

Received November 4, 2020, accepted November 13, 2020, date of publication December 3, 2020, date of current version December 15, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3042342

Heath-PRIOR: An Intelligent Ensemble Architecture to Identify Risk Cases in Healthcare

FELIPE NEVES¹, FERNANDA CAMPOS¹, VICTOR STRÖELE¹, (Member, IEEE),
MIRIAM A. M. CAPRETZ², (Senior Member, IEEE), MORGAN JENNINGS³,
DIANNE BRYANT³, AND MARIO DANTAS¹

¹Department of Computer Science, Federal University of Juiz de Fora, Juiz de Fora 36036-900, Brazil

²Department of Electrical and Computer Engineering, Western University, London, ON N6A 5B9, Canada

³Department of Physical Therapy, Western University, London, ON N6A 5B9, Canada

Corresponding author: Victor Ströele (victor.stroele@ice.ufjf.br)

This work was supported in part by the Coordination for the Improvement of Higher Education Personnel (CAPES), in part by the Federal University of Juiz de Fora (UFJF), and in part by the Emerging Leaders in the Americas Program (ELAP) funded by the Government of Canada.

ABSTRACT Smart city environments, when applied to healthcare, improve the quality of people's lives, enabling, for instance, disease prediction and treatment monitoring. In medical settings, case prioritization is of great importance, with beneficial outcomes both in terms of patient health and physicians' daily work. Recommender systems are an alternative to automatically integrate the data generated in such environments with predictive models and recommend actions, content, or services. The data produced by smart devices are accurate and reliable for predictive and decision-making contexts. This study main purpose is to assist patients and doctors in the early detection of disease or prediction of postoperative worsening through constant monitoring. To achieve this objective, this study proposes an architecture for recommender systems applied to healthcare, which can prioritize emergency cases. The architecture brings an ensemble approach for prediction, which adopts multiple Machine Learning algorithms. The methodology used to carry out the study followed three steps. First, a systematic literature mapping, second, the construction and development of the architecture, and third, the evaluation through two case studies. The results demonstrated the feasibility of the proposal. The predictions are promising and adherent to the application context for accurate datasets with a low amount of noises or missing values.

INDEX TERMS Decision support systems, Internet of Things, machine learning, predictive models, medical conditions.

I. INTRODUCTION

Smart city environments, when applied to healthcare, improve the quality of people's lives, enabling, for instance, disease prediction and treatment monitoring. In medical settings, case prioritization is of great importance, with beneficial outcomes both in terms of patient health and physicians' daily work [1].

The data produced by the Internet of Things (IoT) devices can be integrated [2] with other systems, and Recommender Systems (RS) are an alternative to automatically integrate data with predictive models. Several studies have adopted predictive approaches to diagnosis and ways of treatment [3], risk and detection of chronic diseases [1], [4]–[6], detection

of the possibility of heart disease [7]–[9], and the use of medical notes to make predictions in health centers [10] and to estimate the likelihood of adverse events in postoperative cases [11], [12].

Predictive models aim to understand the factors around a context by using classes (classification models) or numbers (regression models) [13]. Those models are designed to solve specific problems. Depending on the data and context of the application, accuracy is impaired. Some approaches attempt to lessen the possibility of overfitting and prediction error by resorting to ensemble learning methods [14]. Ensemble methods combine multiple learning algorithms to get a more adherent and precise result [15].

Intending to assisting doctors with early diagnosis of diseases by constant monitoring, the present study proposes an architecture for a recommender system based on a Machine

The associate editor coordinating the review of this manuscript and approving it for publication was Xiaolong Li¹.

Learning ensemble model [16], [17], where the analysis of patient data, captured periodically by IoT devices, defines case prioritization [18].

The study primary research question is as follows: Can data produced by IoT devices be integrated into a Machine Learning ensemble model for early prediction of diseases to benefit patients's lives and doctors' daily work?

The objective is to make notifications tailored as per the patient's context and needs. As the core of the system, prediction techniques [19] can be applied by identifying the most opportune moment to provide a recommendation, based on severity and emergency level.

The proposed solution is a prediction architecture based on a Machine Learning ensemble model that can make personalized recommendations focused on patient needs in home environments. We developed an autonomous solution capable of synchronizing and managing various Machine Learning methods into an ensemble model to combine different learners and achieve the final result with higher accuracy [14]. It includes a pre-processing layer capable of cleaning and structuring the data coming from IoT devices. It pre-processes the data coming from IoT devices by removing noise values and replacing the missing ones. It also normalizes the data to prepare the dataset for use. Finally, we evaluated the designed architecture in two case studies involving real health environments.

