An Explainable and Formal Framework for Hypertension Monitoring using ECG and PPG

Abhinandan Panda, Ayush Anand, Srinivas Pinisetty, *Member IEEE* and Partha Roop, *Member IEEE*

***Abstract*—An alarming increase in hypertension is a hazard to global health that poses severe implications for the body's vital organs. To prevent serious repercussions, hypertension should be monitored continuously for early detection. It is well known that physiological signals, such as the photoplethysmogram (PPG) and electrocardiogram (ECG), carry essential information about the vitals of the human body. Considering this, numerous machine learning-based models based on ECG-PPG have been proposed for monitoring hypertension; however, such models are "black boxes" and lack clinical interpretation. This work proposes a formal method-based runtime verification approach for hypertension monitoring using ECG and PPG sensing, which is explainable. The pulse arrival time (PAT) feature extracted using both signals is employed to implement a decision tree to infer hypertension patterns/policies defined in PAT, based on which a runtime monitor is synthesized to classify hypertension. Using the MIMIC II dataset, the proposed scheme's performance is assessed, and the accuracy, sensitivity, and specificity are determined to be 95.7%, 93.9%, and 97.6%, respectively.**

***Index Terms*—Runtime verification, Hypertension, ECG, PPG, Pulse arrival time, Decision tree.**

I. Introduction

Blood pressure (BP) is the vital sign representing the pathological status of the human body. Hypertension (high blood pressure) poses a severe health threat, leading to cardiovascular diseases, which are the primary cause of global fatalities. This suggests the need for continuous monitoring of hypertension. The traditional methods of monitoring BP using auscultation and oscillometric techniques [6] require trained professionals and lack continuous monitoring.

Physiological signals such as ECG and PPG carry key information on the vitals of the body and have been widely studied to infer the BP. It has been found that pulse arrival time (PAT) (which is the time from the R-peak of ECG to the onset of PPG shown in Fig. 2) is a good indicator of BP [15]. Based on this, several deep learning models such as linear regression, decision tree, support vector machine, long short-term memory networks (LSTM), and restricted Boltzmann machine (RBM) models have been proposed [11], [29], [28], [33].

Although the deep learning-based models have demonstrated their efficacy in the hypertension classification tasks, they are regarded as "black boxes" as the internal process is unknown. This calls for the development of dependable, clinically understandable, and explainable monitoring techniques in healthcare [27], [10].

Based on formal methods, the runtime verification (RV) technique [23], [24], [4] monitors if a given set of system policies are satisfied during the execution of the system. In this approach, a monitor (RV monitor) is synthesized from the formalized specifications of the system to monitor the policies both offline and online. The RV monitor is correct-by-construction and lightweight [3], and it can be deployed on a wearable device as an explainable software component for continuous monitoring of the policies.

Recently, RV techniques have been proposed for monitoring healthcare applications such as ECG-PPG correlation [21], pacemakers [19], insulin infusion system [20], and diabetes detection [22].

A. Overview of the proposed approach

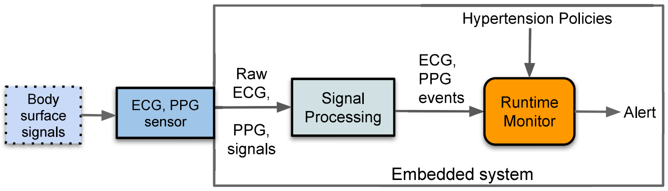

Fig. 1: Proposed hypertension monitoring approach using RV

As shown in Fig. 1, the electrical signals generated by a human heart are recorded by an ECG, and the PPG records the pulse signal. The characteristics of hypertension in pulse arrival time are then defined as policies. Key hypertension patterns are synthesized using a decision tree constructed from pulse arrival time features. Specifying the hypertension specifications as timed automata [1], RV monitors are synthesized automatically. RV allows for the real-time verification of these policies on a wearable device, issuing warnings in the case of hypertension.

