**International Journal of Electrical and Computer Engineering (IJECE)**
Vol. 13, No. 6, December 2023, pp. 6940~6951
ISSN: 2088-8708, DOI: 10.11591/ijece.v13i6.pp6940-6951 

# Design and development of a fuzzy explainable expert system for a diagnostic robot of COVID-19

**Omar El Beggar, Mohammed Ramdani, Mohamed Kissi**
Laboratory of Intelligence Machine, Faculty of Sciences and Techniques of Mohammedia, Hassan II University of Casablanca, Casablanca, Morocco

**Article Info**

***Article history:***

Received Dec 29, 2022
Revised Apr 19, 2023
Accepted Apr 24, 2023

***Keywords:***

Coronavirus disease-2019
Diagnostic robot
Expert system
Explainable artificial intelligence
Fuzzy logic

**ABSTRACT**

Expert systems have been widely used in medicine to diagnose different diseases. However, these rule-based systems only explain why and how their outcomes are reached. The rules leading to those outcomes are also expressed in a machine language and confronted with the familiar problems of coverage and specificity. This fact prevents procuring expert systems with fully human-understandable explanations. Furthermore, early diagnosis involves a high degree of uncertainty and vagueness which constitutes another challenge to overcome in this study. This paper aims to design and develop a fuzzy explainable expert system for coronavirus disease-2019 (COVID-19) diagnosis that could be incorporated into medical robots. The proposed medical robotic application deduces the likelihood level of contracting COVID-19 from the entered symptoms, the personal information, and the patient's activities. The proposal integrates fuzzy logic to deal with uncertainty and vagueness in diagnosis. Besides, it adopts a hybrid explainable artificial intelligence (XAI) technique to provide different explanation forms. In particular, the textual explanations are generated as rules expressed in a natural language while avoiding coverage and specificity problems. Therefore, the proposal could help overwhelmed hospitals during the epidemic propagation and avoid contamination using a solution with a high level of explicability.



***Corresponding Author:***

Omar El Beggar
Laboratory of Intelligence Machine, Faculty of Sciences and Techniques of Mohammedia, Hassan II University of Casablanca
BP 146 Mohammedia 28806, Morocco
Email: omar.elbeggar@fstm.ac.ma

## 1. INTRODUCTION

Medical diagnosis has drawn great interest from researchers and industrialists from different specialties. It consists of determining the disease from which a patient suffers to prescribe the appropriate treatment. In other words, it is based on finding out the causes and symptoms of the disease [1]. Among the dangerous viral diseases that have appeared in the last few years, coronavirus disease-2019 (COVID-19) caused by the SARS-CoV-2 virus [2]. Due to its quick and large transmission, this virus has impacted harmfully public health and the international economy since 2020 [3]. It has affected over 468 million people worldwide and has caused over 6 million deaths by the end of March 2022 [4]. For the same period, in Morocco only, over 1 million confirmed cases and over 16 thousand deaths have been reported by the Moroccan Ministry of Health [5]. According to the World Health Organization (WHO) [6], the symptoms of COVID-19 might vary from one person to another. However, fever, cough, tiredness, and loss of taste and smell are the most common. Other symptoms could be appeared according to the cases' severities. Moreover, the same source claimed that persons aged over 60 years or suffering from chronic diseases are more at risk of contracting COVID-19. They might also develop severe symptoms. Notwithstanding COVID-19 asymptomatic cases, a safe and early diagnosis of persons who show some signs is of great importance. Indeed, making the diagnosis of COVID-19 a safe process is a relevant challenge that should be taken up.

Using medical robots would be among the most efficient ways of that. Nowadays, many robots are used in the healthcare sector. Meanwhile, their first emergence was in the 1980s with robotic surgeries [7]. Over the years, artificial intelligence (AI) has expanded robot capabilities to cover many fields in healthcare like diagnosis, therapy, rehabilitation, and clinic reception. Despite performing a safe process, a diagnostic robot for COVID-19 presents other benefits, like providing impartial medical diagnosis, supporting clinicians and alleviating their workload, and saving time and money. The final objective of our studies is to build this robot and employ it in Moroccan hospitals.

The hardware architecture of the robot is composed of three main blocs (intelligence bloc, navigation bloc, and control bloc) as described in Figure 1. The intelligence block holds the expert system responsible for COVID-19 diagnosis, sound and image processing, and data transmission to the cloud. The navigation bloc ensures the robot's autonomous navigation thanks to the obstacle sensors and the radar signals that map the travel environment. The control bloc is responsible for servomotors' control and energy management supplied by the batteries. Given the key role of the expert system in the intelligence bloc, this study will be focused mainly on designing and developing this system. A diagnostic expert system is a computer-based system that could assist medical specialists or even imitate their behaviors of diagnosing [8]. The proposed expert system infers the likelihood of contracting COVID-19 from a set of input data: symptoms severities, age, possibility of contacting positive cases in the last 14 days, last time immunity has been acquired either from direct infection by COVID-19 or from a vaccine and the maximum risk level deduced from the recently visited places. Besides, diagnosis involves a high degree of uncertainty and vagueness. It is a medical field whereby information is mainly linguistic, particularly during the first consultation. Even worse, there is often confusion between different diseases while diagnosing [9].

Since fuzzy logic is an effective tool that deals with uncertain and vague data [8], it is ideal to use in the design and development of such systems. Furthermore, a fuzzy expert system is not entirely transparent for end-users, and the reasoning behind its outcome is still incomprehensible to humans without an explanation facility [10]. Formerly, the most explanation facilities of the first expert systems are focused on the rule trace-based strategy that answers why and how the system decision is reached [11]. However, expressing the rules that led to a conclusion in a machine language did not provide fully human-understandable explanations. Thereby, an auto-generation of those rules in natural language (NL) is required. This will improve rule expressiveness and enable our future medical robots properly communicate explanations to humans. Nowadays, much attention is paid to explain ability in AI, especially with the increasing use of opaque machine learning algorithms like deep learning [12]. Despite expert systems are not considered opaque, new efforts should be deployed on explainable artificial intelligence (XAI) and its application on those systems to improve their level of transparency and explicability.

In this paper, we attempt to enrich the system explanation facility with a hybrid XAI technique that provides different explanation forms and allows getting a so-called fuzzy explainable expert system (FEES). The rest of the paper is structured as follows. Section 2 discusses the related works. Section 3 presents the proposed method to design and develop the system-to-be and how it addresses the disease diagnosis. Section 4 describes the proposal results and provides some discussions. Finally, conclusions and perspectives are listed in section 5.

