Expert Systems With Applications 225 (2023) 120048

Contents lists available at ScienceDirect

Expert Systems With Applications

journal homepage: www.elsevier.com/locate/eswa

An AI-empowered infrastructure for risk prevention during medical examination

Syed Ihtesham Hussain Shah [a,b,*], Muddasar Naeem [b,d], Giovanni Paragliola [b], Antonio Coronato [d], Mykola Pechenizkiy [c]

[a] *Department of ICT and Engineering, Parthenope University, Naples, Italy*
[b] *ICAR, National Research Council, Italy*
[c] *Department of Mathematics and Computer Science, Eindhoven University of Technology, Netherlands*
[d] *Università Telematica Giustino Fortunato, Italy*

ARTICLE INFO

Keywords:
Reinforcement learning
Deep learning
Risk management
Nuclear medicine department

ABSTRACT

A medical examination at Nuclear Medicine Department (NMD) carries out at multiple stages. Patients are accompanied and guided by nurses during their movements within the NMD to avoid them entering into any hazardous situation. However, even accompanying nurses could be exposed to harmful radiation, which puts their safety at risk. Artificial Intelligence (AI) technologies can address these issues by supporting these processes avoiding risky situations, and preventing patients' and clinicians' safe. This article presents an artificial intelligence-based architecture for risk management during the nuclear medical examination to automatically guide the patients during the medical examination and support injury prevention. The architecture comprises two main components; the first component integrates Deep Learning (DL) techniques and WiFi tools to monitor and verify the patient's position continuously; the second integrates Reinforcement Learning (RL) techniques to guide the patient during his/her examination. Experimental results show the suitability of the proposed architecture. Therefore the proposed risk management system can support the prevention of risks and injuries during medical examination and reduce operational costs.

1. Introduction

Risk management in healthcare has been focused mainly on loss prevention, and patient safety (Di Sarno, Formicola, Sicuranza, & Paragliola, 2013; Kuhn & Youngberg, 2002). Solving the patient safety problem avoids the unnecessary effective incidents that many patients face during their interaction with healthcare services (Naeem & Coronato, 2022), including the nuclear medical examination. It is a fact that interventions to minimize risks during a nuclear medical examination will positively impact patient satisfaction. According to the Centers for Medicare and Medicaid Services (CMS), the primary objective should be to reduce risks during healthcare services and improve patient's quality of life (VanLare & Conway, 2012).

The healthcare systems can produce adverse events like any other complex system if not controlled (Vincent, 2011). An adverse event is, by definition, an event in the form of a complication resulting in disability or unintended injury, prolonged hospital stay, or death caused by healthcare management instead of the patient's underlying disease process (Baker et al., 2004). An intrinsic aspect of clinical care is that, whenever it is provided, patients risk suffering from a disease as an unintended consequence of medical examination (Thomas et al., 2000). Therefore, the probability of errors and adverse events, in general, cannot be ignored in healthcare systems like nuclear medicine. However, the risk of adverse events can be minimized by employing AI based risk management systems.

A nuclear medicine examination process consists of many components. A patient seeking examination may have to pass through many rooms, for example, the acceptance room, waiting room, injection room, hot waiting room, etc. Some of the rooms inside the building are useful, while others may be harmful or undesirable. Such medical examination systems benefit humans if they pass through the system in minimum time without facing any dangerous situations. A person going through the examination process may mistakenly enter the wrong room, resulting in an adverse event.

In most healthcare practices, nurses accompany patients when moving within the department with the view to avoid the patients from making mistakes during the examinations. However, such a scenario

* Corresponding author at: Department of ICT and Engineering, Parthenope University, Naples, Italy.
E-mail address: ihtesham.shah@icar.cnr.it (S.I.H. Shah).
URL: https://www.icar.cnr.it/en/ (S.I.H. Shah).

https://doi.org/10.1016/j.eswa.2023.120048
Received 19 August 2022; Received in revised form 16 March 2023; Accepted 2 April 2023
Available online 10 April 2023
0957-4174/

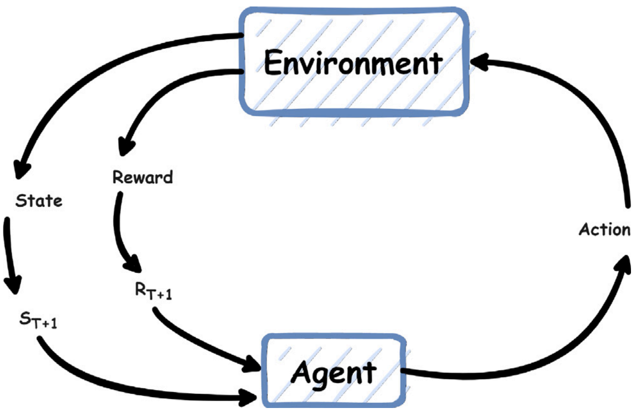


Fig. 1. The reinforcement learning problem.

defines a situation in which both the patient's and nurses' safe are at risk, demanding an efficient risk management system to reduce the time duration of the medical examination as much as possible. Achieving the temporal shortest medical examination would improve the safety of all subjects, such as patients and nurses. To do that, optimizing the leading of the patients' across the nuclear medicine examination process is essential. The motivations of this work are stated in this objective.

The work focuses on developing a risk management system that can assist a person in examining a nuclear medicine department. The proposed approach has a tracker (consisting of a WiFi module and DL) to track the position of a person inside the nuclear medicine department and a RL based controller that guides the person during the examination process inside the building to help them in avoiding harmful events. We have defined a two-step verification mechanism to identify a person's current position. Based on the tracked position, the RL controller guides the patient to the final step through the safest path. In this way, nurses will no longer be necessary for the surveillance of patients within the department. This will have two benefits: lower costs and fewer people exposed to radiation.

The main contribution of the paper concern the definition of *intelligent step-by-step* workflow for risk prevention of nuclear medicine examination processes, where *intelligent* means the integration of two different learning approaches (RL and DL), and *step-by-step* means that the approach models each step of the medical process in accordance with the patient's actions. These two properties make a strong difference compared to the state-of-the-art since they are not concurrently addressed in other works. The proposed workflow does not need any additional infrastructure for the operation since it can adopt the available capabilities of smartphones and personal devices and assist a patient during his/her medical examination. The goodness of the proposed work has been brought to light from an analysis of the experimental results.

The rest of the paper is organized as follows: Related work is described in Section 2 with a discussion on literature. We will discuss a case scenario in Section 3. Section 4 comprises a quick review of background and problem formulation, where introductory concepts about the RL and DL are presented. Then a detailed introduction to system including the major components (Tracker and Controller) is given in Section 5. The discussion about the experiments, dataset and results is reported in Section 6. We summarize the paper in Section 7.

2. Related work

The recent biomedical and technological innovations have led the healthcare sector to implement clinical governance to provide the best quality of healthcare facilities in an increasingly complex environment. Risk management is one of the most relevant aspects of clinical governance, and techniques proposed in the literature highlight the importance of developing such systems (Cagliano, Grimaldi, & Rafele, 2011). The reference standard currently used by manufacturers for risk management is ISO 14971. This standard was designed for traditional medical devices and does not either define a formal methodology to conduct a risk assessment or consider the peculiarities of current medical information architectures (Coronato & Cuzzocrea, 2020). Moreover, the techniques implemented currently by manufacturers normally target to find qualitative Risk Assessment results.

