written in social networks, blogs, some novels, as well as in comics, commercials, some newspapers and popular songs. It was influenced by other languages, such as Berber, French, MSA, Turkish, Italian, Maltese, etc. [12]. This is a result of the position of Tunisia between the two continents (Africa and Europe), as well as the variety of civilizations that ruled it and its openness to neighboring cultures. However, code-switching (between MSA, French and TA) is the main characteristic of the TA [13]. For example, the sentence presented in Table 2 is composed of two French phrases “C’est vrai” and “donc”, a TA phrase “اللي المدير أستاذ أما ما ينجمش” and an MSA phrase “لا يصلح لتسيير الشركة”. This phenomenon allows the introduction of new words (nouns and verbs) derived from foreign languages (e.g. ييراتجى (ybrAtjý)<sup>3</sup> ‘He shares’ derived from the French verb “partager”). Indeed, there are many differences as well as similarity points between TA, Arabic dialects and MSA at different levels: lexical, morphological,

<sup>3</sup> Transliterations of Arabic words are presented in the HSB scheme [83] and are presented between (...). Phonological transcriptions are presented between slashes /.../.

syntactic and phonetic (for more details, see [12]). In addition, TA is distinguished by the presence of words from several other languages. The standard form of the Arabic language is used for some IPAs. While the MSA is the official language of all Arab countries, Arabs do not use the standard form in their daily communications. Colloquial Arabic or dialectal Arabic (DA) is the mother tongue spoken daily by everyone [12].

Table 1. Languages and dialects recognized by devices of four IPAs: Cortana, Siri, Alexa and Google.

Languages	Cortana	Siri	Alexa	Google
English	S + D <sup>4</sup>	S + D	S + D	S + D
Portuguese	S	S	S	
French	S + D	S + D	S + D	S + D
German	S	S + D	S	S + D
Italian	S	S + D	S	S
Spanish	S + D	S + D	S + D	S + D
Chinese	S + D	S + D		
Japanese	S + D	S	S	S
Arabic		S		
Turkish		S		
Thai		S		
Swedish		S		S
Russian		S		
Norwegian		S		S
Hebrew		S		
Korean		S		S
Malay		S		
Cantonese		S		
Danish		S		S
Dutch		S + D		S
Finnish		S		
Hindi			S	S

Table 2. Example of TA sentence.

<b>TA</b>	C'est vrai الذي المدير أستاذ أما ما ينجمش بيراتجى ويوصل المعلومة ولا يصلح لتسيير الشركة
<b>Transliteration</b>	C'est vrai Ally Almdyr ĀstAð ĀmA mA ynjmš ybrAtjý wywSl Almçlwmħ donc lA ySIH ltsyyr Alšrkħ
<b>Translation</b>	<i>It's true that the director is a professor, but he can't share information; so, he is unfit to manage the company.</i>

### 2.1.2 Importance of Using Dialectal Arabic for an IPA

As we said before, an IPA is a software solution designed to help people in their everyday lives to do multiple tasks, going from simple (e.g. checking mail, making calls, etc.) to complex tasks related to smart home (e.g. opening TV, etc.). To be useful to users, the communication mode should be simple. The Arabic language that is currently supported by some IPAs (e.g. Siri on Apple) represents the standard form of Arabic, while some Arabs (not to say the majority of Arab people) are not fluent in MSA for many reasons. Therefore, they even use an IPA in foreign languages (e.g., French for French speakers, English for English speakers, etc.). Indeed, the absence of the colloquial form of Arabic in all APIs makes the use of such types of technologies relative only to intellectualized people. Therefore, the design and development of an IPA speaking dialectal Arabic will help Arabs interact easily and encourage them to use an intelligent assistant.

## 2.2 Challenges in Building Tunisian Arabic IPA

### 2.2.1 Rarity of Resources

The presence of spoken and annotated corpora is important for building an IPA. The automatic processing of TA is a new area of research. Unlike MSA, TA suffers from the scarcity or even the

<sup>4</sup> 'S' means the standard form of the language and 'D' means the dialectal form of the language.



absence of freely available corpora. The few TA resources developed over the last few years are still in the early stages. Their size is relatively limited compared to that of MSA. The only TA spoken corpus accessible is that of [14].

### 2.2.2 Ambiguity

Like MSA, TA is characterized by ambiguity at many levels: phonological, morphological, semantic and syntactic. A word or an expression can be understood differently based on the context. For example, the greeting expression السلام عليكم (AlslAm ȅlykm) is also used for 'the goodbye'. Also, the word باهي (bAhy) can mean 'ok' or 'good'. Ambiguity affects the performance of TA IPA because of the confusion in understanding some questions and/or answers. Some cases of ambiguity can be easily addressed, while others require complex disambiguation methods to improve TA IPA achievement. A few works [15] considered the task of disambiguation by TA. This issue complicates the task of building an IPA for TA.

### 2.2.3 Sub-dialectal Variation

TA is characterized by the existence of many sub-dialectal varieties [15]. The sub-dialects differ at many levels. The same word is pronounced in different ways (e.g. بقرة (baqrah) 'cow' is pronounced as /bagra/ and /baqra/). The phonological differences complicate the development of speech recognition and speech synthesis for TA. Similarly, the sense of a word differs from one sub-dialect to another. For example, the word ربح (rbH) means in some sub-dialects 'salt' and in others 'benefit'.

### 2.2.4 Code Switching

Code switching between TA and other languages causes many problems for the development of TA IPA. First, the SR component is not able to distinguish words in TA from other languages. As a result, it transcribes all words using the same script. This creates ambiguities for the NLU module, because a word in French transcribed in Arabic letters can have a different meaning. For example, the French word "merci" 'thank you' transcribed in Arabic letters can refer to a person's name مرسي (mrsy) 'Morsi'. Tunisians also use some French words in everyday communication without any modification (e.g. "mécanicien"). This can cause, also, some problems for the SS module.

## 3. RELATED WORKS

### 3.1 Dialog System

In general, dialog systems (IPA, chatbot<sup>5</sup>, etc.) can be classified into task-oriented systems and task-non-oriented systems [16]. Task-oriented systems (e.g. IPA) try to help the user achieve certain tasks. Task-non-oriented systems (e.g. chatbot) talk to the user to provide responses and entertainment. While developing a dialog system, four methods are proposed for understanding and generating language according to its goal. For the first type of dialog system; oriented task, Chen et al. (2018) [16] have classified methods into two categories: pipeline methods and end-to-end methods.

The typical structure of a pipeline methods consists of four key components:

- The language-understanding component (NLU) parses the user utterance into pre-defined semantic slots. It classifies the user's intent and the utterance category into one of the pre-defined intents. The NLU component extracts important information, such as named entities and fills the slots. Deep-learning techniques are successfully applied in intent classification. Hashemi et al. [17] have applied Conventional Neuronal Network (CNN) in intent classification, while Sreelakshmi et al. [18] have used Bi-Directional Long Short-Term Memory (Bi-LSTM) networks for intent identification. Slot filling and named-entity extraction are important tasks for NLU components. Deep-belief networks (DBNs) are usually used by some researchers, like [19]. CNN has also been exploited in slot filling by [20]. Pre-trained BERT and BiLSTM have been employed by [21] for intent and argument detection.

<sup>5</sup> A chatbot is a program that allows a human-computer conversation to be conducted *via* auditory or textual methods using natural language[40], [84]. It operates almost as an IPA.

- The dialog state tracker is the main component in a dialog system. It divines the objective of each turn of dialog. For tracking dialog state, [22] exploited rule-based methods. [23] and [24] made use of statistical and deep-learning techniques.
- Dialog policy learning learns the next action based on the current dialog state. Like in previous components, rule-based [25], statistical and deep-learning approaches [26] have been applied.
- Natural Language Generation (NLG) is responsible for generating the response. As illustrated in [16], conventional approaches are widely used in NLG. It transforms the input (i.e., semantic symbols) into an intermediary structure (such as tree-like or template structures) and then, the intermediate structure is transformed into the final response [27]. Deep-learning techniques, such as LSTM-based structure, are proposed by [28] and [29] to NLG. Wen et al. [28] used a forward RNN generator together with a CNN re-ranker and a backward RNN re-ranker [16]. Zhou et al. [30] adopted an encoder-decoder LSTM-based structure to generate correct answers based on the question information, semantic slot values and dialog act type. The sequence-to-sequence approach is used by [31]. It can be trained to produce natural-language strings as well as deep syntax-dependency trees from input dialog acts. Recurrent neural-network language generation (RNNLG) is proposed by [32]. It can learn to generate statements directly from dialog-act pairs of statements with no pre-defined syntax and no semantic alignment.

End-to-end approaches to develop dialog task-oriented systems have been proposed and used by several researchers [33]–[35]. They have combined several methods, like an encoder-decoder model, an end-to-end reinforcement learning technique, an attention-based key-value retrieval mechanism, etc. All of the end-to-end methods use a single module and interact with structured external databases. The input of the model is the user request and the output is the response.

In each presented approach, there are multiple used techniques: parsing, pattern matching, Artificial Intelligence Markup Language (AIML), chatscript, ontologies, Artificial Neural Network Models, etc. Several commercial, free platforms, APIs and libraries have used these techniques for understanding natural language, dialog management and language generation in order to develop conversational systems. Among these platforms, we cite Dialogflow<sup>6</sup> from Google, IBM Watson Assistant<sup>7</sup>, Pandorabots<sup>8</sup>, Rasa [36], etc. Some of these platforms are exploited in developing some Arabic chatbots for MSA [37], [38] and Colloquial Arabic (BOTTA [39], Nabiha [40], etc.). The majority of previous efforts in creating Arabic dialog systems have been focusing on developing task-non-oriented dialog systems. In contrast, in this work, we focus on building a task-oriented dialog system using Rasa platform, which is able to understand TA and DOIng some simple tasks. Furthermore, to the best of our knowledge, there is no work dealing with creating a TA task-oriented dialog system or developing Tunisian IPA, where our work is the first one.

### 3.2 Speech Components

We present in this sub-section some work proposed for Speech Recognition (SR) and Speech Synthesis (SS) for Latin and Arabic languages. As defined by [41], SR is an automatic way to transcribe speech into text. It is used to make machines understand human speech. SS has the inverse task. It transforms text into voice. In general, the SR module receives an IPA user request and the SS gives the response to the user.

#### 3.2.1 Speech Recognition

Like the dialog system, the automatic speech recognition (ASR) can be classified into two approaches: conventional ASR pipeline approach and end-to-end ASR approach. The conventional ASR pipeline includes trained acoustic, pronunciation and language model components which are trained independently. It regroups classical methods: (1) rule-based methods that use phonetic rules in order to convert graphemes into phonemes, (2) probabilistic and data-driven methods ([42], [43], etc.), which are based on a phonetic dictionary, acoustic models and feature-extraction step. These methods utilize Hidden Markov Models (HMMs), Conditional Random Fields (CRFs), Deep-learning techniques, etc. to generate acoustic models and extract features. End-to-end voice recognition goes a long way to

<sup>6</sup> <https://dialogflow.cloud.google.com/>

<sup>7</sup> <https://developer.ibm.com/articles/introduction-watson-assistant/>

<sup>8</sup> <https://www.pandorabots.com/>

simplify the complexity of traditional speech recognition. It is based on deep-learning techniques. No preliminary steps (phonetic rule construction, acoustic dictionary, feature extraction, etc.) are required. In this approach, we need a set of records and their transcriptions and the deep neural network can automatically learn language or pronunciation information. Among the works conducted using this approach, we cite [44]–[49].

Few works are conducted in Arabic due to the scarcity of transcribed speech corpora. [50], [51], etc. have proposed and tested SR classical methods which are based on HMMs. The broadcast news-transcription system proposed by [50] has two main components: an audio partitioner and a word recognizer. Data partitioning is based on an audio stream-mixture model and divides the continuous stream of acoustic data into homogeneous segments [50]. The second component of the proposed system determines the sequence of words for each speech segment. The recognizer utilizes continuous-density HMMs for acoustic modeling and the n-gram statistics for language modeling. Lamel and Gauvain [50] have developed a pronunciation lexicon based on a grapheme-to-phoneme tool. It contains 57k distinct lexical forms from 50 hours of manually transcribed vocalized data. The language model contains up to 4-gram. The same method has been adopted by [51]. Their proposed system is based on Carnegie Mellon University's Sphinx tools. It uses 3-emitting state HMMs for triphone-based acoustic models. The system was trained on 4.3 hours of Arabic broadcast news corpus and tested on 1.1 hour. The phonetic dictionary contains 23,841 definitions, corresponding to about 14,232 words. The language model contains both bi-grams and tri-grams. The same approach has been adopted by [42] for recognizing TA speech. The author has developed a phonetic dictionary for TA based on the CRF algorithm. Then, the dictionary is integrated into the MSA SR system. Ben Ltaief et al. [52] have described an SR system for TA based on the Kaldi toolkit. They have built 2 acoustic models: HMM-GMM (Gaussian mixture models) and HMM-DNN and have trained 3-gram models. Their models are based on the TARIC corpus [53]. Messaoudi et al. [54] have exploited the DeepSpeech architecture [2] to generate a model to recognize TA speech. They have exploited four Arabic corpora (two TA corpora and two MSA corpora) for generating language and deep-learning models. We note that these SR models are not available for testing and using.

The majority of previous efforts in creating Arabic ASR have been focusing on using conventional traditional pipeline. The only work done for TA based on end-to-end approach is proposed by [54]. In this work, we propose to follow the same approach. The main difference between our work and that of [54] is the nature of our corpus, which is based only on TA corpus.

### 3.2.2 Speech Synthesis

There are principally two types of speech-synthesis approaches. Classical approaches are based on Concatenate Speech Synthesis (CSS) and HMM techniques. The principle of CSS is to generate speech by concatenating its units one after the other [55]. The generation requires a corpus composed of utterances with well-annotated phones. The quality of generated speech is related to the quality of the collected corpus. The HMM synthesis techniques, called statistical parametric synthesis of speech [55], extract parameters from the recorded utterances which are then used to generate speech. The quality of the generated voice is maintained, even if the size of the training corpus is small. The classical approaches are used in SS for several languages, such as English ([56], [57]), French ([58], [59]), etc. and MSA ([60]–[63], etc.).

The second approach is based on deep neural architectures that have proved successful at learning the fundamental features of data [64]. Several architectures are proposed. Among the most famous ones, we cite WaveNet [65]; a deep generative model of audio data that operates at the waveform level. The application of this model to SS shows that produced samples surpass many SS systems in subjective naturalness. However, it has some drawbacks. First, the model is not a full end-to-end system. Second, the generation of speech is very slow [64]. Deep voice [66] is another deep neural architecture used for SS. It is an end-to-end neural architecture. Traditional text-to-speech pipelines inspire it, but its components are replaced with neural networks. It is simpler than classical approaches. Any human involvement is required for deep voice model training. Tacotron [67] is another end-to-end architecture for SS. It is a generative model based on a seq.-to-seq. model with an attention mechanism [68] that produces audio waveforms directly from the characters. Tacotron automated some SS tasks, such as

feature engineering and human annotation. Tacotron 2 is an improved version of Tacotron proposed by [69]. It eliminates non-neural network components used to synthesize speech, such as the Griffin-Lim reconstruction algorithm [64]. Shen et al. [69] have used hybrid attention [70] with a recurrent seq.-to-seq. generative model and a modified wavenet acting as a vocoder to synthesize speech signals [64].

The deep approach was proposed and tested for several languages, such as English. In the last few years, a few researchers have tested some architectures for MSA. Tacotron 2 [69] has been tested for a vowel MSA corpus by [64]. Hadj Ali et al. [71] have tested DNN for the task of grapheme-to-phoneme conversion using diacritized texts. Abdelali et al., [72] have also tested Tacotron [67], Tacotron 2 [69] and Model ESPnet Transformer TTS [73] in the Arabic language. To the best of our knowledge, there is no work being done for TA and our work is the first one developing an SS for TA. In Table 3, we present a comparison between speech works (SR and SS) done for Arabic language.

Table 3. Comparison between different Arabic speech systems.

	Ref. No.	Approach	Classification method	Used dataset	Result	MSA/TA
Speech Recognition	[50]	Conventional pipeline	HMM + n-gram	1200 hours of broadcast news data	WER = 0.209	MSA
	[42]	Conventional pipeline	GMM-HMM model + n-gram	10 hours of TA	WER = 0.226	TA
	[51]	Conventional pipeline	HMM	4.5 hours of Arabic TV news	WER = 0.09	MSA
	[52]	Conventional pipeline	HMM- DNN + HMM-GMM	10 hours of TA	WER= 0.368	TA
	[54]	End-to-end	RNN encoder-decoder	61 hours and 34 minutes of MSA and TA	WER = 0.244	MSA + TA
Speech Synthesis	[63]	Classical approach	HMM	598 utterances	MOS = 4.86	MSA
	[64]	Deep approach	Sequence-to-sequence architecture + flow-based implementation of WaveGlow	2.41 hours	MOS = 4.21	MSA
	[71]	Deep approach	DNN	1597 utterances	MAE <sup>9</sup> = 19	MSA
	[72]	Deep approach	Model ESPnet Transformer	9969 utterances male and female voices	MOS = 4.40	MSA

#### 4. TUNISIAN IPA COMPONENTS

An IPA usually operates by the following these steps. First, when it is on and is not used for a certain time, it goes into a “listening mode”. When the user calls the IPA by pronouncing the Trigger Word (TW) (e.g. “Alexa”, “Siri”, “Hey Google”, etc.), the latter wakes up. It waits for the user’s request. Then, the IPA accomplishes the requested task and gives vocal response to the user. Finally, it goes back into the listening mode. Hence, we propose to build two SR modules. The first one (SR-TW), based on a Convolutional Neural Network [1], is responsible for detecting the trigger word (TW). When it is recognized, the second module (SR-R), based on the DeepSpeech architecture [2], is activated for receiving and transcribing users’ requests. Once the request is received, the Language Understanding module is activated. It is responsible for classifying the intents and detecting the entities. The latter are used by the Dialog Management model to decide the next action to do. Then, the Language Generation module prepares and generates the suitable response. For these three components, we propose to apply the RASA dialog framework to generate the response by following the dialog history of the user and generate the response according to user intention. It also accomplishes the requested task. Finally, the generated response will be sent to the Speech Synthesis (SS) model in order to generate the corresponding voice. We apply the Tacotron 2 [69] model to the TA. Figure 2 presents the architecture of our IPA. We note that these components are dedicated to recognizing and understanding the commands of the users relative to four basic IPA skills: greeting and knowledge, weather forecasts, checking email and asking time and date. We present, in the rest of this section, the details of our proposed method.

<sup>9</sup> Mean absolute error is a measure of errors between paired words expressing the same speech.

## 4.1 Speech Recognition

### 4.1.1 Proposed Method

We present in this sub-section our proposed methods for two SR modules. The first one, Speech Recognition-Trigger Word (SR-TW), is responsible for detecting the trigger word (TW). The second module, Speech Recognition-Request (SR-R), is responsible for transcribing users' requests.

**Speech Recognition-Trigger Word (SR-TW):** In order to activate the IPA, a TW should be said by the user. We propose to generate a model that classifies short sounds (1 second) into two classes: TW and non-TW. For classification, we apply the deep neural network, in particular the Convolutional Neural Network (CNN). Its architecture is composed of eight hidden layers: an input layer, four convolutive layers followed each one by a pooling and drop layer, one flatten layer, two dense layers followed each one by the dropout layer and finally, an output layer. This architecture is often used for recognizing and classifying speech. "Hey Cortana", "Hey Google" and "Alexa" are some of the TWs, respectively, used by Microsoft Cortana, Google and Alexa IPAs. We have chosen *عالسلامة* (ǧAlslAmh) 'hello' as a TW.

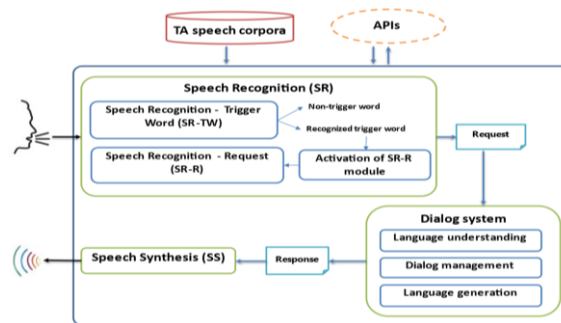


Figure 2. Architecture of our IPA.

**Speech Recognition-Request (SR-R):** For recognizing user requests, we applied the deep-learning architecture proposed by [2], baptized DeepSpeech. We chose to apply this architecture, because it has shown its efficacy for several languages (English [2], Mandarin [74], German [47], etc.). This architecture also has shown its efficacy for TA [54]. We note that the SR module proposed by [54] is not available for testing and using.

DeepSpeech is also robust when it is applied in noisy environments [2]. With deep learning, we do not need to do the extraction of features or generate the phonetic dictionary. The DeepSpeech architecture is composed of five hidden layers [2]. The three first layers of non-recurrent  $h_t^{(1-3)}$  (dense) are fully connected and computed by the ReLu activation function. The fourth layer is a bi-directional recurrent layer. It includes two sets of hidden units: a set with the forward recurrence  $h_t^{(f)}$  and a set with the backward recurrence  $h_t^{(b)}$ . The fifth (non-recurrent)  $h_t^{(5)}$  layer takes both the forward and backward units as inputs. The output layer is a standard softmax function that yields the predicted character probabilities for each time slice  $t$  and character  $k$  in the alphabet. Further, the Connectionist Temporal Classification (CTC) loss function is used to maximize the probability of correct transcription [75].

We do not modify the architecture of DeepSpeech, because it has shown its performance for complex languages with a large number of characters, like Mandarin [45]. For training an automatic SR system based on a DeepSpeech system [2], two main components are used: a Recurrent Neural Network (RNN) and a language model. We used a set of audio files with their corresponding transcriptions in order to train the RNN model. For the language model, we have used KenLM to generate an n-gram model [76]. We have used the same values of parameters as proposed in [54]. We have generated a 3-gram model with an alpha value of 1 and a beta value of 1.5. We have used an alphabet composed of 38 characters and omitted the short vowels of the alphabet.

### 4.1.2 Dataset

We exploited in the development of IPA components the freely available spoken corpora for TA. To the best of our knowledge, the Spoken Tunisian Arabic Corpus (STAC) [14] is the only publicly available

spoken corpus. It is composed of 5 transcribed hours collected from different Tunisian TV channels and radio stations. It contains spontaneous speech, less spontaneous speech and prepared speech and a large number of speakers (about 70 speakers) in order to make the dataset a representative sample of the TA. We have exploited a part of STAC (2 hours and 31 minutes). Some pre-processing steps are done on this corpus. First, we have removed all types of annotations, such as disfluencies, etc. We have, then, corrected some orthographic errors. We have corrected them according to the CODA convention [12]. Next, we have removed unclear speeches, music, superposed sounds and long pauses. After that, we have subdivided the audio files into small audio waves (less than or equal to 10 seconds). Finally, we have converted files into wave format with a mono audio channel and a sample rate of 16,000 Hz in order to be read by the DeepSpeech pipeline. We obtained, after pre-processing, 1 hour and 56 minutes of pure transcription.

We recall that our objective is to build an IPA for TA. So, we have enriched the corpus with some transcriptions of an IPA user command, such as greeting and acknowledgement, providing weather forecasts, asking for the date and time and finally, checking new email. We have recorded commands for 28 persons (3 men and 25 women). We have augmented the transcriptions by adding noise and modifying the pitch. We have obtained a total of 50 minutes. We have also enriched the corpus with some other transcriptions (Tunisian dialect stories and some read chapters from the Tunisian constitution in TA). The total size of these transcriptions is 1 hour and 27 minutes. We have augmented this corpus by adding noise and modifying the pitch (down and up) of its files. Table 4 summarizes the details of our corpus.

Table 4. Size of our corpus used in SR modules.

Corpus	Size
A part of STAC	1 hour and 56 minutes
IPA corpus	50 minutes
Other transcriptions	1 hour and 27 minutes
Augmentation	3 hours and 24 minutes
<b>Total</b>	<b>7 hours and 37 minutes</b>

For training and testing SR-TW, we have collected, from the corpus, transcriptions that contain the word *عالسلامة* (çAlslAmħ) 'hello'. The duration of each transcription is about 1 second. We have collected multiple pronunciations (10 persons) of the trigger word. We have augmented the corpus by adding noise and modifying the pitch. We obtained about 16 minutes of different pronunciations of the trigger word. For the class non-trigger word (NTW), we have collected different sounds that have a duration equal to 1 second, pronouncing different words in TA. We have obtained about 33 minutes. To train and test the TW-SR model, we have divided the corpus into 70-30%. In contrast, we used the division 80-10-10% for the training, validation and testing of the SR-R models. For generating n-gram models, we have exploited the TA corpora used in [77], composed of 260,364 words.

#### 4.1.3 Evaluation

To measure the performance of our module, we have calculated the Word Error Rate (WER) and the Character Error Rate (CER) for SR-R. A Lower WER respectively CER is often used to indicate that the Speech Recognition model is more precise in recognizing speech. A higher WER respectively CER, then, often associated with a lower accuracy. Since the SR-TW classifies short sounds into TW and Non-TW, we have calculated the accuracy measure to test the accomplishment of this component. Formulae of the following measures are presented below, where  $N_w$  is the number of words in reference text,  $S_w$  is the number of words substituted (a word in the reference text is transcribed differently),  $D_w$  is the number of words deleted (a word is completely missing) and  $I_w$  is the number of words inserted. We note that the formula for CER is the same as that of WER, but CER operates at the character level instead. Table 5 presents the results of the two models. The accuracy value of SR-TW is an encouraging result. The errors are related to some homophones, such as *عالسلامة* (çAlslAmħ) 'hello' et *بالسلامة* (bAlslAmħ) 'bye'. For SR-R, the evaluation results of [54] are better than our results. This is due to the size of their used corpus that contains STAC corpus and other speech TA corpora. By analyzing the results, transcription errors are caused by the insertion of some extra letters. The presence of homophones, disfluencies, etc. are the principal causes of failure cases.

$$(1) \text{ Accuracy} = \text{Number of correct predictions} / \text{Total number of predictions}$$

$$(2) WER = \frac{(Sw + Dw + Iw)}{Nw} \quad (3) CER = \frac{(Sc + Dc + Ic)}{Nc}$$

Table 5. Evaluation results of the two SR models.

Model	WER	CER	Accuracy
SR-TW	-	-	0.97
SR-R	0.41	0.30	-
Tunisian DeepSpeech [54]	0.322	0.204	-

## 4.2 Natural Language Understanding, Dialog Management and Response Generation

### 4.2.1 Proposed Method

Over the last decade, there has been a focus on using statistical and machine-learning methods in language understanding, dialogue management and language generation rather than traditional technologies (i.e., rule-based method). Indeed, we propose to apply a statistical method to understand requests, manage dialogs, generate a suitable response and do the task. Therefore, we have used the Rasa framework [36]: an open-source framework which allows developers to create a machine learning-based conversational system (especially a chatbot). Rasa proposes two main modules: Rasa NLU and Rasa Core. Rasa NLU analyzes the user's request. It classifies it based on the appropriate intent and then extracts the entities. Rasa Core chooses the action that the dialog system should take based on the output of the Rasa NLU (structured data in the form of intents and entities) using a probabilistic model. Rasa leads to creating and generating models DOing simple and complicated tasks in an efficient way, even with minimal initial training data [36]. It regroups a set of components that make up the NLU pipeline (tokenization, entity extraction, intent classification, response selection, pre-processing and more) and works in succession to process the user input into a structured output. It also has a set of policies that manage conversation actions. Both policies and components are based on machine learning (e.g. SVM, CRF, RNN, LSTM, etc.) and rule-based techniques. We have chosen to use Rasa for many reasons. First, it is free and an open-source tool. It can run locally [78]. Second, the use, implementation and bootstrapping of Rasa are relatively easy [79]. Since Rasa NLU does not support the Arabic language, we have applied the pre-configured NLU pipeline. It is composed of eight components. Rasa Core uses policies to decide the next action in a dialog conversation. It provides rule-based and machine-learning policies. In our work, we have also used pre-configured policies. Figure 2 presents the components of the pre-configured pipeline and pre-configured policies. The full description of the components and policies is presented in the documentation of Rasa [80].

Our training data is composed of a list of messages that IPA expects to receive. These messages are annotated with intents and entities that the RASA NLU learns to extract. As we said before, our IPA is limited to four services: "greeting and knowledge", "weather forecasts", "checking email" and "asking time and date". Therefore, our corpus includes intents for these services. We added other basic intents; namely, "affirm", "goodbye", "thanks", "person identification" and "city identification" to ensure a good conversation. Our training data also contains a set of responses that the user expects to receive. We have defined five types of responses: "bye", "end", "start", "first conversation" and "thanks". We have added four customizable responses to the intents: "ask mail", "provide weather forecasts", "person identification" and "ask date and time". We identified and annotated nine entities; namely, "date", "component of the date", "mail", "person's name", "time", "Tunisian city", "weather specification", "weekday" and "hijri date". In addition, the data contains a list of entities' synonyms. Table 6 and Table 7 present, respectively, some examples of intents and entities and IPA responses.

The main function of the Tunisian IPA is to provide answers to several inquiries about the weather, time, date and email box. We have prepared several possible stories that simulate a real conversation between a user and an IPA. A story is a representation of a conversation between a user and an IPA transformed into a particular format. The user request is expressed as intent (entities when necessary) and the assistant's responses and actions are expressed as action names [80]. Stories are used to train models that are able to generalize to unseen conversation paths. We identified 24 possible stories for requesting services in Tunisian. Figure 3 presents an example of a story. It is composed of a set of user requests (i.e., intent: greet, intent: ask\_email and intent: thanks) and actions which the IPA should do (i.e., action: utter\_start, action: action\_mail, action: utter\_thanks.). In a story, we mark entities which the IPA should

identify and save. The attribute “slot\_was\_set” is used to this end. Table 8 presents a real example of the story presented in Figure 3. For generating the suitable response for some intents (i.e., provide\_weather\_forecasts, ask\_mail and ask\_time\_date), we have extracted the suitable information from three APIs: Accuweather API<sup>10</sup>, google\_api\_python\_client<sup>11</sup> and ummalqura.hijri\_date API<sup>12</sup>.

```

pipeline:
- name: WhitespaceTokenizer
- name: RegexFeaturizer
- name: LexicalSyntacticFeaturizer
- name: CountVectorsFeaturizer
- name: CountVectorsFeaturizer
  analyzer: char_wb
  min_ngram: 1
  max_ngram: 4
- name: DIETClassifier
  epochs: 100
  constrain_similarities: true
  model_confidence: linear_norm
- name: EntitySynonymMapper
- name: ResponseSelector
  epochs: 100
  constrain_similarities: true
- name: FallbackClassifier
  threshold: 0.3
  ambiguity_threshold: 0.1

policies:
- name: MemoizationPolicy
- name: TEDPolicy
  max_history: 5
  epochs: 100
  constrain_similarities: true
- name: RulePolicy

```

Figure 3. The pre-configured RASA pipeline and polices.

Table 6. Examples of intents and entities.

Intent	Example	Entity
greet	عالسلامة (çAlslAmħ) ‘Hello’	-
affirm	ياهي (bAhY) ‘Ok’	-
goodbye	بالسلامة (bAlslAmħ) ‘Good bye’	-
thanks	صحبت (SHyt) ‘Well-done’	-
provide_weather_forecasts	بالله شنوة احوال الطقس؟ (bAllh šnwħ AHwAl AlTqs) ‘How is the weather?’	-
	شنية احوال الطقس في الجّم المهدية (šnyħ AHwAl AlTqs fy Aljm~ Almhdyħ) ‘How is the weather in Djem Mahdia?’	[المهدية] {‘entity’: ‘tun_city’, ‘value’: ‘Mahdia’} [الجّم] {‘entity’: ‘tun_city’, ‘value’: ‘Mahdia’}
ask_mail	أقرأ لي ليزمايل متاعي (ÂqrA ly lyzmAyl mtAçy) ‘Please read my emails’	[ليزمايل] {‘entity’: ‘mail’, ‘value’: ‘mail’}
ask_date	اليوم في قداه (Alywm fy qdAh) ‘What is the date of today?’	[اليوم] {‘entity’: ‘date’, ‘role’: ‘Day’}
	أحنا قداش في العام الهجري؟ (ÂHnA qdAš fy AlçAm Alhjry?) ‘What is the Hijri date today?’	[العام الهجري] {‘entity’: ‘date’, ‘role’: ‘Year_Hejri’}

#### 4.2.2 Dataset

To use RASA NLU, we prepared a training dataset to recognize intents and extract entities. The training data includes about 722 sentences with marked entities to train the RASA NLU. More specifically, there are 84.63% of sentences with entities which are presented according to weather specifications, date, time and email. The training data contains both interrogative and declarative sentences. We have also defined 46 synonyms for several entities (e.g. Tunisian cities, wind, clouds, etc.). We have used the person’s name lexicon, composed of 538 entries.

#### 4.2.3 Evaluation

First, to evaluate the performance of our NLU module, we have measured the numbers of intents and

<sup>10</sup> <http://dataservice.accuweather.com/forecasts/>

<sup>11</sup> <https://pypi.org/project/google-api-python-client/>

<sup>12</sup> <https://github.com/borni-dhifi/ummalqura>



Table 7. Examples of pre-defined responses.

Response type	Example	Signification
utter_city_identification	(ĀçTyny AlwlAyħ mtAçk) أعطيني الولاية متاعك <i>Give me the name of your state.</i>	City identification
utter_bye	بالسلامة (bAlslAmħ) 'Bye'	Bye
utter_thanks	'You are welcome,' (mn γyr mzyħ) مزية من غير	Thanks
utter_start	(çAlslAmħ) {person_name} عالسلامة <i>'Hello {person_name}'</i>	Start the conversation
utter_start_first	عالسلامة أنا المساعد الشخصي متاعك. تنجم تعرفني بيك؟ شنو اسمك؟ (çAlslAmħ ĀnA AlmsAçd AlšxSy mtAçk. tnjm tçrfny byk? šnw Asmk?) <i>'Hello. I am your personal assistant. Can I recognize you? What is your name?'</i>	First conversation

```

- story: ask_mail_story_1
steps:
- intent: greet
- action: utter_start
- intent: ask_mail
entities:
- mail: mail
- slot_was_set:
- mail: mail
- action: action_mail
- intent: thanks
- action: utter_thanks

```

Figure 4. Example of Tunisian dialect story.

entities correctly classified. Due to the small size of the collected corpus, we have applied 5-cross validation and 10-cross validation to evaluate our NLU module. We have calculated the accuracy, F1-score and precision measures. Table 9 shows that entity-extraction accuracy is generally good. The accuracy value is 0.97 for both 10- and 5-cross validation. The results show that the F1-score scales from 0.74 for cloud entity extraction to 1 for multiple entities (e.g. month name). The failure in the classification of some entities can be explained by the presence of some entities composed of two words or more (e.g. عام العربي (çAm Alçrby) 'Hejri Year'). The DIET classifier [81] is not able to detect the whole components of an entity. Sometimes, it detects the first or second part of the entity. In other cases, it fails to detect all the parts. When it comes to intents, the accuracy is 0.951 for 10-cross validation and 0.947 for 5-cross validation (See Table 9). There are some intent-classification mistakes related to greeting, denying and bye intents. By analyzing errors, we observe that the classifier makes an error for closely related utterances like عالسلامة (çAlslAmħ) 'hello' and بالسلامة (bAlslAmħ) 'bye'. In addition, some Tunisians use the same utterances for greeting and good-bye (i.e., السلام عليكم (AlslAm çlykm) to say 'hello' and 'goodbye'). In our future work, to avoid some errors, we propose to apply some pre-processing steps (e.g. tokenization, parsing, base phrase chunking, etc.) to requests before classification steps.