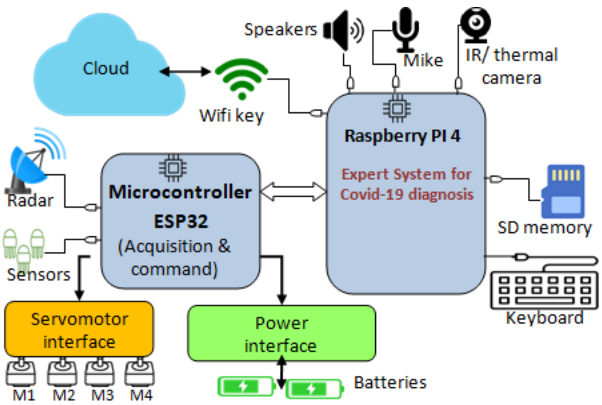



Figure 1. The hardware architecture for the diagnostic robot of COVID-19

## 2. RELATED WORKS

Expert systems have been widely used in medicine to diagnose different diseases as stated in [8]. COVID-19 is no different because many works have been performed since the epidemic's emergence. In this section, we will focus only on research that has proposed an expert system to diagnose this disease.

Nema *et al.* [13] developed a knowledge-based system for COVID-19 diagnosis in Iraq using internet of things (IoT) technologies. IoT is used to determine contact and location history. However, the authors did not treat uncertainty and vagueness in the diagnosis process. Instead, Shatnawi *et al.* [14] proposed a fuzzy inference system based merely on symptoms to diagnose COVID-19. Unfortunately, this system has developed using MATLAB and could not be employed directly by healthcare specialists.

Al Hakim *et al.* [2] proposed an android-based expert system to make self-diagnoses of this disease in Indonesia. Besides, their approach applied certainty factors (CFs) to facts and rules to overcome the uncertainty problem. However, these CFs are still crisp values and should be calculated using mathematical formulas (the authors used Microsoft Excel for that).

Raihan *et al.* [15] a framework consisting of a web and mobile application has been developed to allow COVID-19 prognostication. Rules and features have been deduced using classification algorithms. The objective of this computer-aided framework is COVID-19 risk prediction thanks to the input symptoms and the clinical measures, like respiratory and pulse rates. However, the lack of such measurable data might lead to an error in the final prediction. Furthermore, using opaque or black-box models to build the expert system will not allow getting fully interpretable system conclusions.

Banjar *et al.* [16] proposed a prototype for the diagnosing and monitoring of COVID-19 cases in Saudia Arabia. This prototype comprises an expert system responsible for recommendations providing, such as risk classification, treatment plans, and the required laboratory tests and chest images. The knowledge base is adaptively updated by learning new medical guidelines. The authors chose the following system inputs: symptoms, hospitalization history, epidemiological information, and contact exposure. They also consecrated an interface called "knowledge inbox" to present system explanations, although they are not well-detailed.

Overall, most of the studied expert systems used the COVID-19 symptoms revealed by WHO, personal information and location history as common inputs. Nonetheless, they have ignored the pertinent factor of immunity that has to be included. Indeed, a long period of acquiring immunity either from infection or from vaccine increases the possibility of contracting COVID-19, although symptoms severities are slight. To the most of our knowledge, almost studied systems offer no explanation facility compared to our approach. The main reason claimed by Dhaliwal and Tung [17] is that the development of explanation facilities has received lesser attention than the development of the expert systems themselves.

In particular, our approach and that of Banjar *et al.* [16] give explanations of the system decisions. However, the proposed explanations in [16] are limited to expressing rules in a pseudo-NL, and plots of fuzzy sets corresponding to user inputs. The paper also lacks details about the fuzzification process, including the method adopted to generate fuzzy rules and their expressiveness in NL.

Another drawback of these works is related to the number and length of the rules that might be presented as explanations of the system outcomes. Despite the coverage and specificity problems decreasing significantly the system explicability, any of the studied expert systems have been interested in these problems. The main similarities and differences between the studied approaches and our proposal are provided in Table 1.

Table 1. Comparison between the studied expert systems for COVID-19 diagnosis

| Approach | Criteria | | | | | |
|---|---|---|---|---|---|---|
| | Diagnosis factors | Uncertainty support | Epidemic data update | Dynamic extensibility | Decision explicability | Tool |
| Nema *et al.* [13] | Symptoms contact history location history | No | No | No | No | python +CLIPS |
| Shatnawi *et al.* [14] | Symptoms | Fuzzy logic | No | No | No | MATLAB toolbox |
| Al Hakim *et al.* [2] | Symptoms, contact history, location history age | Certainty factors | No | No | No | Mobile App |
| Raihan *et al.* [15] | Symptoms, measures | Triangular fuzzy numbers | No | No | No | Mobile and web apps |
| Banjar *et al.* [16] | Symptoms hospitalization history, epidemiological info, contact exposure | Fuzzy logic | No | Yes | Clinical rules + fuzzy sets plots | Web App |
| Our approach | Symptoms, contact history, location history, age, immunity period (vaccine/infection) | Fuzzy logic | Yes | Yes (new rules, variables) | Yes (hybrid XAI) | Morfees-C19 |

## 3. PROPOSED METHOD

The proposal aims to develop a FEES for early diagnosis of COVID-19. It contains three main components: a knowledge base, an inference engine, and user interfaces. The knowledge base includes facts and rules. Rules are represented in form of IF-THEN statements and either defined by experts or collected from reliable sources. The inference engine is the rule interpreter that provides inferences by selecting the rules from the knowledge base that should be executed with regard to their priorities or weights. The user interface is essential for users to interact with the system and could be extended with an explanation facility to supply user explanations [10]. The FEES starts with the fuzzification of the crisp input data. Next, it evaluates the fuzzy rules using the inference engine and aggregates their outputs. Finally, it defuzzifies the outcome and provides explanations.

Our methodology to design and implement this system contains three main phases described as: i) definition of fuzzy variables: creating membership functions associated with input and output variables, ii) definition of fuzzy rules: expressing fuzzy rules gathered from discussions with experts and external sources, and iii) generation of explanations using a hybrid XAI technique: providing different forms of explanations related to the system conclusions.