A system for risk transition management in e-Healthcare services is proposed in Wiboonrat (2011). The proposed model is discussed using many cases of healthcare services transition projects in Thailand to develop and manage transition plans. Risk management is deployed during the transition process to reduce project failure. The work presented in Ham, Hwang, Kim, and Lee (2009) carried out a questionnaire and concluded the need for effective patient risk management in a nuclear medicine department. The results of the study showed that it is possible to minimize safety accidents during the examination process.

A qualitative study of two embedded cases was done in two sections of a hospital's nuclear medicine department to understand clinicians' relationship to the risks of exposure to low doses and their procedure to combine the logic of patient care/cure concerning self-protection (Lonceint, Bodéré, & Geffroy, 2019). The study includes 23 interviews and ten weeks of observations with several health professionals in the department. The study shows the coexistence of care/cure and radiation protection logic to be a source of contradictions for professionals in a nuclear medicine department.

A deep RL technique for fall risk reduction using mobile assistant robots is proposed in Namba and Yamada (2018). The authors have collected data regarding past incidents and then used it as the input data to analyze fall risks and to assess the examples of risk reduction measures. A Q-learning-based methodology is employed in Paragliola, Coronato, Naeem, and De Pietro (2018) and Paragliola and Naeem (2019) to assist customers in a nuclear medicine department. However, it was assumed that the patient is equipped with short-range Radio Frequency Identification (RFID) readers, and the position of the patient (state) was emulated.

A multi-agent RL risk management system for distributed agile software projects is designed in Adel, Harb, and Elshenawy (2021). The system is implemented to use a dynamic policy. The model is applied as an experiment to definite numbers for a set of risk factors, for example, software development life cycle risk, project management risk, communication, and coordination risk.

In Liu, Hu, and Lin (2021), authors applied data mining techniques on a hospital's electronic medical records database comprising a nursing information system to construct inpatient-fall-prediction models for use during various stages of inpatient care. An adaptive differential evolution scheme for resource allocation by utilizing generalized opposition-based learning and belief space is proposed in Deng, Ni, Liu, Chen, and Zhao (2022).

The cited works provide an overview of the application of traditional machine-learning-based approaches to support clinical assistance in e-Health locations. However, it is possible highlighting a few issues:

– None of the papers aimed to directly interact with the subject since the models are built to support only the clinical decision process.
– All the related works concern the lack of a verification step confirming the correctness of the patient's activities during the examination
– None of the papers propose solutions based on the combination of different learning approaches since most papers adopt machine-learning-based techniques without combining them with other models.

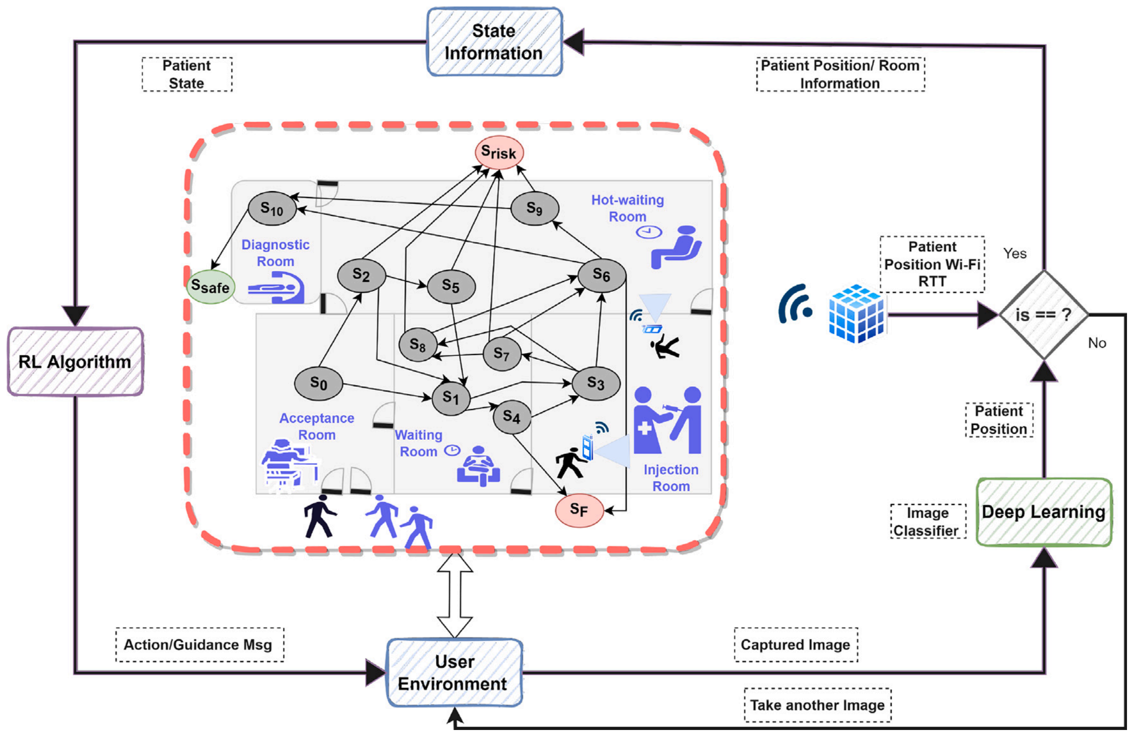


Fig. 2. Workflow of proposed risk management system.

The proposed paper addresses all the previous issues by defining a step-by-step verification process that identifies the current position of a subject and using RL-based controller to guide a subject during his/her examination process inside a nuclear medicine department to help him/her avoid harmful events. In detail, the novel contributions of the paper are:

– the definition of an RL-based recommendation system that directly suggests to the patient his/her next move following the current state of his/her medication and current position in the nuclear department.
– the definition of a DL-based model to verify the correct proceeding of the examination in order to guarantee that the patient correctly follows the RL-based recommendation system
- the definition of a hybrid approach based on the combination of both RL and DL techniques to define the tracking task, the recommendation task, and the verification task.

Our system defines a step verification process that identifies the current position and a RL based controller based on the tracker's input, guiding the person during the examination process inside the building to help them avoid harmful events.

3. Use case scenario

An overview of the nuclear medicine department can be visualized inside the red boundary of Fig. 2. We consider a nuclear medicine building that is consist of different locations as described below:
– A Reception Room (RR) where the patients are admitted into the department;
– A Waiting Room (WR) is a place where the patients wait for the injection of a radio-pharmaceutical;
– An Injection Room (IR) where the patients are injected with the required substance;
– A Hot Waiting Room (HWR) where the patients have to wait for the examination until the radiation level reaches the target range after having been injected;
– A Diagnostic Room (DR) where the examinations are performed. The patient is equipped with a smartphone (tracker) so that the system may track him/her. For the successful completion of the medical expatriation, the patient has to follow this path: $RR \rightarrow WR \rightarrow IR \rightarrow HWR \rightarrow DR$.