The methodological process followed three main steps:

- (i) A Systematic Literature Mapping step to identify the state of the art of eHealth recommender system architectures and the most applied predictive models. Models, algorithms, techniques, stages, and flow to be followed were defined for the recommendation process.
- (ii) A Development step to design and build an architecture that supports multiple learning algorithms and prioritizes severity level cases.
- (iii) An Evaluation step to measure the efficiency of the proposed architecture in terms of adherence in the health domain and its accuracy in predictions.

Our main contributions can be summarized as follows:

- We apply an ensemble of predictive models to assist patients and doctors in the early detection of disease or prediction of postoperative worsening through constant monitoring.
- Using real data from two different health domains, scenarios, and datasets, Chronic Kidney Disease (CKD) and Wounds Healing Control, prioritizing emergency cases, we demonstrate the virtues of our parallel processing proposal over non-ensemble predictive models.
- The approach can help healthcare apply and develop better prioritizing emergency cases policies.

This paper is organized as follows: Section II describes the background. Section III presents related work on prediction in Recommender Systems within the context of healthcare and on users' diseases. Section IV describes the Health-PRIOR conceptual architecture and its Ensemble model. Section V

introduces the case studies in Chronic Kidney Disease and Wounds Healing Control, in addition to our proposed solutions. Finally, in Section VI, we summarize our contributions.

II. BACKGROUND

Regarding smart cities, Kitchin in 2014 [20] argues that "a city that monitors and integrates conditions of all of its critical infrastructure, including roads, bridges, tunnels, rails, subways, airports, seaports, communications, water, power, even major buildings, can better optimize its resources, plan its preventive maintenance activities, and monitor security aspects while maximizing services to its citizens". This definition fits our context because it shows the connection between public and private agents, working together to get a higher quality of services.

IoT devices can capture actions, temperature, location, among other information, and transmit such data to systems that consume them [21]. Primarily used in monitoring contexts, these systems will be widely used for health care in citizens' residences [22].

A typical IoT system consists of sensors, communication interfaces, advanced algorithms, and a cloud interface. These sensors are used to collect data from different devices and may be interconnected. Several types of algorithms, proprietary or not, are used to process data and analyze everything significant enough through APIs or applications.

Data production through these devices is quite fast and specific. The diversity of information thereof promotes systems that depend on sensitive data to act correctly. Among these systems are Recommender Systems. In eHealth, recommender systems have gained greater importance [2], for instance:

- Comprehensive: whether people use IoT for health, exercise, safety or beauty reasons, it has a holistic solution for everyone needs.
- Integration with different technologies: IoT in eHealth enables different technologies to work together seamlessly without concerning the complexity of technology integration.
- Big Data processing and analytics: IoT eHealth can effectively process, analyze, and manipulate the massive volume of multi-scale, distributed, and heterogeneous data sets produced by connected sensors. This allows extracting useful and reliable information from health data.
- Ability to personalize content or service: IoT data analytics can significantly expand the possibilities of meeting the need for personalized healthcare and treatments.
- Lifetime monitoring: patients can receive comprehensive additive data on their past, present, and future health by predicting diseases or worsening in their health status.

These characteristics, aggregated in a single database, enable analyses by intelligent models in order to work as a predictive basis of smart decision support systems. Intelligent wearable systems, according to [23], are already used to

monitor patients 24 hours a day, at home and outdoors, according to preventive medicine protocols. A monitoring system is connected to a technical service because the measured parameters are transmitted continuously or intermittently.

Recommender systems are intended for recommending content to users based on their profile and context, considering their preferences, needs, and interests [24], [25]. Identifying the semantic context when predicting an individual's preferences and needs is a challenge, which poses great difficulty in assertively identifying the semantic meaning of such needs and preferences. It is worth noting that the individual's context identification is the predictive basis itself [26].

Achieving greater precision through increased accuracy is another challenge related to prediction models [4]. The volume of data can influence results, and old data generates less accurate results and, when the amount of data is too large, distortions and missing values can influence the predictions [27].

Our Systematic Literature Mapping¹ revealed that a high number of studies use Machine Learning as predictive approaches. Variations of models have been addressed, such as Deep Learning and Support Vector Machines (SVM). Those models draw on data training to pursue predictions and analyze the data pattern to classify them into a group.