Throughout the rest of the paper, the proposed FEES will be referred to as the Moroccan fuzzy explainable expert system for diagnosing COVID-19 (MORFEES-C19). The main functionality of MORFEES-C19 is the diagnosis of COVID-19. However, we have tried to produce our solution with additional features that make it more attractive and easier to use. In fact, MORFEES-C19 is multilingual and supports English, French and Arabic languages. The system also supplies user explanations and helps patients to get quick medical assistance or notify the medical authorities by email. Moreover, actualizing epidemiological data through communication with external sources allows a reliable diagnosis. As well, the MORFEES-C19 could be extended with additional rules making thus the system even more scalable. Overall, the main functionality concerning COVID-19 diagnosis is performed by MORFEES-C19 according to the following three steps:

Step 1: in a chosen language, the patient enters his temperature and birth date. The system deduces his age and records his visited places or localizations in the last 14 days. A location could concern a living, a working, or a travelling place. The patient can submit his locations as many as he desires during this period. The objective is to check whether he has visited a region with a high risk of contamination. For each location, the system calculates the corresponding risk level from a dedicated table "region_stats", and assigns it to the submitted localization. This table is updated periodically from the external source "COVID19-geomatic.ma" [18]. The website provides a GeoJSON file containing data on the epidemic evolution in Morocco by region. The data herein concern the confirmed cases and the number of deaths recorded daily by region. A region risk *R* is calculated per 100K people for the last 14 days using the (1),

$$R = \frac{100\,000}{Pop} \sum_{j=1}^{14} \left( NConfirmedcases_j + Ndeaths_j \right) \tag{1}$$

where $NConfirmedcases_j$ and $Ndeaths_j$ represent the number of confirmed cases and deaths recorded in a region on a $j^{th}$ day respectively. *Pop* is the region's total population. In this way, the system determines which location visited by the patient presents the maximum risk of contamination.

Step 2: the second step consists of entering the symptoms' severities according to [6], as well as the contact exposure in the last 14 days, and the period separating either the possible infection by COVID-19 or the vaccination against this disease.

Step 3: MORFEES-C19 fuzzy files all entries and fires next the appropriate fuzzy rules to provide the patient's likelihood level of contracting COVID-19 and the different explanations related to this result. The different performed diagnoses are saved in a relational data base. The diagnosis functionality could be synthesized by algorithm 1. Besides, Figure 2 shows the software architecture of MORFEES-C19.

### 3.1. Definition of fuzzy variables

The first phase of the proposed method consists of creating membership functions required to convert the input data into fuzzy sets. In this paper, the triangular (2), Gaussian (3) and trapezoidal (4) functions are used to this purpose:

$$\mu_A(x) = \begin{cases} \frac{x-a}{b-a}, a \leq x \leq b \\ \frac{c-x}{c-b}, b \leq x \leq c \\ 0, x < a or x > c \end{cases} \tag{2}$$

$$\mu_A(x) = exp\left[-\frac{1}{2}\left(\frac{x-c}{\sigma}\right)^2\right] \quad (3)$$

$$\mu_A(x) = \begin{cases} \frac{x-a}{b-a}, & a \le x \le b \\ 1, & b \le x \le c \\ \frac{d-x}{d-c}, & c \le x \le d \\ 0, x < a or x > d \end{cases} \quad (4)$$

The fuzzy variables used in the proposal are symptoms, age, Max_Locs_Risk, Pos_Contact, and Time_ Immunity. The following is a brief explanation of them:

a. Symptoms: In 2021, WHO classified COVID-19 symptoms into three categories [6]: i) main symptoms are listed as fever, dry cough, tiredness, and loss of taste and smell; ii) less common symptoms include diarrhea, headache, sore throat, rashes, muscle pains, and conjunctivitis; and iii) Serious Symptoms comprise confusion, chest pain, and dyspnea. Three levels are defined to describe the severity of those symptoms: slight, moderate, and severe. Figure 3(a) illustrates the fuzzy variable fever according to the corporal temperature. The severities of the other symptoms are represented in a range of 0-3 using the Gaussian function, such as headache shown in Figure 3(b).
b. Age: it could be described by five linguistic terms: child, young, middle-aged, old, and senile as described in Figure 3(c).
c. Max_Locs_Risk: indicates the maximum risk deduced from the visited places in the last 14 days using (1). The following terms: very low, low, medium, high, and very high have been defined for scoring and ranking this variable, as shown in Figure 3(d).
d. Pos_Contact: indicates the possibility of contact with confirmed cases in the last 14 days. Like Figure 3(b), this variable is ranked using the Gaussian function into three linguistic terms: unlikely, likely, and very likely.
e. Time_Immunity: indicates the last time a person developed immunity to COVID-19 from either a direct infection or a vaccination. Four linguistic terms are used to represent it Figure 3(e): very small, small, average, and long.

Moreover, Cov19_Likelihood is the fuzzy output variable that will be derived from the fuzzy input variables previously described. It represents a likelihood level of contracting COVID-19 expressed as a percentage and ranked into seven linguistic terms as seen in Figure 3(f): extremely low, very low, low, medium, high, very high, and extremely high. Table 2 describes the all-fuzzy variables of the proposal.

Algorithm 1. Diagnosis functionality of MORFEES-C19

```
Input: bdate, syms, locs, temp, posc, timm /*birth date, symptoms, visited
        localizations, temperature, possible contact and time immunity*/
Output: likelihood   //a percentage of possible contracting of COVID-19
Data: maxRisk, age
Procedure submitInputs(bdate,syms,locs,temp,posc,timm)
     //check either input is out of range or not and get data (age and MaxRisk)
     age=calculateAge(bdate)
     foreach L in locsdo
        // get risk form Table Region stats
        L.risk=findRiskfromDB(L.name)
           // locs implements a Comparator by risk
     maxRisk=Collections.max(locs)
Procedure fuzzifyInputs(age,syms,posc,maxRisk,timm)
     Fuzzy_age=membership_age(age)
     fuzzy_fever=membership_temp(temp)
     fuzzy_MaxRisk=membership_risk(MaxRisk)
     fuzzy_posc=membership_contact(posc)
     fuzzy_timm=membership_immunity(timm)
     fori=1→syms.sizedo
        fuzzy_Si.sev=membership_severity(Si.sev)
Function calculatelikelihood():real
          // using a Mamdani model
     outputsrules=fireRules(fuzzy_ age, fuzzy_ fever, fuzzy_ MaxRisk,
             fuzzy_ posc,fuzzy_timm,fuzzy_S1.sev,...,fuzzy_Sn.sev)
     fuzzy_likelihood=aggregate(Outputsrules)
     likelihood=defuzzify(fuzzy_ likelihood)
return likelihood
Procedure generateExplanations()
     convertToNL(Rule)
     displayVisulisations()
```