The goal is to automatize the complete examination process, where the automatize indicates that hospital staff is not responsible for monitoring patients inside the department. This scenario exposes the patient to radioactive agents and may cause severe injury. If a patient does not follow the correct sequence, they enter a hazardous situation, as shown in Fig. 3. Moreover, Coronato and Pietro (2010) focuses on a real case example, which consists of a pervasive application for the department of nuclear medicine. The description of each state representing a particular situation is given in Table 1.

For example, the state s_4 indicates the situation when a patient goes in and out of HWR without being injected, and correspondingly two directions (actions) are available such as (1). the patient is in the HWR without injection and he may move to WR; (2). The patient may move to an undesired location (risk state). The controller part of the proposed system learns all these dynamics and can eventually guide a patient through the examination process.

According to what clinicians have reported in the description of the workflow, the patient receives the substance by injection and he/she gets a certain degree of radiation. However, even people who stand close to the radioactive substance before the injection receive a little amount of radiation. For this reason, clinical staff is equipped with a sensor that measures the level of radiation in the air and is supposed to stay within the injection room for a predetermined maximum time slot.

Definitively, entering by mistake the injection room may cause the retirement of some radiation, which would be preferred to be avoided even if of very small quantity and thus with limited risk. We have developed an intelligent Risk Management System (RMS) that consists of RL and DL agents to minimize the risk of patients entering into dangerous situations during their medical examination. The RMS receives the patient's position and according to the current position, it provides a guidance message to the patient to safely lead him/her through the department's building.

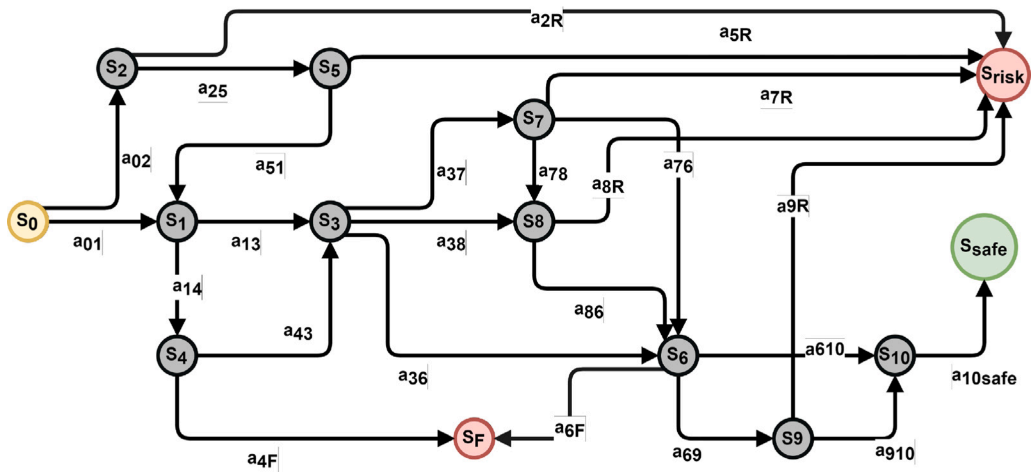


Fig. 3. Risk case scenario: States action representation of risk management environment.

Table 1
Description of states and actions.

States	Description	Actions	Description
s_0	Patient Admitted for medical examination	a_{01} a_{02}	You are in Acceptance Room, Please move to waiting Room You are in Acceptance Room, Please move to Hot waiting Room
s_1	A patient waits in WR and not injected	a_{13} a_{14}	You are in waiting Room, Please move to Injection Room You are moving in the same waiting Room
s_2	Patient waits in HWR and not injected	a_{25} a_{2R}	BE CAREFUL !, You are moving to HWR without being injected You are in a Risk State
s_3	Patient is available for injection in IR	a_{36} a_{37} a_{38}	You are in the injected, Please move to Hot Waiting Room You are moving back to the Waiting Room, Please move to HWR Your are injected and in WR, please move to the HWR
s_4	A patient exits the WR but enters in the WR again	a_{4F}	System does not work
s_5	The patient goes in and out of HWR without being injected	a_{51} a_{5R}	You are in the HWR without injection, Please move to WR You are in a Risk State
s_6	The patient waits in HWR and injected	a_{6F} a_{69} a_{610}	System does not work you are still in HWR, Please move to the diagnostic Room You are in HWR, Please move to the diagnostic Room
s_7	The patient waits in WR and injected	a_{7R} a_{76} a_{78}	You are in a Risk State Your are Injected and in WR, please move to the HWR Be CAREFUL! Your are Injected and in WR, please move to the HWR
s_8	The patient goes in and out of WR after being injected	a_{8R} a_{86}	You are in a Risk State Your are Injected and in WR, please move to the HWR
s_9	The injected patient waits in HWR instead of moving into DR	a_{9R} a_{910}	You are in a Risk State Please move to the diagnostic Room for inspection
s_{10}	Patient under examination	a_{10Safe}	You are in diagnostic room, examination process is underway
Terminal states			
s_{safe} s_F s_R	A patient exits from the DR System Failure Risk Position		

4. Technical background

This section provides a brief introduction to different AI tools that have been used during experiments.

4.1. Reinforcement learning

RL is a sub-field of Machine Learning (ML) where an agent interacts with an environment to achieve a goal, and learning takes place interaction after interaction. This section introduces some basic mechanisms and terminology. A detailed presentation of RL can be found in Sutton and Barto (2018).

In RL an **Agent** is an entity (algorithm/robot/player, etc.) that interacts with a given environment (problem/smart space/game, etc.) by performing actions, and receives feedback (penalty/reward) from the environment after any action selected as described in Fig. 1. The reward is the mechanism that enables the agent to understand whether the action selected has produced a positive or a negative effect concerning the final goal.

A **policy** is a strategy that indicates to the agent which action to select in every state of the environment. The agent has to learn the optimal policy; that is, the one that maximizes the cumulative reward over the long run.

A RL problem is defined as a **Markov Decision Process (MDP)**. A MDP is a tuple (S, A, P_a, R_a, γ), where S is a set of states, A is a set of actions, $P_a = P_r(s_{t+1} = s'|s_t = s, a_t = a)$ is the transition probability (i.e., the probability of achieving s' at time $t+1$, after having selected a in s at time t), $R_a(s, s')$ is the expected reward or immediate reward obtained when transitioning from state s to state s' when action a was taken respectively, and γ is a discount factor.

For any state–action (s, a) pair, the probability of resulted state and the corresponding reward (s', r) is given as in Eq. (1):

$$p(s', r|s, a) \doteq Pr\{S_{t+1} = s', R_{t+1} = r|S_t = s, A_t = a\} \tag{1}$$

Informally, the target of the RL agent is to maximize the reward. This is to say, with the list of rewards R_{t+1}, R_{t+2}, … after time period t, the goal is to maximize the reward function as given in Eq. (2):

$$G_t = R_{t+1} + R_{t+2} + \cdots + R_T \tag{2}$$

where T is the last time interval.