Other predictive approaches are also considered, whose predictive basis is similarity. Classical model-based approaches suit this predictive category, such as k-Nearest Neighbour (KNN), algebraic similarity, logistic regression, and Euclidean Distance [28]–[30]. Similarity can be pursued between items or individuals. That depends on the filtering model used by the Recommender System. Bayesian ranking and Fuzzy techniques appear in [30]–[32]. The method that uses Fuzzy logic is based on subdivision of sets, which treats groups of objects to analyze similarly.

The commonly used ensemble techniques are bagging, boosting, voting, and stacking due to their bias, variance, and prediction accuracy. For N. Mahendran *et al.* [33], ensemble learning is more robust than the individual learner algorithms because of its generalization ability.

Bagging (bootstrap aggregating) combines several classifiers trained on different subsamples of the dataset, where the training sets are randomly selected with replacement from the original instances. It is possible that several records may appear more than once as a result of resampling, while others may not be present in the training set. The disadvantage with the traditional bootstrap method is that training subsets produced by random selection with replacement are not especially concentrated on misclassified instances. Tuysuzoglu and Birant [34] propose a novel modified version of bagging, named enhanced bagging (eBagging), which uses a new bootstrapping method, referred to as prediction error-based bootstrapping (eBootstrapping).

Boosting is an iterative approach combining various weak learners, which results in low training error [33], [35]. It is performed by sequentially updating selected instances to the ensemble subspace by giving more weight to difficult examples, i.e., the most informative instances, which are not correctly classified in the previous steps Tuysuzoglu and Birant [34]. Weighted majority voting is applied as the combination rule for the ensemble outputs. It facilitates the reduction of otherwise stable learners' bias, such as univariate decision trees, also known as decision stumps or linear classifiers [34]. eBagging [34] generates training sets from the original dataset in parallel, so it is not an iterative approach as boosting, and it does not assign weight values to each instance as boosting; instead, all difficult examples are copied into all training sets.

Voting based methods operate in a way that depending on classifier choice, the ensemble then chooses the results that receive the largest total vote. "Under the condition that the classifier outputs are independent, it can be shown the majority voting combination will always lead to a performance improvement for a sufficiently large number of classifiers" [36]. Voting based methods represent a class of consensus methods.

According to Mahendran *et al.* [33] stacking model implements different lower-level learners and then combines them using a high-level meta-base learner and forms a stacking generalization model. This model prediction accuracy is superior to the individual classifiers, but it isn't easy to analyze theoretically.

Approaches such as ontology [3], neural networks [37] and predictions based on heuristics [38] are focused on modelling, using it as the system predictive core, trying to get new information through data [3], [38].

The most common method identified for the evaluation of accuracy is the Precision Metric one. Its calculation is simple and considers the index of correctness produced in the recommendation process. Recall and Mean Absolute Error (MAE) metrics are largely used as well.

The main challenges with ensemble classifiers in the field of machine learning are to select the base classifiers and to combine the outputs. Based on the Voting Method [36], we used the weighted average of the model results to define the final classification. We believe that at least half of the classifiers classify an instance correctly, which will minimize the difference between predictions and maximize assertiveness.

Our study focuses on combining the benefits of a smart city context with the power of recommender systems, aiming to improve assistance services offered to doctors and patients in the healthcare sector. We use predictive approaches to select patients who need attention based on their real-time health conditions.

III. RELATED WORK

Several related studies have dealt with models to predict healthcare and users' diseases. This section presents research related to this paper, highlighting the prediction systems,

¹www.ufjf.br/pgcc/dissertacoes/

focusing on Machine Learning and deep learning models, ensemble solutions, data caption, and recommendation. In the end, we made a comparative analysis among them and brought up this paper's contributions.

Lo Gullo *et al.* [1] research focuses on patients with locally advanced breast cancer undergoing neoadjuvant chemotherapy (NAC). Accurate prediction of treatment response has the potential to improve patient care by improving prognostication, enabling de-escalation of toxic treatment that has little benefit, facilitating upfront use of novel targeted therapies, and avoiding delays to surgery. The authors provide an overview of the machine learning and deep learning models applied to predict treatment response early in the course of or even before the start of NAC. They describe seven works, and despite the encouraging results, they conclude that the field of machine learning using multiparametric breast MRI for early prediction of NAC treatment response is still in its infancy. Most of the studies have been retrospective, single-institutional, and included relatively small numbers of patients, limiting the studies' statistical power and may compromise the generalizability of the results. Finally, they recognized that it is necessary to train models on extremely large datasets that are large and diverse enough to span the biological heterogeneity of the diseases and outcomes they seek to classify.