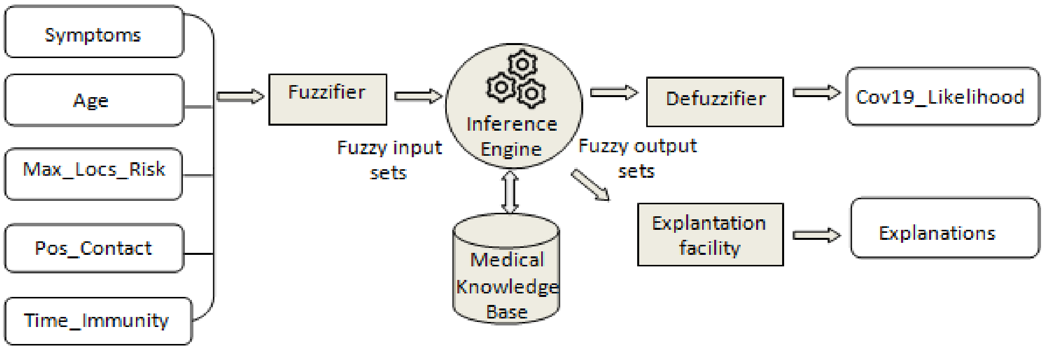



Figure 2. The software architecture of MORFEES-C19 and its different components

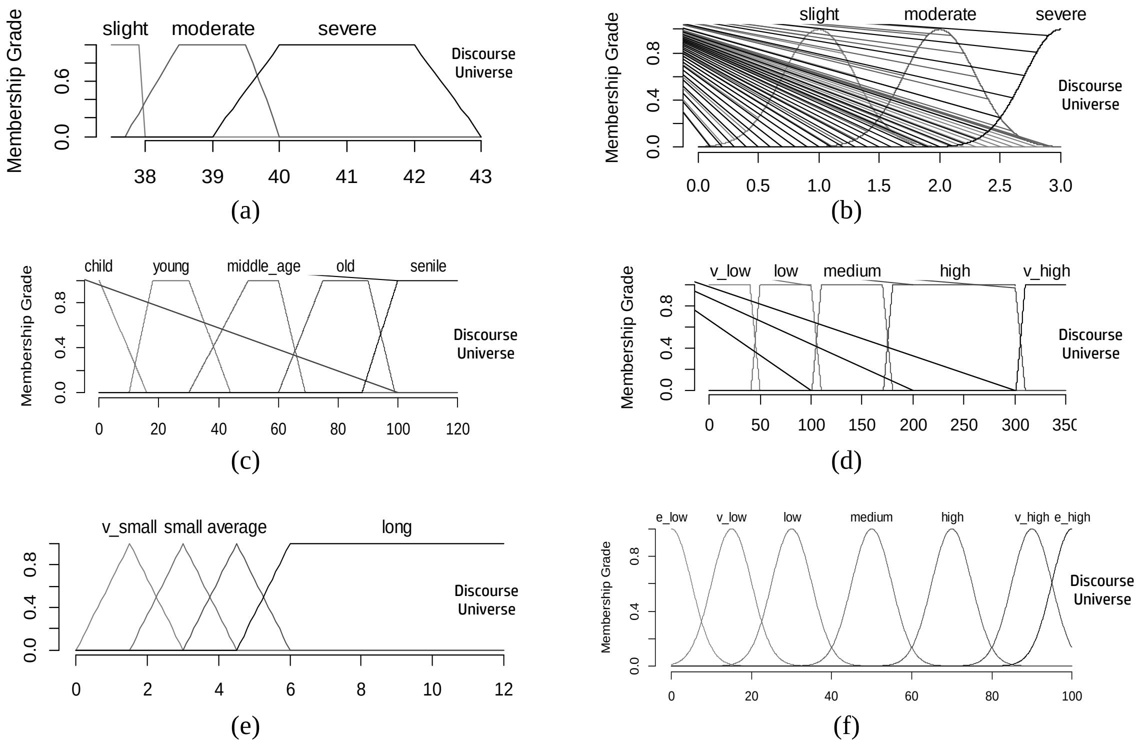



Figure 3. The used membership functions of the fuzzy variables:(a) fever, (b) headache, (c) age, (d) max risk, (e) time immunity, and (f) likelihood of contracting COVID-19

Table 2. Fuzzy variables of the proposal

| Linguistic variable | Linguistic terms | Universe of discourse |
|---|---|---|
| Fever | slight, moderate, and severe | From 36 to 43 ℃ |
| The rest of symptoms | slight, moderate, and severe | from 0 to3 |
| Age | child, young, middle-aged, old, and senile | greater than 0 years old |
| Max locs risk | very low, low, medium, high, and very high | greater than 1 case per 100K |
| Pos Contact | unlikely, likely, and very likely | from 0 to 3 |
| Time Immunity | Very small, small, average, and long | from 0 to 12 months |
| Cov19_Likelihood | extremely low, very low, low, medium, high, very high, and extremely high | from 0% to 100% |

### 3.2. Definition of fuzzy rules

The proposed system is a Mamdani system [19] with a forward chaining inference engine [20]. This kind of system is well-fitting to human inputs and more interpretable than other ones [21]. In a Mamdani model, the output of each fuzzy rule is also a fuzzy set. The form of a fuzzy rule according to this model:

$$IF\ x\ is\ A\ and\ y\ is\ B\ THEN\ z\ is\ C \tag{5}$$

where *A* and *B* are fuzzy sets of the rule antecedents, while *C* is a fuzzy set of the rule consequent. In Mamdani model, the conjunction (AND) between rules' antecedents and implication are evaluated with the t-norm operator. Meanwhile, t-conorm is used for disjunction (OR) and aggregation [19].

The second phase of our proposal consists of defining the fuzzy rules according to the Mamdani model. Based on the fuzzy variables already defined in the previous phase, the fuzzy rules are designed and built with the assistance of a domain expert (an internist) and expressed using fuzzy control language (FCL) [22]. FCL supports the IEC 61131-7 specifications, and allows getting portable fuzzy rules. Our choice is motivated by the language independency of systems suppliers and the ability to exchange programs control between different platforms [23]. We can also assign a weight and support degrees to a fuzzy rule, representing its importance and support respectively. Both degrees lie between 0 and 1. The support degree is deduced from the evaluation of all memberships present in the rule antecedents and the connection methods used between them (*MIN*, *MAX*, and *PROD*). Some machine learning algorithms could be used to adjust these degrees and optimize the FEES accuracy. A sample of the defined fuzzy rules in FCL is shown in Figure 4.