The return G_t is the sum of discounted rewards obtained after time t.

$$G_t = \sum_{k=0}^{T} \gamma^k R_{t+k+1} \tag{3}$$

A policy π defined in Eq. (4) tells an agent which action to take in a given state.

$$\pi(a|s) \doteq P[A_t = a|S_t = s] \tag{4}$$

Having the policy π and the return G_t, two value functions can be defined, i.e., state–value and the action–value functions. The state–value function $v_\pi(s)$ is the expected return starting from a state s and following the policy π as given in Eq. (5).

$$v_\pi(s) \doteq E_\pi[G_t|S_t = s] = E_\pi[\sum_{k=0}^{\infty} \gamma^k R_{t+k+1}|S_t = s] \tag{5}$$

The action–value function $q_\pi(s, a)$ is the expected return starting from a state s, taking action a, by following the policy π.

The optimal value function is one that obtains the best gains in terms of returns, as given in Eq. (6).

$$v_*(s) = \max_\pi v_\pi(s), \forall s \in S \tag{6}$$

RL schemes are normally categorized into two major types, model-free, and model-based algorithms. Model-based RL algorithms need a precise description of the dynamics of the environment in terms of the state-transition probability distribution. These methods (e.g., Dynamic Programming (DP)) compute the optimal policy by solving systems of equations. Whereas model-free RL techniques are adopted when there is not a precise description of the model or its solution is too complicated. This class of algorithms interacts directly with the environment (or with an emulator) using Trial&Error schemes to learn the optimal policy. In inverse RL (Shah & Coronato, 2021a, 2021b; Shah, De Pietro, Paragliola and Coronato, 2022), we study an agent's objectives, values, or rewards with the help of employing insights into its behavior. Several methods are available (e.g., M Monte Carlo (MC), Temporal Difference (TD), etc.). An overview of such methods is reported in Naeem, Coronato, and Paragliola (2019), Naeem, Rizvi, and Coronato (2020) and Paragliola et al. (2018), whereas a guideline useful to help to choose the algorithm depending on the kind of problem is defined in Coronato, Naeem, De Pietro, and Paragliola (2020) and Shah, Coronato, Naeem and De Pietro (2022).

4.2. Deep learning

DL has revolutionized many research areas with its ability to learn better models from huge volumes of data (Coronato, de Pietro, & Paragliola, 2013; Sarker, 2021; Zhao et al., 2022). Such technology relies on a new generation of Artificial Neural Networks (ANNs) called Deep Neural Networks (DNNs). Before approaching the next section, this subsection presents a brief overview of DL techniques.

The premise is that the performance of a DNN is generally superior to the one of a classic ANN at the cost of more excellent training time that, however, can be reduced by using advanced hardware (e.g., GPU) and/or special techniques (e.g., Transfer Learning) (Li et al., 2022; Mathew, Amudha, & Sivakumari, 2020; Paragliola & Coronato, 2013). The design of a DNN is crucial for success. We start this subsection by discussing some of the most used DL architectures.

Convolutional Neural Networks are ANNs with a much higher number of layers and nodes. They are typically adopted for image classification. A Convolutional Neural Network (CNN) needs less pre-processing as compared to other classification schemes. Relevant filters are used in CNNs to capture the temporal and spatial dependencies in the image (Le et al., 2015; Yamashita, Nishio, Do, & Togashi, 2018). The most common CNN architectures are: ZFNet, ResNet, GoogleNet, VG-GNet, AlexNet, and LeNet (Li, Liu, Yang, Peng, & Zhou, 2021).

Resnet is a short form of residual networks, and ResNet50 is a fifty-layer deep convolutional Neural Network (NN) that was initially introduced in 2015 (He, Zhang, Ren, & Sun, 2016). It consists of five stages, and each stage has multiple convolution layers. This architecture gives us 50 layers (in total) deep convolutional network. There are over 23 million trainable parameters that the ResNet50 model has.

This framework can be used on many computer vision tasks such as object detection, object localization (Khan, Jalil, Haq, & Shah, 2021), image classification etc. ResNet50 can also be deployed to solve non-computer vision tasks to benefit from the depth and reduce the computational cost.

In this article, we have applied ResNet50 architecture for image classification. Images from the patient's cell phone are collected and then classified through a pre-trained ResNet50 model. This classification helps us verify the patient's current position in a nuclear medicine department.

5. System model

The proposed system model is shown in Fig. 2 is consist of a WiFi indoor positing system, DL, and RL learning techniques. The WiFi indoor positing system and DL model work together to track a patient's position. The WiFi module monitors the current position of a person while DL verifies that position. We considered the WiFi indoor positioning system as a reference. If the output of the DL does not match the output of the WiFi indoor positioning system then another picture will be taken from the device that the patient is holding, it will be classified from the DL and again the position of the patient will be compared with the one received from WiFi indoor positioning system as demonstrated in Fig. 2. Once a patient's position has been identified, it is input to the controller (as state information to the RL algorithm). A detailed description of each component is presented next. We start with a detailed description of the use case scenario.

5.1. Tracker

The tracker is used to locate the current position of the patient. We employ two-position locator methods to increase the accuracy. First, we find the patient's position by using the WiFi indoor position system, and then in the next step, verification is done by employing the DL method. Therefore, it is a two-way verification system (WiFi indoor position system and DL classifier).

- **WiFi indoor positioning system**
  The WiFi indoor positioning system uses WiFi access points that transmit certain information about the coordinates. This system defines the patient's current position by evaluating the router's MAC address and the RSSI (received signal strength indicator).
  WiFi standard IEEE_802.11mc is a new advancement. It provides a measurement protocol to calculate the distance between the transmitter and receiver using Round Trip Time (RTT) rather than the received signal strength indicator (RSSI). WiFi RTT (round trip time) is a comparatively new technology with low latency, and high accuracy (Bai, Kealy, Retscher, & Hoden, 2020). It measures the distance between WiFi routers and a device

Algorithm 1 SARSA algorithm for Controller

Given: Environment parameters (i.e Set of states S, set of actions A).
Initialize: Learning rate lr, Discount factor γ, greedy selection ϵ, No. of episodes, starting state = s_0, End state = s_{safe}
set: $R(:,:,:) = 0$, $R(s,a,s_{safe}) = 100$, $R(s,a,s_R) = -100$ and $Q(s,a) = 0$
for ($i = 1$; $i \leq No.of\ Episodes$; i^{++}) **do**:
  state = starting state
  if ($np.random.uniform(0,1) < epsilon$) **then**: randomly choose action
  else: action = np.argmax (Q[state,:])
  end if
  while (state != End state) **do**:
    choose next state (state, action)
    if (np.random.uniform(0, 1) < $epsilon$) **then**: choose next action randomly
    else: next action = $np.argmax(Q[state,:])$
    end if
    predict = $Q[state, action]$
    target= $R(s,a) + \gamma * Q[next\ state,\ next\ action]$
    $Q[state, action]$= $Q[state, action] + lr * (target - predict)$
    state ← next state
    action ← next action
  end while
end for
Return: $Q(s,a)$

Table 2
Q-Table represents the Q-values of each action at each state.