Table 8. Example of conversation between the user and IPA according to the story presented in Fig. 3.

User	عالسلامة 'Hello' (çAlslAmħ)
IPA	عالسلامة 'Hello Ines' (çAlslAmħ ĀynAs)
User	تشوف لي عنديشي مايل جديد (tšwf ly çndyšy mAyl jdyd) <i>'Can you tell me if I have new email?'</i>
IPA	عندك زوز مايلوات جدد 'You have two new emails.' (çndk zwz mAylwAt jdd)
User	يعيشك مرسي 'Thank you' (myrsy yçyšk)
IPA	من غير مزية 'You are welcome,' (mn γyr mzyħ)

Moreover, to evaluate the quality of a full-dialog system; namely, NLU, DM and NLG modules, we have evaluated dialogs end-to-end by running through test stories. For this purpose, we used 15 stories. We obtained an accuracy of about 60%. Some errors are related to misclassification of some intents and/or entities. We have also evaluated the action level of the RASA core. The action-level results

Table 9. Entities' and intents' classification results.

	Intents		Entities	
	5-cross validation	10-cross validation	5-cross validation	10-cross validation
<b>Accuracy</b>	0.947	0.951	0.974	0.971
<b>F1-score</b>	0.945	0.95	0.966	0.966
<b>Precision</b>	0.951	0.954	0.977	0.978

measure the numbers for each intent-entity extraction prediction in all of the test stories. We obtained the following results: 0.899, 0.894 and 0.915, respectively, for F1-score, precision and accuracy.

### 4.3 Speech Synthesis

#### 4.3.1 Proposed Method

End-to-end neural network architectures are widely used in many SS tasks. Unlike pipeline-based techniques, they are structured as a single component. End-to-end architectures learn all the steps between the initial input phase and the final output result and generate a single model. They reduce the need for expensive domain expertise and arduous feature engineering and require only minimal human annotation [64]. Among the famous and successful proposed end-to-end architectures proposed for SS, we cite Tacotron 2 [69]. Tacotron 2 is composed of two components: a sequence-to-sequence architecture spectrogram prediction network with attention and a flow-based implementation of WaveGlow [64]. For TA, we applied the Tacotron 2 architecture, which was updated by [64] in order to synthesize MSA. According to [64], the sequence-to-sequence spectrogram consists of an encoder and a decoder. The encoder takes a phonetized text as input and produces a hidden feature vector representation, which goes to the decoder and generates the mel-spectrograms of the given input characters. Then, the spectrograms are passed to a five-layer post-net. Finally, the WaveGlow vocoder, a flow-based generative network, is trained alongside using the mel-spectrograms and generates the voice as the output. We have used the open-source phonetization algorithm proposed by Nawar Halabi<sup>13</sup> to phonetize the input text.

#### 4.3.2 Dataset

As the first attempt for our SS model, we have decided to train our model using one speaker transcriptions. Hence, we have trained Tacotron 2 on TA transcriptions, which contain about 1 hour and 33 minutes of speech composed of 2180 utterances. The corpus is composed of a pair of audio files and their transcriptions. We collected text from some Tunisian stories and some chapters from the Tunisian constitution in TA. We have divided them into short sentences which, then, have been recorded by a Tunisian woman (native speaker) in a silent environment. We manually recorded speech audio files using Audacity software<sup>14</sup>. We have used the Buckwalter transliteration<sup>15</sup> for the input text. Due to the unavailability of diacritization system for the TA and the slowness of manual transcription, we have decided to ignore all diacritics. Like SR corpus, we have converted files into wave format with a mono audio channel with a sample rate of 22050 Hz in order to be read by the Tacotron 2 pipeline. This dataset is used for training and validating the model. For testing our model, we have used a set composed of 2445 utterances, which consists of possible responses that the TA IPA can return to the user. The utterances include greetings, bye, weather forecasts, as well as time and date information.

#### 4.3.3 Evaluation

Qualitative analysis was realized by using human ratings. We have calculated the subjective Mean Opinion Score (MOS), a rating of how good the synthesized utterances are for audio naturalness and comprehensiveness. Each utterance is evaluated by two raters. A score ranging from 1 to 5 was given to each utterance. 1 is given to bad audio, while 5 is given to the most natural audio. Table 10 presents the evaluation results. We have obtained an average MOS of 3.08. The score is encouraging for a first attempt to generate an SS model for TA. It shows that our SS can generate voice (the output of the IPA), which is almost natural and understandable. The analysis of the SS output shows that the majority of

<sup>13</sup> <https://github.com/nawarhalabi/Arabic-Phonetiser>

<sup>14</sup> <https://www.audacityteam.org/>

<sup>15</sup> <http://www.qamus.org/transliteration.htm>

mistakes are related to some phonemes that our model is not able to pronounce. Also, it is not able to synthesize some words. We can explain this failure by the absence of some phonemes in the training corpus. For example, the word *ثلاثاشر* (θltTAš) 'thirteen' is mispronounced due to the absence of the phoneme related to the letter *ش*. We remark that our MOS score is lower than the [64] score. First, they used a diacritized corpus. The presence of short vowels helps learn the pronunciation. Also, their corpus is relatively bigger than ours.

Table 10. Evaluation results

Raters	MOS average
1	3.32
2	2.84
<b>Average</b>	<b>3.08</b>
MSA Tacotron 2 model [64]	<b>4.21</b>

## 5. CONCLUSION AND FUTURE WORK

In this paper, we present an attempt to create an Intelligent Personal Assistant (IPA) for the Tunisian dialect. We studied different approaches proposed for developing a dialog system, in particular, a task-oriented system. We prepared the basic components of an IPA: speech components (speech recognition model and speech synthesis model) and a dialog system core (natural-language understanding, Dialog management and language generation). We have applied deep-learning techniques: CNN [1], RNN encoder-decoder [2] and end-to-end approaches for creating IPA speech components (Speech Recognition and Speech Synthesis). In addition, we have explored the available and free dialog platform for understanding and generating the suitable response in TA for a request. Despite the lack of TA resources and as the first attempt, the evaluation results of some components are acceptable. We have proved the feasibility of creating an IPA with free resources while the language is under-resourced.

For future work, we have two main objectives. First, we intend to improve the quality of the proposed components. We intend to expand the size of all corpora by adding code-switching utterances and test other deep architectures. For speech components, we plan to add diacritics for our transcriptions. Their roles are important for the Arabic language. For speech recognition, we also plan to add more speakers to our corpus in order to recognize different speakers. We aim to augment the size of the corpus for the core of the dialog system by adding more services to the IPA. We intend to apply Transformer [82] in order to build some TA NLP tools and integrate them into the Rasa pipeline. The model will be able in the future to detect more complex entities and perform more complicated tasks. The second objective is the integration and testing of the developed components in the open-source vocal IPA, "openjarvis". It is designed to be executed on an energy-saving system, like the Raspberry Pi. It is a customizable IPA and the integration of new components is relatively easy.

## REFERENCES

- [1] K. O'Shea and R. Nash, "An Introduction to Convolutional Neural Networks," CoRR, vol. abs/1511.0, 2015.
- [2] A. Hannun et al., "Deep Speech: Scaling up End-to-end Speech Recognition," arXiv1412.5567v2 [cs.CL], pp. 1–12, 2014.
- [3] I. Lopatovska, "Overview of the Intelligent Personal Assistants," Ukr. J. Libr. Inf. Sci., no. 3, pp. 72–79, DOI: 10.31866/2616-7654.3.2019.169669, 2019.
- [4] K. Jokinen and M. McTear, "Spoken Dialogue Systems," Synthesis Lectures on Human Lang. Technol., Synthesis., Morgan & Claypool Publishers, DOI: 10.2200/S00204ED1V01Y200910HLT005, 2010.
- [5] N. Goksel-Canbek and M. E. Mutlu, "On the Track of Artificial Intelligence: Learning with Intelligent Personal Assistants," Int. J. Hum. Sci., vol. 13, no. 1, pp. 592–601, DOI: 10.14687/ijhs.v13i1.3549, 2016.
- [6] A. V. Román, D. P. Martínez, Á. L. Murciago, D. M. Jiménez-Bravo and J. F. de Paz, "Voice Assistant Application for Avoiding Sedentarism in Elderly People Based on IoT Technologies," Electronics, vol. 10, no. 980, 2021.
- [7] Y. Matsuyama, A. Bhardwaj, R. Zhao, O. J. Romero, S. A. Akoju and J. Cassell, "Socially-aware Animated Intelligent Personal Assistant Agent," Proc. of the 17<sup>th</sup> Annual Meeting in Special Interest Group on Discourse and Dialogue (SIGDIAL 2016), pp. 224–227, DOI: 10.18653/v1/w16-3628, 2016.

- [8] J. Santos, J. J. P. C. Rodrigues, B. M. C. Silva, J. Casal, K. Saleem and V. Denisov, "An IoT-based Mobile Gateway for Intelligent Personal Assistants on Mobile Health Environments," *J. Netw. Comput. Appl.*, vol. 71, pp. 194–204, DOI: 10.1016/j.jnca.2016.03.014, 2016.
- [9] M. T. Talacio, Development of an Intelligent Personal Assistant to Empower Operators in Industry 4.0 Environments, M.Sc. Thesis, School of Technology and Management of Bragança. University of Bragança, 2020.
- [10] E. Balci, "Overview of Intelligent Personal Assistants," *Acta INFOLOGICA*, vol. 3, no. 1, pp. 22–33, DOI: 10.26650/acin.454522, 2019.
- [11] K. Zdanowski, "Language Support in Voice Assistants Compared," *Summa Linguae Technologies*, Accessed on: Aug. 01, 2022, [Online], Available: <https://summalinguae.com/language-technology/language-support-voice-assistants-compared/>, 2021.
- [12] I. Zribi, R. Boujelbane, A. Masmoudi, M. Ellouze, L. Belguith and N. Habash, "A Conventional Orthography for Tunisian Arabic," *Proc. of the 9<sup>th</sup> International Conference on Language Resources and Evaluation (LREC'14)*, vol. Proc., pp. 2355–2361, Reykjavik, Iceland, 2014.
- [13] A. Bouzemni, "Linguistic Situation in Tunisia: French and Arabic code switching," *INTERLINGUISTICA*, pp. 217–223, 2005.
- [14] I. Zribi, M. Ellouze, L. H. Belguith and P. Blache, "Spoken Tunisian Arabic Corpus 'STAC': Transcription and Annotation," *Resarch in Computing Science*, vol. 90, pp. 123–135, 2015.
- [15] I. Zribi, M. Ellouze, L. H. Belguith and P. Blache, "Morphological Disambiguation of Tunisian Dialect," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 29, no. 2, pp. 147–155, 2017.
- [16] H. Chen, X. Liu, D. Yin and J. Tang, "A Survey on Dialogue Systems: Recent Advances and New Frontiers," *arXiv:1711.01731v3*, no. 1, 2018.
- [17] H. B. Hashemi, A. Asiaee and R. Kraft, "Query Intent Detection Using Convolutional Neural Networks," *WSDM QRUMS 2016 Workshop*, DOI: 10.1145/1235, 2016.
- [18] K. Sreelakshmi, P. C. Rafeeqe, S. Sreetha and E. S. Gayathri, "Deep Bi-directional LSTM Network for Query Intent Detection," *Procedia Computer Science*, vol. 143, pp. 939–946, 2018.
- [19] A. Deoras and R. Sarikaya, "Deep Belief Network Based Semantic Taggers for Spoken Language Understanding," *Proc. Interspeech 2013*, pp. 2713–2717, DOI: 10.21437/Interspeech.2013-623, 2013.
- [20] P. S. Huang, X. He, J. Gao, L. Deng, A. Acero and L. Heck, "Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data," *Proc. of the 22<sup>nd</sup> ACM Int. Conf. on Information & Knowledge Management (CIKM '13)*, pp. 2333–2338, DOI: 10.1145/2505515.2505665, 2013.
- [21] W. A. Abro, A. Aicher, N. Rachb, S. Ultes, W. Minker and G. Qi, "Natural Language Understanding for Argumentative Dialogue Systems in the Opinion Building Domain," *Knowledge-Based Syst.*, vol. 242, DOI: 10.1016/j.knosys.2022.108318, 2022.
- [22] J. D. Williams, "Web-style Ranking and SLU Combination for Dialog State Tracking," *Proc. of the 15<sup>th</sup> Annu. Meet. Spec. Interes. Gr. Discourse Dialogue (SIGDIAL 2014)*, pp. 282–291, DOI: 10.3115/v1/w14-4339, 2014.
- [23] S. Sharma, P. K. Choubey and R. Huang, "Improving Dialogue State Tracking by Discerning the Relevant Context," *Proc. of the Conf. of the North American Chapter of the Association for Computational Linguistics: Human Lang. Technol. (NAACL HLT 2019)*, vol. 1, DOI: 10.18653/v1/n19-1057, 2019.
- [24] Q. Xie, K. Sun, S. Zhu, L. Chen and K. Yu, "Recurrent Polynomial Network for Dialogue State Tracking with Mismatched Semantic Parsers," *Proc. of the 16<sup>th</sup> Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 295–304, DOI: 10.18653/v1/w15-4641, Prague, Czech Republic, 2015.
- [25] Z. Yan, N. Duan, P. Chen, M. Zhou, J. Zhou and Z. Li, "Building Task-oriented Dialogue Systems for Online Shopping," *Proc. of the 31<sup>st</sup> AAAI Conf. on Artificial Intell. (AAAI-17)*, pp. 4618–4625, 2017.
- [26] H. Cuayáhuatl, S. Keizer and O. Lemon, "Strategic Dialogue Management *via* Deep Reinforcement Learning," *arXiv:1511.08099v1*, pp. 1–10, 2015.
- [27] A. Stent, R. Prasad and M. Walker, "Trainable Sentence Planning for Complex Information Presentation in Spoken Dialog Systems," *Proc. of the 42<sup>nd</sup> Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pp. 79–86, DOI: 10.3115/1218955.1218966, Barcelona, Spain, 2004.
- [28] T. H. Wen et al., "Stochastic Language Generation in Dialogue Using Recurrent Neural Networks with Convolutional Sentence Reranking," *Proc. of the 16<sup>th</sup> Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 275–284, DOI: 10.18653/v1/w15-4639, Prague, Czech Republic, 2015.
- [29] T. H. Wen, M. Gašić, N. Mrkšić, P. H. Su, D. Vandyke and S. Young, "Semantically Conditioned LSTM-based Natural Language Generation for Spoken Dialogue Systems," *Proc. of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1711–1721, DOI: 10.18653/v1/d15-1199, 2015.
- [30] H. Zhou, M. Huang and X. Zhu, "Context-aware Natural Language Generation for Spoken Dialogue Systems," *Proc. of the 26<sup>th</sup> Int. Conf. on Computational Linguistics: Technical Papers*, pp. 2032–2041, Osaka, Japan, 2016.
- [31] O. Dušek and F. Jurcicek, "Sequence-to-sequence Generation for Spoken Dialogue *via* Deep Syntax Trees

- and Strings," Proc. of the 54<sup>th</sup> Annual Meeting of the Association for Computational Linguistics, vol. 2: Short Papers, pp. 45-51, DOI: 10.18653/v1/p16-2008, Berlin, Germany, 2016.
- [32] T. H. Wen and S. Young, "Recurrent Neural Network Language Generation for Spoken Dialogue Systems," *Computer Speech & Language*, vol. 63, DOI: 10.1016/j.csl.2019.06.008, 2020.
- [33] T. H. Wen et al., "A Network-based End-to-end Trainable Task-oriented Dialogue System," Proc. of the 15<sup>th</sup> Conf. of the European Chapter of the Association for Computational Linguistics, vol. 1: Long Papers, pp. 438-449, Valencia, Spain, April 3-7, 2017.
- [34] A. Bordes, Y. Lan Boureau and J. Weston, "Learning End-to-end Goal-oriented Dialog," Proc. of the 5<sup>th</sup> Int. Conf. Learn. Represent. (ICLR 2017), 2017.
- [35] C. Li, L. Li and J. Qi, "A Self-attentive Model with Gate Mechanism for Spoken Language Understanding," Proc. of the 2018 Conference on Empirical Methods in Natural Language Processing, pp. 3824-3833. DOI: 10.18653/v1/D18-1417, 2018.
- [36] T. Bocklisch, J. Faulkner, N. Pawlowski and A. Nichol, "Rasa: Open Source Language Understanding and Dialogue Management," Proc. of NIPS 2017 Conversational AI Workshop, pp. 1-9, Long Beach, USA, 2017.
- [37] B. A. Shawar, "A Chatbot as a Natural Web Interface to Arabic Web QA," *Int. J. Emerg. Technol. Learn. (IJET)*, vol. 6, no. 1, pp. 37-43, DOI: 10.3991/ijet.v6i1.1502, 2011.
- [38] S. M. Yassin and M. Z. Khan, "SeerahBot: An Arabic Chatbot about Prophet's Biography," *Int. J. Innov. Res. Comput. Sci. Technol. (IJIRCST)*, vol. 9, no. 2, DOI: 10.21276/ijircst.2021.9.2.13, 2021.
- [39] D. Abu Ali and N. Habash, "Botta : An Arabic Dialect Chatbot," Proc. of the 26<sup>th</sup> Int. Conf. on Comput. Linguist.: Sys. Demonstrat. (COLING 2016), pp. 208-212, Osaka, Jpn, 2016.
- [40] D. Al-ghadhban and N. Al-twairsh, "Nabiha : An Arabic Dialect Chatbot," *Int. J. of Advanced Computer Sci. and App. (IJACSA)* vol. 11, no. 3, pp. 452-459, 2020.
- [41] A. A. Abdelhamid, H. Alsayadi, I. Hegazy and Z. T. Fayed, "End-to-end Arabic Speech Recognition: A Review," Proc. of the 19<sup>th</sup> Conf. of Language Engineering (ESOLEC'19), Bibliotheca Alexandrina, 2020.
- [42] A. M. Dammak, "Approche Hybride Pour la Reconnaissance Automatique de la Parole Pour la Langue Arabe," *Environnements Informatiques pour l'Apprentissage Humain, Université du Maine, Français*, (NNT : 2016LEMA1040), 2016.
- [43] S. Dua et al., "Developing a Speech Recognition System for Recognizing Tonal Speech Signals Using a Convolutional Neural Network," *Appl. Sci.*, vol. 12, no. 12, p. 6223, DOI: 10.3390/app12126223, 2022.
- [44] A. Y. Hannun, D. Jurafsky, A. L. Maas and A. Y. Ng, "First-pass Large Vocabulary Continuous Speech Recognition Using Bi-directional Recurrent DNNs," arXiv 1408.2873v2 [cs.CL], pp. 1-7, 2014.
- [45] Y. Peng and K. Kao, "Speech to Text System: Pastor Wang Mandarin Bible Teachings (Speech Recognition)," CS230: Deep Learning, Stanford Univ., CA., 2020.
- [46] N. Zeghidour et al., "Fully Convolutional Speech Recognition," arXiv:1812.06864v2, pp. 25-29, 2019.
- [47] A. Agarwal and T. Zesch, "German End-to-end Speech Recognition Based on DeepSpeech," Proc. of the 15<sup>th</sup> Conf. on Natural Language Processing (KONVENS 2019), pp. 111-119, 2019.
- [48] V. Pratap et al., "Wav2Letter++: The Fastest Open-source Speech Recognition System," Proc. of the 2019 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), pp. 2-6, Brighton, UK, 2018.
- [49] S. Qin, L. Wang, S. Li, J. Dang and L. Pan, "Improving Low-resource Tibetan End-to-end ASR by Multilingual and Multilevel Unit Modeling," *Eurasip J. Audio, Speech, Music Process.*, vol. 2022, no. 1, DOI: 10.1186/s13636-021-00233-4, 2022.
- [50] L. Lamel and J. Gauvain, "Automatic Speech-to-text Transcription in Arabic," *ACM Transactions on Asian Language Information Processing*, vol. 8, no. 4, DOI: 10.1145/1644879.1644885, 2009.
- [51] M. Elshafei and H. Al-Muhtaseb, "Speaker-independent Natural Arabic Speech Recognition System," Proc. of the Int. Conf. on Intelligent Systems., [Online], Available: [https://www.researchgate.net/publication/303873329\\_Natural\\_speaker\\_independent\\_arabic\\_speech\\_recognition\\_system\\_based\\_on\\_HMM\\_using\\_sphinx\\_tools](https://www.researchgate.net/publication/303873329_Natural_speaker_independent_arabic_speech_recognition_system_based_on_HMM_using_sphinx_tools), 2010.
- [52] A. Ben Ltaief, Y. Estève, M. Graja and Lamia Hadrach Belguith, "Automatic Speech Recognition for Tunisian Dialect," *Language Resources and Evaluation*, vol. 52, no. 1, pp.249-267, DOI: 10.1007/s10579-017-9402-y, hal-01592416, 2018.
- [53] A. Masmoudi, M. Ellouze Khmekhem, Y. Esteve, L. Hadrach Belguith and N. Habash, "A Corpus and Phonetic Dictionary for Tunisian Arabic Speech Recognition," Proc. of the 9<sup>th</sup> Int. Conf. Lang. Resour. Eval., vol. 3, no. 1, pp. 306-310, 2014.
- [54] A. Messaoudi, H. Haddad, C. Fourati et al., "Tunisian Dialectal End-to-end Speech Recognition Based on DeepSpeech," *Procedia Comput. Sci.*, vol. 189, pp. 183-190, DOI: 10.1016/j.procs.2021.05.082, 2021.
- [55] S. N. Kayte, M. Mundada, S. Gaikwad and B. Gawali, "Performance Evaluation of Speech Synthesis Techniques for English Language," *Adv. Intell. Syst. Comput.*, vol. 439, no. June, pp. 253-262, 2016.

- [56] C. Quillen, "Autoregressive HMM Speech Synthesis," Proc. of the 2012 IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), DOI: 10.1109/ICASSP.2012.6288800, Kyoto, Japan, 2012.
- [57] M. Shannon and W. Byrne, "Autoregressive HMMs for Speech Synthesis," Proc. of the 10<sup>th</sup> Int. Conf. of the Int. Speech Comm. Associa. (Interspeech 2009), DOI: 10.21437/interspeech.2009-135, 2009.
- [58] S. Roekhaut, S. Brognaux, R. Beaufort and T. Dutoit, "eLite-HTS: A NLP Tool for French HMM-based Speech Synthesis," Proc. Annu. Conf. Int. Speech Commun. Assoc. (Interspeech 2014), Singapore, 2014.
- [59] S. Le Maguer, N. Barbot and O. Boeffard, "Evaluation of Contextual Descriptors for HMM-based Speech Synthesis in French," Proc. of the 8<sup>th</sup> ISCA Work, Speech Synth., HAL Id: hal-00987809, version 1, 2013.
- [60] K. M. Khalil and C. Adnan, "Arabic Speech Synthesis Based on HMM," Proc. of the 15<sup>th</sup> IEEE Int. Multi-Conf. on Systems, Sig. & Devic. (SSD), DOI: 10.1109/SSD.2018.8570388, Hammamet, Tunisia, 2018.
- [61] A. Amrouche, A. Abed and L. Falek, "Arabic Speech Synthesis System Based on HMM," Proc. of the 6<sup>th</sup> IEEE Int. Conf. on Electrical and Electronics Eng. (ICEEE), DOI: 10.1109/ICEEE2019.2019.0022, Istanbul, Turkey, 2019.
- [62] H. Al Masri and M. E. Za'ter, "Arabic Text-to-speech (TTS) Data Preparation," arXiv:2204.03255v1, [Online], Available: <http://arxiv.org/abs/2204.03255>, 2022.
- [63] K. M. Khalil and C. Adnan, "Arabic HMM-based Speech Synthesis," Proc. of the IEEE 2013 Int. Conf. on Electri. Eng. and Soft. Appl., DOI: 10.1109/ICEESA.2013.6578437, Hammamet, Tunisia, 2013.
- [64] F. K. Fahmy, M. I. Khalil and H. M. Abbas, "A Transfer Learning End-to-end Arabic Text-to-speech (TTS) Deep Architecture," arXiv:2007.11541v1 [eess.AS], 2020.
- [65] A. van den Oord et al., "WaveNet: A Generative Model for Raw Audio Based on PixelCNN Architecture," arXiv:1609.03499, 2016.
- [66] S. Arik et al., "Deep Voice: Real-time Neural Text-to-speech," Proc. of the 34<sup>th</sup> Int. Conf. Mach. Learn. (ICML 2017), vol. 1, no. Icml, pp. 264–273, 2017.
- [67] Y. Wang et al., "Tacotron: Towards End-to-end Speech Synthesis," arXiv:1703.10135v2, pp. 1–10, 2017.
- [68] I. Sutskever, O. Vinyals and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," arXiv:1409.3215, 2014.
- [69] J. Shen et al., "Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions," DOI: 10.1109/ICASSP.2018.8461368, Proc. of the IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), Calgary, AB, Canada, 2018.
- [70] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho and Y. Bengio, "Attention-based Models for Speech Recognition," arXiv:1506.07503, 2015.
- [71] I. Hadj Ali, Z. Mnasri and Z. Lachiri, "DNN-based Grapheme-to-phoneme Conversion for Arabic Text-to-speech Synthesis," Int. J. Speech Technol., vol. 23, pp. 569–584, DOI: 10.1007/s10772-020-09750-7, 2020.
- [72] A. Abdelali, N. Durrani, C. Demiroglu, F. Dalvi, H. Mubarak and K. Darwish, "NatiQ: An End-to-end Text-to-speech System for Arabic," arXiv:2206.07373v1, 2022.
- [73] N. Li, S. Liu, Y. Liu et al., "Neural Speech Synthesis with Transformer Network," Proc. of the 33<sup>rd</sup> AAAI Conf. on Artificial Intelligence (AAAI-19), pp. 6706–6713. DOI: 10.1609/aaai.v33i01.33016706, 2019.
- [74] D. Amodei et al., "Deep Speech 2: End-to-End Speech Recognition in English and Mandarin," arXiv 1512.02595v1 [cs.LG], pp. 1–28, 2015.
- [75] A. Graves, S. Fernandez, F. Gomez and J. Schmidhuber, "Connectionist Temporal Classification: Labelling Unsegmented Sequence Data with Recurrent Neural Networks," Proc. of the 23<sup>rd</sup> Int. Conf. on Machine Learning (ICML '06), pp. 369–376, DOI: 10.1145/1143844.1143891, 2006.
- [76] K. Heafield, "KenLM: Faster and Smaller Language Model Queries," Proc. of the 6<sup>th</sup> Workshop on Statistical Machine Translation, pp. 187–197, Edinburgh, Scotland, 2011.
- [77] A. Mekki, I. Zribi, M. Ellouze and L. H. Belguith, "Sentence Boundary Detection of Various Forms of Tunisian Arabic," Language Resources and Evaluation, vol. 56, pp. 357–385, DOI: 10.1007/s10579-021-09538-4, 2022.
- [78] N. Thi, M. Trang and M. Shcherbakov, "Enhancing Rasa NLU Model for Vietnamese Chatbot," Int. J. of Open Information Technologies (INJOIT), vol. 9, no. 1, pp. 31–36, 2021.
- [79] Y. Windiatmoko, A. F. Hidayatullah and R. Rahmadi, "Developing FB Chatbot Based on Deep Learning Using RASA Framework for University Enquiries," CoRR, vol. abs/2009.1, [Online], Available: <https://arxiv.org/abs/2009.12341>, 2020.
- [80] V. Vlasov, J. E. M. Mosig and A. Nichol, "Rasa Open Source Documentation," RASA DOCS, [Online], available: <https://rasa.com/docs/rasa/>, 2022.
- [81] T. Bunk et al., "DIET: Lightweight Language Understanding for Dialogue Systems," arXiv:2004.09936v3, [Online], Available: <https://arxiv.org/pdf/2004.09936.pdf>, 2020.
- [82] A. Chernyavskiy, D. Ilvovsky and P. Nakov, "Transformers: 'The End of History' for Natural Language Processing?," arXiv:2105.00813, [Online], Available: <http://arxiv.org/abs/2105.00813>, 2021.
- [83] N. Habash, A. Soudi and T. Buckwalter, "On Arabic Transliteration," Arabic Computational Morphology, Part of the Text, Speech and Language Technology Book Series, vol. 38, pp. 15–22, 2007.
- [84] S. Hussain, O. A. Sianaki and N. Ababneh, "A Survey on Conversational Agents," Proc. of the Workshops

**ملخص البحث:**

تم اقتراح أنظمة ذكية على نحو مكثف، بدفع من تقنيات الذكاء الاصطناعي، لمساعدة الناس في أداء مهام متنوعة. ومن هذه الأنظمة الذكية المساعد الشخصي (IPA). في هذه الورقة، نعرض محاولة لإيجاد مساعد شخصي ذكي يتفاعل مع المستخدمين عبر اللهجة التونسية، وهي اللهجة العامية المتداولة في تونس.

نقترح ونطبق طريقة سهلة التنفيذ لبناء المكونات الأساسية للمساعد الشخصي الذكي بالعربية التونسية، ونستخدم تقنيات التعلم العميق (CNN)، والترميز-فك الترميز RNN، وطريقة من الطرف الى الطرف) من أجل إيجاد مكونات المساعد الشخصي الذكي، وهي: (تمييز الكلام، وتحليل الكلام).

بالإضافة الى ذلك، نستكشف مدى التوفر ومجانية منصات الحوار لفهم وتوليد الإجابة المناسبة للطلب باللهجة التونسية. ولهذا الغرض، نقوم بإيجاد واستخدام نصوص باللهجة العامية المتداولة في تونس من أجل إنشاء النماذج ذات العلاقة.

وقد أسفر تقييم النظام المقترح على نتائج مشجعة، اعتبرت مقبولة كمحاولة أولى تبقى مرشحة للتحسين والتطوير في الأبحاث المستقبلية.

# FEATURE LEVEL FUSION FRAMEWORK FOR BRAIN MR IMAGE CLASSIFICATION USING SUPERVISED DEEP LEARNING AND HAND CRAFTED FEATURES

Prashantha S. J.<sup>1</sup> and H. N. Prakash<sup>2</sup>

(Received: 16-Jun.-2022, Revised: 18-Aug.-2022, Accepted: 12-Sep.-2022)

## ABSTRACT

In this paper, we propose an efficient fusion framework for brain magnetic resonance (MR) image classification using deep learning and handcrafted feature extraction methods; namely, histogram of oriented gradients (HOG) and local binary patterns (LBPs). The proposed framework aims to: (1) determine the optimal handcrafted features by Genetic Algorithm (GA) (2) discover the fully connected (FC) layers' features using fine-tuned convolutional neural network (CNN) (3) employ the canonical correlation analysis (CCA) and the discriminant correlation analysis (DCA) methods in feature-level fusion. Extensive experiments were conducted and the classification performance was demonstrated on three benchmark datasets; viz., RD-DB1, TCIA-IXI-DB2 and TWB-HM-DB3. Mean accuracy of 68.69%, 90.35% and 93.15% from CCA and 77.22%, 100.00% and 99.40% from DCA was achieved by the Support Vector Machines (SVM) sigmoid kernel classifier on RD-DB1, TCIA-IXI-DB2 and TWB-HM-DB3, respectively. The obtained results of the proposed framework outperform when compared with other state-of-the-art works.

## KEYWORDS

Deep-learning features, Handcrafted features, Canonical-correlation analysis, Discriminant-correlation analysis, Support vector machines.

## 1. INTRODUCTION

A brain tumour is one of the most significant health problems in the human body. The accurate diagnosis and assessment of disease depend on computerized tools involved in diagnostic tasks. Computer-vision and machine-learning strategies promote early detection to identify the disorder of an individual based on imaging systems. In medical imaging, the acquisition and interpretation of images have improved substantially over recent years. Nowadays, magnetic-resonance imaging (MRI) is a widely used medical imaging method that assists in the detection of brain tumors [1]-[6]. The MRI scan gives detailed imaging information to distinguish cancerous structures from healthy ones, but it takes a long time to diagnose. Hence, to overcome this drawback, an automated approach is needed.

A typical pattern-recognition system consists of feature-extraction, selection and classification methods. Over the past decades, a large number of feature-extraction approaches have been developed, such as histogram of oriented gradients (HOG) [7], wavelet [3][8], convolutional neural network (CNN) [7], [9]-[11], Local Binary Patterns (LBPs) [12], ...etc. On the other hand, feature selection uses several approaches, including ant-colony optimization (ACO), particle-swarm optimization (PSO), genetic algorithms (GAs) [13] [3] and so on. Sometimes, if we use more than one approach, neither a feature extraction nor selection can lead to multi-discriminatory features for the automated medical-diagnosis system. However, these multi-discrimination features hardly interact among themselves, which further limits their semantic relatedness. In this context, fusion methods are essential to generating rich features through fused representation in the automated system. In the process of fusion, the result can occur at the pixel level [14], feature level [15]-[16] and decision level [17] and that ensures the salient features that can improve recognition accuracy. Feature-level fusion has two advantages: first, it can eliminate redundant information between the cross-domain features; second, it may collect non-identical discriminatory features from different cross-domain feature sets.