**RULE 1**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight THEN Cov19_ Likelihood IS extremely low WITH 0.1;
**RULE 2**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND (Age IS old OR Age IS senile) THEN Cov19_ Likelihood IS very low WITH 0.5;
**RULE 3**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND Time_ Immunity IS long THEN Cov19_ Likelihood IS very low WITH 0.5;
**RULE 4**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND (Pos_Contact IS likely OR Pos_ Contact IS very likely) THEN Cov19_ Likelihood IS low;
**RULE 5**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND (Max_ Locs_ Risk IS high OR Max_ Locs_ Risk IS very high) THEN Cov19_ Likelihood IS low;
**RULE 6**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND (Diarrhoea IS NOT slight OR Conjunctivitis IS NOT slight OR Headache IS NOT slight OR Muscle pains IS NOT slight OR Sore throat IS NOT slight OR Rashes IS NOT slight) THEN Cov19_ Likelihood IS very low WITH 0.5;
**RULE 7**: IF Fever IS slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight AND (Confusion IS NOT slight OR Chest_ pain IS NOT slight OR Dyspnea IS NOT slight) THEN Cov19_ Likelihood IS low;
**RULE 8**: IF Fever IS NOT slight AND Tiredness IS slight AND Dry_cough IS slight AND Loss_taste_smell IS slight THEN Cov19_ Likelihood IS very low WITH 0.1;
**RULE 9**: IF Fever IS NOT slight AND Tiredness IS NOT slight AND Dry_cough IS slight AND Loss_taste_smell IS slight THEN Cov19_ Likelihood IS low WITH 0.1;
**RULE 10**: IF Fever IS NOT slight AND Tiredness IS NOT slight AND Dry_cough IS NOT slight AND Loss_taste_smell IS slight THEN Cov19_ Likelihood IS medium WITH 0.1;

Figure 4. The sample defined fuzzy rules

To create the fuzzy inference rules, we have based on the basic principle that considers symptoms of the first category [6] as determinist factors of the likelihood of contracting COVID-19. As far as the remaining inputs only increase this likelihood. Besides, weights are gradually assigned to fuzzy rules to prioritize those that provide high likelihood values in the final aggregation. The inference engine generates the output fuzzy sets from the fuzzy rules whose premises match the input fuzzy sets. All output fuzzy sets are aggregated next into a single fuzzy set. At this stage, the combined output fuzzy set constitutes the diagnostic result. This fuzzy result cannot be interpreted easily by healthcare specialists. Thereby it should be converted into a crisp value using the defuzzifier component. There are many techniques for defuzzification, such as the center of gravity or Maxima [24]. In this paper, we have chosen the center of gravity presented in (6).

$$c = \frac{\int_{min}^{max} x\mu(x)dx}{\int_{min}^{max} \mu(x)dx} \tag{6}$$

### 3.3. Generation of explanations with a hybrid XAI technique

Explainable Artificial Intelligence (XAI) consists of understanding the decisions made by machines and making their behaviors more intelligible to humans [25]. The literature distinguishes between transparent or white-box models that are interpretable by design, and those considered opaque or black-box models that can be explained through XAI techniques [26]. Although rule-based systems are considered transparent by design, an explanation facility module is required to increase their level of transparency. Indeed, the explanation facility is the FEES component that might interact with the user interface and knowledge base to provide explanations. These explanations should take a form understandable by users. As stated by [26]–[28], different explanation forms could be generated in XAI: text explanations [29], visualizations [30], local explanations [28], Counterfactual explanations [27], explanations by comparison [31], explanations by example [26], explanations by simplification [26] and feature relevance [32].

Explainability in an expert system could be confronted with two major problems that are coverage and specificity [26]. Coverage corresponds to the number of rules. As long as the expert system uses a lot of rules, its performance will be increased to the detriment of its explicability. The specificity problem relates to the rule length. When the rule's antecedents or consequents are too-long, the explainability also diminishes.

The last phase of the proposal consists of providing an explanation facility that adopts a hybrid XAI technique and solves the coverage and specificity problems. Our hybrid XAI comprises text and visual explanations as well as feature relevance of local rules. Local rules describe the rules that led to a given prediction [28], while their associated weights and degrees of support determine their impact or relevance in the final prediction. Despite accommodating fuzzy logic with rules' expressiveness, this is not enough to provide explanations in a fully-comprehensible manner. Automatic generation of rules in NL could make system decisions more familiar to users and easy to understand. For that, the explanation facility converts the fuzzy rules expressed in Mamdani form to an NL form according to the following template:

I found patient's $IN\_VAR_1$ is $Term\_VAR_1$ from user input
And I also found patient's $IN\_VAR_2$ is $Term\_VAR_2$ from user input
And I also found patient's $IN\_VAR_n$ is $Term\_VAR_n$ from user input
Therefore, the *OUT_VAR* is *Term_VAR* from the activation rule *N*.

$Term\_VAR_i$ and *Term_VAR* represent the linguistic terms associated with input and output variables respectively. $N$ is the number or the noun of the locale rule. An example of expressing rule 1 according to this template:

*I found patient's fever is severe from user input*
*And I also found patient's tiredness is severe from user input*
*And I also found patient's dry cough is moderate from user input*
*And I also found patient's loss of taste and smell is moderate from user input*
*Therefore the likelihood of contracting COVID-19 is high from the activation rule 1.*

The functions *convertToNL(R:Rule)*, *getRightTerm(V:Variable)* and *getActiveVariable(vars:Set)* are used to support conversion. Indeed, *convertToNL(RuleR)* is the main function of conversion and the other ones are considered helpers. It is important to note that the coverage problem is solved using the operators OR and NOT that allow building combined premises, minimizing thus the number of rules. However, this is at the stake of compromising with the specificity problem, since rules could become longer. For this reason, the aforementioned helpers solve the problem of specificity. The function *getRightTerm(V:Variable)* returns the appropriate linguistic term when the operators OR or NOT are present in the antecedent or the consequent of a rule. It selects the term representing the maximum membership value according to the user input as seen in algorithm 2. For example, the expression "Age IS NOT young", with 65 years old as user input, will be "Age is middle-aged". Also, the function *getActiveVariable(Vars:Set)* returns the set of variables that correspond to the symptoms whose associated premises are satisfied as seen in algorithm 3. Another example concerning the rule antecedent with the expression "AND Confusion IS NOT slight OR Chest pain IS NOT slight OR Dyspnea IS NOT slight, with 2.7 as an input of Dyspnea severity, will be "Dyspnea IS severe".

Algorithm 2. Conversion of a premise with NOT/OR operator
```
Function getRightTerm(V:Variable):string
      ch, md, ol, se, maxi: real;     term : string
      /* obtain membership values from the user input for each fuzzy set */
     ch=V .getMembershipValue("child")
     md=V .getMembershipValue ("middle-aged")
     ol=V .getMembershipValue ("old")
     se=V .getMembershipValue("senile")
     maxi = max{se, max{ol, max{ch, md}}}/* get the maximum membership value */
    if maxi== se then
         term="senile"
    else if maxi == ol then
        term="old"
    else if maxi==md then
        term="middle-aged"
    else
        term="child"
     return term
```

Concerning the visual explanations, the facility explanation of MORFEES-C19 provides different plots. For instance, the aggregated fuzzy set and its center of gravity are shown in a plot to highlight the system outcome. Other plots are also available to discover the different ranges and fuzzy sets equivalent to user inputs. Besides, a bar and pie charts are proposed to allow comparisons between locations 'risks and rules supports respectively. Some FEES could contain input variables whose values come from external sources and could be updated over time, such as Max_Locs_Risk. Monitoring the evolution of this variable over time is very important for users to recognize the system's conclusions. Understanding further when such time-observable variables have repeated behaviors or spikes might improve the system's explicability.