Actions	Values at each state													
	s_0	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_{10}	s_{safe}	s_F	s_R
a_{01}	**55.37**	−6.76	0.00	−4.90	0.00	0.00	25.02	−6.01	0.00	0.00	0.00	100	−100	−100
a_{02}	15.53	−0.06	0.00	−0.15	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{13}	0.00	**66.13**	0.00	0.00	0.00	0.00	0.00	7.36	0.00	0.00	0.00	100	−100	−100
a_{14}	0.00	58.02	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{25}	0.00	0.00	**53.19**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{2R}	0.00	0.00	−5.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{36}	0.00	0.00	0.00	**86.39**	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{37}	0.00	0.00	0.00	12.68	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{38}	0.00	0.00	0.00	13.86	0.00	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{43}	0.00	0.00	0.00	0.00	**60.74**	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{4F}	0.00	0.00	0.00	0.00	−20.32	0.00	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{51}	0.00	0.00	0.00	0.00	0.00	**76.11**	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{5R}	0.00	0.00	0.00	0.00	0.00	−75.09	0.00	0.00	0.00	0.00	0.00	100	−100	−100
a_{6F}	0.00	0.00	0.00	0.00	0.00	0.00	−13.96	0.00	0.00	0.00	0.00	100	−100	−100
a_{69}	0.00	0.00	0.00	0.00	0.00	0.00	30.57	0.00	0.00	0.00	0.00	100	−100	−100
a_{610}	0.00	0.00	0.00	0.00	0.00	0.00	**87.00**	0.00	0.00	0.00	0.00	100	−100	−100
a_{7R}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	−12.74	0.00	0.00	0.00	100	−100	−100
a_{76}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**47.87**	0.00	0.00	0.00	100	−100	−100
a_{78}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	18.59	0.00	0.00	0.00	100	−100	−100
a_{8R}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	−16.57	0.00	0.00	100	−100	−100
a_{86}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**65.71**	0.00	0.00	100	−100	−100
a_{9R}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	−17.74	0.00	100	−100	−100
a_{910}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**73.73**	0.00	100	−100	−100
a_{10Safe}	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	**90.73**	100	−100	−100

(smartphone) which is way more accurate. The WiFi position system's accuracy depends on the building's topology, the number of access points, and the type of smartphone. Modern Android-based smartphones can provide accuracy up to 1 m (Retscher, 2020).

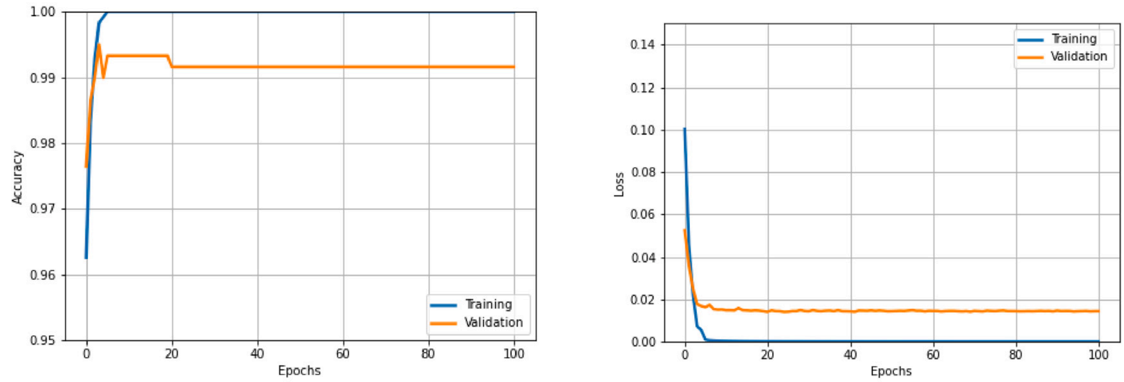
WifiRttLocator is an Android-based application that uses WiFi RTT technology. It allows users to track their position on their smartphones. We used this platform to get the accurate position of the patient.

Setting up the configuration for this App takes two steps. 1. Selecting configuration file: It describes the locations of all the WiFi RTT-capable access points. 2. Importing Overlay file: This file contains the overlay map of the location. We used the blueprint of the nuclear medicine department for this purpose. The WifiRttLocator application scans and detects nearby access points and starts tracking indoor navigation. We send this information to the database, where we compare it with the patient's position obtained from the deep learning model.

- **Deep Learning Model** The deep learning model using the pre-trained ResNet50 model is employed as a second step verification tool for more accurately obtaining the patient's position inside the nuclear medicine department (NMD). The idea is to track the patient's position by getting the image of the patient's surroundings. The image is then sent to the base station, where information about the patient's current room (position) is obtained.

  The deep learning model ResNet50 is used for image classification. It is a conventional neural network (CNN) model that is 16 layers deep. We trained this model for six categories (acceptance room, waiting room, gallery, injection room, hot waiting room, and diagnostic room).

  We are getting live video streaming from the patient's mobile camera. After five seconds, a frame from the video streaming is



(a) Training and validation accuracies

(b) Training and validation loss

Fig. 4. Loss and accuracies of deep learning model.

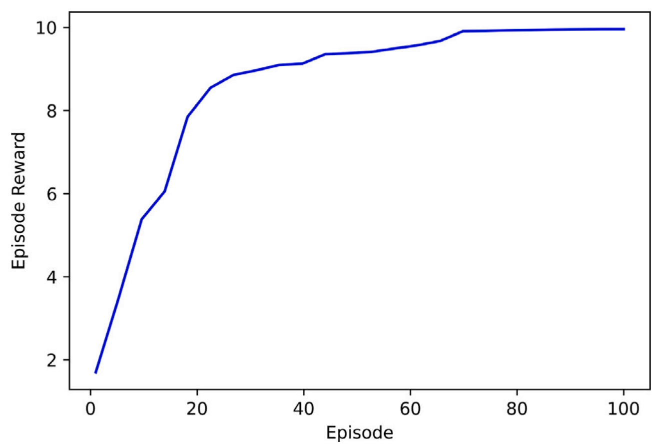


Fig. 5. Learning curve for SARSA algorithm.

Table 3
Deep learning model parameters.

Name of parameters	Values
Input image size	224,224
Number of training epochs	100
Batch size	32
Training/validation split	0.8/0.2
Seed	123
Pooling	avg
Activation function for hidden layer	relu
Activation function for output layer	softmax
Loss function	cross entropy

collected and classified through the ResNet50 classifier. It gives us the most recent location of the patient.

5.2. Controller

The controller is an intelligent agent empowered with a reinforcement learning algorithm. State–Action–Reward–State–Action (SARSA) was introduced in Rummery and Niranjan (1994) as a modified version of Q-learning. Being an online learning method, the agent interacts with the environment (emulated nuclear medicine department) and updates policy based on selected action (Sutton & Barto, 1998). The action–value function Q is updated by an error and adjusted by the learning rate lr as given in Eq. (7).