The key contributions of this work are:

- This work presents formal runtime verification approach for hypertension monitoring using ECG and PPG.
- Synthesis of pulse arrival time policies pertained to hypertension.
- The proposed approach paves the way for designing a wearable monitor for hypertension monitoring.

The paper is structured as follows: an overview of the ECG, PPG, and BP, as well as the datasets, the signal processing methodology, and features used in this study, is presented in Section II. The extraction of policies for hypertension monitoring is discussed in Section III. The fundamentals of timed automata, as well as RV monitors, are discussed in Section IV, along with some examples. In Section V, we use an example involving BP monitoring to demonstrate the behavior of RV monitors. In Section VI, we discuss the performance of the suggested technique, followed by conclusion in Section VII.

Abhinandan Panda was with the School of Electrical and Computer Sciences, IIT Bhubaneswar, India.

Ayush Anand and Srinivas Pinisetty are currently with the School of Electrical and Computer Sciences, IIT Bhubaneswar, India, Email: a23cs09003@iitbbs.ac.in; spinisetty@iitbbs.ac.in

Partha Roop is currently with the Department of Electrical, Computer and Software Engineering, University of Auckland, New Zealand, Email: p.roop@auckland.ac.nz

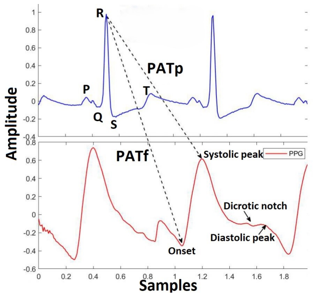

Fig. 2: ECG, PPG, and PAT

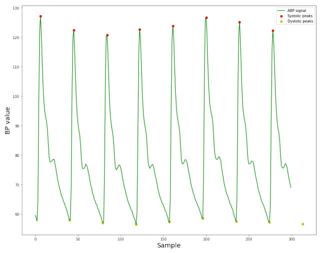

Fig. 3: ABP signals

II. Overview

This section briefly presents the basics of ECG, PPG, and BP, along with the considered dataset, processing of signals, and the features under study.

A. ECG, PPG, BP, and pulse arrival time

When the heart pumps blood, the force of the blood pressing against the walls of the arteries is known as blood pressure. The maximum and minimum blood pressures during successive heartbeats are known as systolic blood pressure (SBP) and diastolic blood pressure (DBP), respectively. An SBP of between 140 and 159 mmHg is interpreted as the first stage of hypertension, whereas an SBP greater than 159 mmHg is the second stage.

A typical ECG (the top signal in Fig. 2) represents the systole and diastole phases of the heart. The highest peak of ECG, known as the *R-peak* of ECG, represents ventricular contraction, as shown in Fig. 2. A PPG signal (the bottom signal in Fig. 2) indicates the variations in the blood flow (volume) during the systole phase of the heart. The peak of the PPG is known as the *systolic peak* and the foot is called the *onset* of the PPG. [1]

Pulse arrival time (PAT), i.e., the duration between the R-peak of the ECG and the arrival of the pulse wave, has been considered a predictive marker for arterial stiffness and cardiovascular diseases [12]. According to Chan et al. [5], the PAT and blood pressure are inversely related and have proposed a linear model to measure BP. In [25], Pon et al. discuss the relationship (nonlinear) between PAT and BP. According to Mukkamala et al. [16], PAT is inversely related to BP. In this study, we explore patterns from the pulse arrival time features that are indicators of hypertension.

a) Dataset considered: In this work, we study the Physionet MIMIC II dataset published by Kachuee et al. [11]. The dataset includes 942 patients' simultaneous ECG, PPG, and ABP (arterial blood pressure, BP signal) signals captured at 125 Hz. The subjects are of various ages and gender groups.