Over the last few years, many research works on the MR brain-tumor diagnosis categorized brain imaging into two different types: (1) classifying the brain image as either abnormal or normal and (2)

---

1. Prashantha S. J. is with Department of Computer Science and Engineering, AIT Chikkamagaluru, Visvesvaraya Technological Uni., Belagavi, India. Email: prasi.sjp@gmail.com  
2. H. N. Prakash is with Department of Computer Science and Engineering, RIT Hassan, Visvesvaraya Technological Uni., Belagavi, India. Email: prakash.hn98@gmail.com



classifying the abnormal brain image into various types of brain tumors. Kharrat et al. [3] proposed an automated diagnosis and classification approach for Magnetic Resonance (MR) human-brain images. This work used wavelet transform (WT) as an input-feature module to the Genetic Algorithm (GA) and Support Vector Machine (SVM). It separates MR brain images into normal and abnormal ones. Sethy P. K. and Behera S. K. [4] investigated the use of deep-classification methods with deep-learning features to identify tumorous brain MR images. They used the VGG19, VGG16 and Alex Net pre-trained network, combined with SVM for detecting the brain tumor using the 2D brain MRI slices. The work aim was to evaluate the performances of these methods. Ichrak Khouliqi and Najlae Idrissi [18], presented a method of pre-trained Deep Convolutional Neural Networks (DCNNs) based on Transfer Learning (TL) for cervical-cancer detection and classification using MRIs to classify the MRIs into two classes: benign or malign. In [19], the authors have introduced a novel technique for bias-field estimation and correction in MR images to enhance segmentation results. It comprises a modified expectation maximization clustering; the bias field is fitted as a hyper-surface in a 4D hyper-space.

The work proposed by H. H. Sultan et al. [20] classified different brain-tumor types by convolutional neural networks. The proposed method is comprised of two studies. The first study classifies tumors into different types; namely, meningioma, glioma and pituitary tumors. The second study is based on the differentiation between the three glioma grades. Saxena et al. [1] presented a study to classify brain MRI scans into two classes using Resnet-50, VGG-16 and Inception-V3 pre-trained models. However, the Inception-V3 model suffered from overfitting and is slightly better than a random classifier with an accuracy of 0.55. Moreover, the new state-of-the-art architectures are needed by using transfer-learning techniques to improve accuracy. S. Oreski & Oreski, G. [13] proposed a method to identify an optimum feature subset by the hybrid genetic algorithm with neural networks (HGA-NN). Chen et al. [21] proposed to address feature-selection problems through GAs for feature clustering, where a GA was used to optimize the cluster center values of a clustering method to group features into different clusters. Some of the researchers presented feature-level fusion to be more effective. The well-known (1) serial and (2) parallel methods are the most widely used in feature-fusion methods [15]. M. Haghighat et al. [16] presented the correlation analysis-based feature-set fusion for multi-modal biometric recognition. The method demonstrated the effectiveness in the fusion of feature sets extracted from a single modality. However, it uses the class associations of the samples by discriminant-correlation analysis (DCA).

The above-stated research works [1], [3]-[4], [20] have focused only on the identical feature type to build a classification approach of brain MR images. Hence, it is reasonable to propose a study of a multiple-feature fusion framework for brain MR-image classification task. In this work, we proposed an MR brain-image classification model based on a handcrafted and deep-feature fusion approach. The summary of this research work is as follows: (1) A novel proposed method including three significant steps: (i) Handcraft and deep-learning features are extracted from brain MR images, (ii) An optimal handcraft feature set is selected by a GA and (iii) Feature-level fusion operation is performed using CCA and DCA methods. (2) Conduction of experiments is extensively carried out on three benchmark datasets. (3) The robustness of the proposed strategy is evaluated.

In this paper, the contents of the research work are structured as follows. In Section 2, we proposed a novel feature-level fusion model that uses deep and handcrafted features to classify images as either normal or abnormal. In Section 3, experiments conducted are presented. We utilized three publicly available datasets and compared our work with five state-of-the-art works. In Section 4, a conclusion and future-work horizons are presented.

## 2. PROPOSED METHOD

Figure 1 describes the proposed method of feature-level fusion framework for brain MR-image classification. The methodology begins with a pre-processing step, in which the input brain MR-image is resized, normalized by the min-max method and enhanced by the bit-plane slicing method. Next, feature extraction and selection are done in two ways. First, local binary patterns (LBPs) and histogram of oriented gradients (HOG) methods as feature extractors and an optimal subset of features are determined by a Genetic Algorithm (GA), named as handcrafted features. Second, the fine-tuned CNN model acts as a feature extractor and selects deep features from two fully-connected layers (FC1

and FC2). Then, relevant features to the best-fitting feature space of a specific dimension are constructed using principal component analysis. The feature-level fusion techniques have been employed by DCA and CCA methods based on the combination of feature vectors. Finally, classification is performed using a support vector machine (SVM) classifier with a sigmoid function to recognize whether the given MR image is either normal or abnormal.

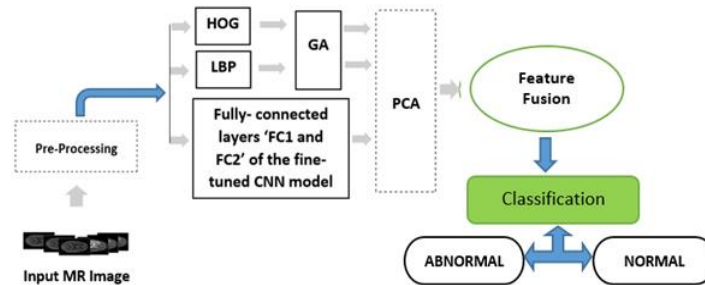


Figure 1. Architecture of the proposed methodology.

## 2.1 Pre-processing

Before classification, a pre-processing step is applied to MR brain images for the proposed methodology. All images are resized to 240x240 pixel dimensions using the bilinear method and image normalization is employed based on the min–max method. Meanwhile, the bit-plane slicing method [12] was applied, which determines whether a bit-plane contains significant information.

## 2.2 Feature Extraction and Selection

We extract the handcrafted (HOG and LBP) and fine-tuned CNN features. The HOG has to capture the edges and corners, whereas LBP has to identify the local micro-structure pattern. The HOG and LBP methods are applied for computing the features based on 32x32 cells, where 324 dimensions of HOG features and 531 dimensions of LBP features are obtained. We adapt a feature selection by Genetic Algorithm (GA) [3] to find the optimal handcrafted feature subset of HOG and LBP. But, despite that, fine-tuned CNN model is used as a trainable feature detector, which can extract low-level, high-level and highly adaptive features. We expand the CNN architecture of Fig.1 in the proposed framework described in Table 1. There are nine layers ordered as I, C1, P1, C2, P2, C3, P3, F1 and F2 in the sequence CNN model, where, I, C, P and F are denoted as the input, convolutional, pooling and fully connected layers, respectively. However, the architecture design of CNN was optimized using a trial and error approach. We extract layer-wise feature sets, out of which two fully connected layers (namely FC1 and FC2) are deep-feature sets having 768 dimensions of richer high–level features obtained.

Table 1. Architecture of CNN.

Layer Name	Type of layer	Kernel size	Feature map
I	Input	-	240x240x1
C1	Conv1+ ReLU	5x5 , 32 filters	240x240x32
P1	Max- Pooling	2x2, stride 1	120x120x32
C2	Conv2+ ReLU	5x5 , 48 filters	120x120x48
P2	Max- Pooling	2x2, stride 1	60x60x32
C3	Conv3+ ReLU	5x5 , 64 filters	60x60x64
F1	Fully Connected (FC1) ReLU	1x384 -	1x384 1x384
F2	Fully Connected (FC2)	1x384	1x384

## 2.3 Fusion and Dimensionality Reduction

The feature-fusion process involves a high-feature space that is highly complicated. The proposed model uses handcrafted (HOG and LBP) and deep features of the fusion task. For this reason, the fusing of two or more inhomogeneous feature vectors leads to conflict. More features can allow for the chance of over-fitting. We used principal component analysis (PCA) to tackle the curse of dimensionality among HOG, LBP and deep features. The dimensionality-reduction process maps the

original predictor space to the best-fitting space of a specific dimension. Next, the fusion operation of an image by CCA and DCA methods is performed to determine the discrimination power on the future combinations.

## 2.4 Classification

In this work, we adopted the support vector machine (SVM) classifier for the automated brain MR-image classification. The CCA and DCA methods of feature fusion are used to train a binary-classification classifier. We used the sigmoid kernel function in the SVM algorithm. The classification results are evaluated and reported in terms of accuracy, sensitivity, specificity, precision, recall and F-measure metrics.

## 3. RESULTS

### 3.1 MR Dataset

The proposed method was applied and tested on three well-known, publicly available benchmark datasets. The acquisition protocol of each dataset includes T2- weighted MR images. The first dataset of brain MR images was downloaded from the radiopaedia.org website [22], labeled as RD-DB1. The RD-DB1 dataset comprises 100 images from 41 subjects or cases, out of which 50 images contain abnormalities and the remaining 50 images are normal. The second dataset was downloaded from The Cancer Imaging Archive (TCIA) [23] and IXI-dataset [24], named as TCIA-IXI-DB2. It comprises 200 images, out of which 50 for the TCGAGBM collection and 50 for the TCGA-LGG collection are tumors, while the remaining are from normal, healthy subjects in the IXI-dataset. The third dataset consists of 350 MR images from the Whole Brain Atlas-Harvard Medical School [25], designated as TWB-HM-DB3, which includes 140 abnormal and 210 normal images.

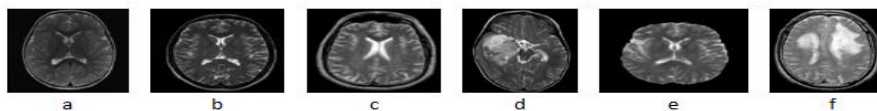


Figure 2. Sample of brain MR images: (a-c) Normal (d-f) Abnormal.



Figure 3. Results of the bit-plane method: (a-c) Normal (d-f) Abnormal.

### 3.2 Experimental Setup and Results

We conducted extensive experiments using three datasets of brain MR images. We used a bit-plane approach as pre-processing to provide feasible improvement. The bit-plane technique results are shown in Figure 3. We set up experiments based on composites of LBP, HOG and deep-feature vectors. The extraction of features was done from the two groups of image descriptors. The first group had two handcrafted features - the HOG and the LBP. The second group had the image feature representation learned by a fine-tuned CNN. Figure 4 shows the results of each handcrafted feature group. Meanwhile, original handcrafted features (HOG and LBP) are input to the Genetic Algorithm for feature selection, to determine the optimal features based on the fitness cost. Figure 5 shows the results of the best feature cost *versus* iteration plot by GA to select the optimal feature set. We used three datasets individually in this paper to select the  $k$  best significant features, including, (1) RD-DB1: The dataset consists of 41 cases with 100 images. Three features are extracted from these images. They are handcrafted LBP with dimension 142, HOG dimension 157 and deep features with dimension 768. (2) TCIA-IXI-DB2: The dataset contains 200 images. The three features extracted are: LBP with dimension 153, HOG with dimension 157 and 768 dimensions of deep features and (3) TWB-HM-DB3: The dataset contains 350 images. Both handcrafted and deep features are extracted from these images. They are HOG with dimension 161, LBP with dimension 146 and deep-learning features with dimension 768. To eliminate bias induced by unequal dimensions of feature groups, we utilize PCA to lower the dimensions of the features on a group-by-group basis.

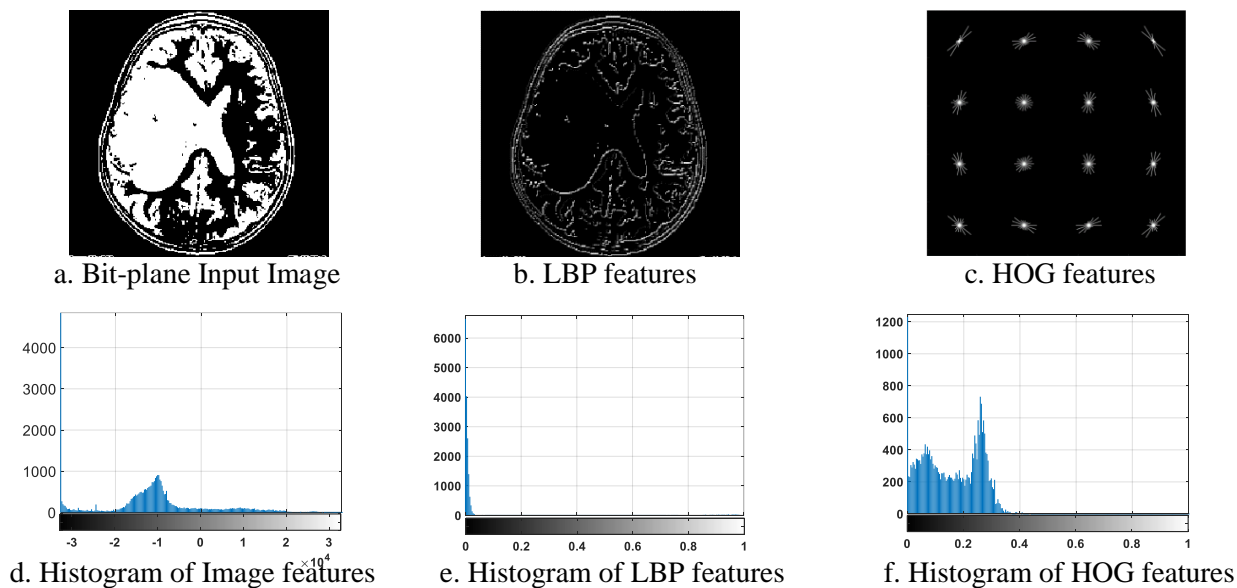


Figure 4. Results of feature extraction.

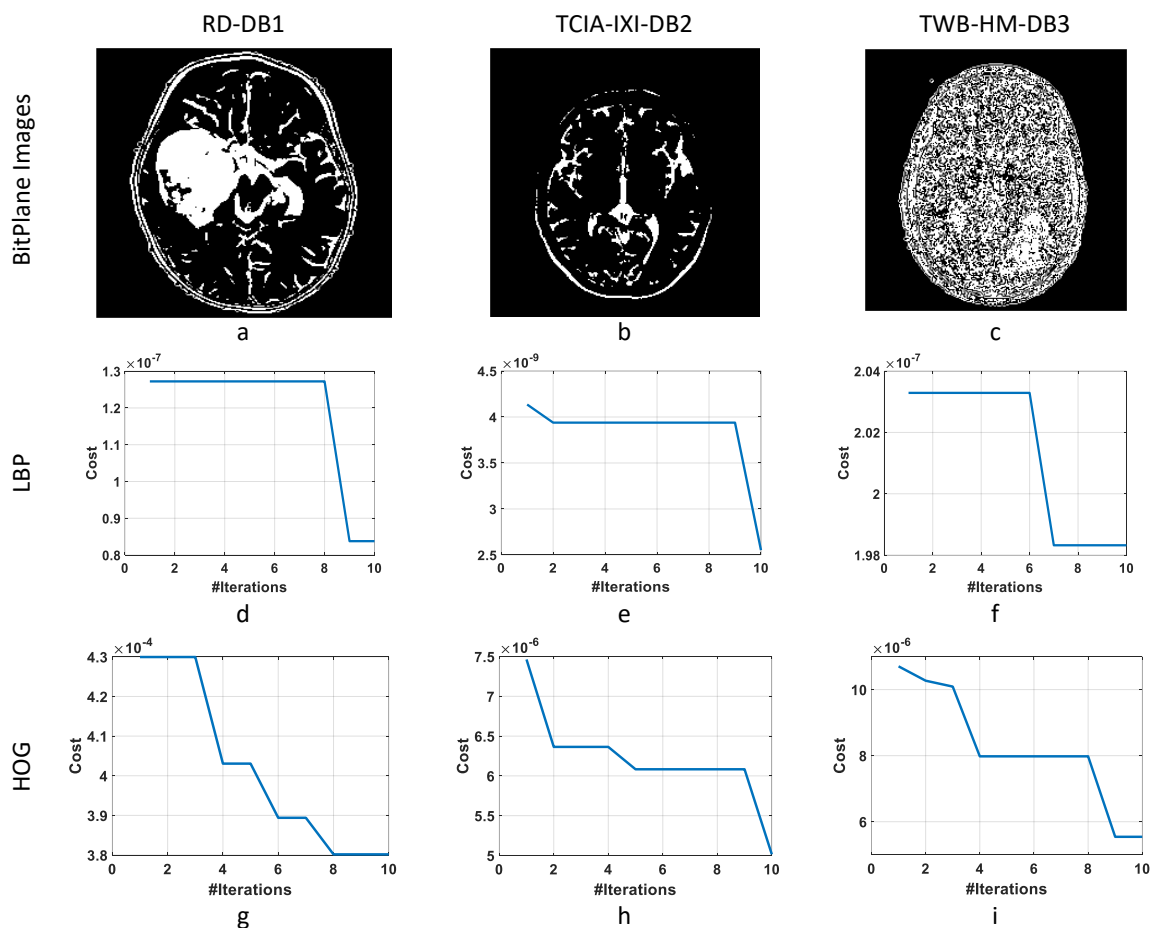


Figure 5. Feature-selection results of (a-c) Bit-plane images (d-f) LBP and (g-i) HOG feature best cost plot by GA.

Figure 4a and Figure 4d demonstrate the bit-plane input image and histogram of image features. Bit-plane slicing is a method of representing an image with one or more bits of the byte used for each pixel. It is converting a gray-level image into a binary image, making the informative region and the noise-like region. Figure 4b and Figure 4e, show an example of computing and visualizing a full LBP 2D array. LBP is a texture descriptor used for the property of high discrimination power. LBP labels each pixel in an image by comparing the gray level with the neighboring pixels and then assigning a

binary number. Finally, using histograms, we represent the frequency of LBP pattern that occurs in the image. In Figure 4c and Figure 4f, we clearly visualize the parts that have strong HOG features of the brain. The HOG feature focuses on the structure or the shape of an object. For the regions of the image, it generates histograms using the magnitude and orientation of the gradient to compute the features as a HOG-feature vector.

### 3.3 Discussion

The classification result for each dataset is based on a training set (70%, 80% and 90% of the total dataset) and a testing set (30%, 20% and 10% of the total dataset). An appropriate classifier is required to test the performance of the proposed classification method. In this approach, we use SVM with sigmoid kernel classifier in the recognition of brain images. During testing, the confusion matrix displays the classification results [26]. Here, TP (True Positive): correctly classified abnormal or positive cases; TN (True Negative): correctly classified normal or negative cases; FP (False Positive): incorrectly classified normal or negative cases; FN (False Negative): incorrectly classified abnormal or positive cases. We used mean accuracy, sensitivity, specificity, precision, recall and F-measure as evaluation metrics. The results in terms of average accuracy over all 3 datasets of test images are shown in Table 2. The proposed-approach fusion-performance results with respect to different metrics are shown in Table 3 and Table 4.

Table 2. Performance results by SVM classifier.

Feature combination HOG/LBP/DL	RD-DB1		TCIA-IXI-DB2		TWB-HM-DB3	
	CCA	DCA	CCA	DCA	CCA	DCA
	Acc (%)	Acc (%)	Acc (%)	Acc (%)	Acc (%)	Acc (%)
<b>LBP+HOG</b>	68.89	77.78	91.27	100.00	95.74	100.00
<b>LBP+DL</b>	70.56	76.11	92.00	100.00	95.45	98.21
<b>HOG+DL</b>	66.67	77.77	87.78	100.00	89.87	100.00
<b>Average</b>	68.70	77.22	90.35	100.00	93.67	99.40

Acc: Accuracy.

Table 3. CCA-based fusion-performance results.

Dataset	Feature Type	Sensitivity	Specificity	Precision	Recall	F-measure
RD-DB1	LBP+HOG	0.6793	0.7041	0.7111	0.6793	0.6930
	LBP+DL	0.7079	0.7129	0.7222	0.7079	0.7099
	HOG+DL	0.6627	0.6718	0.7222	0.6627	0.6883
TCIA-IXI-DB2	LBP+HOG	0.8764	0.9335	0.9389	0.8764	0.9064
	LBP+DL	0.9222	0.9222	0.9222	0.9222	0.9222
	HOG+DL	0.8757	0.8818	0.8889	0.8757	0.8816
TWB-HM-DB3	LBP+HOG	0.9087	0.9277	0.9574	0.9418	0.9418
	LBP+DL	0.9027	0.9277	0.9545	0.9405	0.9405
	HOG+DL	0.9320	0.8632	0.8944	0.9320	0.9111

Table 4. DCA-based fusion-performance results.

Dataset	Feature Type	Sensitivity	Specificity	Precision	Recall	F-measure
RD-DB1	LBP+HOG	0.8412	0.7589	0.7333	0.8412	0.7683
	LBP+DL	0.6934	0.9722	0.9777	0.6934	0.8075
	HOG+DL	0.7936	0.8677	0.8444	0.7936	0.7870
TCIA-IXI-DB2	LBP+HOG	1	1	1	1	1
	LBP+DL	1	1	1	1	1
	HOG+DL	1	1	1	1	1
TWB-HM-DB3	LBP+HOG	1	1	1	1	1
	LBP+DL	0.9705	1	1	0.9705	0.9850
	HOG+DL	1	1	1	1	1

The outcomes of experiments revealed a greater discriminative feature in all cases. The proposed CCA-based and DCA-based feature-fusion approaches perform extremely well in various feature vectors together. This might be because these approaches reduce the amount of redundant information in the two input-feature vectors. Surprisingly, the proposed method stipulates a more powerful feature vector from CCA and DCA-based fusion frameworks for the classification target. We sent all the three dataset images separately to a classifier and then recorded the computation time of training and testing. The average computation time consumed by our proposed system with RD-DB1 of classification in CCA is about 0.571 s (training), 0.010s (testing) and for DCA is about 0.574 s (training), 0.012 s (testing). The training time of TCIA-IXI-DB2 is about 0.58s by CCA and 0.57s by DCA, testing time is about 0.014s in CCA and 0.013s in DCA. Furthermore, in TWB-HM-DB3, the classification time is 0.58s (training), 0.015s (testing) in CCA, while in DCA, it is 0.601s (training), 0.013s (testing).

### 3.4 Comparative Analyses

In this sub-section, we compare the results of the proposed model to those of other state-of-the-art models of MR brain-image classification. So, our intention is to present at this point the results reported in the works [1], [3]-[6] along with the obtained results in the proposed model. In practice, the classification of brain-tumor MR images is done in two ways. The first way is to identify whether the brain MR images are normal or abnormal. The second way is to classify abnormal brain MR images into different tumor types. Table 5 compiles the best (highest) accuracy reported for different approaches. We say that we did not implement or test the other models, but we present the best results of those existing in [1], [3]-[6]. Subsequently, the comparison reveals that the application of the proposed approach directly to the projected fusion features shows improved performance when compared to the original features. However, the efficiency of our model remains higher than those of several previous state-of-the-art works.

Table 5. Performance comparison with state-of-the-art methods.

Study	Feature Extraction	Classification Method	Number of MR Images	Accuracy
Saxena et al., 2019 [1]	CNN	CNN with transfer learning	253	95.00%
Kharrat et al., 2010 [3]	Wavelet-based features	Genetic algorithm with SVM	83	98.14%
Ullah et al., 2020 [4]	DWT	Feed-forward neural network	71	95.80%
B. Ural, 2018 [5]	KMFCM	Probabilistic neural network	25	90.00%
Hemanth et al., 2019[6]	CNN	CNN	220	94.50%
Proposed method (CCA)	HOG+LBP+DL	SVM	100	68.69%,
Proposed method (CCA)	HOG+LBP+DL	SVM	200	90.35%
Proposed method (CCA)	HOG+LBP+DL	SVM	350	93.15%,
Proposed method (DCA)	HOG+LBP+DL	SVM	100	77.22%
Proposed method (DCA)	HOG+LBP+DL	SVM	200	100.00%
Proposed method (DCA)	HOG+LBP+DL	SVM	350	99.40%

Table 5 describes the comparison results of the proposed classification method with state-of-the-art methods. It is clear that the proposed structure gives better prediction results compared to structures given in other related previous studies, which demonstrates the reliability of the proposed model. In contrast, Saxena et al. [1] used feature engineering to extract features and then reduced their dimensions to use them in another stage for classification. In [3], genetic algorithm with SVM was used as a classification method based on wavelet features and achieved 98.14% accuracy; however, the model involves less number of MR images. In [4]-[6], the authors used pathological images to train the network, using DWT, KMFCM and CNN feature-extraction methods with less accuracy.

## 4. CONCLUSION

In this work, we have presented a brain MR-image classification approach jointly using deep and handcrafted features to classify images as either normal or abnormal. The approach has been used for T2-weighted brain MR images only. We made a successful attempt and explored the applicability of correlation analysis to two groups of features. We found that HOG, LBP and deep features are complementary for image representation and that two or more features are better than a single one. The main limitation of feature-extraction models was the high dimensions of features. However, we addressed this issue with PCA-based dimension reduction. We introduced the canonical and discrimination-correlation analysis of fusion features and achieved very good average-classification rates. The effectiveness of the proposed classification system is validated through well-known measures. In future work, we will explore further improvements in the classification approach with symbolic-representation schemes and better ways to handle more features.

## ACKNOWLEDGEMENTS

The authors would like to thank all anonymous reviewers for their valuable suggestions to improve the quality of the research paper.

## REFERENCES

- [1] P. Saxena, A. Maheshwari, S. Tayal and S. Maheshwari, "Predictive Modeling of Brain Tumor : A Deep Learning Approach," *Innovations in Computational Intelligence and Computer Vision, Part of the Advances in Intelligent Systems and Computing Book Series*, vol. 1189, pp. 275–285, 2019.
- [2] H. M. Rai and K. Chatterjee, "Detection of Brain Abnormality by a Novel Lu-Net Deep Neural CNN Model from MR Images," *Machine Learning with Applications*, vol. 2, p. 100004, 2020, DOI: 10.1016/j.mlwa.2020.100004, 2020.
- [3] A. Kharrat, K. Gasmi and M. B. E. N. Messaoud, "A Hybrid Approach for Automatic Classification of Brain MRI Using Genetic Algorithm and Support Vector Machine," *Leonardo Journal of Sciences*, vol. 17, pp. 71–82, 2010.
- [4] Z. Ullah, M. Umar, S. Lee and D. An, "A Hybrid Image Enhancement Based Brain MRI Images Classification Technique," *Medical Hypotheses*, vol. 143, no. May, p. 109922, 2020.
- [5] B. Ural, "A Computer-based Brain Tumor Detection Approach with Advanced Image Processing and Probabilistic Neural Network Methods," *J. of Medical Biological Eng.*, vol. 38, pp. 867–879, 2018.
- [6] D. J. Hemanth, J. Anitha, A. Naaji, O. Geman, D. E. Popescu and L. H. Son, "A Modified Deep Convolutional Neural Network for Abnormal Brain Image Classification," *IEEE Access*, vol. 7, pp. 4275–4283, DOI: 10.1109/ACCESS.2018.2885639, 2018.
- [7] T. J. Alhindi, S. Kalra, K. H. Ng, A. Afrin and H. R. Tizhoosh, "Comparing LBP , HOG and Deep Features for Classification of Histopathology Images," *Proc. of the 2018 IEEE Int. Joint Conf. on Neural Networks (IJCNN)*, DOI: 10.1109/IJCNN.2018.8489329, Rio de Janeiro, Brazil, 2018.
- [8] P. K. Sethy and S. K. Behera, "A Data Constrained Approach for Brain Tumour Detection Using Fused Deep Features and SVM," *Multimedia Tools and Applications*, vol. 80, no. 19, pp. 28745–28760, 2021.
- [9] B. Athiwaratkun and K. Kang, "Feature Representation in Convolutional Neural Networks," *arXiv: 1507.02313*, pp. 6–11, 2015.
- [10] N. Tajbakhsh et al., "Convolutional Neural Networks for Medical Image Analysis: Fine Tuning or Full Training ?" *IEEE Trans. on Medical Imaging*, vol. 35, no. 5, pp. 1299–1312, 2016.
- [11] Z. Zhou, J. Shin, L. Zhang, S. Gurudu, M. Gotway and J. Liang, "Fine-tuning Convolutional Neural Networks for Biomedical Image Analysis," *Proc. of the 2017 IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 4761–4772, DOI: 10.1109/CVPR.2017.506, Honolulu, USA, 2017.
- [12] S. Maheshwaria, V. Kanhangada, R. Pachoria, B. Sulatha V. and U. R. Acharyac, "Automated Glaucoma Diagnosis Using Bit-plane Slicing and Local Binary Pattern Techniques," *Computers in Biology Medicine*, vol. 105, pp. 72–80, 2019.
- [13] S. Oreski and G. Oreski, "Expert Systems with Applications Genetic Algorithm-based Heuristic for Feature Selection in Credit Risk Assessment," *Expert Systems with Applications*, vol. 41, no. 4-2, pp. 2052–2064, 2013.
- [14] S. Li, X. Kang, L. Fang, J. Hu and H. Yin, "Pixel-level Image Fusion: A Survey of the State of the art," *Information Fusion*, vol. 33, pp. 100–112, DOI: 10.1016/j.inffus.2016.05.004, 2016.
- [15] J. Yang, J. Y. Yang, D. Zhang and J. F. Lu, "Feature Fusion: Parallel Strategy vs. Serial Strategy," *Pattern Recognition*, vol. 36, no. 6, pp. 1369–1381, 2003.
- [16] M. Haghigat, M. Abdel-Mottaleb and W. Alhalabi, "Discriminant Correlation Analysis: Real-time

- Feature Level Fusion for Multimodal Biometric," IEEE Transactions on Information Forensics and Security, vol. 11, no. 9, pp. 1984 - 1996, 2016.
- [17] S. Roheda and H. Krim, "Decision Level Fusion : An Event Driven Approach," Proc. of the 26<sup>th</sup> IEEE European Signal Processing Conf. (EUSIPCO), vol. 9560, pp. 2598–2602, Rome, Italy, 2018.
- [18] I. Khouli and N. Idrissi, "Cervical Cancer Detection and Classification Using MRIs," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 08, no. 02, pp. 141-158, DOI: 10.5455/jjcit.71-1640595124, June 2022.
- [19] D. Azzouz and S. Mazouzi, "A Hyper –surface –based Modelling and Correction of Bias Field in MR Images," Jordanian Journal of Computers and Information Technology (JJCIT), vol. 07, no. 03, pp. 223-238, DOI: 10.5455/jjcit.71-1617051919, Sep. 2021.
- [20] H. Sultan, N. Salem and W. Al-Atabany, "Multi-classification of Brain Tumor Images Using Deep Neural Network," IEEE Access, vol. 7, pp. 69215–69225, DOI:10.1109/ACCESS.2019.2919122, 2019.
- [21] D. Chen, K. C. Chan and X. Wu, "Gene Expression Analysis Using Genetic Algorithm Based Hybrid Approaches," Proc. of the IEEE Congress on Evolutionary Computation, pp. 963–969, Hong Kong, china, 2008.
- [22] Radiopaedia, "Cases," [Online], Available: <https://radiopaedia.org/cases/>, Accessed on 26 June 2020 .
- [23] National Cancer Institute, "Cancer Imaging Archive," [Online], Available: <https://www.cancerimagingarchive.net/>, Accessed on 30 June 2020.
- [24] Biomedical Image Analysis Group, "IXI Dataset," [Online], Available: <http://brain-development.org/ixi-dataset>, Accessed on 15 June 2020.
- [25] K. A. Johnson and J. Alex Becker, "The Whole Brain," Atlas, [Online], Available: <http://www.med.harvard.edu/AANLIB/>, Accessed on 18 June 2021.
- [26] Wikipedia, "Confusion Matrix," [Online], Available: [http://en.wikipedia.org/wiki/Confusion\\_matrix](http://en.wikipedia.org/wiki/Confusion_matrix), Accessed on 20 Aug 2021.

### ملخص البحث:

في هذه الورقة، نقترح إطاراً فعالاً للاندماج على مستوى السمات من أجل تصنيف صور الرنين المغناطيسي للدماغ، باستخدام التعلم العميق، واستخلاص السمات؛ وبالذات مخطط الميول الموجّهة (HOG)، والأنماط الثنائية المحلية (BLPs). ويهدف الإطار المقترح إلى: تحديد السمات المثالية باستخدام خوارزمية جينية (GA)، واكتشاف سمات الطبقات المتصلة كلياً باستخدام شبكة عصبية التفاضلية (CNN) دقيقة الضبط، وتطبيق التحليل الارتباطي القانوني (CCA) والتحليل الارتباطي التمييزي (DCA) في إحداث الاندماج على مستوى السمات.

تم إجراء التجارب المكثفة لإظهار مستوى الأداء التصنيفي للنظام المقترح على ثلاثة من مجموعات البيانات المرجعية (RD-DB1)، و (TCIA-IXI-DB2)، و (TWB-HM-DB3). وكانت نسب الدقة لمجموعات البيانات المذكورة 68.69%، و 90.35%، و 93.15% على الترتيب عند استخدام تقنية (CCA)، بينما بلغت تلك النسب 77.22%، و 100%، و 99.40% على الترتيب عند استخدام تقنية (DCA). ويمكن استنتاج أن الإطار المقترح تفوق على العديد من النماذج ذات العلاقة المستخدمة في الدراسات السابقة.



# COMBINATION OF DEEP-LEARNING MODELS TO FORECAST STOCK PRICE OF AAPL AND TSLA

Zahra Berradi<sup>1</sup>, Mohamed Lazaar<sup>2</sup>, Oussama Mahboub<sup>1</sup>, Halim Berradi<sup>3</sup> and Hicham Omara<sup>4</sup>

(Received: 20-Jun.-2022, Revised: 27-Aug.-2022, Accepted: 19-Sep.-2022)

## ABSTRACT

*Deep Learning is a promising domain. It has different applications in different areas of life and its application on the stock market is widely used due to its efficiency. Long Short Term Memory (LSTM) proved its efficiency in dealing with time-series data due to the unique hidden unit structure. This paper integrated LSTM with attention mechanism and sentiment analysis to forecast the closing price of two stocks; namely, APPL and TSLA, from the NASDAQ stock market. We compared our hybrid model with LSTM, LSTM with sentiment analysis and LSTM with Attention Mechanism. Three benchmarks were used to measure the performance of the models; the first one is Mean Square Error (MSE), the second one is Root Mean Square Error (RMSE) and the third one is Mean Absolute Error (MAE). The results show that the hybridization is more accurate than the LSTM model alone.*

## KEYWORDS

*Deep learning, Hybrid model, LSTM, Attention mechanism, Sentiment analysis.*

## 1. INTRODUCTION

Since the beginning of the stock market, forecasting the price of stocks is still one of the most challenging tasks for every investor. Searching for and developing new effective technologies are essential. The volatility and non-stationary aspect of the stock market push researchers to find the most rewarding AI technologies to predict its behavior. Based on the published research papers over the last years, we can easily watch the evolution of artificial intelligence and its applications in all aspects of life, especially in the development of stock-market forecasting. Its appearance and improvement go along with AI development. We mentioned in this paper the most significant waves that have affected the stock market in the 20<sup>th</sup> and the beginning of the 21<sup>st</sup> century.

We notice the statistical approach as a first wave used for the prediction problem. For example, in 1901, the mathematician Karl Pearson created the method of Principal Component Analysis to reduce the dimension of the input data. Then, other models appeared continuously, such as Auto-regressive Moving Average (ARMA) [1], Auto-regressive Integrated Moving Average (ARIMA) [2], the Generalized Auto-regressive Conditional Heteroskedasticity (GARCH) [3] and Quadratic Discriminant Analysis (QDA) [4].