Algorithm 3. Conversion of a rule antecedent containing OR operator

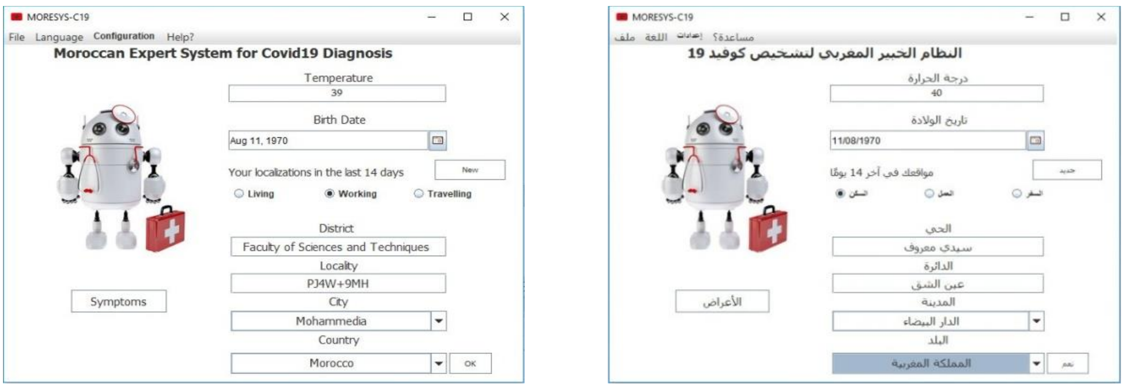
```
Function getActiveVariable(V ars:Set): Set
   max,mo,se,sl: float; actives: Set; // Diagnostic.getDetails() returns the diagnostic
   details, i.e symptoms and their severities entered by the end-user
      foreach D in Diagnostic.getDetails() do
            foreach V in vars do
                              /* lingterm is the linguistic term used in the premise */
                if D.Symptom.name == V.name AND V.getMembershipValue(lingterm)>0 then
                    actives.add(V);
   return actives
```

Within this context, time will be an additional explanatory element. In considering this element, we have proposed a time series chart to illustrate the evolution of this kind of variable at equal intervals of time. It is well-known that a time series is a potential tool for data analysis and forecasting. The main features of time series are level, trend, seasonality, and noise which could be analyzed to make a deep understanding of variables. Overall, the early expert systems such as MYCIN [11] were based on the paradigm why/how or rule-trace to provide explanations. In the proposal, thanks to XAI techniques, we have extended this paradigm with new questions "How much" and "When". In fact, how much relevance is made on a rule to get the output value? The support degrees and weights assigned to rules give a clear answer to this question. When a variable has a remarkable change? This question is answered using the time series chart.

## 4. RESULT AND DISCUSSION

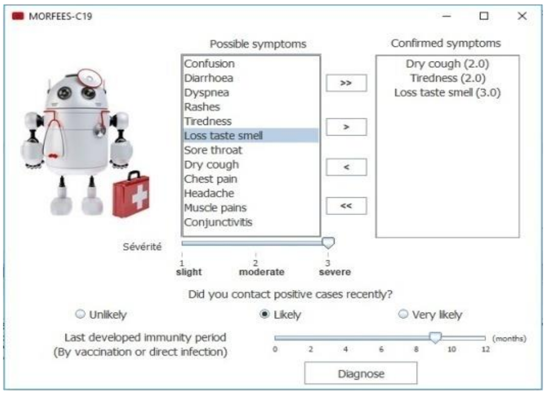
MORFEES-C19 is developed with Java language and uses different Java API, such as JavaMail, Gson, jfreeChart, JPA, and jFuzzyLogic. The latter is an open-source API to integrate fuzzy logic in Java applications [22]. It supports developing fuzzy expert systems and defining fuzzy rules with FCL. Figure 5(a) represents the home frame of MORFEES-C19 displayed in English, while Figure 5(b) shows the Arabic version.

This frame contains the following inputs: the patient temperature, birth date and visited locations in the last 14 days. The next step consists of submitting the symptoms' severities, the possibility of contact exposure to positive cases and the last period of developed immunity. Herein, three main symptoms: fever, dry cough, and tiredness are defined with moderate severity, while loss of taste and smell is severe. Besides, the immunity is developed from nine months with a likely possibility of contacting positive cases. Figure 6(a) illustrates these input values submitted to MORFEES-C19. The latter predicts in the last frame as shown in Figure 6(b) a likelihood of contracting COVID-19 of 92.97% and generates the locale rules in English NL with the associated visualizations.

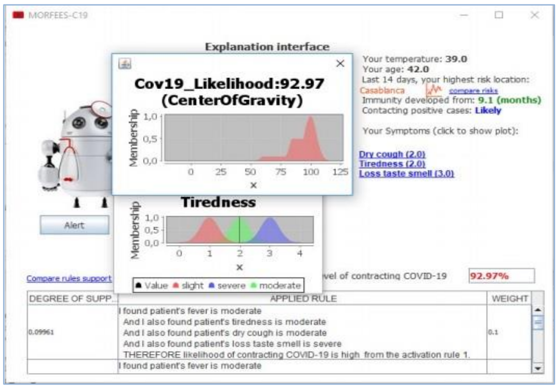



(a) (b)

Figure 5. Home frames of MORFEES-C19 in (a) English and (b) Arabic languages



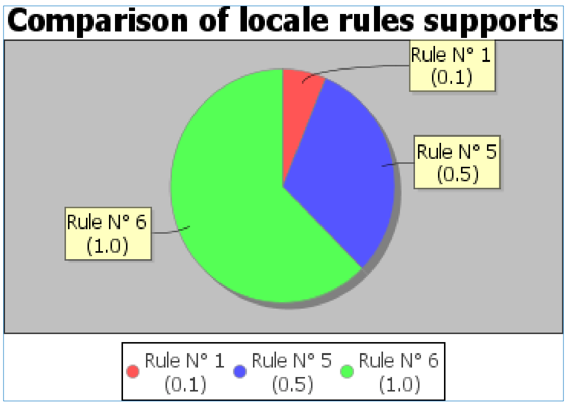
(a)