b) Signal processing: The ECG, PPG, and ABP signals are processed using the Neurokit2 tool [14] in Python. Using the Pan-Tompkins algorithm [18] and Elgendi's algorithm [8], the R-peaks of ECG and the systolic peak and onset of PPG are extracted. The SBP and DBP are computed from the peak and onset of the ABP signal, respectively, as shown in Fig. 3.

c) Features considered: In this work, we have considered the following pulse arrival time features:

- PAT_f interval (time interval between the R-peak of ECG and the onset of PPG)
- PAT_p interval (time interval between the R-peak of ECG and the systolic peak of PPG)

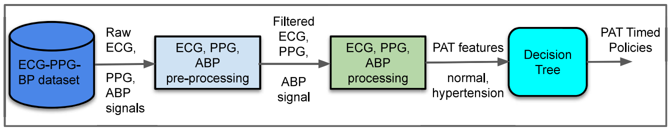

Fig. 4: Steps for policy mining

III. Hypertension Policies Mining

As discussed above, we extract PAT features and BP values from each synchronized cycle of ECG, PPG, and ABP signals. In order to extract patterns and policies on PAT, we implement a decision tree [26] that has been widely utilized in data mining to extract rules and patterns from datasets [17]. The tree is built from the PAT features PAT_f and PAT_p, where each feature is assigned a class, namely hypertension or normal based on the BP values. The work-flow for policy extraction is shown in Fig. 4. Once the tree is constructed, the PAT policies are deduced by going from the tree's root to its leaves. A Python module is implemented to construct the decision tree from the PAT features using the scikit-learn package of Python. The dataset and source code for generating the decision tree is available in [2].

a) Inference of policies from the decision tree: Classifying a feature (into hypertension or normal) is the result of following a path from the root of the tree to one of its leaf nodes. Every path that leads to a leaf of the decision tree corresponds to a production rule of the form $(X_1 \wedge X_2 \wedge \ldots) \rightarrow A$, where A is the class of the leaf. A condition X_i may be deleted from the tree if its deletion does not reduce the accuracy of the entire decision tree. X_i can also be deleted if its presence appears to enhance the accuracy of the tree, but this accuracy may be peculiar to the training set used in generating the decision tree. We therefore reduce the tree from 15 leaves to just 3 rules/policies [26]. The generated decision tree is also available for reference in [2].

The following patterns/policies are generated from the tree corresponding to hypertension.

- φ_{PAT1}: If PAT_f > 420 ms and $PAT_f \leq$ 468 ms, then it indicates hypertension.
- φ_{PAT2}: If PAT_f > 312 ms and $PAT_f \leq$ 420 ms and PAT_p > 556 ms and $PAT_p \leq$ 628 ms, then it indicates hypertension.
- φ_{PAT3}: When $PAT_f \leq$ 312 ms and PAT_p > 536 ms and $PAT_p \leq$ 620 ms, then hypertension is present.

Note: Regarding the correctness of the policies, the decision tree can be verified formally concerning the input-output mappings and violation ranges following the approaches mentioned in [31]. In addition, to verify the correctness of policies, we synthesize the policies from 70% of the dataset and evaluate them against the remaining 30% of the dataset detailed in Section VI.

[1] Enlarged version of figures 2, and 3 are available in the repository [2].

IV. Formalising Hypertension Policies as Timed Automata and the Runtime Verification Monitor

The sequential timed events in a pulse arrival time allow for formalizing the PAT policies discussed above as timed automata [1]. This section briefly discusses timed automaton to present a policy and explains the basics of a runtime verification monitor.

A. Timed Automata (TA)

Definition 1 (Timed automata): A *timed automaton* $\mathcal{A} = (S, s_0, C, \Sigma,\ \Delta, F)$ is a tuple, where: S is a finite set of *locations*, $s_0 \in S$ is the *initial location*, C is a finite set of *clocks*, Σ is a finite set of *events*, $\Delta \subseteq S \times \mathcal{G}(C), \Sigma \times 2^C \times S$ is the *transition relation*, $F \subseteq S$ is a set of *accepting locations*.