The second wave is machine learning. It is an extension of AI that uses the input and the output of a specific system as features to be able to solve a prediction problem or a classification problem. It permits a machine to learn automatically without needing to program it explicitly. Many models exist in literature and have proved their efficiency, such as Support Vector Machine (SVM) [5], Genetic Algorithm (GA) and Multi-layer Perceptron (MLP) [6]. Besides those mentioned, there is a famous one: Neural Networks (NN). It is used to solve time-series data problems and has gained popularity among researchers, because it can perfectly handle non-stationary systems, including the stock market.

The development of NN leads to another branch of ML called Deep Learning (DL). It is characterized by a specific hidden-node calculation in the hidden layer. The power of DL remains in the ability to control time-series data with more precision than NN. It shows its efficiency throughout time in many areas, such as speech recognition, image classifications, face recognition [7], natural language processing, sentiment analysis, translation and health care [8]. Some of the well-known types of DL

- 
1. Z. Berradi and O. Mahboub are with National School of Applied Sciences, Abdelmalek Esaadi University, Tetouan, Morocco. Emails: berradi.zahra@gmail.com and mahbouboussama@gmail.com
  2. M. Lazaar is with ENSIAS, Mohammed V University in Rabat, Morocco. Email: mohamed.lazaar@ensias.um5.ac.ma
  3. H. Berradi is with Telecommunication Systems Laboratory, National School of Applied Sciences, Ibn Tofail University, Kenitra, Morocco. Email: halimberradi@gmail.com
  4. H. Omara is with Faculty of Sciences, Abdelmalek Esaadi University, Tetouan, Morocco. Email: Hichamomara@gmail.com

models are Deep Multi-layer Perceptron (DMLP), Recurrent Neural Networks (RNNs), Long Short Term Memory (LSTM) [9], Gated Recurrent Unit (GRU), Convolutional Neural Networks (CNNs), Restricted Boltzmann Machines (RBMs), Deep Belief Networks (DBNs), Autoencoders (AEs) and Deep Reinforcement Learning (DRL).

DMLP is a model containing three layers: the first is an input layer, the second contains more than one hidden layer and the third is an output layer. RNN differentiates from DMLP by a recurrent unit, but it suffers from the vanishing-gradient problem. Long Short Term Memory (LSTM) was the solution to the vanishing problem. It has demonstrated its capability on speech recognition as mentioned in the works of [10], text classification [11], natural language processing and the stock market fluctuation [12] [13]-[14]. Gated Recurrent Unit (GRU) is a variation of RNN. It is similar to the LSTM unit with some differences in how to calculate the hidden-unit value [15]. CNN was first introduced in image processing, then its application expanded in other fields [16]. It contains several layers; the first is an input layer, the second is a convolutional layer, the next is a max-pooling layer, then a fully connected layer with dropout, while the last layer is the output layer.

The model RBM is used for unsupervised learning, which is applied in classification and dimension reduction. The particularity of RBM is to extract hidden patterns in the system [17]. Deep Belief Network (DBN) is also an unsupervised-learning model composed of RBMs. The model autoencoder (AE) is mainly used for feature extraction [18] and dimensionality reduction. It gets more accurate results when it is combined with LSTM [19]-[20]. In addition, Deep-learning models have proved their efficiency in areas beyond finance. Their efficiency expands into other non-stationary, volatile domains, such as education, robotics, smart cities [21] and health care.

Pattern recognition searches for repetitive patterns in the stock market over time. To extract these patterns, investors may rely on Attention Mechanism (AT), Convolution Neural Network (CNN) and Wavelet Transform (WT) to reach an accurate prediction. To make use of this type of prediction, traders will consider using volume, candlestick charts, as well as Simple and Exponential Moving Averages (SMA, EMA).

The third wave is sentiment analysis (SA). People's opinion on social media, such as Twitter, reflects the behavior and the new direction of the prices on the stock market. By the massive daily information, SA has become more desirable as mentioned by [22]-[23]. The core idea is to get from a sentence or an article the polarity; whether the sentence is positive or negative. CNN is an example of a deep-learning model that is used to get this polarity.

The last wave is hybridization. Data analysts frequently integrate the positive aspects of two, three or more models to create a reliable forecast. This type of model is a powerful tool for predicting the next stock-price movements. For example, LSTM works perfectly with historical prices, but the market is influenced by many other factors, such as the investors' opinion and the internal and external political factors of the country. So, limiting focus only on the historical data will not lead to a better prediction. That is the reason behind combining multiple models.

In other words, two primary business models are applied nowadays to predict stock prices which traders use to make good decisions.

- 1) Fundamental analysis: it is based on many factors, such as the company's evolution and performance, the politics and the news.
- 2) Technical analysis: technical analysis depends on the historical prices, the volume and other indicators that depend on the historical price.

In this paper, we are interested in making predictions of stock prices per day using both business models by using the historical price and the news from Twitter. By considering the huge effectiveness of deep learning, our research is about integrating the sentiment analysis, attention mechanism (AM) and LSTM are used to forecast the stock price of both AAPL and TSLA. The aim is to get a more accurate prediction by adding the sentiment of the tweets to the LSTM model with AM and using the results to support the investors' strategy.

This paper is arranged as follows: Section 2 is about related work. In Section 3, we explain the deep-learning model; namely, LSTM, SA and AM. Section 4 is about the experimental results and discussion. As for Section 5, it provides a summary of the findings.

## 2. RELATED WORK

Researchers always tried to find a developed model to predict the stock market. Many research papers proved the power of deep learning, sentiment analysis and other models to analyze the financial market.

The hybrid model has been widely used in recent years due to the beneficial results of getting more than one strength point of each model for better performance. Jin et al. [24] combined LSTM, sentiment analysis and Empirical Modal Decomposition EMD to forecast the closing price of AAPL. Berradi and Lazaar [25] integrated PCA and RNN to predict the daily closing price of some stocks on the Casablanca Stock Exchange. Ismail and Awjan [26] combined Empirical Modal Decomposition with the Exponential Smoothing Method to improve the forecasting results. Qiu et al. [27] used LSTM, Attention Mechanism and Wallet Transform to predict the stock prices. The proposed model outperforms LSTM, GRU and LSTM with Wallet Transform. Jin et al. [28] proposed a hybrid model to predict the stock price, using LSTM and sentiment analysis applied on Shanghai Stock Exchange (SSE).

On the other hand, the news is a factor that can affect the fluctuation in the stock market beyond other factors, like economy, natural phenomena and politics. The information that can impact the market is characterized by two factors: perfect timing and reliability. For example, on June 7, 2021, the Food and Drug Administration (FDA) admitted Biogen's Alzheimer's treatment. This information directly affected the Biogen (BIIB) stock price; the opening price was 295.35\$ and the closing price was 395.85\$ on the same day. So, the fact that the historical data of stock prices is the only reasonable way to have an accurate prediction is not valid. The correlation between the stock price and the news has been proven in many research papers.

Rakhi Batra and Sher Muhammad Daudpota [29] integrated StockTwits with sentiment analysis to predict the behavior of AAPL. They used the SVM model (a supervised machine-learning algorithm) to predict the next day's closing price. Mohan et al. [30] improved the predictions of S&P500 by gathering the news related to the mentioned index and the historical data for three years. They used LSTM and Facebook Prophet as models. Kirange et al. [31] emphasized the idea of the news effect on the stock price. They used SVM, Naïve Bayes and KNN to prove the correlation between stock-price movement and the news. Abraham et al. [32] presented an approach to predict the changes in Bitcoin and Ethereum based on Twitter and google trends' data.

Liu et al. [33] proposed a model containing CNN and LSTM to analyze the quantitative strategy in stock markets. In addition, Vargas et al. [34] proposed a hybrid model with CNN and LSTM for the intraday directional movements of S&P500 index using the news and seven technical indicators. Lee et al. [35] suggested a hybrid model called Recurrent Convolutional Neural Network (RCN) that combines CNN, sequence modeling and word embedding. The main work of this model is to extract the polarity from the news, then add technical indicators for stock-price forecasting.

Li et al. [36] stated a hybrid model which combined LSTM and Naïve Bayes. The Naïve Bayes was used for extracting the sentiment from the forum and LSTM for the prediction. Pang et al. [12] proposed an LSTM neural network with an embedded layer and LSTM with an Automatic Encoder neural network to forecast Shanghai A-share composite index and Sinopec. The accuracy of the two models was 57.2% and 56.9%, respectively. The input data is multi-stock high-dimensional historical data. Yan Hongju and Hongbing Ouyang [37] combined wavelet analysis with Long Short Term Memory (LSTM) to forecast the daily closing price of the Shanghai Composite Index. They compared the proposed models with other machine-learning techniques: Multi-layer Perceptron (MLP), Support Vector Machine (SVM) and K-nearest neighbors. They concluded that LSTM with wavelet analysis gives better accuracy. Chen et al. [38] used LSTM with Attention Mechanism to predict the daily return ratio of the HS300 index, while they used the embedding layer to extract the convenient features. Kim et al. [39] proposed a hybrid model with LSTM and GARCH-type models to forecast the volatility of the KOSPI 200 index.

Shen et al. [40] proposed a new model composed of Deep Belief Network (DBN) and Continuous Restricted Boltzmann Machines (CRBMs) to forecast currency exchange rates. They compared their model with Feed Forward Neural Network (FFNN) and found that the proposed model performs better than FFNN.

### 3. PROPOSED MODEL

The power of the hybrid model lies in the combination of different crucial points of each model, which produces a more reliable and robust model. In this section, we provide the mathematical background of each model.

#### 3.1 LSTM Model

LSTM is a type of RNN. [41] invented LSTM to solve the problem of long-term dependencies caused by the traditional RNN. From recently published papers, LSTM is more widely used compared to other deep-learning models. It is applied in various domains, such as speech recognition, sentiment analysis and time-series problems. It is composed of one input layer, one hidden layer and one output layer. The hidden layer is formed of LSTM nodes (as shown in Figure 1).

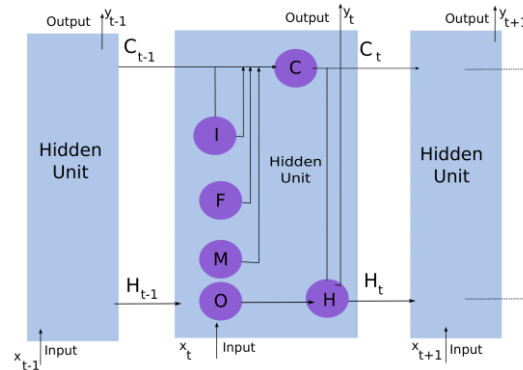


Figure 1. The composition of a hidden node of long short term memory (LSTM).

Each node value is calculated using Equations 1-6:

$$\text{The forget gate is: } F_t = \sigma(W_F x_t + U_F h_{t-1} + B_F) \quad (1)$$

$$\text{The second equation, called the output gate: } O_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (2)$$

$$\text{The input gate: } i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (3)$$

$$\text{The state gate: } S_t = \tanh(W_s x_t + U_s h_{t-1} + b_s) \quad (4)$$

$$\text{The hidden state: } h_t = O_t * \tanh c_t \quad (5)$$

$$\text{The cell state: } c_t = c_{t-1} * F_t + S_t * i_t \quad (6)$$

where  $x_t$  the input vector, the function  $\tanh$  is the hyperbolic tangent function,  $\sigma = \frac{1}{1+e^{-x}}$  is the sigmoid function, where  $x \in \mathbb{R}$ ,  $*$  is the element-wise product.

The other parameters in the hidden node  $F_t$ ,  $O_t$ ,  $i_t$ ,  $S_t$ ,  $h_t$  and  $c_t$  are forget gate, output gate, input gate, state gate, hidden state and cell state, respectively, at time  $t$ .

where  $B_F$ ,  $b_o$ ,  $b_i$ ,  $b_s$  are bias.  $W_F$ ,  $W_o$ ,  $W_i$ ,  $W_s$ ,  $U_F$ ,  $U_o$ ,  $U_i$ ,  $U_s$  are the weight matrix. The choice of hyper-parameters is crucial for better performance. LSTM has many hyper-parameters that affect its performance: the number of hidden layers, the number of nodes in each hidden layer, the weight initialization, the learning rate, the activation function, the number of epochs, the bias initialization, the optimization algorithms and the decay rate.

#### 3.2 Sentiment Analysis

Sentiment Analysis is an approach used to extract the polarity from sentences. Many artificial models find the polarity from text, such as CNN (Conventional Neural Network). It is used in computer vision and was invented to deal with image treatments, such as image classification. CNN model is composed of many layers. The first one is the input layer, an input matrix of fixed dimensions. The second one is the convolution layer, the third one is the max-pooling layer, the fourth layer is the fully connected layer with dropout and the last one is the output layer.

Sentiment analysis is used to find whether the sentiment from the tweeter is positive (bullish) or negative (bearish). The first step is to get the tweets from the tweeter. The second step is the pre-processing step, which is about cleaning the text, because it contains a lot of noise, such as URL, emoji's, hashtags and numbers. The third step is word vector embedding. In this step, the method GloVe [42] was used for word representations. The architecture of the CNN is as follows.

### 3.3 Attention Mechanism

The attention mechanism was inspired by the natural-eye functions which focus only on a specific thing while looking at the whole image. It was first introduced in the Human Vision, then in Natural-language Processing (NLP) [43]-[44]. It also has proved its effectiveness in time-series data, such as speech recognition [45]. This fact led researchers to apply it to the stock-price prediction problem. There are two types of attention: hard attention and soft attention. The first one (hard attention) gives importance to one input element and does the training, which means that more training is required for better accuracy. The soft attention mechanism gives different attention weights to each of the input elements based on their importance (as shown in Figure 3). The work of [44] supposed that each output  $y_t$  is a probability of the previous output  $y_{t-1}$ , the hidden state  $s_t$  (see Equation 7) and the context vector  $c_t$  (as shown in Equation 8), where  $t \in \{0; \dots; T\}$  and  $f$  are nonlinear function.

$$s_t = f(c_t, s_{t-1}, y_{t-1}) \tag{7}$$

where  $s_{t-1}$  is the previous recurrent hidden state. The determination of the context vector is made by summation of  $a_{t,i}$  and  $h_i$ .

$$c_t = \sum_{i=1}^{i=T} a_{t,i} h_i \tag{8}$$

The value of  $a_{t,i}$  represents the weight of each hidden  $h_i$ , which means how important the  $h_i$  can be for our model (as shown in Equation 9).

$$a_{t,i} = \frac{\exp(e_{t,i})}{\sum_{k=1}^T \exp(e_{t,k})} \tag{9}$$

$e_{t,i}$  represents the *alignment model*, as shown in Equation 10.

$$e_{t,i} = g(s_{t-1}, h_i) \tag{10}$$

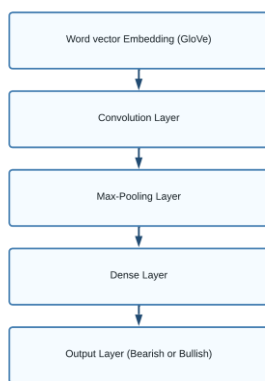


Figure 2. CNN architecture.

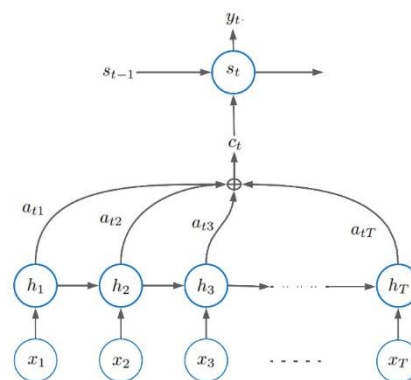


Figure 3. The pattern of attention mechanism taken from [44].

### 3.4 The Hybrid Model

Long Short Term Memory is widely considered the solution to long-term dependencies. On the other hand, the Attention Mechanism is capable of keeping the most reliving information from the input. Based on the work of Berradi et al. [46], hybridization of many basic models gives more accurate results. The work of Hollis et al. [47] explained that the combination of LSTM and AM models outperforms LSTM alone. These facts lead us to combine multiple deep-learning models to get a more robust one. The proposed model contains the input layer, the LSTM layer, the attention layer, the dense layer and the output layer. Figure 4 represents the different types of layers defined in our deep model.

In the following, the pseudo code used for the hybrid model is presented.

**Algorithm 1** The algorithm of the hybrid model

H

```

1: Input  $x = [x_1, \dots, x_8]$ ,  $x_i \in \mathbb{R}^{512}$ 
2: Define the attention mechanism model (AttentionDecoder)
3: Build the hybrid model
model = Sequential()
model.add(LSTM(20, activation = 'tanh', input_shape = (8, 1), return_sequences = True))
model.add(AttentionDecoder(16, 20))
model.add(Dense(1, activation = 'tanh'))
adam = keras.optimizers.Adam(lr = 0.001, beta1 = 0.9, beta2 = 0.999, epsilon = None, decay = 0.0)
model.compile(loss='mse', optimizer='adam')
4: Predict the closing price of the stock.

```

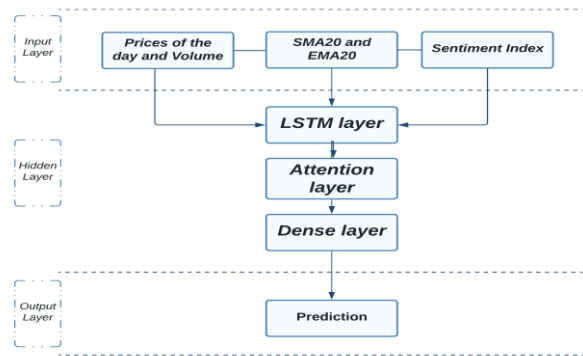


Figure 4. The deep-learning structure.

**3.5 Examination Tools**

To compare the models, we used benchmarks such as the mean square error ( $E_{MSE}$ ), the root mean square error ( $E_{RMSE}$ ) and the mean absolute error ( $E_{MAE}$ ). The closer the error to zero, the better the model performance.

The mean square error:

$$E_{MSE} = \frac{1}{M} \sum_{i=1}^M \|\hat{Y}_i - Y_i\|^2 \quad (11)$$

The root mean square error:

$$E_{RMSE} = \sqrt{\frac{1}{M} \sum_{i=1}^M \|\hat{Y}_i - Y_i\|^2} \quad (12)$$

The mean absolute error:

$$E_{MAE} = \frac{1}{M} \sum_{i=1}^M |\hat{Y}_i - Y_i| \quad (13)$$

where  $Y_i$  and  $\hat{Y}_i$  are the estimated output and the actual output, respectively, and  $M$  is the last time step.

**4. EXPERIMENTAL RESULTS AND DISCUSSION**

The script was executed in Windows 10 (x64), Intel(R) Core(TM) i5-8250U processor running at 1.80 GHz (four CPUs), with 8 GB of RAM, 256 Hard Disk type SSD.

This paper aims to search for new factors and intelligent models that will enhance the prediction of AAPL and TSLA. The effect of social media on the fluctuation of prices is paramount. So, the first attempt was to get tweets from Twitter using the keywords AAPL and TSLA. After collecting the tweets from Twitter for five months using API, we found that the MSE of LSTM with sentiment analysis related to AAPL was close to 1. Therefore, instead of being a helping factor, it represents a perturbation. After several simulations, we found that MSE in train data was 0.0182478 and MSE in test data was 0.0269575. The relevance of the tweets that we collected plays a massive impact on our prediction model. This finding makes us search for a new way to extract only the most relevant tweets.

Dealing with the false tweets and differentiating between what is real and fake is very important. The

tweets should be from a confidential source. Otherwise, it can affect the real price in a wrong way and lead to a bad decision. We found that the website "www.stockstwit.com" has a policy of getting only the relevant information. We intend to combine LSTM with AM and add the sentiment analysis for better forecasting. Figure 5 represents the general steps used to forecast the closing prices of TSLA and APPL.

#### 4.1 Data

We collected all the tweets from the website *www.stockstwit.com* from 07/03/2019 to 30/04/2021. The keywords were "AAPL" and "TSLA". It took almost three days for the script to get the needed data. For the historical data related to the price of AAPL, we collected it from Yahoo! Finance *www.yahoo.com*. Our previous work [48] was performed on APPL data; we applied LSTM, ANN and GRU to predict the closing stock prices of APPL. This work is an extension of it. We decided to compare the LSTM, LSTM+SI and the hybrid model. But, having only one stock is not enough; thus, we added the stock TSLA, because it's in the same stock market (Nasdaq) to see how the model would work.

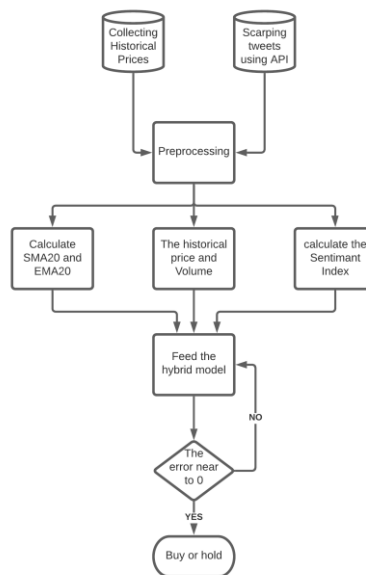


Figure 5. Flowchart of the forecasting steps.

The data needs a pre-processing step to remove all the noise. Table 1 and Table 2 show a sample of the technical indicators used. The data was divided into two semi-groups: 80% training data and 20% testing data.

Table 1. Sample of technical indicators of TSLA after pre-processing.

Date	Open	High	Low	Close	Adj Close	Volume
2019-04-18	54.24	54.96	53.95	54.65	54.65	29381500
..	..	..	..	..	..	..
2021-04-29	699.51	702.25	668.5	677	677	28845400

Table 2. Sample of technical indicators of AAPL after pre-processing.

Date	Open	High	Low	Close	Adj Close	Volume
2019-04-18	50.77	51.03	50.63	50.96	49.91	96783200
..	..	..	..	..	..	..
2021-04-29	136.47	137.07	132.44	133.47	133.47	151101000

First, we calculated the SMA20 and EMA20 based on [25], then we extracted the sentiment of tweets of each day (bullish or bearish). After that, we calculated the Sentiment Index (SI) based on the following equation which was taken from [24]:

$$SI = \ln \left( \frac{1+\alpha}{1+\beta} \right) \quad (14)$$

where  $\alpha$  is the number of bullish tweets per day and  $\beta$  is the number of bearish tweets per day. The features are: opening price, highest price, lowest price, adjusted price, volume, SMA20, EMA20 and SI. All these features are different; so the normalization step was essential. We transformed the original data into numbers between 0 and 1 to improve the training. The implementation was developed using Python3. These hyper-parameters were chosen based on our previous work [48].

## 4.2 Results and Discussion

Three metrics (MSE, MAE and RMSE) were used to validate the accuracy of the models. The closer the value of the error to 0, the better the performance.

Table 3. The hyper-parameters of the models.

Hyper parameters	Values
Number of hidden layers	1
Number of nodes	20
Batch size	10
Epochs	100
The activation function	<i>tanh</i>
Optimizer	Adam
Learning rate	0.001

Table 4. Errors of the models related to TSLA stock data.

	<i>LSTM</i>	<i>LSTM with SI</i>	<i>LSTM with AM</i>	<i>Hybrid model</i>
$E_{MSE}$ (train)	0.0027509	0.0026156	0.0003709	0.0007796
$E_{MSE}$ (test)	0.0135314	0.0126432	0.0063939	<b>0.0058008</b>
$E_{RMSE}$ (train)	0.0480684	0.0629436	0.0198917	0.0224225
$E_{RMSE}$ (test)	0.1108812	0.1335756	0.0771000	<b>0.0765036</b>
$E_{MAE}$ (train)	0.0449642	0.0465490	0.0154761	0.0183373
$E_{MAE}$ (test)	0.1144399	0.11970902	0.07385018	<b>0.0693858</b>

For TSLA stock prices, the lowest error was of Mean Square Error (MSE) using the LSTM model. It was with a value of 0.0135314. For the model LSTM with SI, MSE was the lowest error too with 0.0126432. For the model LSTM with AM, the lowest error was 0.0063939. For the hybrid model, the lowest error was 0.0058008. We conclude that the hybrid model performs better than the other models (as shown in Table 4). In other words, the accuracy of the LSTM model is 98.64%, while the accuracy of the hybrid model is 99.42%. We notice the impact and the improvement of the model's performance using hybridization.

Table 5. The Error of the models related to APPL stock.

	<i>LSTM</i>	<i>LSTM with SI</i>	<i>LSTM with AM</i>	<i>Hybrid model</i>
$EMSE$ (train)	0.0031031	0.0025217	0.0005482	0.0007796
$EMSE$ (test)	0.0127536	0.0107397	0.0033089	<b>0.0032915</b>
$ERMSE$ (train)	0.0493602	0.0597210	0.0246819	0.0278273
$ERMSE$ (test)	0.1082771	0.1194630	0.0528034	<b>0.0548668</b>
$EMAE$ (train)	0.0538247	0.1077374	0.0199947	0.0216156
$EMAE$ (test)	0.1214208	0.1748030	0.0535530	<b>0.0506833</b>

For APPL stock prices (as shown in Table 5), the lowest Mean Square Error (MSE) was using the LSTM model. It was with a value of 0.0127536. For the model LSTM with SI, MSE was the lowest error with 0.0107397. The MSE for the model LSTM with AM was 0.0033089. For the hybrid model, it performs better than the other models, because the MSE was 0.0032915. We conclude that the hybrid model performs better for APPL Stock than other models.



As shown in Figure 6 and Figure 7, it represents the prediction of the closing price of AAPL using the LSTM model and the hybrid model, respectively. The prediction of the closing price using the hybrid model between April 2019 and April 2020 proved to be close to the actual closing price. But, from April 2020 to April 2021, the gap between the two curves of the predicted price and the actual price of APPL extends. We can explain this due to other factors, such as politics and rumors. This fact leads us to believe that technical analysis alone is not enough; using fundamental analysis is also essential to eliminate the risk of losing money in long-term trading.

We noticed also that the SA didn't have a significant impact on the prediction, because the real problem is in the text itself, whether real or fake, a sarcasm or rumors, which makes the sentiment inaccurate. We get this conclusion based on our first attempts. After collecting the tweets from Twitter for five months using API, we found that the MSE of LSTM with sentiment analysis related to AAPL was close to 1. Therefore, instead of being a helping factor, it represents a perturbation. This finding leads us to conclude that SA has several limitations on the implementation side and needs more work in the future for obtaining better performance.

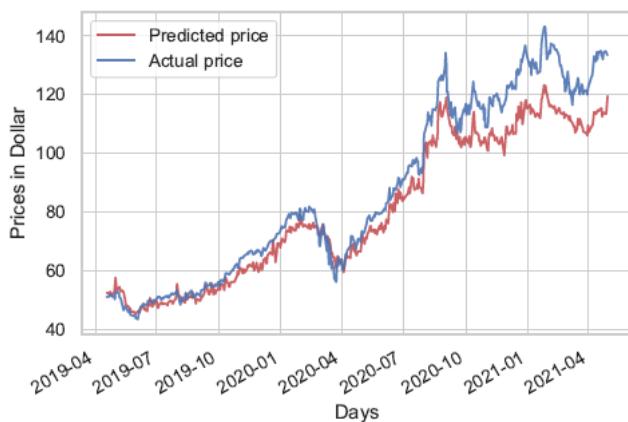


Figure 6. The prediction of AAPL using LSTM.



Figure 7. The prediction of AAPL using the hybrid model.

Figure 8 and Figure 9 represent TSLA and AAPL predictions of the closing price, respectively, using LSTM, LSTM with attention mechanism (LSTM+AM) and the hybrid model (LSTM+AM+SI). For both stocks, AAPL and TSLA, we can observe that between 2019 and 2020, the prediction was accurate compared with the period between 2020 and 2021. The movement of AAPL and TSLA closing prices in 2021 was chaotic. We can explain this perturbation by the beginning of the lockdown in 2021, which represents a very chaotic period in the whole world. People tended to spend more time on social media; thus, the impact of fake news and rumors is more powerful than in the period from April 2019 to April 2020.

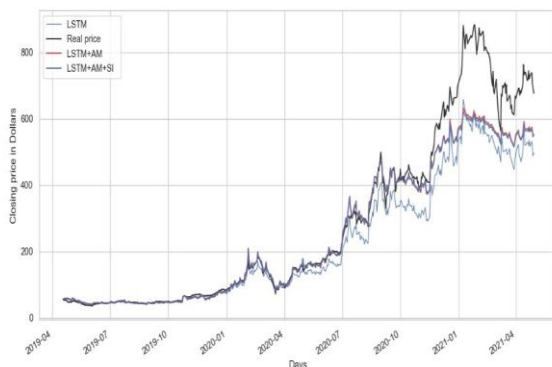


Figure 8. TSLA prediction using LSTM, LSTM+AM and the hybrid model.

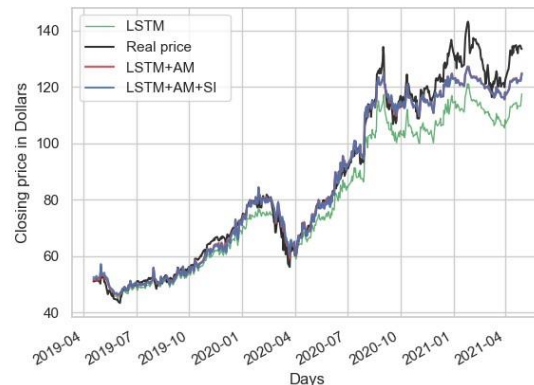


Figure 9. APPL prediction using LSTM, LSTM with AM and the hybrid model.

However, if we compare the prediction of LSTM and LSTM+SI with that of the hybrid model, we conclude that the hybrid model surpasses the other models. Taking a decision based on the proposed model is beneficial, because it minimizes the risk of losing money while trading. The model is not perfect, but it will help traders make good decisions about buying or holding stocks.

## 5. CONCLUSION AND FUTURE WORKS

This paper is about using a hybrid model to forecast the stock prices of AAPL and TSLA. The chaotic movement of these stocks makes the prediction problem very challenging. We integrate LSTM, Attention Mechanism and Sentiment Analysis (the hybrid model) to solve this issue. The attention Mechanism with LSTM makes the prediction accuracy better than with the LSTM alone. In our case, the sentiment analysis with LSTM did not bring too much value to the prediction. Otherwise, the integration of LSTM with AM makes the prediction perform better than LSTM alone and LSTM with sentiment analysis. In conclusion, the hybrid model has higher accuracy than the other models. We also conclude that the news affects the results, but with minor improvement in this experiment. Therefore, this issue leads us to look for new methods to extract only the most relevant ones. Future work will be about integrating our proposed model into a simulator platform, such as Ninja trader or Meta trader to help traders predict the next move of the stock prices and make the right decision.

## ACKNOWLEDGEMENTS

The authors would like to thank the editor and the three anonymous reviewers for their insightful comments, which helped improve the paper.

## ETHICS AND IMPLICATIONS

The authors declare that they have no conflicts of interest to disclose. We are not responsible for anyone who uses these results in real-life trading without consulting professionals.

## REFERENCES

- [1] M. M. Rounaghi and F. N. Zadeh, "Investigation of Market Efficiency and Financial Stability between S&P 500 and London Stock Exchange: Monthly and Yearly Forecasting of Time Series Stock Returns Using ARMA Model," *Physica A: Statistical Mechanics and its Applications*, vol. 456, pp. 10–21, 2016.
- [2] A. A. Ariyo, A. O. Adewumi and C. K. Ayo, "Stock Price Prediction Using the ARIMA Model," *Proc. of the 16<sup>th</sup> IEEE International Conference on Computer Modeling and Simulation (UKSim-AMSS)*, pp. 106–112, Cambridge, UK 2014.
- [3] P. H. Franses and D. V. Dijk, "Forecasting Stock Market Volatility Using (Non-linear) Garch Models," *Journal of Forecasting*, vol. 15, no. 3, pp. 229–235, 1996.
- [4] R. A. K. Cox and G. W.-Y. Wang, "Predicting the US Bank Failure: A Discriminant Analysis," *Economic Analysis and Policy*, vol. 44, no. 2, pp. 202–211, 2014.
- [5] W. Huang, Y. Nakamori and S.-Y. Wang, "Forecasting Stock Market Movement Direction with Support Vector Machine," *Computers & Operations Research*, vol. 32, no. 10, pp. 2513–2522, 2005.
- [6] A. V. Devadoss and T. A. A. Ligor, "Forecasting of Stock Prices Using Multi Layer Perceptron," *International Journal of Computing Algorithm*, vol. 2, pp. 440–449, 2013.
- [7] N. Singhal, V. Ganganwar, M. Yadav, A. Chauhan, M. Jakhar and K. Sharma, "Comparative Study of Machine Learning and Deep Learning Algorithm for Face Recognition," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 7, no. 3, pp. 313–325, 2021.
- [8] F. Assiri and M. Alrehaili, "Development of Ensemble Machine Learning Model to Improve COVID-19 Outbreak Forecasting," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 08, no. 2, pp. 48 – 58, DOI: 10.5455/jjcit.71-1640174252, 2022.
- [9] M. Hiransha, E. Ab Gopalakrishnan, V. K. Menon and K. P. Soman, "NSE Stock Market Prediction Using Deep-learning Models," *Procedia Computer Science*, vol. 132, pp. 1351–1362, 2018.
- [10] A. Graves, N. Jaitly and A.-R. Mohamed, "Hybrid Speech Recognition with Deep Bidirectional LSTM," *Proc. of the 2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, pp. 273–278, Olomouc, Czech Republic, 2013.
- [11] C. Zhou, C. Sun, Z. Liu and F. Lau, "A C-LSTM Neural Network for Text Classification," *arXiv preprint, arXiv: 1511.08630*, 2015.
- [12] X. Pang, Y. Zhou, P. Wang, W. Lin and V. Chang, "An Innovative Neural Network Approach for Stock Market Prediction," *The Journal of Supercomputing*, vol. 76, no. 3, pp. 2098–2118, 2020.