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + lr[r_{t+1} + \gamma Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)] \tag{7}$$

The pseudo-code is presented in algorithm 1 where the agent takes action in the initial state, observes the reward and moves a step onward where it observe next state and next action. In this stage, the Q-function is updated, and it continue taking the steps until reaches the final state. The policy is updated at each episode by taking action with a maximum Q value. By utilizing the eligibility traces to state–action pairs may speed up convergence in SARSA. The eligibility traces are updated in $SARSA(\lambda)$ as given in Eq. (8).

$$\begin{aligned} e_t(s,a) &= \gamma\lambda et - 1(s,a) + 1 \quad \text{if} \quad s = s_t \\ &\qquad \text{and} \quad a = a_t \\ e_t(s,a) &= \gamma\lambda et - 1(s,a) \qquad \text{otherwise} \end{aligned} \tag{8}$$

And the resulting update rule for the SARSA algorithm by using trace can be written as given in Eq. (9).

$$Q_{t+1}(s,a) = Q_t(s,a) + \alpha\delta_t e_t(s,a) \text{ for all } s \in S \tag{9}$$

Convergence in SARSA is guaranteed when all the pairs (state–action) are observed for an infinite number times (Russell, Norvig, Canny, Malik, & Edwards, 2003). To consider all states and actions, we can use $\epsilon-$ greedy policy that randomly chooses the action with small probability ϵ and otherwise takes action with high values as given in Eq. (10).

$$\pi(s) = \begin{cases} \operatorname{argmax}_a Q(s,a) & \text{if } \sigma > \epsilon \\ a \approx A(s) & \text{if } \sigma \leq \epsilon \end{cases} \tag{10}$$

where $0 \leq \sigma \leq 1$.

6. Experiment

This section first presents an overview of the dataset used during experiments, then the performance of both tracker and controller, and an overall discussion on the proposed system's suitability.

6.1. Dataset

We have collected images from University Hospital Federico-II, Naples, Italy, to train the DL model. Initially, the dataset contains 500 images of each room captured manually in different orientations and light conditions. Later, we adopted data augmentation to increase the dataset, and for that purpose, each image is rotated, shifted, zoomed in/out, distorted, and shaded with a hue. In the end, we obtained 1000 images for each class/room. We resized images to (224,224) for the training of the model. We used 80% and 20% ratios for training and validation, respectively. The training dataset is used to make the model learn hidden features/patterns in the data. On the other hand, a validation dataset is used to validate the performance of the trained model.

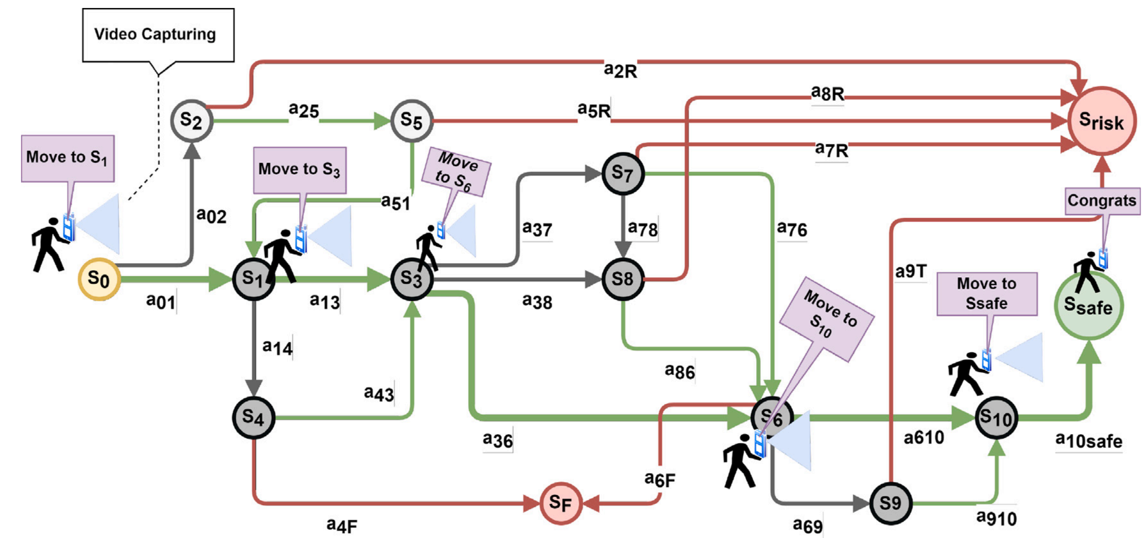
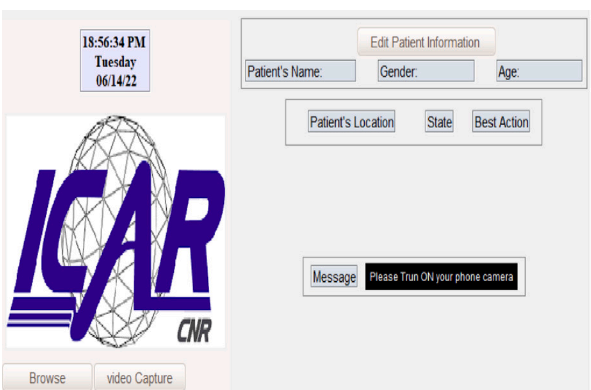
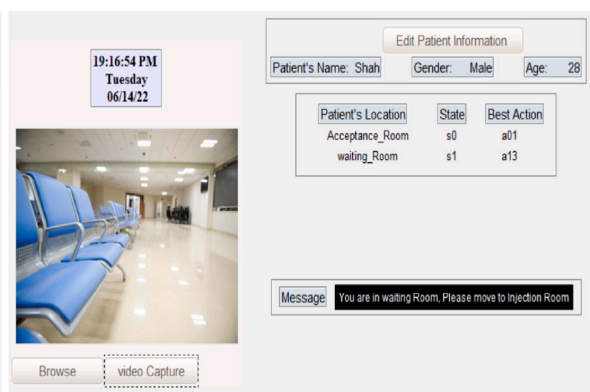


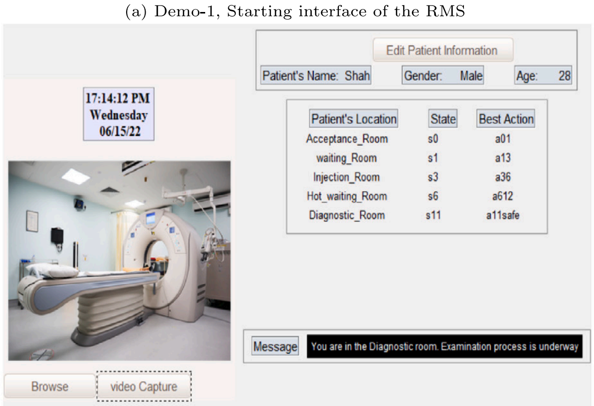
Fig. 6. At each state, actions in green color represent the optimal action, those in red represent the worse actions and optimal path for a patient is represented in thick green color.



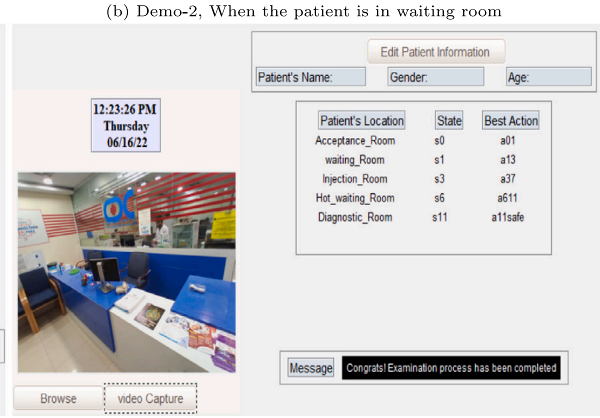
(a) Demo-1, Starting interface of the RMS



(b) Demo-2, When the patient is in waiting room



(c) Demo-3, When the patient is in diagnostic room



(d) Demo-4, When medical examination is completed

Fig. 7. Demonstration of the proposed system.