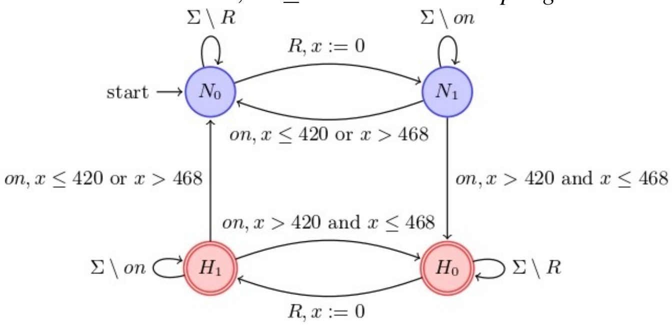

Fig. 5: Policy φ_{PAT1} represented by a timed automaton

Example 1: Consider the timed policy φ_{PAT1}: If $PAT_f >$ 420 ms and $PAT_f \leq$ 468 ms then it indicates hypertension.

The timed automata in Fig. 5 represents the policy φ_{PAT1}, where N_0, N_1 and H are the locations with N_0 as the initial location. Here, N_0 and N_1 are the locations indicating no hypertension. H_0 and H_1 are the accepting locations indicating that the policy φ_{PAT1} is satisfied (and hence indicating that hypertension is present). $\Sigma = \{R, on, sp\}$, represent the set of events, where R denotes the R-peak of the ECG and on, and sp, respectively, denote the beginning (onset) and systolic peak of the PPG. Here there is only one clock variable, x that measures the time between the events R of ECG and on of PPG. An operation on the clock variable initialized to 0 is known as a reset of the clock (here x:=0). Also, we can put constraints on the clocks called "guards"; here, $x \leq 420$ is a guard on the clock x.

The trace/timed word (σ) processed by the TA is a sequence of events along with time, for example, $\sigma = (e_1, t_1) \cdot (e_2, t_2) \cdots (e_n, t_n)$, where e_i is an event and t_i is the time of occurrence of the event. Given a finite alphabet Σ, the set of timed words over Σ is denoted by $tw(\Sigma)$.

B. Runtime verification (RV) monitor

Definition 2: Consider a monitoring policy $\varphi \subseteq tw(\Sigma)$ is formalized as a timed automata $\mathcal{A}_\varphi$, then the verification monitor synthesized from $\mathcal{A}_\varphi$ can be represented as a function $M_\varphi : tw(\Sigma) \rightarrow \mathcal{D}$, where $\mathcal{D} = \{c\_True, c\_False\}$. The RV monitor is defined as follows considering $\sigma \in tw(\Sigma)$ as the current observation:

$$M_\varphi(\sigma) \quad = \begin{cases} c\_True & \text{if} \quad \sigma \in \varphi \\ c\_False & \text{if} \quad \sigma \notin \varphi \end{cases}$$

The monitor M_φ for the policy φ takes σ as input and emits a verdict from the set $D = \{c\_True, c\_False\}$, where c_True stands for *currently true* and c_False for *currently false*. After reading the timed word σ, if the policy is satisfied with the current observation, the monitor emits the verdict c_True, otherwise, c_False.

V. Monitoring Hypertension

We follow the approaches described [23], [24], [4] to develop a RV monitor M_φ, given a timed automaton (TA) $\mathcal{A}_\varphi$ expressing the policy φ. The `RV_Monitor` module is implemented in Python 2.7. The framework parses the timed automata using the `UPPAAL DBM` libraries [7].

The PAT policies φ_{PAT1}, φ_{PAT2} & φ_{PAT3} provided in Section III are formally expressed as TA. The RV monitors Mφ_{PAT1}, Mφ_{PAT2} & Mφ_{PAT3} are synthesized for each policy, respectively. At each step, the RV monitors sense the generated ECG/PPG events, verify the policies, and provide verdicts. The final verdict is computed by merging the results of each monitor [2].