- [13] J. Huang, J. Chai and S. Cho, "Deep Learning in Finance and Banking: A Literature Review and Classification," *Frontiers of Business Research in China*, vol. 14, pp. 1–24, 2020.
- [14] D. Shah, H. Isah and F. Zulkernine, "Stock Market Analysis: A Review and Taxonomy of Prediction Techniques," *International Journal of Financial Studies*, vol. 7, no. 2, DOI: 10.3390/ijfs7020026, 2019.
- [15] J. Chung, C. Gulcehre, K. H. Cho and Y. Bengio, "Empirical Evaluation of Gated Recurrent Neural Networks on Sequence Modeling," arXiv preprint, arXiv: 1412.3555, 2014.
- [16] A. Vidal and W. Kristjanpoller, "Gold Volatility Prediction Using a CNN-LSTM Approach," *Expert Systems with Applications*, vol. 157, DOI: 10.1016/j.eswa.2020.113481, 2020.
- [17] Y. Bengio, "Deep Learning of Representations for Unsupervised and Transfer Learning," *Proc. of ICML Workshop on Unsupervised and Transfer Learning*, vol. 27, pp. 17–36, 2012.
- [18] N. Bahadur and R. Paffenroth, "Dimension Estimation Using Autoencoders," arXiv preprint, arXiv: 1909.10702, 2019.
- [19] A. Essien and C. Giannetti, "A Deep Learning Framework for Univariate Time Series Prediction Using Convolutional LSTM Stacked Autoencoders," *Proc. of the 2019 IEEE Int. Symposium on Innovations in Intelligent Systems and Applications (INISTA)*, Sofia, Bulgaria, pp. 1–6, 2019.
- [20] W. Bao, J. Yue and Y. Rao, "A Deep Learning Framework for Financial Time Series Using Stacked Autoencoders and Long-short Term Memory," *PLOS One*, vol. 12, no. 7, p. e0180944, 2017.
- [21] G. H. Merabet, M. Essaïdi, M. Ben Haddou et al., "Intelligent Building Control Systems for Thermal Comfort and Energy Efficiency: A Systematic Review of Artificial Intelligence-assisted Techniques," *Renewable and Sustainable Energy Reviews*, vol. 144, DOI: 10.1016/j.rser.2021.110969, 2021.
- [22] V. S. Pagolu, K. N. Reddy, G. Panda and B. Majhi, "Sentiment Analysis of Twitter Data for Predicting Stock Market Movements," *Proc. of the 2016 IEEE Int. Conf. on Signal Processing, Communication, Power and Embedded System (SCOPEs)*, pp. 1345–1350, Paralakhemundi, India, 2016.
- [23] J. Smailović, M. Grčar, N. Lavrač and M. Žnidaršič, "Predictive Sentiment Analysis of Tweets: A Stock Market Application," *Proc. of the Int. Workshop on Human-computer Interaction and Knowledge Discovery in Complex Unstructured, Big Data (HCI-KDD 2013)*, vol. 7947 pp. 77–88, 2013.
- [24] Z. Jin, Y. Yang and Y. Liu, "Stock Closing Price Prediction Based on Sentiment Analysis and LSTM," *Neural Computing and Applications*, vol. 32, pp. 9713–9729, 2020.
- [25] Z. Berradi and M. Lazaar, "Integration of Principal Component Analysis and Recurrent Neural Network to Forecast the Stock Price of Casablanca Stock Exchange," *Procedia Computer Science*, vol. 148, pp. 55–61, 2019.
- [26] M. Ismail and A. M. Awajan, "A New Hybrid Approach EMD-EXP for Short-term Forecasting of Daily Stock Market Time Series Data," *Electronic Journal of Applied Statistical Analysis*, vol. 10, no. 2, pp. 307–327, 2017.
- [27] J. Qiu, B. Wang and C. Zhou, "Forecasting Stock Prices with Long-short Term Memory Neural Network Based on Attention Mechanism," *PLOS One*, vol. 15, no. 1, p. e0227222, 2020.
- [28] N. Jing, Z. Wu and H. Wang, "A Hybrid Model Integrating Deep Learning with Investor Sentiment Analysis for Stock Price Prediction," *Expert Systems with Applications*, vol. 178, p. 115019, 2021.
- [29] R. Batra and S. M. Daudpota, "Integrating Stock Twits with Sentiment Analysis for Better Prediction of Stock Price Movement," *Proc. of the 2018 IEEE Int. Conf. on Computing, Mathematics and Engineering Technologies (iCoMET)*, pp. 1–5, Sukkur, Pakistan, 2018.
- [30] S. Mohan, S. Mullapudi, S. Sammeta, P. Vijayvergia and D. C. Anastasiu, "Stock Price Prediction Using News Sentiment Analysis," *Proc. of the 2019 IEEE 5<sup>th</sup> Int. Conf. on Big Data Computing Service and Applications (BigDataService)*, pp. 205–208, Newark, CA, USA, 2019.
- [31] D. K. Kirange, R. R. Deshmukh et al., "Sentiment Analysis of News Headlines for Stock Price Prediction," *Composoft: An Int. J. of Advanced Computer Technol.*, vol. 5, no. 3, pp. 2080–2084, 2016.
- [32] J. Abraham, D. Higdon, J. Nelson and J. Ibarra, "Cryptocurrency Price Prediction Using Tweet Volumes and Sentiment Analysis," *SMU Data Science Review*, vol. 1, no. 3, p.1, 2018.
- [33] S. Liu, C. Zhang and J. Ma, "CNN-LSTM Neural Network Model for Quantitative Strategy Analysis in Stock Markets," *Proc. of the Int. Conf. on Neural Information Processing (ICONIP 2017)*, vol. 10635, pp. 198–206, 2017.
- [34] M. R. Vargas, B. De Lima and A. G. Evsukoff, "Deep Learning for Stock Market Prediction from Financial News Articles," *Proc. of the 2017 IEEE Int. Conf. on Computational Intelligence and Virtual Environments for Measurement. Sys. and Appl. (CIVEMSA)*, pp. 60–65, Annecy, France, 2017.
- [35] C.-Y. Lee and V.-W. Soo, "Predict Stock Price with Financial News Based on Recurrent Convolutional Neural Networks," *Proc. of the 2017 IEEE Conf. on Technologies and Applications of Artificial Intelligence (TAAD)*, pp. 160–165, Taipei, Taiwan, 2017.
- [36] J. Li, H. Bu and J. Wu, "Sentiment-aware Stock Market Prediction: A Deep Learning Method," *Proc. of the 2017 IEEE Int. Conf. on Service Systems and Service Management*, pp. 1–6, Dalian, 2017.
- [37] H. Yan and H. Ouyang, "Financial Time Series Prediction Based on Deep Learning," *Wireless Personal Communications*, vol. 102, no. 2, pp. 683–700, 2018.
- [38] Y. Chen, J. Wu and H. Bu, "Stock Market Embedding and Prediction: A Deep Learning Method," *Proc.*

- of the IEEE 2018 15<sup>th</sup> Int. Conf. on Service Systems and Service Management (ICSSSM), pp. 1–6, Hangzhou, China, 2018.
- [39] H. Y. Kim and C. H. Won, "Forecasting the Volatility of Stock Price Index: A Hybrid Model Integrating LSTM with Multiple Garch-type Models," *Expert Systems with Applications*, vol. 103, pp. 25–37, 2018.
- [40] F. Shen, J. Chao and J. Zhao, "Forecasting Exchange Rate Using Deep Belief Networks and Conjugate Gradient Method," *Neurocomputing*, vol. 167, pp. 243–253, 2015.
- [41] S. Hochreiter and J. Schmidhuber, "LSTM Can Solve Hard Long Time Lag Problems," *Proc. of the 9<sup>th</sup> International Conference on Neural Information Processing Systems (NIPS'96)*, pp. 473–479, 1997.
- [42] J. Pennington, R. Socher and C. D. Manning, "GloVe: Global Vectors for Word Representation," *Proc. of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, Doha, Qatar, 2014.
- [43] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, "Learning Phrase Representations Using RNN Encoder-Decoder for Statistical Machine Translation," *arXiv preprint*, arXiv: 1406.1078, 2014.
- [44] D. Bahdanau, K. Cho and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," *arXiv preprint*, arXiv: 1409.0473, 2014.
- [45] J. Chorowski, D. Bahdanau, D. Serdyuk, K. Cho and Y. Bengio, "Attention-based Models for Speech Recognition," *arXiv preprint*, arXiv: 1506.07503, 2015.
- [46] Z. Berradi, M. Lazaar, O. Mahboub and H. Omara, "A Comprehensive Review of Artificial Intelligence Techniques in Financial Market," *Proc. of the 2020 6<sup>th</sup> IEEE Congress on Information Science and Technology (CiSt)*, pp. 367–371, DOI: 10.1109/CiSt49399.2021.9357175, 2020.
- [47] T. Hollis, A. Viscardi and S. Eun Yi, "A Comparison of LSTMS and Attention Mechanisms for Forecasting Financial Time Series," *arXiv preprint*, arXiv: 1812.07699, 2018.
- [48] Z. Berradi, M. Lazaar, H. Omara and O. Mahboub, "Effect of Architecture in Recurrent Neural Network Applied on the Prediction of Stock Price," *IAENG International Journal of Computer Science*, vol. 47, no. 3, pp. 436–441, 2020.

### ملخص البحث:

يُعدّ التعلّم العميق ميداناً واعداءً من ميادين البحث والتطوير، وله تطبيقات عديدة في شتى مجالات الحياة. والجدير بالذكر أنّ تطبيقاته في أسواق الأسهم منتشرة على نطاق واسع نظراً لما يتمتع به من فاعلية. وقد أثبتت تقنية ذاكرة المدى القصير والطويل (LSTM) فاعلية في التعامل مع بيانات السلاسل الزمنية نظراً لبنيتها الفريدة ذات الوحدات المخفية.

وتجمع هذه الورقة بين تقنية ذاكرة المدى القصير والطويل (LSTM) المذكورة وبين كل من آلية الانتباه (AM) وتحليل المشاعر (SA) للتعنبؤ بسعر الإغلاق في كل من بورصة AAPL وبورصة TSLA التابعتين لسوق ناسداك المالي (NASDAQ). وقد تمت مقارنة النموذج الهجين المقترح مع كل من نموذج (LSTM)، ونموذج (LSTM) مع تحليل المشاعر (SA)، ونموذج (LSTM) مع آلية الانتباه (AM). وقد استعملت ثلاث علامات مرجعية لقياس أداء كل من تلك النماذج هي: MSE، و RMSE، و MAE. وبينت النتائج أنّ النظام الهجين تفوق في الأداء على غيره من النماذج.

# EARLY PREDICTION OF CERVICAL CANCER USING MACHINE LEARNING TECHNIQUES

Mohammad Subhi Al-Batah<sup>1</sup>, Mazen Alzyoud<sup>2</sup>, Raed Alazaidah<sup>3</sup>, Malek Toubat<sup>4</sup>,  
Haneen Alzoubi<sup>2</sup> and Areej Olaiyat<sup>5</sup>

(Received: 28-Aug.-2022, Revised: 16-Oct.-2022, Accepted: 30-Oct.-2022)

## ABSTRACT

According to recent studies and statistics, Cervical Cancer (CC) is one of the most common causes of death worldwide and mainly in the developing countries. CC has a mortality rate of around 60%, in poor developing countries and the percentages could go even higher, due to poor screening processes, lack of sensitization and several other reasons. Therefore, this paper aims to utilize the high capabilities of machine-learning techniques in the early prediction of CC. In specific, three well-known feature selection and ranking methods have been used to identify the most significant features that help in the diagnosis process. Also, eighteen different classifiers that belong to six learning strategies have been trained and extensively evaluated against primary data consisting of five hundred images. Moreover, an investigation regarding the problem of imbalance class distribution which is common in medical datasets is conducted. The results revealed that LWNB and RandomForest classifiers showed the best performance in general and considering four different evaluation metrics. Also, LWNB and logistic classifiers were the best choices to handle the problem of imbalance class distribution which is common in medical diagnosis tasks. The final conclusion which could be made is that using an ensemble model which consists of several classifiers such as LWNB, RandomForest and logistic classifiers is the best solution to handle this type of problems.

## KEYWORDS

Cervical cancer, Classification, Feature selection, Machine learning, Medical diagnosis.

## 1. INTRODUCTION

According to the Jordanian Ministry of Health (MoH) statistics, cancerous diseases are the second cause of death in Jordan. Globally, huge efforts from all nations have been implicated in the last century into building a strong understanding of pathophysiology, genetic changes and clinical presentation of different cancers and recruiting this knowledge in developing new methods of treatment, new screening methods and improving prognosis among cancer patients [1].

CC is a gynaecological malignancy that occurs mainly in middle-aged women, due to unregulated division of cells in cervical mucosa of females' reproductive system. Usually, females come to the clinic with chief complaints of vaginal bleeding and abnormal vaginal discharge [2].

CC almost exclusively develops in cervical cells with pre-existing human papilloma which induces dysplasia (abnormal cell growth that is premalignant) that remains latent with no symptoms for decades before developing into absolute CC [3].

Human Papilloma Virus (HPV) is a sexually transmitted infection that occurs mainly in individuals with multiple sex partners and who have not been vaccinated against carcinogenic HPVs [3]. Early detection of this dysplasia before developing into cancer is the cornerstone in fighting against CC [4].

Although CC-related incidences and deaths have dramatically decreased in developed countries-thanks to huge improvements in screening procedures [4], CC is still a huge challenge, especially to developing countries. It is the most deadly type of cancer in women in developing countries that cannot overcome the problem of lacking sufficient number of health-care professionals who are well

1. M. S. Al-Batah is with Department of Computer Science, Faculty of Science and Information Technology, Jadara University, Irbid, Jordan. Email: [albatah@jadara.edu.jo](mailto:albatah@jadara.edu.jo)

2. M. Alzyoud and H. Alzoubi are with Faculty of Information Technology, Al-al-Bayt University, Jordan. Emails: [malzyoud@aabu.edu.jo](mailto:malzyoud@aabu.edu.jo) and [haneen123shada@gmail.com](mailto:haneen123shada@gmail.com)

3. R. Alazaidah is with Computer Science Department, Faculty of Information Technology, Zarqa University, Jordan. Email: [razaidah@zu.edu.jo](mailto:razaidah@zu.edu.jo)

4. M. Toubat is with Faculty of Medicine, JUST, Irbid, Jordan. Email: [mhtoubat19@med.just.edu.jo](mailto:mhtoubat19@med.just.edu.jo)

5. A. Olaiyat is with Faculty of Pharmacy, Yarmouk University, Irbid, Jordan. Email: [2017507139@sec.yu.edu.jo](mailto:2017507139@sec.yu.edu.jo)

trained in implementing this procedure for high-risk populations [4]. This signifies the importance of developing a computerized screening test using artificial intelligence and machine learning strategies [5].

Therefore, this paper aims to achieve the main following objectives:

1. To identify the most relevant and significant features that highly facilitate early prediction of CC.
2. To determine the best classifier that could be used to classify and predict the existence of CC among the large number of classifiers that belong to different learning strategies and use several evaluation metrics.
3. To determine the best classifier in handling the problem of imbalance class distribution which is common and familiar in the medical diagnostic field.

The main motivation for this research is to determine the best classification algorithms to use when attempting to predict CC; hence utilizing these algorithms in designing and programming a tool to automate the prediction of CC.

The rest of this paper is organized as follows: Section 2 surveys the related work dedicated to the prediction of CC. Section 3 describes the main steps of the conducted research and discusses the results obtained. Section 4 concludes the paper and lists out some future research-work horizons.

## 2. RELATED WORK

Classification is one of the main supervised learning tasks in machine learning. This task aims to accurately predict the class label for unseen instance [6]. In general, classification is divided into two main types: Single Label Classification (SLC) and Multi Label Classification (MLC) [7]. The former enforces each instance or example in the dataset to be linked to only one class label. Therefore, class labels in SLC are always mutually exclusive [7].

The latter allows instances in the dataset to be linked or associated with one class label or more. Hence, class labels in MLC are not mutually exclusive and have some kind of correlation among them, since they share the same values of features [8].

Moreover, SLC is divided into two sub-types: Binary Classification (BC) and Multi Class Classification (MCC). The former considers datasets with two class labels only, while the latter considers datasets with more than two class labels [9]-[10].

Classification as a machine-learning task has been utilized in several research papers related to CC. In [11], an attempt to combine the conventional diagnosis procedures and tests with machine learning to early predict abnormal cells, which highly increases the parentage of the complete cure of CC. This paper considered a large number of pap-smear test images which have been trained using deep learning techniques. The final proposed model was capable of predicting abnormal cells related to CC with accuracy of 74.04% only.

Ilyas and Ahmad (2021) [12] attempted to increase the accuracy of predicting CC by depending on an ensemble model. Therefore, eight different classifiers from different learning approaches have been utilized in predicting CC. Their study showed the significance of depending on several classifiers compared to depending on only one classifier when attempting to predict CC. This study could be improved by considering more classifiers and more learning strategies.

In [13], an ant-colony optimization algorithm has been proposed. The proposed algorithm has been trained on a dataset collected by the University of California. Support Vector Machine (SVM) has been used as the base classifier and showed a good performance (accuracy = 95.45%) compared with other algorithms which have been trained on the same dataset. The proposed algorithm has been evaluated using only one evaluation metric (accuracy). Also, the proposed algorithm should be evaluated against a larger number of algorithms.

A recent research that aimed to predict CC using MRI images has been conducted in [14]. Two main objectives have been achieved in this research. The first objective considered proposing an automatic system for early prediction of CC using image-processing techniques. The second objective aimed to enhance the performance of pre-trained Deep Convolutional Neural Networks (DCNNs) using Transfer

Learning (TL). In this paper, five classifiers were used to classify the input-image dataset into two class labels: benign or malign. Also, five evaluation metrics have been used in the evaluation phase of the five considered classifiers. Finally, according to the evaluation results, RandomForest (RF) classifier showed a better performance than the other four classifiers.

Another research that utilized machine-learning techniques in the early prediction of CC can be found in [15]. This research utilized the high capabilities of machine-learning techniques in the feature-selection step and the classification step. Unfortunately, the best evaluation result of the proposed algorithm was very low (best result for Area Under the Curve (AUC) metric was less than 0.69%) compared with other state-of-the-art algorithms.

A data-driven CC prediction model has been proposed in [16]. The proposed model not only aimed to predict CC, but also considered the problems of outliers and over-sampling. The prediction model only considered RF as a classifier. The model has been deployed through a mobile application that collects significant features related to CC and uses them in the prediction step of CC. The evaluation phase of the proposed model considered several evaluation metrics such as accuracy, precision, recall and F1-score. One of the main shortcomings of this model according to the authors themselves is the slow performance and the need to high memory during running the mobile-application software.

An ensemble model which combined the results of three different machine-learning algorithms to predict CC using Pap-smear test was proposed in [17]. The proposed model managed to predict CC using K-Nearest Neighbor (KNN), Support Vector Machine (SVM) and Multi-layer Perceptron (MLP) with a high accuracy rate (accuracy = 97.83%). The research concluded with a great potential of machine learning to highly and accurately predict CC. One of the main limitations for this research was depending on only the accuracy metric in the evaluation step while ignoring other significant evaluation metrics, such as precision, recall and F1-score.

In [18], an empirical analysis to determine the best classification algorithm among three classification algorithm has been performed. The paper considered Naïve Bayes (NB), Iterative Dichotomiser3 (ID3) and C4.5 classifiers. The analysis has been carried out using only one dataset and considering accuracy only as an evaluation metric. The paper concluded that NB outperformed the two other classifiers with accuracy being equal to (81%).

In [19], a research model that consisted of four main phases has been proposed. This research model consists of data pre-processing step, predictive model selection and pseudo-code. Also, several classifiers, such as KNN, Random Forest, SVM, Logistic Regression (LR), have been evaluated using three evaluation metrics. The research concluded the significance of using Random Forest, Decision Tree and several other classifiers in the prediction phase of CC.

### 3. METHODOLOGY, RESULTS AND ANALYSIS

In this section, a comprehensive description regarding the methodology, results and analysis is presented. Firstly, in Section A, the research methodology is presented. Secondly, in Section B, the dataset is described. Thirdly, in Section C, the steps of feature selection and ranking are introduced. Finally, in Section D, the classifiers and evaluation metrics considered are introduced with the results obtained and their analysis.

#### A. Research Methodology

The methodology of this research is illustrated in Figure 1. As can be seen from Figure 1, the methodology consists of seven main steps. The first main step considers the collecting of data from hospitals and several specialized medical centres. Then, the segmentation process is performed as

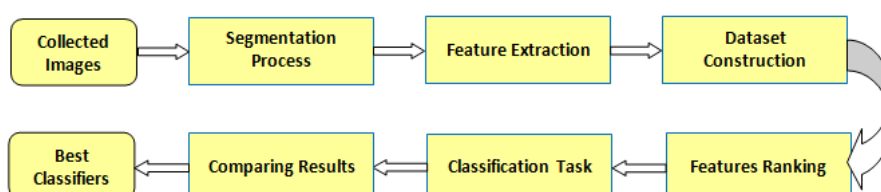


Figure 1. Research-methodology main steps.

explained in Section B. After that, several related features are extracted from the collected images as illustrated in Section C. The next step aims to construct a single-label dataset based on the data collected from the previous step. Then, three different feature-ranking techniques are applied on the dataset. The final steps aim to classify the data, obtain the results and identify the best classifiers among eighteen different classifiers based on several evaluation metrics, as extensively discussed in Section D. More information regarding these main steps can be found in the following sub-sections.

## B. Dataset Description

One dataset has been considered in this research. This dataset has been constructed after performing several steps. Firstly, 500 images have been collected from different hospitals and specialized medical centres in Jordan. All images in this research have been captured using an automatic glass capturing system which has been designed specifically for this purpose. This system consists mainly of three main components: a high-resolution digital camera, a high-quality digital microscope and a personal computer. All images have been captured using 100X and 400X magnification, as recommended by both pathologists and cytologists. Each image has been labelled as Normal, Low-grade Squamous Intra-epithelial Lesion (LSIL) or High-grade Squamous Intra-epithelial Lesion (HSIL) by three domain experts and the final class of the image is determined by considering the majority. Figure 2 depicts a sample of the captured images. Figure 2.a represents a "Normal" class, Figure 2.b represents an "HSIL" class and Figure 2.c represents an "LSIL" class.

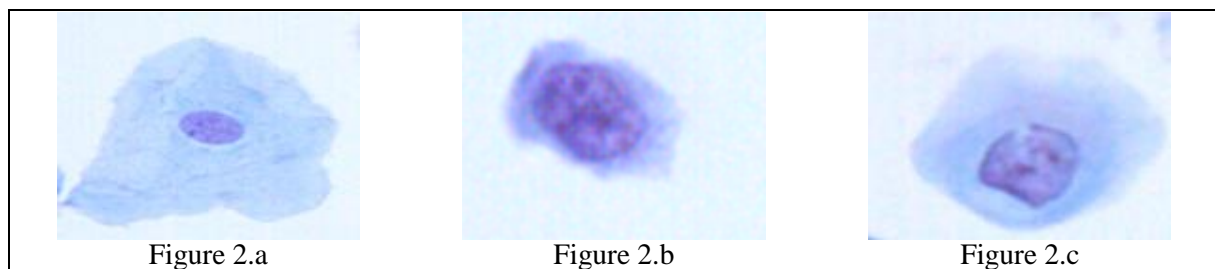


Figure 2. Sample of the captured images.

Secondly, a segmentation process has been applied on the collected images using Adaptive Fuzzy Moving K-means (AFMKM) clustering algorithm [20]. The main goal for applying AFMKM on the collected images is to differentiate the main three parts of the CC cell image: nucleus, cytoplasm and background. Thirdly, nine features are extracted from each CC cell image using both the nucleus and the cytoplasm parts. These features are: size, grey level, perimeter, red, green, blue, intensity1, intensity2 and saturation. Intensity1 and saturation were computed using Equations (1) and (3), respectively [21]. Intensity 2 was computed using Equation (2) [22].

$$Intensity1 = \frac{1}{3}(Red + Green + Blue) \quad (1)$$

$$Intensity2 = (0.299 Red) + (0.587 Green) + (0.114 Blue) \quad (2)$$

$$Saturation = \sqrt{c_1^2 + c_2^2} \quad (3)$$

where;

$$c_1 = Red - 0.5 Green - 0.5 Blue \quad (4)$$

$$c_2 = \frac{\sqrt{3}}{2} Green + \frac{\sqrt{3}}{2} Blue \quad (5)$$

Therefore, in total, the constructed dataset consists of eighteen features and five hundred instances. Each instance has been assigned to one class label only from three different class labels. These class labels are: Normal, LSIL and HSIL.

It is worth mentioning that the frequency of the three class labels: Normal, LSIL and HSIL was: 376, 79 and 45, respectively. Hence, as in most medical-diagnostic datasets, the considered dataset in this



research suffers from the problem of imbalance class distribution. Therefore, this fact should be highly considered when attempting to identify the best classifier to deal with such kind of data.

Table 1 depicts a description of the features used in the considered dataset, such as data type, minimum and maximum values, average and standard deviation. It is worth mentioning that the original dataset consists of eighteen different features.

Table 1. Characteristics of the features of the CC dataset.

No.	Name	Data Type	Minimum Value	Maximum Value	Average	Slandered Deviation
1	Nucleus Area	Integer	33.00	5845.00	607.47	567.82
2	Cytoplasm Area	Integer	151.00	33222.00	13104.05	7958.00
3	Nucleus Grey Level	Real	83.14	195.14	141.87	17.59
4	Cytoplasm Grey Level	Real	130.90	220.53	192.49	15.26
5	Nucleus Perimeter	Integer	22.00	1272.00	140.40	112.44
6	Cytoplasm Perimeter	Integer	81.00	13816.00	2957.13	2482.46
7	Nucleus Red	Real	83.14	195.14	141.87	17.59
8	Cytoplasm Red	Real	130.90	220.53	192.49	15.26
9	Nucleus Green	Real	86.73	189.93	140.19	20.40
10	Cytoplasm Green	Real	101.04	233.32	212.31	17.73
11	Nucleus Blue	Real	134.55	252.36	224.29	23.03
12	Cytoplasm Blue	Real	158.72	254.99	252.52	8.45
13	Nucleus Intensity1	Real	119.54	204.19	168.78	16.93
14	Cytoplasm Intensity1	Real	148.37	234.55	219.11	11.89
15	Nucleus Intensity2	Real	102.31	192.33	150.28	17.43
16	Cytoplasm Intensity2	Real	132.82	230.40	210.97	14.58
17	Nucleus Saturation	Real	43.38	133.39	86.09	15.23
18	Cytoplasm Saturation	Real	32.39	107.81	54.68	12.83

### C. Feature Selection and Ranking Step

One of the main objectives of this research is to identify the best classifier to handle the CC dataset when using all features, 75% of the features and 50% of the features. Therefore, the step of feature selection and ranking is crucial to this research.

Three different techniques have been used to rank the features. These techniques are InfoGainAttributeEval [23], ClassifierAttributeEval [23] and GainRatioAttributeEval [23]. All these techniques have been trained on the considered dataset using WEKA [23]. WEKA is short for Waikato Environment for Knowledge Analysis. WEKA is an open-source software that is used widely in data analysis in the domains of data mining and machine learning.

Regarding InfoGainAttributeEval, this technique evaluates the worth of an attribute by measuring the information gain with respect to the class. The ClassifierAttributeEval technique evaluates the worth of an attribute by using a user-specified classifier. Finally, the GainRatioAttributeEval technique depends on the gain ratio to evaluate the worth of an attribute with respect to the considered class. More information regarding these attribute evaluators and other feature-ranking techniques can be found in [23].

Table 2 depicts the ranking of the features after applying the three previously mentioned ranking techniques on the considered dataset.

Table 2. Feature-selection evaluation step using three attribute evaluators.

No.	Attribute	Ranking Using InfoGainAttributeEval	Ranking Using ClassifierAttributeEval	Ranking Using GainRatioAttributeEval
1	Cytoplasm Area	1	8	4
2	Cytoplasm Green	2	11	3
3	Cytoplasm Perimeter	3	4	7

4	Cytoplasm Intensity2	4	13	2
5	Cytoplasm Intensity1	5	15	1
6	Cytoplasm Blue	6	17	10
7	Cytoplasm Saturation	7	1	11
8	Cytoplasm Grey	8	5	5
9	Cytoplasm Red	9	9	6
10	Nucleus Perimeter	10	2	8
11	Nucleus Area	11	18	9
12	Nucleus Saturation	12	6	12
13	Nucleus Red	13	3	13
14	Nucleus Grey Level	14	7	14
15	Nucleus Intensity1	15	16	16
16	Nucleus Green	16	10	15
17	Nucleus Intensity2	17	14	17
18	Nucleus Blue	18	12	18

Table 3 depicts the features of the dataset after ranking. Features have been ranked using the summation of the ranks of the three considered ranking techniques. The feature with the least sum is ranked first and the feature with the highest sum is ranked last.

Table 3. Attributes' ranking using three attribute evaluators.

Order	Attribute	Ranking Using InfoGainAttributeEval	Ranking Using ClassifierAttributeEval	Ranking Using GainRatioAttributeEval	Sum
1	Cytoplasm Area	1	8	4	13
2	Cytoplasm Perimeter	3	4	7	14
3	Cytoplasm Green	2	11	3	16
4	Cytoplasm Grey Level	8	5	5	18
5	Cytoplasm Intensity2	4	13	2	19
6	Cytoplasm Saturation	7	1	11	19
7	Nucleus Perimeter	10	2	8	20
8	Cytoplasm Intensity1	5	15	1	21
9	Cytoplasm Red	9	9	6	24
10	Nucleus Red	13	3	13	29
11	Nucleus Saturation	12	6	12	30
12	Cytoplasm Blue	6	17	10	33
13	Nucleus Grey Level	14	7	14	35
14	Nucleus Area	11	18	9	38
15	Nucleus Green	16	10	15	41
16	Nucleus Intensity1	15	16	16	47
17	Nucleus Intensity2	17	14	17	48
18	Nucleus Blue	18	12	18	48

Based on Table 3, the classifiers considered in this research are trained on three versions of CC dataset. The first version consists of all features (18 features). The second version consists of the best ranked 75% of the features (12 features). The third version consists of the best ranked 50% of the features (9 features). The considered classifiers are evaluated based on their performance on the three versions and using several evaluation metrics.

#### D. Evaluation of the Considered Classifiers

The main objective of this research is to early predict CC using machine-learning techniques as accurately as possible. Therefore, many classifiers should be considered to identify the best one. Hence, eighteen different classifiers have been considered and extensively evaluated. These eighteen classifiers belong to six well-known learning strategies.

From Bayes learning strategy, the following three classifiers have been considered: BayesNet [21], NaiveBayes [24] and NaiveBayesUpdateable [24]. The function-learning strategy has been

represented through three classifiers: Logistic [25], SMO [26] and SimpleLogistic [27]. The Lazy learning strategy has been represented also using three classifiers: Instance-based Learning (IBL) [28], KStar [29] and Locally Weighted Naive Bayes (LWNB) [30].

For Meta learning strategy, the following classifiers have been considered: AdaBoostM1 [31], LogitBoost [32] and MultiClassClassifier [23]. Also, three different classifiers have been used to represent the Rule-based learning strategy. These classifiers are: DecisionTable [32], JRip [34] and PART [35]. Finally, the Tree learning strategy has been represented by RandomTree [23], RandomForest [36] and J48 [37] classifiers.

The previously mentioned classifiers have been evaluated using four different evaluation metrics: Accuracy, Precision, Recall and F1-Measure (F1-Score), using the following equations.

$$Accuracy = (TP + TN) / (P+N) \quad (6)$$

$$Precision = TP / (TP + FP) \quad (7)$$

$$Recall = TP / (TP + FN) \quad (8)$$

$$F1-Measure = 2 * (precision * recall) / (precision + recall) \quad (9)$$

where:

TP = True Positive, TN = True Negative, FP = False Positive, FN = False Negative. P and N are total positive and negative classes.

Table 4 depicts the evaluation results of the eighteen classifiers grouped by learning strategy and using the Accuracy metric. The evaluation considers all features, 75% of the features and 50% of the features, respectively.

Table 4. Evaluation results using the Accuracy metric.

Learning Strategy	Classifier	All Features	75 % of Features	50 % of Features
Bayes	BayesNet	82.000	82.600	82.200
	NaiveBayes	81.200	81.600	82.000
	NaiveBayesUpdateable	81.200	81.600	82.000
	<b>Average</b>	81.467	81.933	82.067
Functions	Logistic	<b>91.400</b>	86.000	86.600
	SMO	84.800	84.000	84.200
	SimpleLogistic	87.800	85.600	84.200
	<b>Average</b>	88.000	85.200	85.000
Lazy	IBL	86.200	85.800	86.200
	KStar	88.600	89.000	86.200
	LWNB	85.400	85.400	85.400
	<b>Average</b>	86.730	86.733	85.933
Meta	AdaBoostM1	84.800	84.800	84.800
	LogitBoost	89.000	88.400	87.000
	MultiClassClassifier	<b>91.400</b>	87.200	87.200
	<b>Average</b>	<b>88.400</b>	86.733	86.333
Rules	DecisionTable	85.200	85.200	85.200
	JRip	87.800	86.600	82.800
	PART	88.000	86.000	86.200
	<b>Average</b>	87.000	85.933	84.733
Trees	RandomTree	84.000	85.800	87.000
	RandomForest	<b>91.400</b>	<b>89.800</b>	<b>88.400</b>
	J48	88.000	87.200	87.400
	<b>Average</b>	87.800	<b>87.600</b>	<b>87.600</b>

According to Table 4, RandomForest showed the best results considering all features, 12 features and 9 features. Logistic and MultiClassClassifier showed an identical result to RandomForest when considering all features. Moreover, Tree as a learning strategy showed the best result with 12 and 9 features, while Meta learning strategy showed the best performance when considering all features.

It is worth mentioning that NaiveBayes and NaiveBayesUpdateable showed an identical performance on the three datasets (all features' dataset, 75% of the features' dataset and 50% of the features')

dataset).

Table 5 depicts the evaluation results of the eighteen classifiers grouped by learning strategy and using the Precision metric. The evaluation considers using all features, 75% of the features and 50% of the features. From Table 5, it can be clearly seen that LWNB classifier showed the best performance considering the Precision metric on the three considered cases (all features, 75% of the features, 50% of the features).

Considering learning strategies, Meta as a learning strategy showed the best performance on the three considered cases. Also, Lazy learning strategy showed an identical result to Meta learning strategy when considering 75% of the features. It is worth mentioning that NaiveBayes and NaiveBayesUpdateable showed an identical performance on the three datasets (all features' dataset, 75% of the features' dataset and 50% of the features' dataset).

Table 5. Evaluation results using the Precision metric.