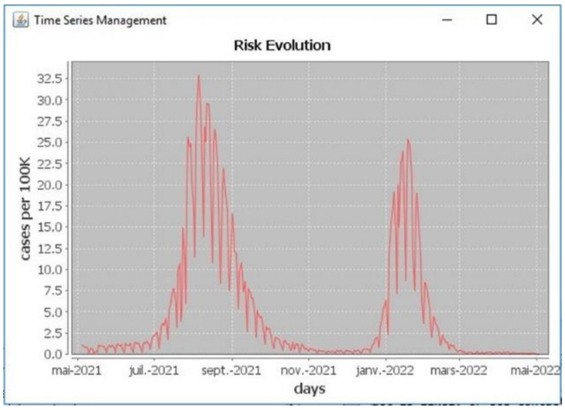
(b)

Figure 6. Input and output frames of MORFEES-C19 (a) user input frame and (b) output frame with explanations

MORFEES-C19 provides important visual explanations thanks to the available hypertexts in the explanation interface. For instance, the hypertext “compare rules support” allows comparing the support of local rules through a pie chart as seen in Figure 7(a). When clicking on the time series icon that marks the visited location with high risk, it appears as shown in Figure 7(b). This chart draws the evolution of the location risk in the last six months. The data is extracted in real-time from a GeoJSON file provided by the website [18]. About symptoms hypertexts, they show the corresponding fuzzy sets plots and the current user inputs.



(a)



(b)

Figure 7. Visual explanations (a) pie chart of rules supports and (b) time series of risk evolution

## 5. CONCLUSION

In this article, we have proposed a fuzzy explainable expert system for early diagnosis of COVID-19. This system infers the likelihood of contracting COVID-19 based on the patient’s symptoms, personal information and activities. This expert system is a medical robotic application that we will attempt to integrate into medical robots. The objective is to assist overwhelmed hospitals with COVID-19 cases and protect the lives of both clinicians and patients. This robotic application integrates fuzzy logic to handle uncertainty and vagueness during diagnosing. For this reason, different membership functions are defined for the system variables in addition to the fuzzy rules required to infer the output variables.

Furthermore, our proposal adopts a hybrid XAI technique that provides different explanation forms, mainly textual and visual ones. The text explanations are generated in NL through an automatic rule-generation algorithm. In the meantime, the coverage and specificity problems are solved using specified helpers. This fact improves rule expressiveness and gives robots the ability to communicate rules to humans in a proper manner. Moreover, the visual explanations are represented by different plots that depict system inputs, outcomes, and inferred rules. In particular, plots drawing the local rules relevance and risk evolution by localization provide a new explanation paradigm "how much/when". The latter extends the rule trace-based paradigm "how/why" of old expert systems to get thus a solution with a high level of explicability.

In future works, we plan to install and deploy this medical robotic application on the robot operating system Raspbian and test it in some clinical settings. The system can be extended with some clinical measures, such as blood pressure, respiratory rate, oxygen saturation, and chest images. However, this implies that the system- to-be should be connected with the clinic information system. As well, one of the interesting future directions is to apply machine learning (ML) algorithms to auto-generate the fuzzy rules.

## REFERENCES


[1] P. R. Innocent and R. I. John, "Computer aided fuzzy medical diagnosis," *Information Sciences*, vol. 162, no. 2, pp. 81–104, 2004, doi: 10.1016/j.ins.2004.03.003.

[2] R. R. Al Hakim, E. Rusdi, and M. A. Setiawan, "Android based expert system application for diagnose COVID-19 disease: cases study of Banyumas regency," *Journal of Intelligent Computing and Health Informatics*, vol. 1, no. 2, Sep. 2020, doi: 10.26714/jichi.v1i2.5958.

[3] P. C. L. Silva, P. V. C. Batista, H. S. Lima, M. A. Alves, F. G. Guimarães, and R. C. P. Silva, "COVID-ABS: an agent-based model of COVID-19 epidemic to simulate health and economic effects of social distancing interventions," *Chaos, Solitons and Fractals*, vol. 139, 2020, doi: 10.1016/j.chaos.2020.110088.

[4] WHO, "Weekly epidemiological update on COVID-19," *World Health Organization*. 2020. Accessed: Apr. 01, 2022. [Online]. Available: https://www.who.int/publications/m/item/weekly-epidemiological-update-on-COVID-19---22-march-2022

[5] Moroccan Ministry of Health, "Statistics of coronavirus in Morocco," *Reuters*. 2022. Accessed: Mar. 20, 2022. [Online]. Available: https://www.reuters.com/graphics/world-coronavirus-tracker-and-maps/countries-and-territories/morocco/

[6] WHO, "COVID-19: symptoms and severity," *World Health Organisation*. 2022. Accessed: Apr. 01, 2022. [Online]. Available: https://www.who.int/westernpacific/emergencies/COVID-19/information

[7] R. H. Taylor, A. Menciassi, G. Fichtinger, P. Fiorini, and P. Dario, "Medical robotics and computer-integrated surgery," in *Springer Handbooks*, Springer International Publishing, 2016, pp. 1657–1684, doi: 10.1007/978-3-319-32552-1_63.

[8] H. A. M. Motlagh, B. M. Bidgoli, and A. A. P. Fard, "Design and implementation of a web-based fuzzy expert system for diagnosing depressive disorder," *Applied Intelligence*, vol. 48, no. 5, pp. 1302–1313, Nov. 2017, doi: 10.1007/s10489-017-1068-z.

[9] D.-M. Filimon and A. Albu, "Skin diseases diagnosis using artificial neural networks," in *2014 IEEE 9th IEEE International Symposium on Applied Computational Intelligence and Informatics (SACI)*, 2014, pp. 189–194, doi: 10.1109/SACI.2014.6840059.

[10] L. Swinney, "The explanation facility and the explanation effect," *Expert Systems with Applications*, vol. 9, no. 4, pp. 557–567, Jan. 1995, doi: 10.1016/0957-4174(95)00024-0.

[11] M. R. Wick and J. R. Slagle, "An explanation facility for today's expert systems," *IEEE Expert*, vol. 4, no. 1, pp. 26–36, 1989, doi: 10.1109/64.21897.

[12] T. De, P. Giri, A. Mevawala, R. Nemani, and A. Deo, "Explainable AI: a hybrid approach to generate human-interpretable explanation for deep learning prediction," *Procedia Computer Science*, vol. 168, pp. 40–48, 2020, doi: 10.1016/j.procs.2020.02.255.

[13] B. M. Nema, Y. M. Mohialden, N. M. Hussien, and N. Ali Hussein, "COVID-19 knowledge-based system for diagnosis in Iraq using IoT environment," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 21, no. 1, pp. 328–337, Jan. 2021, doi: 10.11591/ijeecs.v21.i1.pp328-337.