Bhardwaj and Hooda [16] proposed a Deux Machine Learning framework, implementing a double ensemble of Machine Learning algorithms for building an optimized and efficient solution to predict a complete pathological response of patients after Neoadjuvant Chemotherapy. The framework's performance is measured using a multi-criteria decision-making technique known as weighted simple additive weighting (WSAW). The WSAW total performance score is calculated by considering ten evaluation metrics, namely, Accuracy, Mean Absolute Error, Root Mean Square Error, TP Rate (rate of true positives), FP Rate (rate of false-positive), Precision, Recall, F-Measure (combined measure for precision and recall), MCC (a measure of the quality of binary classifications), and Receiver Operating Characteristics (ROC). They achieved an accuracy of 99.08%. The proposed system acts as a clinical support system and has an important future in prognostication and decision-making.

Yong-HongKuo *et al.* [18] apply the Machine Learning models to tackle the real-time and personalized waiting time prediction in emergency departments. They also introduce the concept of systems thinking to enhance the performance of the prediction models. Four popular algorithms were applied: stepwise multiple linear regression, artificial neural networks, support vector machines, and gradient boosting machines. A linear regression model served as a baseline model for comparison. The computational experiments were based on a dataset collected from an emergency department in Hong Kong. All four machine learning algorithms with the use of systems knowledge outperformed the baseline model. The stepwise multiple linear regression reduced the mean-square error by almost 15%. The other three algorithms

had similar performances, reducing the mean-square error by approximately 20%. Reductions of 17-22% in mean-square error due to the utilization of systems knowledge were observed. The authors concluded that Machine Learning models are more effective than linear regression models for predicting patient waiting time. The knowledge of the system effectively enhances the performance of the prediction models.

Ali *et al.* [17] proposed a healthcare monitoring framework based on the cloud environment and a big data analytics engine to store precisely and analyze healthcare data and improve classification accuracy. The proposed big data analytics engine is based on data mining techniques, ontologies, and bidirectional long short-term memory (Bi-LSTM). It was applied to patients' classification using their healthcare data related to diabetes, blood pressure (BP), mental health, and drug reviews. The framework integrates different information sources, such as smartphones, wearable sensors, medical records, and social networks. In addition, a big data cloud repository is utilized to store the extracted data, and MapReduce is applied to handle and process structured and unstructured data. The big data analytics engine is also used to accurately handle healthcare data containing inconsistencies that have missing values, noise, different formats, a large size, and high dimensionality. It uses artificial intelligence approaches to extract useful features from big data that eventually reduce data dimensionality.

Thong *et al.* in [30] proposed a hybrid model named HIFCF (Hybrid Intuitionistic Fuzzy Collaborative Filtering) using picture fuzzy clustering and intuitionistic fuzzy recommender system for medical diagnosis. The overall idea is to make an implication such as $R_{PS}, R_{SD} \rightarrow R_{PD}$, where PS represents the association between patient and symptom, and SD represents the association between symptom and disease. Patients are classified according to their relationships. The patients are grouped based on their information and medical history, and then they are fuzzified. The collaborative filtering is applied to identify the predicted disease.

A study conducted by Ali *et al.* in 2018 [3], focused on Type-2 fuzzy logic and fuzzy ontology to automate the overall process of foods and drugs recommendation for IoT-healthcare systems. The proposed architecture contained two main layers: a security layer and a Type-2 fuzzy ontology-based decision-making knowledge layer. The first layer manages access and prevents unauthorized access to a smart refrigerator and medical devices and investigates the real patient condition before the recommendation. The second layer extracts the patient's risk factor values via wearable sensors and determines the patient's health condition using Type-2 fuzzy logic. It is also responsible for retrieving drug and food information from the fuzzy ontology and recommends prescription for a smart medicine box and foods for a smart refrigerator, according to the patient's condition. These ontologies represent decision-making knowledge, i.e., foods and drugs more adherent to the patient's health condition.

It also retrieves the patient's personal information and disease history from the database.

Thanh *et al.* in 2017 [39] proposed a hybrid recommender system for medical diagnosis named Neutrosophic Recommender System. It is based on a neutrosophic set and neutrosophic clustering. A neutrosophic dataset has data characterized by a true, indeterminate, and false membership function. The authors propose a methodology with three main steps. First, they investigated a neutrosophic recommender system's characteristics with neutrosophic similarity measures that use algebraic operations and their theoretic properties. Second, the neutrosophic clustering method was implemented to identify the patient's neighbors who share common characteristics. Next, they established the prediction formula that used the neutrosophic algebraic similarity measure and the neighbors that resulted from the previous step. The historical data were used for training in the neutrosophication process, and the similarity through patient, symptom, and disease data were calculated separately. The clustering process was used to identify all patients who share characteristics and symptoms with others to group them. Finally, deneutrosophication was pursued to predict values to identify the disease.