The following example illustrates the behavior of the RV monitor Mφ_{PAT1} for policy φ_{PAT1} using a sample ECG-PPG trace (other RV monitors show similar behavior).

Example 2: Consider the policy φ_{PAT1} discussed previously: *The time interval between R-peak of ECG and onset of PPG should be greater than 420 ms and less than or equal to 468 ms*. To monitor this policy, the RV monitor is input with the TA specifying policy φ_{PAT1} and an ECG-PPG trace σ. If the input trace σ satisfies the policy (indicating that hypertension is present), then the monitor will emit the verdict c_True. Else, the monitor will emit c_False (indicating that hypertension is not present).

Consider a sample ECG-PPG trace: $(R, 30) \cdot (on, 600) \cdot (sp, 750) \cdot (R, 800) \cdot (on, 1250) \cdot (sp, 1400) \cdot (R, 1450) \cdot (on, 1850)$, where R denotes the R-peak of the ECG and on and sp, respectively, denote the onset and systolic peak of the PPG. The RV monitor processes the events sequentially. For the experiment, we input the trace, where the event timestamp is the delay with the prior event or system start. Table I presents the output of the RV monitor for the sample trace.

The monitor reads the initial event R at time $t_1 = 30$ and outputs c_False as the policy is not satisfied at the moment. The monitor emits c_False for the event on at time $t_2 = 600$ (does not satisfy the policy as PAT_f interval is greater than 468 ms). At time $t_3 = 750$, the policy is not satisfied as the monitor gets the event sp, and emits c_False. Similarly, the monitor outputs c_False for the R event at time $t_4 = 800$. When the event on is observed at time $t_5 = 1250$, the policy is satisfied as the PAT_f interval falls between 420 and 468 ms, and the monitor emits c_True (indicating that hypertension is present). At time $t_6 = 1400$, the monitor reads sp, and emits the verdict c_True. On receiving R at time $t_7 = 1450$, the monitor emits c_True. When the monitor receives on at time $t_8 = 1850$, the policy φ_{PAT1} is violated, and the verdict by the monitor is c_False.

VI. Experimental Results

We evaluated the monitoring policies against a large dataset (derived from MIMIC II). Due to inconsistencies (like missing signals, signals being too short, high noise content, etc.) in the

[2] The final verdict is the ‘or’ computation of the three monitors. That is, the final monitor emits the verdict as *currently true*, only if at least one of the monitor emits *currently true*.

TABLE I: Sample behavior of RV monitor

σ	M$\varphi_{PAT1}(\sigma)$
$(R,30)$	CF
$(R,30)\cdot(on,600)$	CF
$(R,30)\cdot(on,600)\cdot(sp,750)$	CF
$(R,30)\cdot(on,600)\cdot(sp,750)\cdot(R,800)$	CF
$(R,30)\cdot(on,600)\cdot(sp,750)\cdot(R,800)\cdot(on,1250)$	CT
$(R,30)\cdot(on,600)\cdot(sp,750)\cdot(R,800)\cdot(on,1250)\cdot(sp,1400)$	CT
$(R,30)\cdot(on,600)\cdot(sp,750)\cdot(R,800)\cdot(on,1250)\cdot(sp,1400)\cdot(R,1450)$	CT
$(R,30)\cdot(on,600)\cdot(sp,750)\cdot(R,800)\cdot(on,1250)\cdot(sp,1400)\cdot(R,1450)\cdot(on,1850)$	CF

original data, the policies depicted 72% accuracy. However, for policy formation and evaluation, we have used a much smaller but consistent subset of the original dataset. This subset consisted of 288 data points extracted from the ECG, PPG and ABP signals of over 25 patients. Each data point consisted of the time stamp of three characteristic points (R_{peak} of ECG, $onset$ and $peak$ of PPG) and the SBP value. The proposed monitoring policies are inferred from 70% of the dataset and evaluated against the remaining 30% of the dataset. We calculate the following performance metrics: accuracy, sensitivity, and specificity defined as follows:

$$accuracy(\%) = (TP+TN)/(TP+TN+FP+FN)\times 100$$
$$sensitivity(\%) = [TP/(TP+FN)]\times 100$$
$$specificity(\%) = [TN/(TN+FP)]\times 100$$

where TP is true positive, TN is true negative, FP is false positive, and FN is false negative, respectively. The ability to differentiate between normal and hypertensive subjects denotes the accuracy of the RV framework. The proportion of hypertension samples that were correctly identified out of all the samples is represented by the RV framework's sensitivity. Similarly, specificity shows what percentage of overall samples the RV framework classifies as healthy ones. Results show that the RV framework has 95.7% accuracy, 93.9% sensitivity, and 97.6% specificity. Table II provides a performance comparison of our framework with a few other classification-based techniques.

TABLE II: Comparison with existing models

Authors	**Dataset**	**Features**	**Methods**	**Accuracy**
Zhang et al.(2018) [32]	own dataset	PTT, HR	Classification trees	90%
Tjahjadi et al. (2020) [30]	PPG-BP dataset	time-frequency	Bidirectional LSTM	97.33%
Fitriyani et al. (2019) [9]	Golinos	personal features	Ensemble Method	85.73%
Luo et al. (2018) [13]	MIMIC II	subject features	CNN	89.95%
Our work RV framework	MIMIC II	PAT	Policy based	95.7%

VII. Conclusion and Future Work

Hypertension is a common health issue that is life-threatening. We propose a formal runtime monitor for monitoring hypertension that is explainable. Such systems are quite useful for clinical interpretation because their internal workings are defined as white-box systems. The proposed approach shows comparable accuracy to the existing models.

Future work: The incorporation of new policies may improve the predictive capability of the framework concerning hypertension. The viability of the proposed monitoring system may be tested against other data sources.

References

[1] R Alur and DL Dill. A theory of timed automata. *Theoretical computer science*, 126(2):183–235, 1994.

[2] Anonymous. Supplementary repository, 2024. https://github.com/TimedProperties/BPMonitoringUsingECG-PPG.

[3] E Bartocci and Y Falcone. *Lectures on Runtime Verification*. Springer, 2018.

[4] Andreas Bauer, Martin Leucker, and Christian Schallhart. Runtime verification for LTL and TLTL. *ACM Trans. Softw. Eng. Methodol.*, 20(4):14:1–14:64, September 2011.

[5] KW Chan, K Hung, and YT Zhang. Noninvasive and cuffless measurements of blood pressure for telemedicine. In *2001 IEEE EMBS*, volume 4, pages 3592–3593. IEEE, 2001.

[6] S Daochai, W Sroykham, Y Kajornpredanon, and C Apaiwongse. Noninvasive blood pressure measurement: Auscultatory method versus oscillometric method. In *The 4th 2011 Biomedical Engineering International Conference*, pages 221–224. IEEE, 2012.

[7] Alexandre David. Uppaal dbm library programmer's reference, 2006.

[8] M Elgendi, I Norton, M Brearley, D Abbott, and D Schuurmans. Systolic peak detection in acceleration photoplethysmograms measured from emergency responders in tropical conditions. *PloS one*, 8(10):e76585, 2013.

[9] NL Fitriyani, M Syafrudin, G Alfian, and J Rhee. Development of disease prediction model based on ensemble learning approach for diabetes and hypertension. *IEEE Access*, 7:144777–144789, 2019.

[10] A Gastounioti and D Kontos. Is it time to get rid of black boxes and cultivate trust in ai? *Radiology: Artificial Intelligence*, 2(3), 2020.

[11] M Kachuee, MM Kiani, H Mohammadzade, and M Shabany. Cuffless blood pressure estimation algorithms for continuous health-care monitoring. *TBME*, 64(4):859–869, 2016.

[12] S Laurent et al. Aortic stiffness is an independent predictor of all-cause and cardiovascular mortality in hypertensive patients. *Hypertension*, 37(5):1236–1241, 2001.