Learning Strategy	Classifier	All Features	75 % of Features	50 % of Features
Bayes	BayesNet	0.838	0.838	0.842
	NaiveBayes	0.828	0.839	0.841
	NaiveBayesUpdateable	0.828	0.839	0.841
	<b>Average</b>	0.831	0.839	0.841
Functions	Logistic	0.914	0.860	0.866
	SMO	0.837	0.952	0.949
	SimpleLogistic	0.878	0.859	0.842
	<b>Average</b>	0.876	0.890	0.886
Lazy	IBL	0.861	0.860	0.865
	KStar	0.883	0.886	0.859
	LWNB	<b>0.978</b>	<b>0.978</b>	<b>0.978</b>
	<b>Average</b>	0.907	<b>0.908</b>	0.901
Meta	AdaBoostM1	0.967	0.967	0.967
	LogitBoost	0.889	0.881	0.869
	MultiClassClassifier	0.912	0.873	0.872
	<b>Average</b>	<b>0.923</b>	<b>0.908</b>	<b>0.903</b>
Rules	DecisionTable	0.847	0.847	0.840
	JRip	0.883	0.869	0.835
	PART	0.878	0.857	0.871
	<b>Average</b>	0.869	0.858	0.849
Trees	RandomTree	0.839	0.852	0.878
	RandomForest	0.907	0.888	0.885
	J48	0.884	0.880	0.879
	<b>Average</b>	0.877	0.873	0.881

Table 6 depicts the evaluation results of the eighteen classifiers grouped by learning strategy and using the Recall metric. The evaluation considers using all features, 75% of the features and 50% of the features.

Table 6. Evaluation results using the Recall metric.

Learning Strategy	Classifier	All Features	75 % of Features	50 % of Features
Bayes	BayesNet	0.820	0.826	0.822
	NaiveBayes	0.812	0.816	0.820
	NaiveBayesUpdateable	0.812	0.816	0.820
	<b>Average</b>	0.815	0.819	0.821
Functions	Logistic	<b>0.914</b>	0.860	0.866
	SMO	0.848	0.840	0.842
	SimpleLogistic	0.878	0.856	0.842
	<b>Average</b>	0.880	0.852	0.850
Lazy	IBL	0.862	0.858	0.862

	KStar	0.886	<b>0.890</b>	0.862
	LWNB	0.854	0.854	0.854
	<b>Average</b>	0.867	0.867	0.859
<b>Meta</b>	AdaBoostM1	0.848	0.848	0.848
	LogitBoost	0.890	0.884	0.870
	MultiClassClassifier	<b>0.914</b>	0.872	0.872
	<b>Average</b>	<b>0.884</b>	0.867	0.863
<b>Rules</b>	DecisionTable	0.852	0.852	0.852
	JRip	0.878	0.866	0.828
	PART	0.880	0.860	0.862
	<b>Average</b>	0.870	0.859	0.847
<b>Trees</b>	RandomTree	0.840	0.858	0.870
	RandomForest	<b>0.914</b>	0.888	<b>0.884</b>
	J48	0.880	0.872	0.874
	<b>Average</b>	0.878	<b>0.873</b>	<b>0.876</b>

From Table 6, RandomForest showed the best performance on all features' dataset and 50% features' dataset. KStar showed the best result on the dataset with 75% of the features. Also, Logistic and MultiClassClassifier showed the best results on all features' dataset along with RandomForest Classifier.

Regarding to the best learning strategy, as can be seen from Table 6, Trees showed the best performance on the dataset with 75% of the features and the dataset with 50% of the features, while Meta learning strategy showed the best performance on the dataset with all features.

It is worth mentioning that NaiveBayes and NaiveBayesUpdateable showed an identical performance on the three datasets (all features' dataset, 75% of the features' dataset and 50% of the features' dataset).

Table 7 depicts the evaluation results of the eighteen classifiers grouped by learning strategy and using the F1-Measure (F1-Score) metric. The evaluation considers using all features, 75% of the features and 50% of the features. According to Table 7, LWNB classifier has a superior constant performance compared with the other seventeen classifiers. LWNB achieved the best results on all features' dataset, 75% of the features' dataset and 50% of the features' dataset.

Considering the learning strategy, Meta as a learning strategy showed the best performance on the dataset with all features, the dataset with 75% of the features and the dataset with 50% of the features. Also, Lazy learning strategy showed the best performance on the dataset with 50% of the features.

It is worth mentioning that NaiveBayes and NaiveBayesUpdateable showed an identical performance on the three datasets (all features' dataset, 75% of the features' dataset and 50% of the features' dataset).

Table 7. Evaluation results using the F1-Measure metric

Learning Strategy	Classifier	All Features	75 % of Features	50 % of Features
<b>Bayes</b>	BayesNet	0.823	0.822	0.821
	NaiveBayes	0.818	0.823	0.828
	NaiveBayesUpdateable	0.818	0.823	0.828
	<b>Average</b>	0.820	0.823	0.826
<b>Functions</b>	Logistic	0.914	0.860	0.866
	SMO	0.824	0.947	0.945
	SimpleLogistic	0.878	0.857	0.841
	<b>Average</b>	0.872	0.888	0.884
<b>Lazy</b>	IBL	0.862	0.859	0.864
	KStar	0.885	0.888	0.860
	LWNB	<b>0.962</b>	<b>0.962</b>	<b>0.962</b>
	<b>Average</b>	0.903	<b>0.903</b>	0.895

<b>Meta</b>	AdaBoostM1	0.958	0.958	0.958
	LogitBoost	0.888	0.881	0.869
	MultiClassClassifier	0.913	0.870	0.871
<b>Average</b>		<b>0.920</b>	<b>0.903</b>	<b>0.899</b>
<b>Rules</b>	DecisionTable	0.843	0.844	0.839
	JRip	0.879	0.867	0.831
	PART	0.879	0.858	0.865
<b>Average</b>		<b>0.867</b>	<b>0.856</b>	<b>0.845</b>
<b>Trees</b>	RandomTree	0.840	0.855	0.874
	RandomForest	0.904	0.888	0.884
	J48	0.882	0.875	0.876
<b>Average</b>		<b>0.875</b>	<b>0.873</b>	<b>0.878</b>

Table 8 summarizes the results obtained from Table 4 to Table 7 by identifying the best classifier with respect to the considered metric and the number of features being used.

Table 8. Summarization of the best classifier with respect to evaluation metric and number of the considered features.

<b>Metric</b>	<b>All Features</b>	<b>75 % of Features</b>	<b>50 % of Features</b>
<b>Accuracy</b>	Logistic		
	MultiClassClassifier RandomForest	RandomForest	RandomForest
<b>Precision</b>	LWNB	LWNB	LWNB
<b>Recall</b>	Logistic		
	MultiClassClassifier RandomForest	KStar	RandomForest
<b>F1-Measure</b>	LWNB	LWNB	LWNB

According to Table 8, LWNB classifier is the best classifier among all considered classifiers. LWNB classifier achieved the best performance six times. RandomForest classifier is the second best classifier, since it achieved the best performance five times. LWNB classifier is the optimal choice when there is a need to optimize Precision and F1-Measure metrics. RandomForest classifier is the best choice when there is a need to optimize Accuracy and Recall metrics. Moreover, Logistic and MultiClassClassifier showed an excellent performance when considering all features with Accuracy and Recall metrics.

Table 9 depicts the best learning strategy with respect to the considered evaluation metric and the number of features being used. Table 9 summarizes the results from Table 4 to Table 7.

Table 9. Summarization of the best learning strategy with respect to evaluation metric and number of the considered features.

<b>Metric</b>	<b>All Features</b>	<b>75 % of Features</b>	<b>50 % of Features</b>
<b>Accuracy</b>	Meta	Trees	Trees
<b>Precision</b>	Meta	Meta Lazy	Meta
<b>Recall</b>	Meta	Trees	Trees
<b>F-Measure</b>	Meta	Meta Lazy	Meta

From Table 9, It is obvious that Meta as a learning strategy is the dominant strategy. Meta showed the best performance considering the four evaluation metrics. Trees learning strategy is the second best learning strategy and Lazy learning strategy is the third best strategy according to Table 9.

In general, medical datasets like the dataset considered in this research usually suffer from the problem of imbalance class distribution. For example, in the CC dataset, the dominant class is the "Normal" class with a frequency equal to 376. For "LSIL" class, the frequency is 79, while the frequency of

“HSIL” class is 45, as mentioned previously. One of the main characteristics of the optimal classifier is the ability to handle the problem of imbalance class distribution.

Therefore, it has been decided to evaluate the eighteen classifiers considered in this research based on how accurate they can predict the least frequent, but most significant, classes (LSIL and HSIL). True Positive (TP) metric has been used to accomplish this task. TP metric calculates the percentage at which the classifier correctly predicts the positive classes.

Table 10 depicts the evaluation results of the eighteen considered classifiers using the TP metric and grouped by the learning strategy. It is worth mentioning that for the TP metric, the higher the value, the better the performance of the classifier.

Table 10. Evaluation results with respect to the TP metric for “HSIL” and “LSIL” classes using all features.

Learning Strategy	Classifier	HSIL	LSIL
Bayes	BayesNet	0.200	0.747
	NaiveBayes	0.600	0.405
	NaiveBayesUpdateable	0.600	0.405
	<b>Average</b>	<b>0.467</b>	<b>0.519</b>
Functions	Logistic	<b>0.711</b>	0.734
	SMO	0.022	0.861
	SimpleLogistic	0.667	0.620
	<b>Average</b>	<b>0.467</b>	<b>0.738</b>
Lazy	IBL	0.600	0.557
	KStar	0.578	0.633
	LWNB	0.000	<b>0.899</b>
	<b>Average</b>	<b>0.393</b>	<b>0.696</b>
Meta	AdaBoostM1	0.000	0.848
	LogitBoost	0.422	0.747
	MultiClassClassifier	0.689	0.722
	<b>Average</b>	<b>0.370</b>	<b>0.772</b>
Rules	DecisionTable	0.156	0.772
	JRip	0.489	0.734
	PART	0.578	0.658
	<b>Average</b>	<b>0.408</b>	<b>0.721</b>
Trees	RandomTree	0.422	0.532
	RandomForest	0.489	0.810
	J48	0.578	0.658
	<b>Average</b>	<b>0.496</b>	<b>0.667</b>

According to Table 10, Logistic classifier is the best classifier to predict the class label “HSIL” with a TP rate equal to 0.711, while LWNB is the best classifier to predict the class label “LSIL” with a TP rate equal to 0.899.

Considering the learning strategy, Trees is the most suitable learning strategy to predict the class label “HSIL”, while Meta is the most appropriate learning strategy to predict the class labels “LSIL”.

Since no classifier can be the dominant classifier for dealing with the problem of imbalance class distribution, it is highly recommended to adopt an ensemble model to overcome this serious problem. Based on the results of this research, it is recommended to include LWNB, RandomForest and Logistic in any future proposed ensemble models.

#### 4. CONCLUSION AND FUTURE WORK

In this paper, a dataset consisting of 500 images related to CC has been collected from different hospital and specialized medical centers. Also, eighteen different classifiers which belong to six learning strategies have been trained on the collected dataset and evaluated. The evaluation of the classifiers considered four evaluation metrics with respect to all features in the dataset, 75% of the features and 50% of the features. The results revealed that LWNB classifier has achieved the best performance in general. RandomForest showed the second best performance. Also, considering the

learning strategy, Meta learning strategy showed the best overall performance compared with the other five strategies. Moreover, Logistic and LWNB classifiers are the best choice to deal with the problem of imbalance class distribution, which is very common in medical diagnostic datasets. Based on the results of this research, the main recommendation for future work is to adopt an ensemble model that consists of LWNB, RandomForest and Logistic classifiers to achieve high performance in the early prediction of CC.

## REFERENCES

- [1] M. A. Abu-Lubad, A. J. Dua'a, G. F. Helaly et al., "Human Papillomavirus as an Independent Risk Factor of Invasive Cervical and Endometrial Carcinomas in Jordan," *Journal of Infection and Public Health*, vol. 13, no. 4, pp. 613-618, 2022.
- [2] B. Obeidat, I. Matalka, A. Mohtaseb et al., "Prevalence and Distribution of High-risk Human Papillomavirus Genotypes in Cervical Carcinoma, Low-grade and High-grade Squamous Intraepithelial Lesions in Jordanian Women," *European Journal of Gynaecological Oncology*, vol. 34, no. 3, pp. 257-260, 2013.
- [3] S. E. Jordan, M. Schlumbrecht, S. George et al., "The Moore Criteria: Applicability in a Diverse, Non-trial, Recurrent Cervical Cancer Population," *Gynecologic Oncology*, vol. 157, no. 1, pp. 167-172, 2022.
- [4] M. Al Qadire, K. M. Aldiabat, E. Alsayheen et al., "Public Attitudes toward Cancer and Cancer Patients: A Jordanian National Online Survey," *Middle East Journal of Cancer*, vol. 13, DOI: 10.30476/mejc.2020.86835.1381, 2020.
- [5] A. I. Khasawneh, F. F. Asali, R. M. Kilani et al., "Prevalence and Genotype Distribution of Human Papillomavirus among a Sub-population of Jordanian Women," *International Journal of Women's Health and Reproduction Sciences*, vol. 9, no. 1, pp. 17-23, 2021.
- [6] R. Alazaidah, M. A. Almaiah and M. Al-luwaici, "Associative Classification in Multi-label Classification: An Investigative Study," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 7, no. 2, pp. 166 - 179, 2021.
- [7] M. Al-luwaici, A. K., Junoh, W. A. AlZoubi., R. Alazaidah and W. Al-luwaici, "New Features Selection Method for Multi-label Classification Based on the Positive Dependencies among Labels," *Solid State Technology*, vol. 63, no. 2s, pp. 9896-9909, 2020.
- [8] R. Alazaidah, F. A. Ahmad, M. F. M. Mohsin and W. A. AlZoubi, "Multi-label Ranking Method Based on Positive Class Correlations," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 6, no. 4, pp. 377-391, 2020.
- [9] M. Alluwaici, A. K. Junoh and R. Alazaidah, "New Problem Transformation Method Based on the Local Positive Pairwise Dependencies among Labels," *Journal of Information & Knowledge Management*, vol. 19, no. 1, ID. 2040017, 2020.
- [10] R. Alazaidah, F. K. Ahmad and M. F. M. Mohsin, "Multi Label Ranking Based on Positive Pairwise Correlations among Labels," *The International Arab Journal of Information Technology*, vol. 17, no. 4, pp. 440-449, 2020.
- [11] B. J. Priyanka, "Machine Learning Approach for Prediction of Cervical Cancer," *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, vol. 12, no. 8, pp. 3050-3058, 2021.
- [12] Q. M. Ilyas and M. Ahmad, "An Enhanced Ensemble Diagnosis of Cervical Cancer: A Pursuit of Machine Intelligence towards Sustainable Health," *IEEE Access*, vol. 9, pp. 12374-12388, 2021.
- [13] J. Wahid and H. F. A. Al-Mazini, "Classification of Cervical Cancer Using Ant-miner for Medical Expertise Knowledge Management," *Proc. of the Knowledge Management Int. Conf. (KMICe)*, Miri Sarawak, Malaysia, 25 -27 July 2018.
- [14] I. Khouli and N. Idrissi, "Cervical Cancer Detection and Classification Using MRIs," *Jordanian Journal of Computers and Information Technology (JJCIT)*, vol. 8, no. 2, pp. 141-158, 2022.
- [15] K. Fernandes, D. Chicco, J. S. Cardoso and J. Fernandes, "Supervised Deep Learning Embeddings for the Prediction of Cervical Cancer Diagnosis," *PeerJ Computer Science*, vol. 4, e154, DOI: 10.7717/peerj-cs.154, 2018.
- [16] M. F. Ijaz, M. Attique and Y. Son, "Data-driven Cervical Cancer Prediction Model with Outlier Detection and Over-sampling Methods," *Sensors*, vol. 20, no. 10, ID. 2809, 2020.
- [17] V. Mishra, S. Aslan and M. M. Asem, "Theoretical Assessment of Cervical Cancer Using Machine Learning Methods Based on Pap-Smear Test," *Proc. of the 9<sup>th</sup> IEEE Annual Information Technology, Electronics and Mobile Communication Conf. (IEMCON)*, pp. 1367-1373, Vancouver, Canada, 2018.
- [18] R. Vidya and G. M. Nasira, "Predicting Cervical Cancer Using Machine Learning Techniques - An Analysis," *Glob. J. Pure Appl. Math.*, vol. 12, no. 3, 2016.



- [19] N. Al Mudawi and A. Alazeb, "A Model for Predicting Cervical Cancer Using Machine Learning Algorithms," *Sensors*, vol. 22, no. 11, ID. 4132, 2022.
- [20] N. A. M. Isa, S. A. Salamah and U. K. Ngah, "Adaptive Fuzzy Moving K-means Clustering Algorithm for Image Segmentation," *IEEE Trans. on Consumer Electronics*, vol. 55, no. 4, pp. 2145-2153, 2009.
- [21] C. Zhang and P. Wang, "A New Method of Color Image Segmentation Based on Intensity and Hue Clustering," *Proc. of the 15<sup>th</sup> IEEE Int. Conf. on Pattern Recognition (ICPR-2000)*, vol. 3, pp. 613-616, Barcelona, Spain, 2000.
- [22] N. Mustafa, N. A. M. Isa, M. Y. Mashor and N. H. Othman, "Capability of New Features of Cervical Cells for Cervical Cancer Diagnostic System Using Hierarchical Neural Network," *IJSSST*, vol. 9, no. 2, pp. 56-64, 2008.
- [23] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA Data Mining Software: An Update," *ACM SIGKDD Explorations Newsletter*, vol. 11, no. 1, pp. 10-18, 2009.
- [24] G. H. John and P. Langley, "Estimating Continuous Distributions in Bayesian Classifier," *Proc. of the 11<sup>th</sup> Conf. on Uncertainty in Artificial Intelligence (UAI1995)*, pp. 338-345, San Mateo, 1995.
- [25] S. Le Cessie and J. C. Van Houwelingen, "Ridge Estimators in Logistic Regression," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 41, no. 1, pp. 191-201, 1992.
- [26] J. Platt, "Using Analytic QP and Sparseness to Speed Training of Support Vector Machines," *Advances in Neural Information Processing Systems*, vol. 11, 1998.
- [27] N. Landwehr, M. Hall and E. Frank, "Logistic Model Trees," *Machine Learning*, vol. 59, no. 1, pp. 161-205, 2005.
- [28] D. W. Aha, D. Kibler and M. K. Albert, "Instance-based Learning Algorithms," *Machine Learning*, vol. 6, no. 1, pp. 37-66, 1991.
- [29] J. G. Cleary and L. E. Trigg, "K\*: An Instance-based Learner Using an Entropic Distance Measure," *Proc. of the 12<sup>th</sup> Int. Conf. on Machine Learning*, pp. 108-114, Tahoe City, California, July 9-12, 1995.
- [30] E. Frank, M. Hall and B. Pfahringer, "Locally Weighted Naive Bayes," *Proc. of the 19<sup>th</sup> Conf. on Uncertainty in Artificial Intelligence*, pp. 249-256, arXiv:1212.2487, 2003.
- [31] Y. Freund and R. E. Schapire, "Experiments with a New Boosting Algorithm," *Proc. of the 13<sup>th</sup> Int. Conf. on Int. Conf. on Machine Learning (ICML'96)*, vol. 96, pp. 148-156, 1996.
- [32] J. Friedman, T. Hastie and R. Tibshirani, "Additive Logistic Regression: A Statistical View of Boosting," *The Annals of Statistics*, vol. 28, no. 2, pp. 337-407, Stanford University, 1998.
- [33] R. Kohavi, "The Power of Decision Tables," *Proc. of the European Conf. on Machine Learning (ECML)*, pp. 174-189, Springer, Berlin, Heidelberg, 1995.
- [34] W. W. Cohen, "Fast Effective Rule Induction," *Proc. of the 12<sup>th</sup> Int. Conf. on Machine Learning*, pp. 115-123, Tahoe City, California, 1995.
- [35] E. Frank and I. H. Witten, "Generating Accurate Rule Sets without Global Optimization," *Proc. of the 15<sup>th</sup> Int. Conf. on Machine Learning (ICML '98)*, pp. 144-151, 1998.
- [36] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.
- [37] J. R. Quinlan, C4. 5: Program for Machine Learning, Morgan Kaufmann Publishers, Inc., 1993.

### ملخص البحث:

تهدف هذه الورقة الى استغلال الإمكانيات العالية لتقنيات تعلم الآلة من أجل الكشف المبكر عن الإصابة بسرطان عنق الرحم. حيث يتم استخدام ثلاث طرق لاختيار السمات وترتيبها لتحديد السمات الأكثر أهمية التي تُساعد في عملية التشخيص. كذلك تم استخدام ثمانية عشر مُصنّفًا تتبع لِسِت استراتيجيات تعلم بحيث تم تدريبها وتقييمها مقابل بيانات أولية تتكون من 500 صورة. من جهةٍ أخرى، جرى استقصاء مشكلة عدم التوازن في توزيع الأصناف. وبيّنت النتائج أن مُصنّف LWNB ومُصنّف RandomForest حقّقا أفضل أداء بشكلٍ عام وباعتماد أربعة مقاييس للتقييم كان مُصنّف LWNB ومُصنّف Logistic هما الأفضل من حيث معالجة مشكلة عدم التوازن في توزيع أصناف البيانات. ويمكن القول إن الاستنتاج النهائي الذي يُمكن الخروج به في هذا البحث هو أن استخدام نموذج مُجمّع يتألف من عدّة مصنّفات (مثل مُصنّفات LWNB و RandomForest و Logistic) هو الحلّ الأمثل للتعامل مع المشكلات المرتبطة بموضوع البحث، وهو الكشف المبكر عن الإصابة بسرطان عنق الرحم.

# COTA 2.0: AN AUTOMATIC CORRECTOR OF TUNISIAN ARABIC SOCIAL MEDIA TEXTS

Asma Mekki, Inès Zribi, Mariem Ellouze and Lamia Hadrich Belguith\*

(Received: 17-Jun.-2022, Revised: 7-Sep.-2022 and 18-Oct.-2022, Accepted: 10-Nov.-2022)

## ABSTRACT

*In written text, orthographic noise is a common concern for NLP, especially when operating social-network comments and raw documents. This is mainly due to its orthographic conventions and morphological ambiguity. We propose to automatically normalize the social-media dialect corpora by following CODA-TA, the conventional Orthography for TA. The existing system developed for TA «COTA Orthography 1.0» is not able to handle all forms of TA. Therefore, we propose to extend its rules and lexicons to address the peculiarities of social media dialect. In certain words, the COTA Orthography 1.0 system provides the user with several correction possibilities. Therefore, in the new version, we incorporated a trigram language model to automatically select the right correction. Our results show that the system can reduce transcription errors by 95.72%.*

## KEYWORDS

*Orthographic normalization, Tunisian Arabic, COTA Orthography system, CODA-TA.*

## 1. INTRODUCTION

Dialectal Arabic is a linguistic variety that is historically related to classical Arabic and exists side-by-side with Modern Standard Arabic (MSA). In fact, MSA is the official written and spoken language used by the government, media and education. Dialectal Arabic is the spoken variety used in daily communication of the Arabic World and is not generally written [1]. Indeed, it has no standard orthographies.

Since the political Tunisian revolution in 2011, the Internet is taking an increasingly important role in Tunisians' lives with 7,447,000 Facebook users and 1,910,000 Instagram users in Tunisia in January 2019<sup>1</sup>. Generally, Tunisians use their dialect for expressing their opinions and emotions. Researchers have taken advantage of the high number of comments shared on social media, as well as the availability and ease of accessing these tools, to build large corpora for Tunisian Arabic [2]-[3]; [1]; [4]-[7].

Indeed, social-media dialect is characterized by its high level of orthographic heterogeneity, which made its processing a serious challenge for Natural Language Processing (NLP) tools. Despite the efforts of researchers to normalize the orthographic form of dialectal Arabic, most of the existing corpora are not standardized, where the same word is written in several forms (i.e., a word can have dozens of writing forms).

In this paper, we propose to automatically normalize social-media dialect corpora written with Arabic characters into CODA-TA Conventional Orthography for Tunisian Arabic (TA) [8]. We decided to use and expand CODA-TA, because there is already a semi-automatic tool (COTA Orthography [9]) that follows its linguistic guidelines. The process of orthographic normalization was made easier by this tool. Furthermore, many TA corpora have already been normalized using this convention. However, the existing system developed for TA [9] is not able to address all forms of TA (see subsection 3.1). Therefore, we propose to extend its rules and lexicons in order to treat the particularities of social-media dialect. Hence, our contributions can be summarized as follows:

- We started by extending the CODA-TA spelling convention to include social-media dialect features.

---

<sup>1</sup> <http://napoleoncat.com>

- The next step is to enrich the lexicon by the vocabulary used in social networks.
- We also proposed adding new patterns to treat onomatopoeia, accentuations of words, ...etc.
- Thereafter, we trained an n-gram model on a large textual scale based on the three forms of dialect (intellectualized dialect, spontaneous dialect and social media dialect).
- Then, we integrated this model into the first version of the Conventionalized Tunisian Arabic Orthography (COTA Orthography system) to choose the right correction automatically.

We show in the evaluation section the effect of using an automatically normalized corpus on the diverse tools, such as sentence segmentation, POS tagger and parser. It ensures that the number of orthographic errors in a document decreases significantly, which is very helpful for NLP tools. Our system contributes to a significant improvement in this assessment. It can also ensure high quality without wasting time on manual orthographic normalization, because it is a fully automatic tool.

This paper is structured as follows: Section 2 is dedicated to review related work. Section 3 presents the Tunisian dialect forms as well as social-media dialect and COTA orthography automatic normalization challenges. Section 4 details our proposed method. Finally, we present and discuss, in Section 5, the evaluation results.

## 2. RELATED WORK

### 2.1 Orthographic Conventions

The orthographic normalization of Arabic dialects has been the subject of many studies, given its importance for many NLP tools. Indeed, the authors in [10] have proposed CODA (Conventional Orthography for Dialectal Arabic). The goal of this convention is to provide a set of rules standardizing the transcription of dialectal Arabic. The convention is based on the MSA spelling rules for their decisions. [10] defined CODA for the Egyptian dialect. Then, [8] made an extension of the spelling convention for TA. Also, [11] suggested an orthographic convention for Algerian dialect based on CODA. Many other extensions were proposed, such as [12] for the Palestinian dialect, [13] for Gulf Arabic and [14] for Moroccan and Yemeni Arabic. [15] proposed CODA\* that presents a conventional orthography applicable on multiple Arabic dialects at the same time (i.e., from 28 Arab cities).

### 2.2 Orthographic Normalization Systems

In the literature, research on normalization of Arabic orthography can be classified based on the variety treated. Therefore, the remainder of this section shows the related work of both varieties MSA and dialectal Arabic.

#### 2.2.1 Modern Standard Arabic

Several works exploited language modeling within the orthographic normalization of MSA. It is a technique based on contextual information in the decision. It uses the estimation of probabilities of sequences of n words (n-grams). [16] and [17] trained language models based on «n-gram» for error correction in inserting and deleting spaces. They also addressed error detection through two character-based trigram language models to classify words as valid and invalid.

[18] used a character-based 15-gram model to deal with merged word errors. In fact, authors in [18] statistically divided them by space, forming a network. In this network, they employed a heuristic evaluation, using an n-gram probability estimation, on each character to estimate the best path through it. Thus, the sequence of letters and spaces with the highest marginal probability, given by the language model, is selected.

[19] addressed the problem of automatically detecting real-word errors by using an n-gram ( $n \in [1, 3]$ ) statistical language model and an SVM algorithm [20]. For the correction phase, the authors applied an n-gram language model to generate all error-word matches using Damerau-Levenshtein distance [21]. The test set is composed of 10K sentences from the KSU corpus<sup>2</sup> and artificially populates it with context errors using single edit distance and mixed-edit distance. The edit distance between two words

<sup>2</sup> <http://ksucorpus.ksu.edu.sa/ar>

is the minimum number of valid operations required to normalize the word (e.g. insertion, deletion or replacement of a single character). The overall F-measure value was 90.7%.

[22] built an Arabic error detection and correction system using a Bi-LSTM architecture. This classifier allows Boolean predictions rather than inferring error types. Therefore, the authors manually compiled a list of approximately 150 errors, including punctuation, spelling, morphological, syntax and named entity-recognition errors. For evaluation, they developed a corpus of 15M fully inflected Arabic words. The experimental results revealed an F-measure of 93.89%.

### 2.2.2 Dialectal Arabic

[22] proposed a system able to transform spontaneous orthography of the Egyptian dialect into the conventionalized form CODA. The authors start with a pre-processing step that eliminates letters repeated more than twice. For normalizing Egyptian dialect, [23] proposed two techniques: contextual and non-contextual. The first technique builds a unigram model that replaces every word in the spontaneous orthography with its most likely CODA form as seen in the training data based on the word level. In the second technique, a set of transformations is applied on a character level using the k-nearest neighbor algorithm (k-NN) [24]. It does not depend on the character context inside the word. In addition to the techniques discussed above, the authors have used a morphological tagger [25]. The best results come with the combination of the cited approaches with 68.1% of error reduction.

[9] proposed a method similar to that proposed for Egyptian dialect. The authors have proposed a hybrid approach to normalize the spelling of the spontaneous Tunisian Arabic (TA) based on the spelling convention CODA-TA [8]. The first method using k-NN supervised algorithm corrects the attached proclitic with generally several types of errors for the same word. Then, the linguistic method is based on pre-defined patterns and a specific lexicon for each error form.

[26] created a dataset that consists of 185K Algerian texts. The authors began by automatically pre-processing the corpus by eliminating punctuation, emoticons and reducing the number of recurring letters to not more than two. Then, the dataset was manually normalized by experts. The parallel corpus contains 50,456 words and 26,199 unique words to be normalized. [26] introduced two deep-learning models for this task, with the CNN model achieving the best evaluation result with an overall F-score of 64.74%.

Despite the richness and relevance of research, we must point out that the only orthographic normalization tool developed for the TA does not support social-media dialect, which is the most widely available and easiest to collect dialect. Furthermore, COTA orthography [9] remains semi-automatic, requiring user intervention to normalize certain words.

## 3. TUNISIAN ARABIC

In this section, we discuss the characteristics of Tunisian Arabic (TA), its different forms as well as the orthographic errors that can be found in TA writings.

### 3.1 Brief Presentation

Tunisian Arabic (TA) is a North African dialect of Arabic that represents the native language spoken in Tunisia by almost 12 million people [27]. It differs from the Modern Standard Arabic (MSA) in different levels [28]; [11]: morphology, syntax, pronunciation and vocabulary. Its lexicon contains several words from different languages, such as Maltese, Berber, French, English, ...etc. [28]; [8]. TA is classified into three different forms [27]; [3]: intellectualized dialect, spontaneous dialect and social media dialect according to the specificities of each one.

**Intellectualized dialect** [3] is mainly used by intellectuals. This form is a mixture of MSA and TA with a relatively high frequency of MSA words. Its syntactic structure is the closest to the MSA, which makes it the most regular form.

**Spontaneous dialect** is the form of communication dialect that contains the highest mass of TA words with its co-existence of multiple languages, such as Maltese, MSA, Italian and mostly French. It is characterized by the presence of disfluencies (e.g., incomplete words, repetition, filled pause, stuttering, ...etc.). Several papers proposed transcribed corpora from several audio sources, such as

[1]-[2]; [29].

Textual content of social networks represents a combination between the two forms previously cited. However, content in social media generally contains more orthographic errors than the other forms of dialect (intellectualized dialect and spontaneous dialect). This is obvious, since, at the time when tens contribute to the writing of intellectualized-dialect and spontaneous-dialect corpora, each internet user contributed with a very limited number of comments in the corpus. Moreover, each time the number of writers increases, the heterogeneity of the written words increases. In addition, we notice the presence of non-standard abbreviations, onomatopoeia, emoji, accentuation, ...etc. Social-media dialect is divided into two parts according to the alphabet character used whether Arabic or Latin «Arabizi». It is a term used to describe an encoding system that uses the Latin script and substitutes some Arabic letters with Arabic numbers instead. The Arabic numbers fill in for Arabic phonemes that are absent in the Latin language, but resemble Arabic letters and their forms, where each letter represents an Arabic phoneme that corresponds to it in pronunciation [30]. For example, the number 3 stands for the Arabic character (ع, E), the number 7 comes for (ح, H), ...etc. In this paper, we only consider the correction of text with Arabic letters.

Table 1 shows some sentences for each form of dialect with their English translation and Arabic transliteration [31]<sup>3</sup>.

Table 1. Examples of sentences of the three forms of TA.

Sentence	Translation	Script	Form of TA
الحق في القراف مضمون AlHq fy AlqrAf mDmwn	<i>The right to strike is guaranteed</i>	Arabic	Intellectualized dialect
# امم أنا étudiante في # Amm AnA étudiante fy #	<i>Amm I am a student in #</i>	Arabic & Latin	Spontaneous dialect
Bjr Hmd chna7welek enti	<i>Hello, it's OK; what's up?</i>	Latin	Social-media dialect
واو جو كيبير ☺ wAw jwkbyyyr ☺	<i>Waw a lot of fun ☺</i>	Arabic	

### 3.2 Tunisian Orthographic Errors

Most of the textual resources available are not standardized. Therefore, words can be presented in several forms, which greatly increases the error rate of any NLP applications. Table 2 presents an example of the word (ثمة, there is) and some of its different writing forms in TA. For all of the provided mistake instances, we rely on the CODA-TA convention's rules [8].

Table 2. Examples of spelling errors in the study corpus.

CODA-TA spelling	Corpus	Transliteration
ثمة <i>vmp</i>	فمة	fmp
	ثمة	vmp
	ثما	vmA
	فما	fmA
	ثمّا	vm A
	فمّا	fm A
	ثم	vm
	فم	fm
	ثاما	vAmA
	فاما	fAmA
	ثامت	vAmt

[9] detected and presented several types of errors for TA. Some of them are shared with MSA, such as writing errors of some letters (ى, Y; ا, A; ي, y, ة, p and ه, h) and the presence of space between the coordination conjunction (و, w) and the following word, ...etc. According to the CODA-TA orthographic convention, they also specified TA specific errors:

<sup>3</sup> We follow the Arabic transliteration convention: <http://www.qamus.org/transliteration.htm>

- **Space after the negation form:** according to the CODA-TA orthographic convention, they also specified TA specific errors due to the absence of space between the negation form (ما, mA) or (م, m) and the following verb, etymologically spelled, the neglecting of the silent Alif at the end of the third-person plural affix.
- **The attached proclitics:** the TA marks a set of proclitics, such as (هـ, h; ك, k; م, m; ع, E; ف, f) which must be attached to the following name. For example, the word (هلكرسي, hlkrsy) must be written according to CODA-TA as (هلكرسي, hAlkrsy, this chair).
- **Plural Waw:** the affix (وا, wA) is used to express the third-person plural, but the character (ا, A) written at the end of the word is often overlooked. For example, the normalized verb (خرجوا, xrw) is often written as (خرجو, xrw).