Mustaqem *et al.* in 2017 [4], presented a hybrid prediction and recommendation model for the diagnosis and treatment of heart disease patients. Their main objective was to propose an intelligent and adaptive recommender system for patients that have different heart diseases. The recommendation objects are general medical recommendations about the disease identified by the system, depending on the type of disease, risk factor, probability of occurrence, and severity. The features are extracted from the clinical dataset, and their missing values are treated to handle outliers. The prediction model is applied to these features to select the most significant ones. After that, the selected features are used to predict the classes of common heart diseases. The authors used three different types of Machine Learning models to perform the classification (Support Vector Machine, Multi-Layer Perceptron, Random Forest). The knowledge base is addressed from the test dataset, identification of critical exposures, weight assigning, and dataset labeling. A medical expert addressed the ranges of value features that represent severity in risk cases. The recommendation process uses a combination of analysis results. A criterion is derived from the knowledge base, risk, and probability estimations, thus creating an inference-based rule set for generating recommendations.

Yuan *et al.* in 2018 [40], present an architecture for a socialized recommender system that uses the strength of relationships between users to pursue recommendations. It recommends healthcare services that are more adherent to users by using a deep learning approach. The architecture has three modules: the feature extraction one is responsible for extracting the features from relationships between users and the structure of these relationships. Such relationships are represented by a graph, where social theory is applied. The trust strength calculation is another module, where strength is calculated using the deep learning approach. The third

module is the rating prediction one, where services are selected and recommended to users. This model uses the extracted information to estimate the level of strength between users. The deep learning model has six neurons in the input layer, representing the extracted features, and two in the output layer representing the positive trust strength and the negative one, which at the end will be merged to produce the final prediction. The recommendation is performed by calculating the user's rating on a target item. It is done by considering the recommendations given by each recommender to an item, and the trust strength between each recommender and the active user. The model recommends healthcare services with the highest predicted rating score to the active user.

Focusing on the mixture of methods, Deng *et al.* in 2018 [41] proposed a Neural Gaussian Mixture Model (NGMM) for recommender systems. They constructed two parallel neural networks. One focuses on learning user preferences and the other on learning item properties. They use a Gaussian mixture model to model rating information. The proposed model operates to review textual information by applying NGMM. Prediction is made by imitating the rating behavior of users to items; i.e., the items that present a better rating are recommended. The model assumes that each rating is generated from a Gaussian mixture model, which models user preferences over factors of items. The authors designed a Gaussian layer on the neural network to simulate model parameters, mean, and mixture proportions to incorporate the Gaussian mixture model into the neural network.

A. COMPARATIVE ANALYSIS

Considering the prediction focus of the related work and their main application context, we can summarize the related work as: Lo Gullo *et al.* [1] made a literature review about patients with locally advanced breast cancer; Bhardwaj and Hooda [16] focused on predicting a complete pathological response of patients after Neoadjuvant Chemotherapy; Yong-HongKuo *et al.* [18] try to personalize waiting time prediction in emergency departments; Ali *et al.* [17] proposed a monitoring framework to diabetes; Thong *et al.* [30] and Thanh, *et al.* [39] study the prediction of diseases through the identification of patients who share characteristics and symptoms with others; Yuan *et al.* [40] recommend healthcare services; Ali *et al.* [3] automate the process of foods and drugs recommendation; Mustaqem *et al.* [4] work with Heart Disease. The variety of healthcare predictions show the importance of the research. We had the chance to evaluate our Health-PRIOR architecture and its Ensemble model in Chronic Kidney Disease, and Wounds Healing Control, prioritizing emergency cases.

Despite the number of works using machine learning and deep learning models in the proposed solution [1], [4], [16], [18], [30] there are many challenges in the healthcare area. Four algorithms were applied by Yong-HongKuo *et al.* [18]: stepwise multiple linear regression, artificial neural networks,

support vector machines, and gradient boosting machines. The results enhanced the performance of the prediction models. Mustaqeem *et al.* in 2017 [18] used classical Machine Learning models and statistical analysis to predict the type of heart disease. Ali *et al.* [17] framework uses artificial intelligence approaches to extract useful features from big data. Our proposal supports multiple learning algorithms and can prioritize cases by severity level. For predictions in real-time, we propose the ensemble model, composed of six models, in parallel.