There are more than one label class and we expect the label to be specified as an integer. Thus, we used a sparse categorical cross-entropy loss function to measure the loss between label and prediction. All other parameters used for training and testing the Resnet50 model are shown in the Table 3. Moreover, the total number of parameters in our Resnet50 model was 24,639,365, from which 1,051,653 were trainable, and 23,587,712 was non-trainable parameters.

6.2. Results

We trained the Resnet50 model on Tesla-K80 GPU by using the Google-coLAB platform. We used classification accuracy and loss function as performance metrics for our model. Figs. 4(a) and 4(b) show accuracy and loss performance for Resnet50 model respectively. The model has better performance as it has achieved 99 percent accuracy and a loss of nearly 0.02 on the validation dataset.

Furthermore, RL-model(State–Action–Reward–State–Action (SARSA)) is used to get the appropriate action (guidance message) for the current detected position (state) of patients. We trained the SARSA model and the learning curve is shown in Fig. 5. It is evident that the RL agent is able to learn the dynamics of the NMD in a few iterations. Once the RL agent (Controller) learned the environment dynamics, it can provide assistance to the patient through appropriate guidance messages. $Value$ is the maximum expected future reward for action at each state. Table 2 represents the $values$ of actions at each state. Optimal action is the one that has the height $value$. For example, at state s_0 two actions (a_{01}, a_{02}) are possible to select. Action a_{01} has the highest $value$ which is more likely to be selected than action a_{02}. Similarly, all the best possible actions at each state are highlighted in green color in Table 2.

For better understanding, we mapped these $values$ to the layout of the risk management environment as shown in Fig. 6. The $values$ of each action in the figure is represented with different color and thicknesses of the lines from one state to other. Actions with green lines represent the best action, while those with red lines represent the worst action in each state. On the other hand, gray lines represent normal actions (neither best nor worse). An optimal policy $\{(s_0, a_{01}), (s_1, a_{13}), (s_3, a_{36}), (s_6, a_{610}), (s_{10}, a_{10S_{safe}})\}$ is shown with thick green lines.

Fig. 7 shows the real demonstration of the proposed system. We discussed few scenarios here. The description related to each result is given next.

1. The Fig. 7(a) shows the starting interface of the risk management system. The interface contains the necessary information about the patient. Some of the information on the interface is technical (for example, state, and action) and it is used for experimental purposes. The system suggests the user turn on the smartphone camera before entering the building.
2. The second image in Fig. 7(b) presents the situation when a patient is admitted for inspection and waits in the waiting room. We can see that the position is correctly identified using the tracker component (WiFi indoor positing system and DL classifier) of the system and then there is a guidance message from the controller component of the system. The message is that the patient is in the waiting room and he should move to the injection room. Where the patient will be injected. After getting an injection, the patient has to stay in the hot waiting room until the parameters (blood pressure and heartbeat and body temperature, etc.) reach the required level. After that patient has to move to the diagnostic room for further medical inspection. RMS guides the patient during this process.
3. Next we can visualize in Fig. 7(c) that the detected position by the tracker (WiFi indoor positing system and DL classifier) is "diagnostic room" and the corresponding guidance message for the patient is "You are in the diagnostic room and examination is undergoing".
4. The Fig. 7(d) shows the completion of the process when a patient gets out of the diagnostic room after a detailed medical examination.

The demonstration presented in Figs. 7(a) to 7(d) indicates the feasibility of the proposed system. However, it should be noted that the system is intended to provide service to normal patients such that a patient without any visual and audio disabilities. We believe that the proposed solution can minimize the risk of entering into hazardous situations if a patient follows the system-generated live guidance.

7. Conclusions

In this paper we have presented a risk management system that can assist a person in examining a nuclear medicine department with the aim of supporting and supervising patient's activities during the examination process at a nuclear medicine department.

The proposed system consists of WiFi indoor positing system and deep learning method to monitor and verify a patient location and then used reinforcement learning techniques to guide the patient for safest path and avoid entering into dangerous states during the examination process.

The experimental results show that the system is able to get the appropriate guidance messages for the current detected position of patients and lead them to a safety examination. The proposed risk management system not only helps to reduce the risk and injury during medical examination but also minimize the cost.

In future work, we want to apply the proposed approach to more complex environments with multiple possible dangerous situations and investigate the application of both algorithms to deep learning and reinforcement learning techniques to address the more complex scenario.

List of acronyms

RL Reinforcement Learning

DL Deep Learning

NMD Nuclear Medicine Department

AI Artificial Intelligence

MDP Markov Decision Process

TD Temporal Difference

DP Dynamic Programming

RMS Risk Management System

NN Neural Network

CNN Convolutional Neural Network

SARSA State–Action–Reward–State–Action

ANN Artificial Neural Network

DNN Deep Neural Network

ML Machine Learning

TD Temporal Difference

MC Monte Carlo

RFID Radio Frequency Identification

RTT Round Trip Time

CRediT authorship contribution statement

Syed Ihtesham Hussain Shah: Formal analysis, Software. **Muddasar Naeem:** Conceptualization, Resources. **Giovanni Paragliola:** Investigation. **Antonio Coronato:** Validation, Supervision. **Mykola Pechenizkiy:** Supervision, Project administration.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

References

Adel, R., Harb, H., & Elshenawy, A. (2021). A multi-agent reinforcement learning risk management model for distributed agile software projects. In *2021 tenth international conference on intelligent computing and information systems (ICICIS)* (pp. 512–520). IEEE.

Bai, Y. B., Kealy, A., Retscher, G., & Hoden, L. (2020). A comparative evaluation of Wi-Fi RTT and GPS based positioning. In *Proceedings of the international global navigation satellite systems IGNSS 2020 conference, Sydney, Australia* (pp. 5–7).

Baker, G. R., Norton, P. G., Flintoft, V., Blais, R., Brown, A., Cox, J., et al. (2004). The Canadian adverse events study: the incidence of adverse events among hospital patients in Canada. *Cmaj*, *170*(11), 1678–1686.

Cagliano, A. C., Grimaldi, S., & Rafele, C. (2011). A systemic methodology for risk management in healthcare sector. *Safety Science*, *49*(5), 695–708.

Coronato, A., & Cuzzocrea, A. (2020). An innovative risk assessment methodology for medical information systems. *IEEE Transactions on Knowledge and Data Engineering*.

Coronato, A., Naeem, M., De Pietro, G., & Paragliola, G. (2020). Reinforcement learning for intelligent healthcare applications: A survey. *Artificial Intelligence in Medicine*, *109*, Article 101964. http://dx.doi.org/10.1016/j.artmed.2020.101964, URL: http://www.sciencedirect.com/science/article/pii/S093336572031229X.

Coronato, A., & Pietro, G. D. (2010). Formal specification of wireless and pervasive healthcare applications. *ACM Transactions on Embedded Computing Systems (TECS)*, *10*(1), 1–18.

Coronato, A., de Pietro, G., & Paragliola, G. (2013). A monitoring system enhanced by means of situation-awareness for cognitive impaired people. In *Proceedings of the 8th international conference on body area networks* (pp. 124–127). Brussels, BEL: ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), http://dx.doi.org/10.4108/icst.bodynets.2013.253551.