For social-media dialect, we found other types of errors. Among the errors, we can cite:

- **Accentuation of words.** This phenomenon represents the repetition of a letter several times successively (e.g. علااش, ElAAA\$, why). Sometimes, people intend these repetitions to show affirmation or intensification.
- **Interjection.** An interjection is a term that is grammatically independent of the rest of the sentence. It mainly expresses a short and sudden expression of emotion rather than meaning. Internet users write interjections with multiple forms, such as وواو, wAAAw instead of واو, wAw (i.e., they do not use the same number of characters).
- **Onomatopoeia.** To imitate or resemble the sound of an animal, objects or human sounds, Internet users write onomatopoeia with multiple forms. For example, they do not use the same number of characters while writing laughing sound.
- **Tatweel.** Arabic scripts present horizontal strokes. In contrast to white space that creates justification by expanding spaces between words, Tatweel increases the length of a text by elongating characters at certain points (e.g. باهي, b\_Ah\_y Good).

To summarize, several types of orthographic errors can be detected in the three forms of TA. These mistakes show the necessity for a spelling normalization tool to be implemented. In the following section, we'll go over this in more depth.

## 4. NORMALIZATION OF TUNISIAN SOCIAL MEDIA DIALECT

COTA orthography system [9] (Conventionalized Tunisian Arabic orthography) is the only system that semi- automatically corrects Tunisian Arabic (TA) spelling errors. What we propose in this paper is an extension of this system and the orthographic convention CODA-TA [8]. Our goal is to automate the task of standardizing the spelling of the social-media dialect (sub-section 3.1). Figure 1 illustrates the steps of the proposed method.

### 4.1 CODA-TA Extension

CODA-TA [8] is a convention primarily based on MSA spelling rules (see Table 3). It provides an extension of the Arabic dialect orthographic convention CODA (Conventional Orthography for Dialectal Arabic) [32]. Both CODA-TA and CODA\* are based on the convention proposed by [32]. They both share the same objectives and guiding principles. There are a few small variations between the two standards, such as the use of (برشة, br\$P, a lot) in CODA-TA *versus* (برشا, br\$A) in CODA\*. Taking the example of numbers, both conventions add the character (ن, n) at the end of the word. However, CODA\* adds the letter (ع, E) to numbers, such as (ثمانعش, vmAntE\$n, eighteen) despite the fact that Tunisians do not pronounce this character. The same example is written (ثمنطاشن, vmnTA\$n) according to CODA-TA. However, the main distinction is that CODA\* is useful in multi-dialect processing cases, while CODA-TA was designed specifically for the Tunisian dialect. There is already a semi-automatic tool (COTA Orthography) that follows its linguistic guidelines. Furthermore, many TA corpora have already been normalized using this convention.

The CODA-TA extension rules are described in the remaining paragraphs of this sub-section, along with several examples that help to make them clear.

- This convention is mainly based on the consonants and vowels of the Arabic language. For example, non-Arabic phonemes (/V/, /G/ and /P/) are converted into (ف for /f/, ق for /q/ and ب for /b/). Indeed, they generally keep the same MSA spelling rules to choose the right form to use.

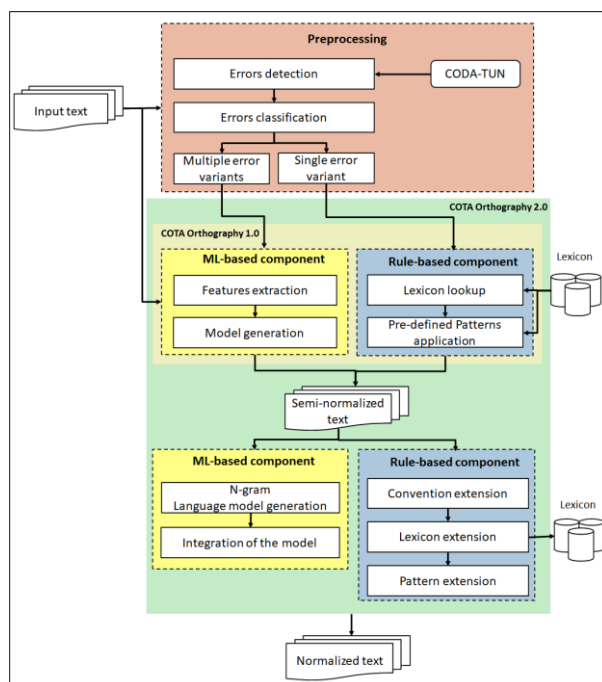


Figure 1. COTA orthography architecture.

Table 3. Most frequent CODA-TA examples.

Category	Tunisian	CODA-TA	Translation
Letter	ق /G/	ق/ق/	
	ف /V/	ف/f/	
	ب /P/	ب/p/	
Word	يقولو , yqwlw	يقول له , yqwl lh	<i>he tells him</i>
	يقول لو , yqwl lw		
	يقوله , yqwlh		
	يقل لو , yql lw		
	يقلو , yqlw		
Number	اسبعتاش باب , AsbETA\$ bAb	سبعطاشن باب , sbETA\$n bAb	<i>seventeen doors</i>
	سبعطاش باب , sbETA\$ bAb		
Enclitic	ع , E	ع , E	<i>On</i>
	م , m	م , m	<i>From</i>
	خ , x	خ , x	<i>Let</i>
Proclitic	ش , \$	ش , \$	<i>Not</i>
	شي , \$y	شي , \$y	<i>Question</i>
	ش , \$		

- Long vowels are written in a long form like in MSA. For example, the word يقول له (he tells him) can be written in TA as (يقوله , yqwlh).
- The letter (ن, n) is added after some numerical structures in CODA-TA (e.g. سبعطاشن كرهبة , sbETA\$n krhbp, seventeen cars).

- TA shares most of the attached clitics with MSA, such as definite article (ال, Al), coordinating conjunction (و, w), ...etc. Other attached clitics are specific to TA such as interrogation (شي, \$y) and negation (ش, \$) enclitics, proclitics (ع, E). Thus, the verb (خلي, xly, let) is sometimes used as a prefix and attached to the following word (e.g. خنمشيو, xnm\$ywa, let's go).

This spelling convention is more practical for word-processing corpora, especially when it comes to a method of adapting the system from MSA to TA. However, it does not take into account the phenomena of social-media dialect. For example, the sentence extracted from social-media comments presented in Table 5 as «raw text». We notice that three words of the sentence have not been corrected (مممم, mmmm), بالارشا, bAAAr\$A) and (تعب, t\_E\_b) (see «before extension» in Table 5). Therefore, we propose a set of rules that will be the subject of an extension of the orthographic convention CODA-TA. Table 4 lists examples for the different proposed patterns.

- **Accentuation of words** - All repeated characters in a word will be eliminated and we keep only one character of them (e.g. علاش, ElA\$ instead of علااش, ElAAA\$).
- **Interjections and onomatopoeia** - We unified all interjections and onomatopoeia written by a single repeated character into three characters. If they are composed of more than one character, where each character or a set is repeated sequentially, we keep only two appearances of each letter. Table 4 presents three examples of orthographic normalization of onomatopoeia.
- **Tatweel** - We proceed to eliminate all instances of Tatweel from words. For example, (باهي, b\_Ah\_y Good) turns into باهي.

After adding these rules to CODA-TA spelling convention, the comment presented above becomes correctly normalized as shown in Table 5.

## 4.2 First Version of COTA Orthography System

### 4.2.1 Method Overview

In order to normalize the spelling of Tunisian spontaneous dialect, Boujelbane et al. started by spotting transcription mistakes using CODA-TA convention. Indeed, they established two categories of errors. The first characterizes errors having several variants for the same word, such as (هالناس, hAlnAs, *these people*) which has several variants, such as ها الناس, hA AlnAs; هلناس, hlnAs; هل الناس, hl AlnAs, ...etc. The second category includes errors having only one variant, such as the word (ثقافة, vkAfp, *culture*) that can be written (ثقافه, vqAfh). COTA orthography is a correction system based on the characteristics of each type. The machine learning-based component was defined to correct the first category of errors. It is inspired by [23]. Indeed, the same list of classes was used. The k- NN correction model was created after establishing the feature set. It determines whether a character should be replaced, removed or added to another character. The linguistic method is mainly composed of two techniques: the application of a set of pre-defined patterns and the lexicon lookup. For the first technique, standardization patterns were assigned for each form of word agglutination. For the lexicon lookup, [9] proposed six sub-lexicons which include 6,063 words for two sub-lexicons of similar spelling errors, 1,632 for etymologically spelled consonants, 1,066 words for the sub-lexicon of the third-person singular pronoun, 111 CODA-TA word list and 5,674 for waw of plurality. [9] tested COTA orthography system with a part of the non-normalized version of STAC [1] that contains 10,236 words (2,640 wrong words). The accuracy result achieved was 86.6%.

Table 4. List of new CODA-TA normalization examples.

Spelling error	Error	CODA-TA	Translation
Accentuation	brrr\$پ برشة	br\$پ برشة	<i>a lot</i>
Onomatopoeia	hhhh هههه	hhh ههه	
	hhxxx ههخخ	hhxx ههخخ	
	hEhEhE ههههه	hEhE هههه	
Tatweel	t_wn_s تونس	twns تونس	<i>Tunisia</i>
Attached clitics	Al mdrsp ل مدرسة	Almdrsp المدرسة	<i>the school</i>



Expression containing the name «Allah»	n\$AAIhh نَشَالله	An \$A All h ن شَالله	<i>Allah willing</i>
Word Each	kAlwAHd كلوَاد	kl wAHd كل وَاَد	<i>Each one</i>

Table 5. Example of TA comment normalized according to CODA-TA before and after the extension.

	Example	Translation
Raw text	مممم الخدمة فيها بالارشا تعب ربي يعينو mmmm Alxdmh fyhA bAAAr\$A t_E_b rby yEynw	<i>Mmm this work is very tiring, may God help him</i>
Before extension	مممم الخدمة فيها بالارشا تعب ربي يعينه mmmm Alxdmp fyhA bAAAr\$A t_E_b rby yEynh	
After extension	ممم الخدمة فيها برشمة تعب ربي يعينه mmm Alxdmp fyhA bAr\$P tEb rby yEynh	

#### 4.2.2 COTA Orthography System Errors

Although the COTA orthography system achieves an encouraging result for the Tunisian spontaneous dialect, we notice several failure cases that were not considered during the system's implementation.

The social-media dialect represents a valuable source of data for researchers. Despite this, the system is unable to detect and correct a variety of errors. For example, the different types of onomatopoeia (e.g. ههه, hhh), accentuation (e.g. كيببير, kbyyyr, big) and Tatweel signs (e.g. نوح, n\_wH, Noah) are not taken into account and the system does not correct them. Moreover, social-media dialect corpora contain words with the character (ا, A) added at the beginning of some words, such as the verb (دخلتوا, dxltwA, you got in) that can be found as (ادخلتوا, AdxltwA). Furthermore, some of the suggested patterns only called for a specific grammatical category. However, several words, generally used in social-media dialect, cannot be detected and corrected by the system (e.g. لا يكيو, lAykyw, like) is an incorrectly written word which is generally used in social-media dialect corpora.

Sometimes, Internet users forget to add a space between the comment's words, which implies two or more words attached. Even native speakers of the dialect are often unable to read sentences that do not contain any spaces to delimit the words (e.g. the two attached words تصير ساعات, tSyrSAEAt which means it happens sometimes). COTA orthography system does not take this type of error into account.

The term (كل, kl, each) is frequently used as quantity noun in TA. Nevertheless, COTA orthography normalizes the different types of errors in the (ك, k) enclitic, which may result in the modification of expressions containing this quantity nouns, creating text distortion. For instance, the rate of comments involving this form of error did not exceed 19% in our study corpus (80% of TAD [6]).

Another form of error caused by [9]'s system arises when changing words containing consonants with multiple pronunciations. For example, the verb (صمن, Smn, solidified) becomes (سمن, smn, gained weight). However, Tunisians use both terms. As a result, we cannot judge whether the Internet user is indicating « solidified » or « has gained weight ».

Moreover, COTA orthography system [9] grants several alternatives for some words separated by «/». These terms require more than a correction regardless of the subject. In other words, each option can be valid in a given context and wrong otherwise. For instance, words ending with (و, w) may mean the third-person singular pronoun (ه, h) or the affix of the third-person plural (وا, wA). It depends on the context of the word in the sentence. Take the example of the word (فهمتو, fhmtw) in TA, which does not follow any spelling convention. When it is considered as the third-person singular pronoun, the word is corrected by: (فهمته, fhmth), which means (I explained to him) or (I get it). However, if it is considered as the affix of the third-person plural, the word will be corrected by (فهمتوا, fhmtwA, you understood). Therefore, COTA orthography system gives the two writing

choices separated by a slash «/» which requires a manual decision between the two propositions depending on the context.

### 4.3 Second Version of COTA Orthography System

In this sub-section, we detail our proposed method for the orthographic normalization of TA errors. We focus on texts from social-media dialect. We start by extending the CODA-TA spelling convention [8]. Subsequently, we propose to automatically extract a lexicon from the Tunisian Treebank «TTB» [33] and a list of predefined patterns. Thus, we create a language model to process multiple-choice words.

#### 4.3.1 Linguistic Techniques' Extensions

To extend the linguistic techniques, we have used three social-media dialect corpora to fix orthographic errors. The first one is TAD (Tunisian Arabic Dialect) [6]. They extracted 73,024 messages of which over 72% are in Latin letters. These messages went through three steps of processing: spam filtering, message division (Arabic or Latin characters) and message classification (dialectal or non-dialectal). TAD is composed of 7,145 messages (151,598 words) written in Arabic letters. The messages are collected from Facebook comments, messages from mobile phones, ...etc. The corpus contains only dialectal texts [6]. This dataset is available by email request to the first author.

The second corpus is of Masmoudi et al.'s corpus [34]. It is a collection of 21,917 words extracted from Tunisian blogs treating various fields (politics, sports, culture, science, ...etc.). Two experts who are native Tunisian Arabic (TA) speakers have validated the comments as TA. They manually translated 3,500 Arabizi words (530 sentences) into Arabic script. We get access to this corpus by emailing the first author.

The TSAC<sup>4</sup> (Tunisian Sentiment Analysis Corpus) [5] contains 17,000 comments that are classified to positive (63,874 words) and negative (49,322 words) polarities. This data was collected from Facebook comments that are written on the official pages of Tunisian radio and television channels (Mosaïque FM, JawhraFM, HiwarElttounsi TV, ...etc.). The version proposed in GitHub is licensed under the GNU-v3.0.

We used 80% of TAD corpus [6] for the extraction of patterns and the enrichment of sub-lexicons. The remaining part was used for the test. Furthermore, we randomly selected 43,247 words from TSAC [5] and Masmoudi et al.'s corpus [34]. Indeed, two native speakers manually normalized them according to CODA-TA [8]. We calculated the inter-annotator agreement to measure how well our two experts can make the same normalization. The obtained kappa value is 0.896, indicating a high degree of concordance. Table 6 presents the size of the corpus and its error rate according to CODA-TA [8].

Table 6. Details about the test set for system evaluation.

Corpus	Number of words	Error rate
TAD (6)	22,740	22.62%
Corpus of (34)	10,371	20.16%
TSAC (5)	10,136	24.85%
<b>Total</b>	<b>43,247</b>	<b>22.55%</b>

**4.3.1.2 Extension of the Lexicon:** [9] collected a set of six sub-lexicons for each error form. It can help in detecting spelling errors (i.e., if the word is not recognized by the system, it will not be corrected). Moreover, for the correction phase, two contributions are possible: the parallel sub-lexicon containing the incorrect word and its normalized equivalent can be used for substituting the wrong detected form by the correct one. Also, it is also used to call certain patterns.

Based on our study corpus, we semi-automatically enriched these lexicons with new words. For example, we added several verbs, such as (برتاڟي, brtAjy, share), (عدل, Edl, adjust), ...etc. Thus, we

<sup>4</sup> <https://github.com/fbougaes/TSAC>

noticed that many Internet users frequently add the character (l, A) to the beginning of words. Therefore, we proposed to add a new sub-lexicon to correct this type of error. This lexicon is collected from various sources (STAC corpus [1], the Tunisian constitution [7], Boujelbane's corpus [35] and Younes's corpus [6]).

Table 7 details the size of the sub-lexicons before and after the update. We extracted 767 verbs, 118 nouns and 44 pronouns automatically from TTB [33]. Then, all the words were validated by native speakers.

Table 7. The new size of sub-lexicons.

Lexicon	Initial size	Final size
List of verbs	5,435	6,202
List of nouns	914	1,032
The third-person singular pronoun	1,066	1,110
Etymologically spelled consonants	1,632	1,514
List of words that start with A	-	7,219

Otherwise, the TA has many consonants with multiple pronunciations. The letter س s can be pronounced as ص S, which inducts multiple spellings. Therefore, [9] built two sub-lexicons for these consonants. For example, the first sub-lexicon is composed of a list of words containing the consonant س and their equivalents in the incorrect writing. By studying these sub-lexicons, we found a set of polysemous examples (i.e., the word proposed as a mistake is correct in another context). For example, the word (سورة, swrp, Surah) can be written incorrectly into (صورة, Swrp, picture) but most likely the writer means Surah<sup>5</sup>. Therefore, to solve this problem in this stage, we propose to eliminate these words from the sub-lexicons. We removed 75 words from the sub-lexicon ص to س, where the total number of words becomes 901 and 43 words from the sub-lexicon that transform the letter س to ص with a new total of 393 words.

**4.3.1.2 Normalization Patterns:** To improve the performance of COTA orthography system, we propose a set of manually implemented patterns to correct social-media dialect spelling errors (presented in Section 3) and to correct others TA errors not covered by [9] (see Table 8).

**Attached Clitics:** [9] generated a set of models that are able to correct errors related to attached clitics. These models are not able to correct spelling mistakes of the following clitics: ـل, Al; ـل, ll; ـل, l; and ب, b. Therefore, we defined a pattern that deletes the space between one of these clitics and the following word (see pattern 1 in Table 8).

**Expressions Containing the Name «Allah»:** Each person writes the expressions containing the name God in his own way (e.g. الحمد لله, AlHmdll h) turns into (الحمد لله, AlHmd l lh, Thank God)). Thus, we created a pattern that detects wrong expressions and corrects them according to the convention CODA-TA (see pattern 2 in Table 8).

**Word كل kl:** The numerical approach proposed by [9] deals automatically with quantity nouns كل, kl as an error of writing (i.e., as the attached clitic ك, k). In the study corpus, we remark that the «k model» of [9] increases the error rate by 83.18%. Therefore, we developed a pattern that covers and avoids all changes of the quantity nouns (كل, kl, each) (see pattern 3 in Table 8).

Table 8. Examples of patterns.

Number	Pattern
1 - Attached clitics	<b>IF</b> len(word) == 1 <b>AND</b> word <b>IS</b> valid_clitic <b>THEN</b> Remove space after word
2 - Expression containing the name "Allah"	<b>IF</b> "Allah" <b>IN</b> word <b>THEN</b> Replace word with normalized form

<sup>5</sup> The Quran is divided into Surahs (chapters). Source: [https://en.wikipedia.org/wiki/List\\_of\\_chapters\\_in\\_the\\_Quran](https://en.wikipedia.org/wiki/List_of_chapters_in_the_Quran)

3 - Word كل kl	<b>IF</b> word <b>IS</b> quantity_nouns <b>THEN</b> Do nothing
4 - Onomatopoeia 1	<b>IF</b> word <b>CONTAINS_ONLY</b> letter <b>AND</b> len(word) > 3 <b>THEN</b> Remove extra letters
5 - Onomatopoeia 2	<b>IF</b> word <b>CONTAINS_ONLY</b> (letter_1 <b>AND</b> letter_2) <b>AND</b> len(word) > 4 <b>THEN</b> Remove extra letters

Table 9. List of new normalization pattern examples.

Pattern	Before normalization	After normalization	Translation
Accentuation	mzyAAAn مزياان	mzyAn مزيان	<i>beautiful</i>
Interjection	Ammm اممم	Amm امم	
Nomatopoeia	hhhAA هههاا	hhAA ههها	
	Hyhyhy هيهيهي	hyhy هيهيهي	
Tatweel	x_wy_A خويا	xwyA خويا	<i>my brother</i>
Attached clitics	Al klyp ال كلية	Alklyp الكلية	<i>the university</i>
Expression containing the name «Allah»	m\$Allh مثالله	mA \$A Allh ما شا الله	<i>machallah</i>
Word each	kAlEbd كالعبد	klEbd كل عبد	<i>each person</i>

**Interjections, Onomatopoeia and Accentuation:** Interjections, onomatopoeia and accentuation are often used in social media dialect. We developed a set of patterns that detect these terms and correct them by removing repeated characters (see patterns 4 and 5 in Table 8).

**Tatweel:** We proposed a pattern that eliminates all Tatweel forms in the input. Table 9 shows some examples of the new normalization patterns.

#### 4.3.2 Language Model

COTA orthography system provides a semi-automatic normalization for terms that imply more than one correction, independently of the context. Therefore, we introduce in this sub-section our method for automating this task. This method relies on the comparison of two language models to fix errors semi-processed by the system while taking advantage of the textual resources already created in favor of TA [3]; [1]; [34]; [5]-[6]; [36] and MSA [37]. Table 10 presents the size of each corpus.

We started with the first step of preparing the collected textual resources that consists in checking the normalization of the TA corpus using the COTA orthography system [9] and manually selecting the correct option from the choices given by COTA orthography system. Our result corpus consists of 7,571 multi-choice words. We have used diverse datasets from different fields and topics to generate the language models. The total size of the corpus in TA is 379,063 words.

Table 10. Size of corpora used for language-model generation.

Corpus	Size
[3]'s corpus	37,964
STAC [1]	42,388
TAD [6]	151,598
TSAC [5]	113,196
Normalized Tunisian constitution [36]	12,000
[34]'s corpus	21,917
KACST corpus [37]	2,207,469
<b>Total</b>	<b>2,586,532</b>

- The first corpus [3] is a transcription of 5 hours and 20 minutes of recordings mainly from a Tunisian television channel. This corpus contains 37,964 words, where 12,207 words come from a TV news program and 25,757 words from programs of political debates. This corpus is available by email request to the first author.
- The second is the STAC<sup>6</sup> (Spoken Tunisian Arabic Corpus) [1] containing 42,388 words (4 hours and 50 minutes of recordings). It is a transcription and annotation of spontaneous TA spoken in various TV and radio channels. It has 97.20% words in TA, 0.37 % in MSA and 2.43 % in French. STAC includes disfluencies. It is licensed under the GNU-v3.0.
- TA constitution [7] is an intellectualized dialect that consists of 12,000 words, normalized by [36]. It is available by emailing the first author.
- TAD, TSAC and Masmoudi et al. datasets [6]; [5]; [34] (see description in sub-section 4.3.1).
- KACST (King Abdulaziz City for Science and Technology) (37) MSA corpus is made up of 2,207,469 words, carefully sampled and its content is classified according to different parameters, such as time, country, field, subject, etc. KACST is licensed under the GNU-v3.0.

We have divided the corpus into training, development and test sets (80/10/10, respectively) according to the number of multiple-choice words.

### N-gram-based Language Model

N-gram models are among the most commonly used language models of spell checking, due to their flexibility and utility. In this type of model, the probability of a word is calculated as a function of its history (the previous n-1 words). These probabilities are determined depending on the count of each sequence detected. The n-gram model was trained on our learning corpus. To obtain an efficient language model, we have configured the basic model generated using the development corpus by suggesting all the possible hypotheses (i.e., offered by the COTA orthography system) for a given sentence. Then, the model assigns a perplexity value to each proposition. As a result, the sentence admitting the lowest perplexity is held as correct.

In the following paragraphs, we describe the process of setting up our model:

- **Fixation of the n-gram.** We trained 6 models with different n-grams to select the most adequate n-grams. The best generated model reached 65.46% using the trigram model.
- **Adjustment of the learning corpus.** We automatically checked the normalization training data using COTA orthography system. Therefore, the training set contains orthographic errors that increase the perplexity of the model. Thus, the elimination of repetitive sentences resulted in 5.72% improvement with an accuracy equal to 71.18%. In the literature, orthographic normalization works using language models are based on large corpora [17]; [16]; [18]. We thus tried to extend the size of our training corpus. Due to the lack of TA textual resources, we decided to add an MSA corpus to our textual base. Indeed, COTA Orthography system relies on MSA spelling rules to correct errors [9]. In addition, MSA supports universally known spelling rules. Therefore, its corpora often do not contain spelling mistakes (i.e., especially written by journalists - case of KACST corpus). Adding KACST [37] to the TA corpus without duplications, raised the accuracy to 72.34%.
- **Choice of options.** Among the available options, -unk was the unique alternative to mark an improvement in the language model. Using the default configuration, Out-Of-Vocabulary (OOV) words are deleted. When using this option, the language model keeps unknown words and treats them as normal words (not OOV). As a matter of fact, it allows keeping the vocabulary open. This addition marked an increase in the accuracy by 9% to reach 81.34%.

### LSTM-based Language Model

GluonNLP [38] is a natural-language processing deep learning-based toolkit. Several models have been supplied by this toolkit for natural-language processing tasks, such as word embedding, language

<sup>6</sup> <https://sites.google.com/site/ineszribi/ressources/corpus>

modeling, machine translation, etc. In this paper, we used GluonNLP to implement a typical LSTM language model architecture. Then, we trained the language model on our training dataset (see Table 10). Several experiments were carried out to improve the LSTM-based language model using the development corpus. We chose the best alternative according to the perplexity result.

Grid search was applied to fine-tune the parameters of the LSTM-based language model. The optimal configuration is based on a batch size of 64, Adam optimizer and Softmax function. The number of epochs is set to 100. At this stage, we get an accuracy of 80.17% using the development set. As for the N-gram-based language model, we tested the effect of adding the MSA corpus to the training set, which improved the result by 1.01%. The best obtained language model reached an accuracy of 81.18%.

We can conclude that the N-gram technique is just 0.16% better. Therefore, we conducted non-parametric tests on the dataset. The p-value for the non-parametric independent Wilcoxon test [39] is 0.031. Since the p-value is less than the threshold of 0.05, we can conclude that the values are statically significant.

## 5. EVALUATION RESULTS

In this part, we provide the language model's experimental results as well as the new version of the COTA orthography system (Conventionalized Tunisian Arabic orthography) while describing the qualitative analysis of these results.

### 5.1 Experimental Results

#### 5.1.1 Language Model Evaluation

We tested our final trigram model using the test corpus (10% of the collected corpus). By entering the input text, COTA orthography system begins by applying the linguistic techniques. This processing gives us a semi-automatic result. The language model is then used to select the best alternative to keep in sentences with several choices. For example, by entering a sentence admitting two choices, we enter two alternatives to the model (i.e., the first sentence contains the first option, whereas the second sentence contains the second option). The model grants perplexity to each sentence and we keep the one with the lowest value of perplexity. This process is fully automatic. Afterwards, we created a reference version of the test corpus to validate the model's output. The accuracy result obtained using our language model is equal to 79.38%.

#### 5.1.2 COTA Orthography System 2.0 Evaluation

In this sub-section, we seek to examine how the complementary patterns and sub-lexicons of social-media dialect can generate additional gains in Tunisian Arabic (TA) automatic normalization.

We present in Table 11 the errors that we treated with their accuracy, their frequency and their percentage in the erroneous part of the test corpus (i.e., the overall results are presented in Table 12). The best accuracy value found achieves 100% for the 6 patterns of interjections, onomatopoeia, word JS, attached clitics, expression containing the name «Allah» and «Tatweel». However, words starting with the character A give the lowest value with 68%.

For the consonants, we detected an accuracy of 72.73% by testing the corpus with the basic system. However, filtering the sub-lexicon increases the results by 50%. Similarly, all interjections, onomatopoeia, accentuation of words or Tatweel errors have not been corrected with the system of [9].

Table 11. System performance for each spelling error with the test corpus.

Orthographic errors	Frequency	Percentage	Accuracy
Accentuation	430	8.4%	96.98%
Interjections and onomatopoeia	329	8.4%	100%
Nouns list	212	4.12%	88.68%
Verbs list	172	3.34%	80.81%
Words starting with A	94	1.83%	67.86%

Word <i>kl</i>	89	1.73%	100%
The third-person singular pronoun	59	1.15%	86.44%
Clitics attached	57	1.1%	100%
Expression of the name of «Allah»	36	0.7%	100%
Tatweel	32	0.62%	100%
Consonants	22	0.43%	72.73%
<b>Total</b>	<b>1,532</b>	<b>31.82%</b>	<b>90.32%</b>

Table 12. Evaluation 1: Results of the normalization system based on the TAD corpus.

Measurement	COTA 1.0	COTA 2.0	Improvement
Number of wrong words	5,143		-
Number of properly corrected words	3,399	5,031	1,632
Number of uncorrected words	1,927	136	1,791
Recall	66.09%	97.28%	+ 31.19%
Precision	63.82%	94.2%	+ 30.38%
F-measure	64.94%	95.72%	+ 30.78%

For the evaluation of our orthographic normalization system, we calculate the measures of recall, precision and F-measure based on the number of properly corrected words.

We tested the new version of the system with the same corpus of spontaneous dialect used for the test [1]. The result showed a 3% improvement over the old version of the system. We obtained 91% of recall, 89% of precision and 90% of F-measure.

We evaluated the system with the corpus of [6] (test part). Our results are presented in Table 12. All of these results are significantly better than those of the old version of COTA orthography system. We conducted additional evaluations with alternative corpora ([34] and [5]). These results are shown in Table 13.

Table 13. Evaluation 2: Results of the normalization system based on Masmoudi et al.'s and TSAC corpora.

Measurement	Masmoudi et al.	TSAC
Number of wrong words	2,091	2,519
Number of properly corrected words	1,933	1,991
Number of uncorrected words	249	617
Recall	92.44%	79.04%
Precision	88.59%	76.34%
F-measure	90.47%	77.67%

## 5.2 Discussion

In the following part, we discuss the results achieved for the trigram-language model as well as version 2.0 of COTA orthography.

### 5.2.1 Language Model Evaluation

The language model manages to correctly choose the right suggestion. For example, the sentence (يعينوا/الله يعينه, All h yEynh/yEynwA) is corrected as follows: (الله يعينه, All h yEynh, may God help him). However, the following sentence: (علاش نشاركو, EIA\$ n\$Arkw, Why are we participating?) is poorly normalized (علاش نشاركه, EIA\$ n\$Arkh). In fact, if the wrong option is made, it is mainly for two reasons. First, the correct choice is not included in the corpus. Second, the probability of appearance of the incorrect choice in the learning corpus is higher than that of the correct choice. Moreover, we notice that sometimes the candidates for standardization get the same perplexity. In fact, this is due to the balance of the probabilities of appearance of the two choices in the learning corpus. The test set includes 6.25% of these sentences. Thus, we set the first option as default.

To reduce the invalid selection of the alternatives, we need to add more of the sentences with

demonstrative pronouns (i.e., feminine (هاكي, hAky) and masculine (هاكه, hAkh)) as well as words ending with (ه, h) and (وا, wA) in order to select more precisely the most appropriate alternative for our context.

### 5.2.2 COTA Orthography System 2.0 Evaluation

In general, the results are encouraging. However, among the errors we have detected, we can cite the example of word خدمتو that is corrected as (خدمته, xdmt, his job). The modification of this word can be true in a defined context, but false in another, depending on the grammatical category of the word. In fact, the word خدمته can indicate the verb «they worked» that should turn into (خدمتوا, xdmtwA). It can also represent the term «his job» which is correctly written.

The error analysis shows that some words are not corrected or wrongly modified due to our lexicon that does not cover some words. For example, the system eliminates the character I from the beginning of the word (ازين, Azyn), while the correct form is (يزين, yzyn, decorate).

The first evaluation using the TAD test corpus shows the best F-measure result with 95.72%. Our system has been improved by over 30% compared to the COTA orthography system. The results obtained are encouraging. Thus, for the second evaluation, the results are higher than 90%. Clearly, the construction of extra-patterns improved the system performance with additional and higher quality sublexicons.

Up to the third evaluation, we tested the system with more difficult cases. TSAC corpus [5] is a corpus dedicated to the analysis of feelings. It is misspelled, which explains that it gives the highest error rate among the three corpora. We shall note that the performance is significantly lower than the corresponding results of the two other corpora, which explains the degradation of the value of F-measure by more than 15%. Therefore, we tried to analyze the failure cases to understand their causes.

First, the most common mistake we have encountered comes from the attached words. It represents more than 20% of uncorrected errors in the corpus TSAC. Internet users can forget to type the space between words. We even found a 100-character sentence without any space to delimit the words. This sentence is not legible even for native speakers of TA. Thus, we can catch other errors, such as missing letters (e.g. الاحترا, AlAHtrA instead of الاحترام, AlAHtrAm, respect), added letters (e.g. تونسيلة, twnsylp instead of التونسية, twnsyp, Tunisian), wrong letters (e.g. غلاش glA\$ instead of علاش, ElA\$, why), ...etc.

Furthermore, the use of a lexicon cannot cover all the words of the TA. Hence, some wrong words in the corpus do not admit any change, since they do not exist in the lexicon.

Table 14. Extrinsic evaluation of the second version of COTA orthography system.

Task	Corpus	Recall	Precision	F-measure
Segmentation	Raw text	69.83%	72.33%	71.06%
	<b>Automatically normalized text</b>	<b>76.9%</b>	<b>83.09%</b>	<b>79.88%</b>
POS Tagging	Raw text	71.93%	74.43%	73.16%
	<b>Automatically normalized text</b>	<b>78.92%</b>	<b>81.49%</b>	<b>80.18%</b>
Parsing	Raw text	47.78%	49.4%	48.58%
	<b>Automatically normalized text</b>	<b>70.31%</b>	<b>68.24%</b>	<b>69.26%</b>

### 5.3 Extrinsic Evaluation

We performed an extrinsic evaluation of our COTA orthography 2.0 system by evaluating the impact of its use on the TA segmenter [40], POS tagger and parser [33].

[40] examined three different methods (deep learning, CRF and SVM) for segmenting TA sentences. Several experiments were carried out in order to enhance the proposed models. Subsequently, the evaluation using a test set of 26.036 words from [1]; [36]; [41]; [6]; [40] revealed that the CRF model produced the highest performance (F-measure = 84,37%), with a 21.47% improvement over deep learning, 18.9% increase over SVM and 23% compared to STAR-TUN system [42].

[33] suggested a semi-automatic annotation method of treebank annotation for the social-media dialect as well as the generation of a parsing model that covers all forms of TA. To enrich the TTB treebank,



the authors annotated a part of the TAD corpus [6]. [33] experimented with different combinations of corpora to generate the best parsing model. This system can be used for POS tagging and parsing tasks. The model is available by email request to the first author.

We prepared automatically segmented, POS tagged and parsed two versions of our test corpus composed of 43K words (see Table 6): the raw corpus that does not follow any spelling convention and the automatically normalized corpus using COTA orthography 2.0. Moreover, two native TA speakers prepared manually POS tagged and parsed versions as a reference for the extrinsic evaluation tasks. Then, both versions were automatically segmented by the TA segmenter. Afterwards, the output provided was compared to the manually segmented version.