Some approaches mix recommendation and prediction models as in [4] and [41] using techniques such as Fuzzy and Neural Networks. Mathematical approaches are addressed in the prediction method, as in [41], using Gaussian distribution. Thanh *et al.* in 2017 [39] use the clustering process to identify all patients who share characteristics and symptoms with others to group them. Thong *et al.* in 2015 [30] use fuzzy clustering to associate patients with symptoms and symptoms with diseases to classify patients according to their relationships. Different ensemble approaches and models usually perform better than when single and can generate more accurate results, which are extremely important for the healthcare area. The extraction and treatment of missing values and textual information are critical, as the data are directly associated with the final performance of the model.

Ali *et al.* in 2018 [3] proposed an architecture that uses the information produced by IoT devices aiming to assist patients in home environments. The application of data generated in real-time health systems demonstrates the need to develop models capable of processing information in a more agile way. In critical contexts, getting faster results can save lives and prevent disease development, where a factor that addresses that is the information and symptoms classification. Approaches as fuzzy logic [3], [30] and neutrosophic [39] sets of information can categorize, in most cases, the correct values. A proper classification of features can generate better predictions.

Thong *et al.* in 2015 [30] concluded that most Machine Learning models may fail to achieve high accuracy of prediction with real medical diagnosis datasets since the relations between patients and symptoms can be vague, uncertain, and imprecise. They reinforced, as did Deng *et al.* in 2018 [41], that the combination of methods, especially fuzzy sets and Machine Learning approaches can mitigate these issues.

Cloud environment and big data analytics can deal with the huge number of patients' data from different information sources. Big data analytics engine is also used to handle data containing inconsistencies, missing values, noise, different formats, a large size, and high dimensionality [17]. Our proposal is designed to collect data provided by IoT devices in real-time and historical data of patient's conditions and needs, e.g., forms, applications, and medical records can also be used to compose the patient's profile.

Most of the above cases were evaluated by means of information retrieval metrics, such as precision, recall, ROC, root-mean-square error (RMSE), MAE, among others. These

metrics can show how accurate the model is and how closely related the recommendation is to the reality of the user's interest and needs. Lo Gullo *et al.* [1] highlight that the studies from the review have been "retrospective, single-institutional, and have included relatively small numbers of patients". The use of large datasets can be a constraint in most experiments. Bhardwaj and Hooda [16] Machine Learning framework calculates the performance score by considering ten evaluation metric that achieves 99.08% accuracy. Having a large amount of data makes a good result possible. We evaluated the proposed architecture efficiency, in terms of adherence in the health domain and accuracy in predictions. To measure the assertiveness of the proposed model, we use classical statistical methods (RMSE and MAE) to evaluate the error and precision, recall, and F-measure measures to evaluate accuracy.

In general, the analyzed models do have applicability in the eHealth sector in terms of assisting in the prediction of diseases and ways of treatment. Different types of predictive approaches were addressed, especially Machine Learning ones. It is a viable alternative that promises results in prediction with recommendation accuracy.

Although the above mentioned studies considered the classification and identification of diseases, none of them focused on case prioritization. Our solution is designed to get information through IoT devices automatically, with the primary objective of proposing a recommender system capable of recommending actions to patients and doctors when patients require immediate attention based on their current health condition. Our proposal can be considered a clinical support system and has an important future in decision-making.

IV. HEATH-PRIOR ARCHITECTURE

Aiming to make recommendations for prioritizing patient cases in the healthcare domain, we designed an architecture dedicated to cases that need prioritization or special attention. The main goal is to use IoT data to compose the patients' profiles and periodically monitor their health status by predicting possible worsening of their overall condition.

The architecture follows the concepts of recommender systems architecture, and it is defined in five layers. The first layer is responsible for data extraction. The second layer is responsible for filtering the information, applying the most adherent filtering types. The third layer contains the predictive models responsible for predicting the resources and their adherence to users. It can be performed by memory-based, model-based, or by hybrid models, representing a combination of both. The resources are identified and chosen with the support of a fourth layer, represented by repositories, with resources, actions, or services that can be recommended to users. Finally, the recommendation layer, the fifth layer, is responsible to present the chosen resource, recommending the most adherent one to the user.