Deng, W., Ni, H., Liu, Y., Chen, H., & Zhao, H. (2022). An adaptive differential evolution algorithm based on belief space and generalized opposition-based learning for resource allocation. *Applied Soft Computing*, *127*, Article 109419.

Di Sarno, C., Formicola, V., Sicuranza, M., & Paragliola, G. (2013). Addressing security issues of electronic health record systems through enhanced SIEM technology. In *2013 international conference on availability, reliability and security* (pp. 646–653). http://dx.doi.org/10.1109/ARES.2013.85.

Ham, J.-H., Hwang, J.-B., Kim, J.-H., & Lee, G.-W. (2009). Effective customer risk management at the nuclear medicine department: Risk managemont MOT development application and producing public relations film. *The Korean Journal of Nuclear Medicine Technology*, *13*(3), 110–122.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 770–778). http://dx.doi.org/10.1109/CVPR.2016.90.

Khan, A. F., Jalil, A., Haq, I. U., & Shah, S. I. H. (2021). Automatic localization of macula and identification of macular degeneration in retinal fundus images. In *2021 international conference on electrical, communication, and computer engineering (ICECCE)* (pp. 1–6). IEEE.

Kuhn, A.-M., & Youngberg, B. J. (2002). The need for risk management to evolve to assure a culture of safety. *BMJ Quality & Safety*, *11*(2), 158–162.

Le, Q. V., et al. (2015). A tutorial on deep learning part 2: Autoencoders, convolutional neural networks and recurrent neural networks. *Google Brain*, *20*, 1–20.

Li, Z., Liu, F., Yang, W., Peng, S., & Zhou, J. (2021). A survey of convolutional neural networks: Analysis, applications, and prospects. *IEEE Transactions on Neural Networks and Learning Systems*, 1–21. http://dx.doi.org/10.1109/TNNLS.2021.3084827.

Li, X., Zhao, H., Yu, L., Chen, H., Deng, W., & Deng, W. (2022). Feature extraction using parameterized multi-synchrosqueezing transform. *IEEE Sensors Journal*.

Liu, C.-H., Hu, Y.-H., & Lin, Y.-H. (2021). A machine learning–based fall risk assessment model for inpatients. *CIN: Computers, Informatics, Nursing*, *39*(8), 450–459, doi:10.1097/cin.0000000000000727.

Lonceint, R., Bodéré, F., & Geffroy, B. (2019). Risk management through an "activity contradictions" lens: Exposure to low doses of radiation in nuclear medicine. *Frontiers in Medicine*, *6*, 228.

Mathew, A., Amudha, P., & Sivakumari, S. (2020). Deep learning techniques: An overview. In *International conference on advanced machine learning technologies and applications* (pp. 599–608). Springer.

Naeem, M., & Coronato, A. (2022). An AI-empowered home-infrastructure to minimize medication errors. *Journal of Sensor and Actuator Networks*, *11*(1), 13.

Naeem, M., Coronato, A., & Paragliola, G. (2019). Adaptive treatment assisting system for patients using machine learning. In *2019 sixth international conference on social networks analysis, management and security (SNAMS)* (pp. 460–465). http://dx.doi.org/10.1109/SNAMS.2019.8931857.

Naeem, M., Rizvi, S. T. H., & Coronato, A. (2020). A gentle introduction to reinforcement learning and its application in different fields. *IEEE Access*, *8*, 209320–209344. http://dx.doi.org/10.1109/ACCESS.2020.3038605.

Namba, T., & Yamada, Y. (2018). Fall risk reduction for the elderly by using mobile robots based on deep reinforcement learning. *Journal of the Robotics Networks and Artificial Life*, *4*(4), 265–269.

Paragliola, G., & Coronato, A. (2013). Intelligent monitoring of stereotyped motion disorders in case of children with autism. In *2013 9th international conference on intelligent environments* (pp. 258–261). http://dx.doi.org/10.1109/IE.2013.12.

Paragliola, G., Coronato, A., Naeem, M., & De Pietro, G. (2018). A reinforcement learning-based approach for the risk management of e-health environments: A case study. In *2018 14th international conference on signal-image technology & internet-based systems (SITIS)* (pp. 711–716). IEEE.

Paragliola, G., & Naeem, M. (2019). Risk management for nuclear medical department using reinforcement learning algorithms. *Journal of Reliable Intelligent Environments*, *5*(2), 105–113.

Retscher, G. (2020). Fundamental concepts and evolution of Wi-Fi user localization: An overview based on different case studies. *Sensors*, *20*(18), 5121.

Rummery, G. A., & Niranjan, M. (1994). *On-line Q-learning using connectionist systems, Vol. 37*. Citeseer.

Russell, Norvig, Canny, Malik, J. M., & Edwards, D. D. (2003). *Artificial intelligence: a modern approach, Vol. 2*. Upper Saddle River: Prentice Hall.

Sarker, I. H. (2021). Deep learning: A comprehensive overview on techniques, taxonomy, applications and research directions. *SN Computer Science*, *2*(6), 1–20.

Shah, S. I. H., & Coronato, A. (2021a). Inverse reinforcement learning through max-margin algorithm. In *Intelligent environments 2021: Workshop proceedings of the 17th international conference on intelligent environments, Vol. 29* (p. 190). IOS Press.

Shah, S. I. H., & Coronato, A. (2021b). Learning tasks in intelligent environments via inverse reinforcement learning. In *2021 17th international conference on intelligent environments (IE)* (pp. 1–4). IEEE.

Shah, S. I. H., Coronato, A., Naeem, M., & De Pietro, G. (2022). Learning and assessing optimal dynamic treatment regimes through cooperative imitation learning. *IEEE Access*, *10*, 78148–78158.

Shah, S. I. H., De Pietro, G., Paragliola, G., & Coronato, A. (2022). Projection based inverse reinforcement learning for the analysis of dynamic treatment regimes. *Applied Intelligence*, 1–13.

Sutton, R., & Barto, A. (1998). *Reinforcement learning: An introduction*. Cambridge MIT Press.

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning: an introduction*. Cambridge, MA, USA: A Bradford Book.

Thomas, E. J., Studdert, D. M., Burstin, H. R., Orav, E. J., Zeena, T., Williams, E. J., et al. (2000). Incidence and types of adverse events and negligent care in Utah and Colorado. *Medical Care*, 261–271.

VanLare, J. M., & Conway, P. H. (2012). Value-based purchasing—national programs to move from volume to value. *New England Journal of Medicine*, *367*(4), 292–295.

Vincent, C. (2011). *Patient safety*. John Wiley & Sons.

Wiboonrat, M. (2011). Risk management in healthcare services. In *Proceedings of the international conference on security and management (SAM)* (p. 1). The Steering Committee of The World Congress in Computer Science, Computer ....

Yamashita, R., Nishio, M., Do, R. K. G., & Togashi, K. (2018). Convolutional neural networks: an overview and application in radiology. *Insights Into Imaging*, *9*(4), 611–629.

Zhao, H., Liu, J., Chen, H., Chen, J., Li, Y., Xu, J., et al. (2022). Intelligent diagnosis using continuous wavelet transform and gauss convolutional deep belief network. *IEEE Transactions on Reliability*.