[14] M. Shatnawi, A. Shatnawi, Z. AlShara, and G. Husari, "Symptoms-based fuzzy-logic approach for COVID-19 diagnosis," *International Journal of Advanced Computer Science and Applications*, vol. 12, no. 4, pp. 444–452, 2021, doi: 10.14569/IJACSA.2021.0120457.

[15] M. Raihan *et al.*, "Development of a smartphone-based expert system for COVID-19 risk prediction at early stage," *Bioengineering*, vol. 9, no. 7, Jun. 2022, doi: 10.3390/bioengineering9070281.

[16] H. R. Banjar, H. Alkhatabi, N. Alganmi, and G. I. Almouhana, "Prototype development of an expert system of computerized clinical guidelines for COVID-19 diagnosis and management in Saudi Arabia," *International Journal of Environmental Research and Public Health*, vol. 17, no. 21, Nov. 2020, doi: 10.3390/ijerph17218066.

[17] J. S. Dhaliwal and L. L. Tung, "Using group support systems for developing a knowledge-based explanation facility," *International Journal of Information Management*, vol. 20, no. 2, pp. 131–149, Apr. 2000, doi: 10.1016/S0268-4012(99)00061-4.

[18] "Moroccan epandemic statistics," *Geomatic*. https://covid19-geomatic.hub.arcgis.com/ (accessed Mar. 30, 2022).

[19] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies*, vol. 7, no. 1, pp. 1–13, Jan. 1975, doi: 10.1016/S0020-7373(75)80002-2.

[20] Z. Xu, K. Gao, T. M. Khoshgoftaar, and N. Seliya, "System regression test planning with a fuzzy expert system," *Information Sciences*, vol. 259, pp. 532–543, Feb. 2014, doi: 10.1016/j.ins.2010.09.012.

[21] E. Sugawara and H. Nikaido, "Properties of AdeABC and AdeIJK efflux systems of Acinetobacter baumannii compared with those of the AcrAB-TolC system of Escherichia coli," *Antimicrobial Agents and Chemotherapy*, vol. 58, no. 12, pp. 7250–7257, 2014, doi: 10.1128/AAC.03728-14.

[22] P. Cingolani and J. Alcala-Fdez, "jFuzzyLogic: A robust and flexible fuzzy-logic inference system language implementation," in *2012 IEEE International Conference on Fuzzy Systems*, Jun. 2012, pp. 1–8, doi: 10.1109/FUZZ-IEEE.2012.6251215.

[23] P. Cingolani and J. Alcalá-Fdez, "jFuzzyLogic: A java library to design fuzzy logic controllers according to the standard for fuzzy control programming," *International Journal of Computational Intelligence Systems*, vol. 6, pp. 61–75, 2013, doi: 10.1080/18756891.2013.818190.

[24] S. Chakraverty, D. M. Sahoo, and N. R. Mahato, "Defuzzification," in *Concepts of Soft Computing*, Singapore: Springer Singapore, 2019, pp. 117–127, doi: 10.1007/978-981-13-7430-2_7.

[25] D. Gunning, M. Stefik, J. Choi, T. Miller, S. Stumpf, and G.-Z. Yang, "XAI—explainable artificial intelligence," *Science Robotics*, vol. 4, no. 37, 2019, doi: 10.1126/scirobotics.aay7120.

[26] A. Barredo Arrieta *et al.*, "Explainable artificial intelligence (XAI): concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, vol. 58, pp. 82–115, Jun. 2020, doi: 10.1016/j.inffus.2019.12.012.

[27] Q. V. Liao, D. Gruen, and S. Miller, "Questioning the AI: informing design practices for explainable AI user experiences," in *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, Apr. 2020, pp. 1–15, doi: 10.1145/3313831.3376590.

[28] J. Bode, M. Schemmer, and T. Balyo, "Explainable AI for constraint-based expert systems," in *17th International Conference on Wirtschaftsinformatik*, 2022, pp. 1–14.

[29] M. Ribeiro, S. Singh, and C. Guestrin, "'Why should i trust you?': explaining the predictions of any classifier," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, Feb. 2016, pp. 97–101, doi: 10.18653/v1/N16-3020.

[30] J. H. Freidman, "Greedy function approximation: a gradient boosting machine," *Institute of Mathematical Statistics*, vol. 29, no. 5, pp. 1189–1232, 2008.

[31] B. Kim, C. Rudin, and J. Shah, "The Bayesian case model: a generative approach for case-based reasoning and prototype classification," *Advances in Neural Information Processing Systems*, vol. 3, pp. 1952–1960, 2014.

[32] D. Baehrens, T. Schroeter, S. Harmeling, M. Kawanabe, K. Hansen, and K. R. Müller, "How to explain individual classification decisions," *Journal of Machine Learning Research*, vol. 11, pp. 1803–1831, 2010.


## BIOGRAPHIES OF AUTHORS

**Omar El Beggar** received the Engineering degree in Computer Science from ENSIAS, University Mohammed V, Morocco in 2002, and his Ph.D. in Computer Science from FSTS, University Hassan I, Morocco in 2013. He obtained afterwards his HDR in Soft Computing and Meta-modelling of decisional support systems from FSTM, University Hassan II, Morocco. Currently, he is a full Professor at the Department of Computer Science at the same faculty and the Pedagogical Director of the engineering department "software engineering and IT systems integration" (ILISI) since 2021. His research interests include green IT, explainable artificial intelligence, MDA, multi-criteria decision aid, and fuzzy logic. He is member of IEEE CIS and author of many publications in relevant international journals and conferences. He can be contacted at email: omar.elbeggar@fstm.ac.ma.

**Mohammed Ramdani** received his Ph.D. in fuzzy machine learning in 1994, and his HDR in perceptual computation in 2001, at the University of Paris VI, France. Since 1996, he is a full Professor at the FSTM, University Hassan II of Casablanca, Morocco. In the same faculty, for the periods 1996-1998 and 2003-2005, he held the position of head of the Computer Science department. Between 2008 and 2014, he was the Pedagogical Director of the engineering department ILISI. Since 2006, he is the Director of the Computer Science Lab. With many relevant publications in indexed journals and conferences, his research interests include explanation in ML, perceptual computation with fuzzy logic and big data mining. He can be contacted at email: mohammed.ramdani@fstm.ac.ma.

**Mohamed Kissi** received in 2004 his Ph.D. in Computer Science from the UPMC, France. Currently, He is a full Professor in the Department of Computer Science, FSTM, University Hassan II, Morocco. His current research interests include machine learning, data and text mining (Arabic) and big data. He is the author of many research papers published in conference proceedings and international journals about Arabic text mining, bioinformatics, genetic algorithms and fuzzy sets and systems. He is member of IEEE CIS and expert of the scientific committee of the faculty. He can be contacted at email: mohamed.kissi@fstm.ac.ma.