Based on the conceptual architecture, the Health-PRIOR architecture includes streaming data aspects, capturing and processing flows, as well as the prioritization approach,

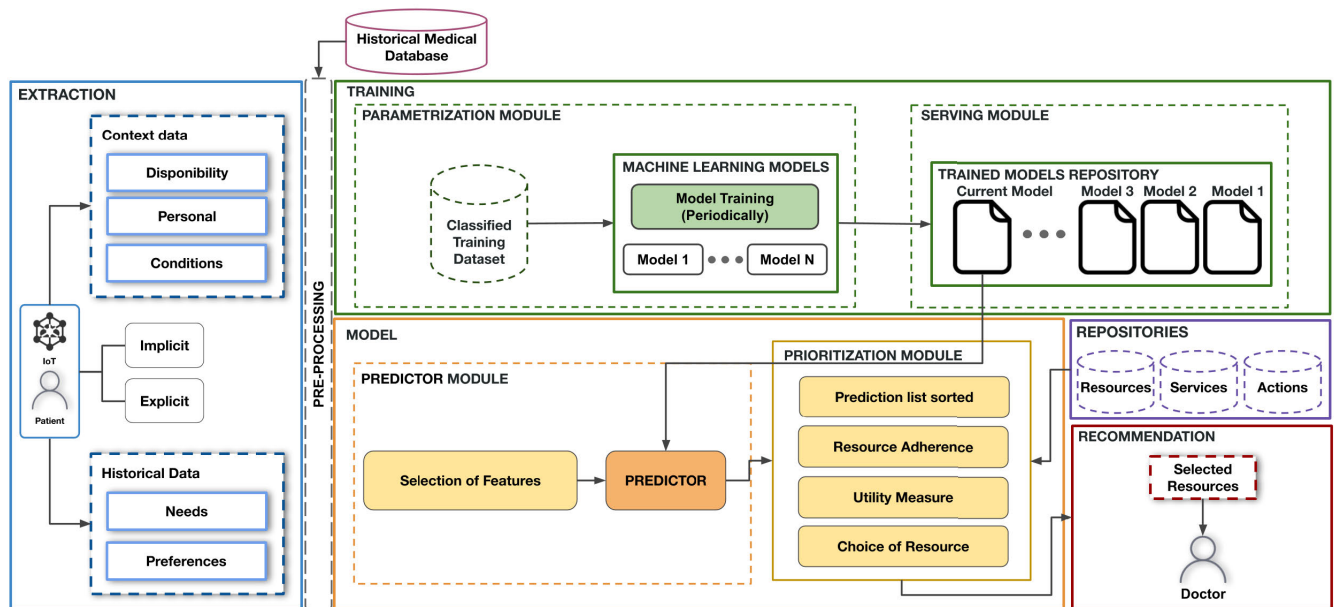


FIGURE 1. Heath-PRIOR conceptual architecture with the six main layers (Extraction, Pre-processing, Training, Model, Repositories, and Recommendation).

which is an important module for the context of the current research. Figure 1 shows the six main layers: Extraction, Pre-processing, Training, Model, Repositories, and Recommendation.

The **Extraction layer** is responsible for capturing information and configuring the user’s profile. The data can be provided by IoT devices connected to people or home environments, and from medical notes. Extraction can be performed implicitly or explicitly. The smart devices allow extraction to be performed in an automated way and be precise and accurate with the information.

The **Pre-processing layer** is responsible for filtering the data. It is where the data are treated by replacing missing values. It converts the raw data received from smart devices into useful data. It also removes noise to construct a cleaned version of the dataset.

The **Training layer** is where the model definition and training are executed. It captures the historical data periodically and trains a new set of models with newer data, providing a better result for the predictions. Predictive models are based on data training to make future predictions; they analyze which pattern the data follow to serve as a predictive basis. The layer captures the historical medical data periodically and trains a new set of models with new data. This layer has two modules: the Parameterization module, which is responsible for finding and setting the best parameters, as well as training the models periodically. The second module is represented by the Serving module, responsible for storing the trained models in a repository and making available the most recently trained model to the Model layer. The architecture considers the most recent historical information to train the models.

The **Model layer** is responsible for the training execution with the received pre-processed data in the **Predictor module**. The data are classified, and the patients are sorted by criticalness in a prioritization list.

The **Repositories layer** stores all recommendation objects, which include resources, actions, or services, depending on the final purpose of the recommender system.

Finally, the **Recommendation layer** presents the recommendation object to the users, allowing them to make a decision or an action. The historical dataset is refueled throughout time, allowing the model to be trained with newer data.