According to [40]'s statistics, 33% of the written audio laughter (ههه, hhh) occurs in the first word of the sentence, while 48% of it is in the end of the sentence. Therefore, the normalization of these terms has a significant impact on the segmentation result. The result of [40]'s system improved by 8.82% using the automatically normalized version of the corpus. For POS tagging and parsing, we evaluated the system by the raw (non-normalized) test corpus. The obtained results were 73.16% for POS tagging and 48.58% for parsing. Using the automatically normalized corpus, we achieved better results (80.18% for POS tagging and 69.26% for parsing) with an improvement between 7% and 20% (see Table 14). These evaluations show that our COTA orthography 2.0 system will contribute to the improvement of TA tools.

We may draw the conclusion that by reducing the orthographic heterogeneity, COTA orthography solves the spelling problems and simplifies the experience. One main benefit of using the proposed system is its accuracy in normalizing texts from all the forms of TA. Running a spell checker ensures that the number of orthographic errors in a document decreases significantly, which is very helpful for several NLP tools, such as text segmentation, POS tagging, parsing, etc. (see Table 14). Since it is an entirely automatic tool, it presents good practice to assure high quality without losing time for manual spell checking.

## 6. CONCLUSION AND FUTURE WORKS

In this paper, we presented an automatic system for orthographic normalization of the Tunisian Arabic. To begin, we expanded the CODA-TA spelling convention [8]. We also extended an existing tool for TA, COTA orthography system [9], by adding new patterns and lexicon. The lexicon was automatically extracted from the Tunisian Treebank «TTB» [33]. Then, a set of patterns was defined to correct social-media dialect errors, such as accentuation, interjections, onomatopoeia, attached clitics, Tatweel, ...etc. We also added a language model that is able to choose the appropriate correction automatically. We experimented with the effect of using several options and different corpora combinations to improve the model. Our experiments show that we can improve the overall system performance by 30.78% and 3%, respectively for social-media dialect and spontaneous dialect. Moreover, the use of our system resulted in an increase of about 9% in the outcomes of automated TA segmentation.

In future works, we plan to correct the problem of attached words. Moreover, we consider experimenting with the impact of applying deep learning techniques. We take into account a comparison of the trigram-language model with other neural techniques and apply a tie-breaking method. We will also investigate incorporating this tool in a platform that contains all the linguistic tools of TA.

## REFERENCES

- [1] I. Zribi, M. Ellouze, L. H. Belguith and P. Blache, "Spoken Tunisian Arabic Corpus STAC: Transcription and Annotation," *Research in Computing Science*, vol. 90, pp. 123-135, 2015.
- [2] A. Masmoudi, M. Ellouze Khmekhem, Y. Esteve, L. Hadrich Belguith and N. Habash, "A Corpus and Phonetic Dictionary for Tunisian Arabic Speech Recognition," *Proc. of the 9<sup>th</sup> Int. Conf. on Language Resources and Evaluation*, vol. 3, no. 1, pp. 306–310, 2014.
- [3] R. Boujelbane, M. Ellouze, F. Béchet and L. Belguith, "De l'arabe Standard *vers* l'arabe Dialectal: Projection de Corpus et Ressources Linguistiques en vue du Traitement Automatique de l'oral dans les Médias Tunisiens," *TAL. 2. Traitement Automatique du Langage Parlé*, vol. 55, pp. 73–96, 2014.

- [4] A. Masmoudi, N. Habash, M. Ellouze, Y. Estève and L. H. Belguith, "Arabic Transliteration of Romanized Tunisian Dialect Text: A Preliminary Investigation," Proc. of the 16<sup>th</sup> Int. Conf. on Computat. Linguistics and Intelligent Text Process. (CICLing 2015), pp. 608–619, Cairo, Egypt, 2015.
- [5] S. Mdhaffar, F. Bougares, Y. Eve and L. Hadrich-Belguith, "Sentiment Analysis of Tunisian Dialect: Linguistic Resources and Experiments," Proc. of the 3<sup>rd</sup> Arabic Natural Language Processing Workshop (WANLP), pp. 55–61, Valencia, Spain, 2017.
- [6] J. Younes, H. Achour and E. Souissi, "Constructing Linguistic Resources for the Tunisian Dialect Using Textual User-generated Contents on the Social Web," Proc. of the 15<sup>th</sup> Int. Conf. on Current Trends in Web Engineering, ICWE 2015 Rotterdam, pp. 3–14, The Netherlands, 2015.
- [7] S. El Klibi, S. El Hamzaoui, H. Ben Abda, C. Kaddes, F. El Horcheni and A. Maalla, *La Constitution en Dialecte Tunisien*. Tunisie: Association Tunisienne de Droit Constitutionnel, 2014.
- [8] I. Zribi, R. Boujelbane, A. Masmoudi, M. Ellouze, L. H. Belguith and N. Habash, "A Conventional Orthography for Tunisian Arabic," Proc. of the 9<sup>th</sup> Int. Conf. on Language Resources and Evaluation, European Language Resources Association (ELRA), pp. 2355–2361, Reykjavik, Iceland, May 2014.
- [9] R. Boujelbane, I. Zribi, S. Kharroubi and M. Ellouze, "An Automatic Process for Tunisian Arabic Orthography Normalization," Proc. of the 10<sup>th</sup> International Conference on Natural Language Processing (HrTAL2016), Dubrovnik, Croatia, 2016.
- [10] N. Habash, M. T. Diab and O. Rambow, "Conventional Orthography for Dialectal Arabic," Proc. of the 8<sup>th</sup> Int. Conf. on Language Resources and Evaluation, European Language Resources Association (ELRA), pp. 711–718, Istanbul, Turkey, May 23–25, 2012.
- [11] H. Saadane and N. Habash, "A Conventional Orthography for Algerian Arabic," Proc. of the 2<sup>nd</sup> Workshop on Arabic Natural Language Processing, pp. 69–79, [Online], Available: <http://www.aclweb.org/anthology/W15-3208>, Beijing, China, July 2015.
- [12] M. Jarrar, N. Habash, F. Alrimawi, D. Akra and N. Zalmout, "Curras: An Annotated Corpus for the Palestinian Arabic Dialect," Language Resources and Evaluation, vol. 51, pp. 745–775, 2016.
- [13] S. Khalifa, N. Habash, D. Abdulrahim and S. Hassan, "A Large Scale Corpus of Gulf Arabic," Proc. of the 10<sup>th</sup> Int. Conf. on Language Resources and Evaluation (LREC 2016), Portorož, Slovenia, May 2016.
- [14] F. Al-Shargi, A. Kaplan, R. Eskander, N. Habash and O. Rambow, "Morphologically Annotated Corpora and Morphological Analyzers for Moroccan and Sanaani Yemeni Arabic," Proc. of the 10<sup>th</sup> Int. Conf. on Language Resources and Evaluation (LREC 2016), pp. 1300–1306, Portorož, Slovenia, 2016.
- [15] N. Habash, F. Eryani, S. Khalifa et al., "Unified Guidelines and Resources for Arabic Dialect Orthography," Proc. of the 11<sup>th</sup> Int. Conf. on Language Resources and Evaluation (LREC-2018), Miyazaki, Japan, 2018.
- [16] M. Attia, M. Al-Badrashiny and M. Diab, "Gwu-hasp-2015@ qalb-2015 Shared Task: Priming Spelling Candidates with Probability," Proc. of the 2<sup>nd</sup> Workshop on Arabic Natural Language Processing, pp. 138–143, Beijing, China, 2015.
- [17] M. Attia, P. Pecina, Y. Samih, K. Shaalan and J. Van Genabith, "Arabic Spelling Error Detection and Correction," Natural Language Engineering, vol. 22, no. 5, p. 751, 2016.
- [18] M. I. Alkanhal, M. A. Al-Badrashiny, M. M. Alghamdi and A. O. Al-Qabbany, "Automatic Stochastic Arabic Spelling Correction with Emphasis on Space Insertions and Deletions," IEEE Transactions on Audio, Speech and Language Processing, vol. 20, no. 7, pp. 2111–2122, 2012.
- [19] A. M. Azmi, M. N. Almutery and H. A. Aboalsamh, "Real-word Errors in Arabic Texts: A Better Algorithm for Detection and Correction," IEEE/ACM Transactions on Audio, Speech and Language Processing, vol. 27, no. 8, pp. 1308–1320, 2019.
- [20] V. N. Vapnik, *The Nature of Statistical Learning Theory*, New York, NY, USA: Springer-Verlag, New York, Inc., 1995.
- [21] F. J. Damerau, "A Technique for Computer Detection and Correction of Spelling Errors," Communications of the ACM, vol. 7, no. 3, pp. 171–176, 1964.
- [22] M. Alkhatib, A. A. Monem and K. Shaalan, "Deep Learning for Arabic Error Detection and Correction," ACM Transactions on Asian and Low-resource Language Information Processing (TALLIP), vol. 19, no. 5, pp. 1–13, 2020.
- [23] R. Eskander, N. Habash, O. Rambow and N. Tomeh, "Processing Spontaneous Orthography," Proc. of the 2013 Conf. of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 585–595, Atlanta, Georgia, June 2013.
- [24] J. Wang and J.-D. Zucker, "Solving the multiple-instance problem: A Lazy Learning Approach," Proc. of the 17<sup>th</sup> Int. Conf. on Machine Learning (ser. ICML'00), pp. 1119–1126, San Francisco, USA, 2000.
- [25] N. Habash, R. Roth, O. Rambow, R. Eskander and N. Tomeh, "Morphological Analysis and Disambiguation for Dialectal Arabic," Proc. of the Human Language Technologies: Conf. of the North American Chapter of the Association of Computational Linguistics, pp. 426–432, Atlanta, USA, 2013.
- [26] W. Adouane, J.-P. Bernardy and S. Dobnik, "Normalizing Non-standardized Orthography in Algerian Code-switched User-generated Data," Proc. of the 5<sup>th</sup> Workshop on Noisy User-generated Text (W-NUT 2019), pp. 131–140, Hong Kong, China, 2019.

- [27] A. Mekki, I. Zribi, M. Ellouze Khmekhem and L. Hadrach Belguith, "Critical Description of TA Linguistic Resources," Proc. of the 4<sup>th</sup> Int. Conf. on Arabic Computational Linguistics (ACLing 2018) & Procedia Computer Science, Dubai, United Arab Emirates, 2018.
- [28] S. Mejri, M. Said and I. Sfar, "Plurilinguisme et Diglossie en Tunisie," Synergies Tunisie, vol. 1, pp. 53–74, 2009.
- [29] M. Graja, M. Jaoua and L. H. Belguith, "Discriminative Framework for Spoken Tunisian Dialect Understanding," Proc. of the 1<sup>st</sup> Int. Conf. on Statistical Language and Speech Processing (SLSP 2013), vol. 7978, pp. 102–110, Tarragona, Spain, July 29–31, 2013.
- [30] W. H. Allehaiby, "Arabizi: An Analysis of the Romanization of the Arabic Script from a Sociolinguistic Perspective," Arab World English Journal, vol. 4, no. 3, 2013.
- [31] T. Buckwalter, "Arabic Transliteration," Available: <http://www.qamus.org/transliteration.htm>, 2002.
- [32] N. Habash, M. T. Diab and O. Rambow, "Conventional Orthography for Dialectal Arabic," Proc. of the 8<sup>th</sup> Int. Conf. on Lang. Resour. and Evaluation (LREC'12), pp. 711–718, Istanbul, Turkey, May 2012.
- [33] A. Mekki, I. Zribi, M. Ellouze and L. Hadrach Belguith, "Treebank Creation and Parser Generation for Tunisian Social Media Text," Proc. of the 17<sup>th</sup> ACS/IEEE Int. Conf. on Computer Systems and Applications (AICCSA), DOI: 10.1109/AICCSA50499.2020.9316462 Antalya, Turkey, 2020.
- [34] A. Masmoudi and F. Bougares, "Automatic Speech Recognition System for Tunisian Dialect," Language Resources and Evaluation, vol. 52, no. 1, pp. 249–267, 2017.
- [35] R. Boujelbane, Traitements Linguistiques Pour la Reconnaissance Automatique de la Parole Appliquée à la Langue Arabe: de L'arabe Standard vers L'arabe Dialectal, Thèse de doctorat, Faculté des Sciences Économiques et de Gestion de Sfax, 2016.
- [36] A. Mekki, I. Zribi, M. E. Khemakhem and L. H. Belguith, "Syntactic Analysis of the Tunisian Arabic," Proc. of the Int. Workshop on Language Processing and Knowledge Management, September 2017.
- [37] A. Al-Thubaity, M. Khan, M. Al-Mazrua and M. Al-Mousa, "New Language Resources for Arabic: Corpus Containing More Than Two Million Words and a Corpus Processing Tool," Proc. of the Int. Conf. on Asian Language Processing, pp. 67–70, Urumqi, China, 2013.
- [38] J. Guo, H. He, T. He et al., "Gluoncv and Gluonnlp: Deep Learning in Computer Vision and Natural Language Processing," J. of Machine Learning Research, vol. 21, no. 23, pp. 1–7, 2020.
- [39] J. Demšar, "Statistical Comparisons of Classifiers over Multiple Datasets," J. of Machine Learning Research, vol. 7, pp. 1–30, 2006.
- [40] A. Mekki, I. Zribi, M. E. Khemakhem and L. H. Belguith, "Sentence Boundary Detection of Various Forms of Tunisian Arabic," Language Resources and Evaluation, vol. 56, pp. 357–385, 2022.
- [41] R. Boujelbane, M. Mallek, M. Ellouze and L. H. Belguith, "Fine-grained POS Tagging of Spoken Tunisian Dialect Corpora," Proc. of the Int. Conf. on Applications of Natural Language to Data Bases/Information Systems (NLDB 2014), vol. 8455, pp. 59–62, 2014.
- [42] I. Zribi, I. Kammoun, M. Ellouze, L. H. Belguith and P. Blache, "Sentence Boundary Detection for Transcribed Tunisian Arabic," Proc. of the 12<sup>th</sup> Workshop on Natural Language Processing (KONVENS 2016), pp. 323–331, Bochum, Germany, September 2016.

### ملخص البحث:

يعدّ الخطأ الإملائي مسألةً مُفكّكة في مجال معالجة اللّغات الطبيعيّة، خصوصاً عند التّعامل مع التّعليقات والنّصوص الخام المأخوذة من وسائل التّواصل الاجتماعيّ. وذلك بسبب استخدام الصّيغ الصّرفيّة الاصطلاحية وعدم الالتزام بالقواعد.

نقترح في هذا البحث نظاماً أوتوماتيكياً لتطبيع النّصوص المكتوبة باللّهجة العاميّة باتّباع طريقة لتصحيح الأخطاء في العربيّة التونسيّة. والجدير بالذّكر أنّ النّظام المعروف باسم (COTA1) لتصحيح الأخطاء الإملائيّة ليس قادراً على التّعامل مع جميع الصّيغ المعروفة للعربيّة التونسيّة. لذا نقترح توسيع قواعد ذلك النّظام ومعالجه لمعالجة الخصوصيات التي تتميّز بها نصوص وسائل التّواصل الاجتماعيّ. وبعبارة أخرى، فإنّ نظام (COTA1) يزود المستخدم بإمكانيات متعدّدة للتّصحيح. من هُنا، فإنّ النّسخة المقترحة في هذا البحث (COTA2) مزوّدة بنظام أوتوماتيكيّ يعمل على إرشاد المستخدم إلى التّصحيح المناسب. ويشير تقييم النّظام المقترح إلى أنّه يعمل على تقليل الأخطاء المتعلّقة بالترجمة بنسبة 95.72%.

## JJCIT Annual List of Reviewers (2022)

Name, Affiliation, Country

Myunghwan Park, *Korea Air Force Academy*,  
Korea

Wen Xu, *Georgia Institute of Technology*,  
USA

Inigo Aldalur, *Mondragon Uni.*,  
Spain

Hieu Tran, *Volgograd State Tech. Uni.*,  
Russia

Le-Minh Nguyen, *JAIST*,  
Japan

Luong Thai Le, *Univ. of Transport and Comm.*,  
Vietnam

Omar Hussain Alhazmi, *Taibah Uni.*,  
KSA

Ako Muhamad Abdullah, *Uni. of Sulaimani*,  
Iraq

M. Rana, *Charles Sturt Uni.*,  
Australia

Luís Alexandre Lopes, *Politécnico de Leiria*,  
Portugal

Hamidreza Bolhasani, *Islamic Azad Uni.*,  
Iran

Omran AlShamma, *Uni. of Inf. Tech. and Comm.*,  
Iraq

Amjad Jaleel Humaidi, *Uni. of Technology*,  
Iraq

Felipe André Zeiser, *UNISINOS*,  
Brazil

Muthana AlAmidie, *Uni. of Missouri*,  
USA

Ahmed AbidAwn Hussain, *Uni. of Missouri*,  
USA

Yannis Haralambous, *IMT Atlantique*,  
France

Izzat Alsmadi, *Texas A&M Uni.*,  
USA

Fawaz S. Al-Anzi, *Kuwait Uni.*,  
Kuwait

Noureddine En Nahnahi, *USMBA*,  
Morocco

Wael Al Etaiwi, *PSUT*,  
Jordan

Ahmed Oussous, *Ibn Tofail Uni.*,  
Morocco

Ashraf Elnagar, *Uni. of Sharjah*,  
UAE

Jerry Chun-Wei Lin, *HVL*,  
Norway

Athanasios Fevgas, *Uni. of Thessaly*,  
Greece

Furqan Rustam, *KFUEIT*,  
Pakistan

Vinh Truong Hoang, *HCMC Open Uni.*,  
Vietnam

Asmma Abbas, *Assiut Uni.*,  
Egypt

Thongchai Surinwarangkoon, *SSRU*,  
Thailand

Wubing Wang, *Ohio State Uni.*,  
USA

Andrew Ferraiuolo, *Cornell Uni.*,  
USA

John Franco, *Uni. of Cincinnati*,  
USA

Paul A. Wortman, *UCONN*,  
USA

Wei Hu, *NWPU*,  
China

Norsang Lama, *Missouri S&T*,  
USA

Abdessamad Ben Hamza, *Concordia Uni.*,  
Canada

CristianDragos Obreja, *UGAL*,  
Romania

Simona Moldovanu, *UGAL*,  
Romania

Geneveffa Tortora, *UNISA*, Italy

Elmar Nöth, *FAU*,  
Germany

Laetitia Jeancolas, *Paris Brain Institute*,  
France

Said Bahassine, *Chouaib Doukkali Uni.*,  
Morocco

Rached N. Zantout, *Rafik Hariri Uni.*,  
Lebanon

Venus Samawi, *Isra Uni.*,  
Jordan

Fatima-zahra El-Alami, *New York Uni.*,  
USA

Floriano De Rango, *Uni. of Calabria*,  
Italy

Stefano Milani, *Sapienza Uni. of Rome*,  
Italy

Mauro Tropea, *UniCal*,  
Italy

Putra Wanda, *Harbin Uni. of Sci. and Tech.*,  
China

Namhun Koo, *SKKU*,  
S. Korea

Ruben De Smet, *ETROVUB*,  
Belgium

Shahab S. Band, *YUNTECH*,  
Taiwan

N. R. Gladiss, *Jeppiaar Institute of Tech.*,  
India

Shih-Ming Wang, *CSIE-CSU*,  
Taiwan

Hung Phuoc Truong, *Sejong Uni.*,  
S. Korea

Yanbang Zhang, *Xianyang Normal Uni.*,  
China

T.B.A. de Carvalho, *UFAP*,  
Brazil

Zhi-Gang Jia, *JSNU*,  
China

Jing Wang, *Southeast Univ.*,  
China

Quanxue Gao, *Xidian Uni.*,  
China

Rajesh Khatri, *Shri G. S. Institute of Tech. and Sci.*,  
India

Sohiful Anuar Zainol, *UNIMAP*,  
Malaysia

Abolfazl Bijari, *Uni. of Birjand*,  
Iran

KunPeng Shang, *Guangdong Uni. of Tech.*,  
China

Omar Hayat, *NUML*,  
Pakistan

Razali Ngah, *UTM*,  
Malaysia

Petros S. Bithas, *Universidad de Málaga*,  
Spain

Turgay İBRİKÇİ, *Adana Alparslan Turkes Sci. and Tech. Univ.*,  
Turkey

Mahdi Jemmali, *Univ. of Monastir*,  
Tunisia

Mariusz Zytniewsk, *Univ. of Economics*,  
Poland

Elaheh Yadegaridehkordi, *National Univ. of Malaysia*,  
Malaysia

Diaa Salama Abd, *Benha Uni.*,  
Egypt

Mohammed A. Al-Ganess, *WHU*,  
China

Abdalhossein Rezaei, *Uni. of Sci. and Culture*,  
Iran

Essam H. Houssein, *Minia Uni.*,  
Egypt

Ahmed A. Ewees, *Damietta Uni.*,  
Egypt

Zhe Xia, *WHUT*,  
China

Hamdi Eltaief, *Uni. of Sousse*,  
Tunisia

Zhenyong Zhang, *ZJU*, China

## JJCIT Annual List of Reviewers (2022)

Name, Affiliation, Country

Selcuk Baktir, *BAU*,  
Turkey

Thippa Reddy Gadekallu, *VIT*,  
India

Lucas Ribeiro, *OULU*,  
Finland

Muhammad Irshad Zahoor, *HRBEU*,  
China

Hirokazu Madokoro, *Iwate-PU*,  
Japan

Gabriele Goletto, *POLITO*,  
Italy

Louahdi Khoudour, *CEREMA*,  
France

Abdul Rehman, *COMSATS Uni.*,  
Pakistan

Taye Girma Debelee, *Addis Ababa Sci. and Tech. University*,  
Ethiopia

Oscar M. Granados, *UTADEO*,  
Colombia

Cuneyt G. Akcora, *UMANITOBA*,  
Canada

Umar Islambekov, *BGSU*,  
USA

Gleudson Sobreira Leite, *Uni. de Fortaleza*,  
Brazil

Chinmay Chakraborty, *Birla Institute of Technology*,  
India

Sina F. Ardabili, *J. Selye Uni.*,  
Slovakia

Natalia V. Yakovenko, *Voronezh State Uni.*,  
Russia

Saqib Ali, *GZHU*,  
China

Qurat-ul-ain Mastoi, *Uni. of Malaya*,  
Malaysia

Ziti Fariha binti Mohd Apani, *TATI Uni. College*,  
Malaysia

Shihao Song, *Drexel Uni.*,  
USA

George A. Kyriacou, *Democritus Uni. of Thrace*,  
Greece

Kunal Srivastava, *Sri Venkateswara College*,  
India

Slawomir Koziel, *Reykjavik Uni.*,  
Iceland

Sharul Kamal Abd. Rahim, *UTM*,  
Malaysia

Yulong Zhao, *UCALGARY*,  
Canada

Raimi DEWAN, *UTM*,  
Malaysia

Saeed Roshani, *Islamic Azad Uni.*,  
Iran

Jie Cui, *NJUST*,  
China

Wen-Cheng Lai, *YUNTECH*,  
Taiwan

Savas Okyay, *Eskisehir Osmangazi Uni.*,  
Turkey

Md Mujibur Rahman, *BUET*,  
Bangladesh

Sercan Aygun, *ITU*,  
Turkey

Sajad Ahmadiana, *ZNU*,  
Iran

Chien-Yao Wang, *SINICA*,  
Taiwan

Chunlei Huo, *NLPR-IA*,  
China

Rui Xiong, *Beihang Uni.*,  
China

Xiongwei Wu, *SMU*,  
Singapore

Yanyan Yang, *Uni. of Portsmouth*,  
UK

Irina Illina, *LORIAINRIA*,  
France

Giuseppe Jurman, *Fondazione Bruno Kessler*, Italy

Daochen Zha, *Texas A&M Uni.*,  
USA

Zhishuo Zhang, *UESTC*,  
China

Soo Fun Tan, *Uni. Malaysia Sabah*,  
Malaysia

Hai Liang, *GUET*,  
China

Nhan Tam Dang, *HCMIU*,  
Vietnam

Xin Liao, *HNU*,  
China

Serdar Solak, *Kocaeli Uni.*,  
Turkey

Yavar Khedmati, *UMA*,  
Iran

Peyman Ayubi, *Islamic Azad Uni.*,  
Iran

Manos Roumeliotis, *UOM*,  
Greece

Stavros Souravlas, *UOM*,  
Greece

Tayyaba Anees, *Uni. of Manag. and Tech.*,  
Pakistan

Erum Mehmood, *Uni. of Manag. and Tech.*,  
Pakistan

Khalid Mohamed Nahar, *Yarmouk Uni.*,  
Jordan

Esra Kaygisiz, *Giresun Uni.*,  
Turkey

Aytuğ Onan, *Celal Bayar Uni.*,  
Turkey

Nhan Cach Dang, *HCMUTRANS*,  
Vietnam

Cut Fiarni, *ITHB*,  
Indonesia

Ali Kartit, *Chouaib Doukkali Uni.*,  
Morocco

Xingyuan Wang, *DLUT*,  
China

Mbarek Marwan, *Chouaib Doukkali Uni.*,  
Morocco

S. P. Raja, *VelTech*,  
India

Stefano Sbalchiero, *UNIPD*,  
Italy

Alexey Rotmistrov, *HSE Uni.*,  
Russia

Sung-Hwan Kim, *PUSAN*,  
S. Korea

Abdur Rasool, *SIAT*,  
China

Christian U. Idemudia, *Lanzhou Uni. of Tech.*,  
China

Chee Meng Benjamin, *Sewon Intelligence*,  
S. Korea

Ramachandran Manikandan, *SASTRA Deemed Uni.*,  
India

Hewei Wang, *UCD Connect*,  
Ireland

Guodong Du, *Xiamen Uni.*,  
China

Ahmad Hammoudeh, *Uni. of Mons*,  
Belgium

Mirko Messori, *ATENEOPV*,  
Italy

Fahed Jubair, *Uni. of Jordan*,  
Jordan

Vilas Baburao Khedekar, *MIT ADT*,  
India

Ewert Bengtsson, *Uppsala Uni.*,  
Sweden

Farhad S. Gharehchopogh, *Islamic Azad Uni.*,  
Iran

Ahmed G. Gad, *Kafrelsheikh Uni.*,  
Egypt

Yu Li, *Henan Uni.*,  
China

Shaymah A. Yasear, *Mustaqbal-College*, Iraq

## JJCIT Annual List of Reviewers (2022)

Name, *Affiliation*, *Country*

- Liu Yongxin, *EmbryRiddle Aeronautical Uni.*,  
USA
- Yanbo ZhU, *Aviation Data Comm. Corporation*,  
China
- Xiaokang Qiu, *Purdue Uni.*,  
USA
- Laith Abualigah, *Al-Ahliyya Amman Uni.*,  
Jordan
- Saeid Barshandeh, *Afagh Higher Edu. Institute*,  
Iran
- Gaurav Dhiman, *Govt. Bikram College of Commerce*,  
India
- Apostolia Karampatea, *AUTH*,  
Greece
- Musa Hussain, *Bahria Uni.*,  
Pakistan
- Mira Bou Saleh, *Antonine Uni.*,  
Lebanon
- Isabelle Huynen, *Uni. Catholique de Louvain*,  
Belgium
- Ruchi Varma, *National Institute of Tech.*,  
India
- Ahmad F. Hidayatullah, *Uni. Islam Indonesia*,  
Indonesia
- Jorge Luis Victria Barbosa, *UNISINOS*,  
Brazil
- Abir Messaoudi, *iCompass*,  
Tunisia
- Siaw-Lang Wong, *Uni. of Malaya*,  
Malaysia
- Kai Hu, *NUIST*, China
- Vahid Rowghanian, *Independent Author*,  
Iran
- Peixian Zhuang *NUIST*,  
China
- Daniela E. Popescu, *Uni. of Oradea*,  
Romania
- Antoanela Naaji, *UVVG*,  
Romania
- Pradeep Kumar Mallick, *KIIT*,  
India
- Bilal Tahir, *KICS*,  
Pakistan
- Soufia Kausar, *Independent Author*,  
Pakistan
- Abdul Majid, *AJKU*,  
Pakistan
- Maaz Amjad, *IPN*,  
Mexico
- Syed A. Ali, *Muhammad Ali Jinnah Uni.*,  
Pakistan
- Nawa Raj Pokhrel, *XULA*,  
USA
- Ramchandra Rimal, *Middle Tennessee State Uni.*,  
USA
- Keshab Raj Dahal, *Truman State Uni.*,  
USA
- Hum Nath Bhandari, *Roger Williams Uni.*,  
USA
- Mohamed Kissi, *UNIVH2C*,  
Morocco
- Rached Nabil Zantout, *Rafik Hariri Uni.*,  
Lebanon
- Qian Li Beihang, *BUAA*,  
China
- Laila Khreisat, *Fairleigh Dickinson Uni.*,  
USA
- Abdelhak Lakhouaja, *Uni. Mohamed First*,  
Morocco
- Hyuk-Chul Kwon, *PUSAN*,  
S. Korea
- Ashraf Suyyagh, *Uni. of Jordan*,  
Jordan
- Gheith A. Abandah, *Uni. of Jordan*,  
Jordan
- Muhammad Syafrudin, *Dongguk Uni.*,  
S. Korea
- Muhammad Adnan Khan, *Gachon Uni.*,  
S. Korea
- Bruno Samways dos, *UTFPR*,  
Brazil
- Marek Piorecký, *FBMI-CVUT*,  
Czech
- Sani Saminu, *UNILORIN*,  
Nigeria
- Sayani Sarkar, *Bellarmino Uni.*,  
USA
- Muhammad Asif Khan, *Qatar University*,  
Qatar
- Seyed Abolfazl Valizadeh, *ETHZ*,  
Switzerland
- Hong Liang, *NWPU*,  
China
- Sonain Jamil, *SJU*,  
S. Korea
- Mohammed B. Abubaker, *PTCDB*,  
Palestine
- Ali Haider Khan, *UMT*,  
Pakistan
- Najlae Idrissi, *USMS*,  
Morocco
- Abdulwahab Alazeb, *Najran Uni.*,  
KSA
- Monika Kosowska, *Bydgoszcz Uni. of Tech.*,  
Poland
- Ichrak Khouli, *Sultan Moulay Slimane Uni.*,  
Morocco
- Huda Kadhim Tayyeh, *UOITC*,  
Iraq
- Aida Mustapha, *UTHM*,  
Malaysia
- Faisal Azam, *CIITWAH*,  
Pakistan
- Mohammed A. Ambusaidi, *UTAS*,  
Oman
- Ebrima Jaw, *Uni. of the Gambia*,  
Gambia
- Fadhl Eryani, *Uni. of Tbingen*,  
Germany
- Rami Doas, *Columbia Uni.*,  
USA
- Salam Khalifa, *New York Uni.*,  
USA
- Maryam Ebrahimi, *ITRC*, Iran
- Jun Yan, *Zhuhai Orbita Aerospace Sci. and Tech.*,  
China
- Qiuying Huang, *Beijing Normal Uni.*,  
China
- Wael Badawy, *Uni. of Hertfordshire*, UK
- ShihChang Hsia, *YUNTECH*,  
Taiwan
- Nasaruddin Nasaruddin, *Uni. Syiah Kuala*,  
Indonesia
- Hoanh Nguyen, *IUH*, Vietnam
- Enas Elgeldawi, *Mansoura Uni.*,  
Egypt
- Soud Larabi Marie-Sainte, *Prince Sultan Uni.*,  
KSA
- Moez Ben HajHmida, *Uni. of Tunis El Manar*,  
Tunisia
- El Habib Nfaoui, *USMBA*,  
Morocco
- Jun-Cheol Jeon, *KUMOH*,  
S. Korea
- Tomás Sureda Riera, *Uni. of Alcalá*,  
Spain
- William Steingartner, *Tech. Uni. of Košice*,  
Slovakia
- Darko Galinec, *Zagreb Uni. for App. Sci.*,  
Croatia
- Nachaat Mohamed, *Uni. Sains Malaysia*,  
Malaysia
- Sarosh Ahmad, *Carleton Uni.*, Canada
- Mohammad Alibakhshkenari, *Uni. of Rome*,  
Italy
- Xiang Long Liu, *Xidian Uni.*,  
China
- Yanal S Faouri, *Uni. of Jordan*, Jordan

المجلة الأردنية للحاسوب وتكنولوجيا المعلومات (JJCIT) مجلة علمية عالمية متخصصة محكمة تنشر الأوراق البحثية الأصيلة عالية المستوى في جميع الجوانب والتقنيات المتعلقة بمجالات تكنولوجيا وهندسة الحاسوب والاتصالات وتكنولوجيا المعلومات. تحتضن وتنشر جامعة الأميرة سمية للتكنولوجيا (PSUT) المجلة الأردنية للحاسوب وتكنولوجيا المعلومات، وهي تصدر بدعم من صندوق دعم البحث العلمي في الأردن. وللباحثين الحق في قراءة كامل نصوص الأوراق البحثية المنشورة في المجلة وطباعتها وتوزيعها والبحث عنها وتنزيلها وتصويرها والوصول إليها. وتسمح المجلة بالنسخ من الأوراق المنشورة، لكن مع الإشارة إلى المصدر.

### الأهداف والمجال

تهدف المجلة الأردنية للحاسوب وتكنولوجيا المعلومات (JJCIT) إلى نشر آخر التطورات في شكل أوراق بحثية أصيلة وبحوث مراجعة في جميع المجالات المتعلقة بالاتصالات وهندسة الحاسوب وتكنولوجيا المعلومات وجعلها متاحة للباحثين في شتى أرجاء العالم. وتركز المجلة على موضوعات تشمل على سبيل المثال لا الحصر: هندسة الحاسوب وشبكات الاتصالات وعلوم الحاسوب ونظم المعلومات وتكنولوجيا المعلومات وتطبيقاتها.

### الفهرسة

المجلة الأردنية للحاسوب وتكنولوجيا المعلومات مفهرسة في كل من:



فريق دعم هيئة التحرير

ادخال البيانات وسكترير هيئة التحرير

المحرر اللغوي

إياد الكوز

حيدر المومني

جميع الأوراق البحثية في هذا العدد متاحة للوصول المفتوح، وموزعة تحت أحكام وشروط ترخيص



[Creative Commons Attribution] (<http://creativecommons.org/licenses/by/4.0/>)

### عنوان المجلة

الموقع الإلكتروني: [www.jjcit.org](http://www.jjcit.org)

البريد الإلكتروني: [jjcit@psut.edu.jo](mailto:jjcit@psut.edu.jo)

العنوان: جامعة الأميرة سمية للتكنولوجيا، شارع خليل الساكت، الجبية، عمان، الأردن.

صندوق بريد: 1438 عمان 11941 الأردن

هاتف: +962-6-5359949

فاكس: +962-6-7295534