[13] Y Luo, Y Li, Y Lu, S Lin, and X Liu. The prediction of hypertension based on convolution neural network. In *2018 IEEE ICCC*, pages 2122–2127. IEEE, 2018.

[14] D Makowski et al. Neurokit2: A python toolbox for neurophysiological signal processing. *Behavior research methods*, 53(4):1689–1696, 2021.

[15] R Mukkamala et al. Toward ubiquitous blood pressure monitoring via pulse transit time: theory and practice. *TBME*, 62(8):1879–1901, 2015.

[16] R Mukkamala and J Hahn. Toward ubiquitous blood pressure monitoring via pulse transit time: Predictions on maximum calibration period and acceptable error limits. *IEEE. Trans. Biomed. Eng.*, 65(6):1410–1420, 2017.

[17] S Nimmala, Y Ramadevi, R Sahith, and R Cheruku. High blood pressure prediction based on aaa++ using machine-learning algorithms. *Cogent Engineering*, 5(1):1497114, 2018.

[18] Jiapu Pan and Willis J Tompkins. A real-time qrs detection algorithm. *TBME*, (3):230–236, 1985.

[19] A Panda, S Pinisetty, and P Roop. Runtime verification of implantable medical devices using multiple physiological signals. In *ACM SAC*, pages 1837–1840, 2021.

[20] A Panda, S Pinisetty, and P Roop. A secure insulin infusion system using verification monitors. In *MEMOCODE*, pages 56–65, 2021.

[21] A Panda, S Pinisetty, and P Roop. A novel mapping of ecg and ppg to ensure the safety of health monitoring applications. *IEEE Embedded Systems Letters*, 2022.

[22] A Panda, S Pinisetty, and P Roop. Policy-based diabetes detection using formal runtime verification monitors. In *CBMS*, pages 333–338. IEEE, 2022.

[23] S Pinisetty, T Jéron, S Tripakis, Y Falcone, H Marchand, and V Preoteasa. Predictive runtime verification of timed properties. *Journal of Systems and Software*, 132:353–365, 2017.

[24] S Pinisetty, PS Roop, V Sawant, and G Schneider. Security of pacemakers using runtime verification. In *MEMOCODE*, pages 1–11. IEEE, 2018.

[25] CCY Poon and YT Zhang. Cuff-less and noninvasive measurements of arterial blood pressure by pulse transit time. In *2005 IEEE Annu. Int. Conf. IEEE Eng. Med. Biol. - Proc.*, pages 5877–5880. IEEE, 2006.

[26] JR Quinlan. Generating production rules from decision trees. In *ijcai*, volume 87, pages 304–307. Citeseer, 1987.

[27] Reyes et al. On the interpretability of artificial intelligence in radiology: challenges and opportunities. *Radiology: artificial intelligence*, 2(3), 2020.

[28] VR Ripoll and A Vellido. Blood pressure assessment with differential pulse transit time and deep learning: a proof of concept. *Kidney Diseases*, 5(1):23–27, 2019.

[29] Peng Su et al. Long-term blood pressure prediction with deep recurrent neural networks. In *2018 IEEE BHI*, pages 323–328. IEEE, 2018.

[30] H Tjahjadi, K Ramli, and H Murfi. Noninvasive classification of blood pressure based on photoplethysmography signals using bidirectional long short-term memory and time-frequency analysis. *IEEE Access*, 8:20735–20748, 2020.

[31] J Törnblom and S Nadjm-Tehrani. Formal verification of input-output mappings of tree ensembles. *Science of Computer Programming*, 194:102450, 2020.

[32] Bing Zhang et al. An empirical study on predicting blood pressure using classification and regression trees. *IEEE access*, 6:21758–21768, 2018.

[33] Bing Zhang et al. Predicting blood pressure from physiological index data using the svr algorithm. *BMC bioinformatics*, 20(1):1–15, 2019.