A. PREDICTION PROCESS

The Predictor module comprises classic Machine Learning models intended to work as an ensemble, capable of autonomously solving classification problems. It is responsible for predicting the resources and their adherence to users. Figure 2 shows the two main layers that contain the data flow process: the trained model and the predictor model itself.

The **Training Layer** is responsible for storing the classified data and training the models. Each Machine Learning model is trained with the same data, generating different predictors. Each model is tested with several parameters aiming to maximize its prediction approach. Once the best parameters are obtained, all models are stored in the Serving module. The Ensemble model is composed of classical Machine Learning models implemented as autonomous services, with reactivity, intelligence, and social characteristics.

The **Model Layer** is responsible for making predictions in real-time. The data received from IoT devices are pre-processed, and their features are selected and processed by the Ensemble model. The **Predictor Module** has a service

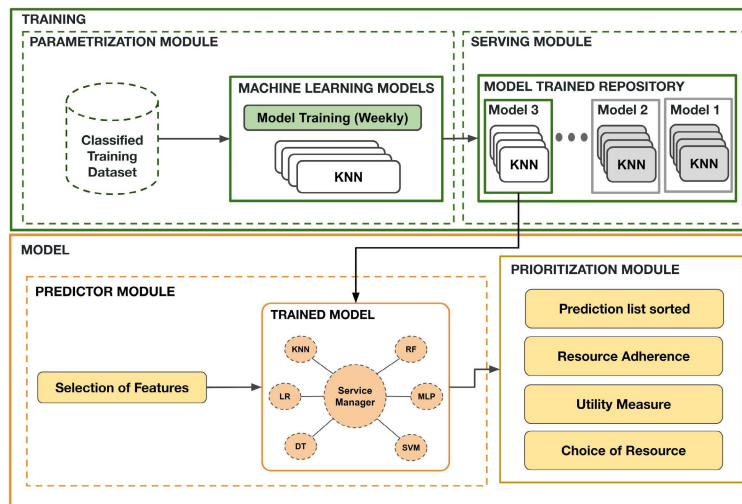


FIGURE 2. Health-PRIOR prediction process with the two main layers and the data flow.

manager that coordinates each model as a service. Aiming to synchronize all of them, the manager triggers all predictors in parallel to compute the input. Each model computes the selected features, and by the accuracy percentage, a Voting Ensemble Method [36] is applied to evaluate the results that most models have in common in order to get the final prediction, with higher certainty.

The **Prioritization Module** sorts the results and receives the recommendation objects that are most adherent to each prioritizing case. After that, it can present the selected notification to the patient, as well as to the doctor. This module can be parameterized to work based on the doctor’s concerns or the patient’s needs.

V. EVALUATION

Wohlin *et al.* in 2012 [42] defined a case study as an empirical investigation, which is based on different sources of evidence, used when the object of study is a contemporary phenomenon challenging to be studied in isolation. We will describe two case studies with different health domains, scenarios, and datasets. They followed three steps: planning, execution, and results.

A. CHRONIC KIDNEY DISEASE CASE STUDY

Chronic Kidney Disease has received increased attention from the international scientific community after studies showed its high prevalence. Non-appearance of symptoms in patients in the early stages of CKD requires physicians to maintain an adequate index of suspicion in all patients, especially in those with medical or sociodemographic risk factors for CKD [43].

1) PLANNING

This case study evaluates the Ensemble model in a chronic kidney disease context in a smart city environment. The focus

is to understand how prediction models can bring more health and quality of life to citizens in a smart environment. We evaluated aspects such as dataset, automated notifications, and training time. Figure 3 presents the proposed architecture with the data and the strategy devised for this specific context.

The extraction layer gets patients’ information such as blood information, historical data, and all the data that configure the profile. They are captured by IoT devices connected to patients. The Training layer considers only historical datasets that have the information and the classifications for each case. The training process was carried out once, as we had only one dataset for training, and all scenarios were based on the same configurations.

The **Data Transformation Module** was introduced to translate the signals captured by IoT devices into raw values, treat noise and missing values, and transfer the captured data to the system core. After that, the data are recognizable and formatted in a way that the predictive model can use them.

2) MODEL PARAMETERIZATION

Its responsibility is to process the transformed data and predict the patient’s possibility of developing CKD. Each Machine Learning model is intended to be an autonomous service capable of dealing with requests and pursuing predictions. We carried out empirical tests and followed some recommendations from the sklearn framework documentation [44]. The parameters were:

Decision Tree: From empirical tests, the maximum tree depth value adopted was 10, and the default values for the other parameters were maintained.

KNN: We set $K = 5$, which represents the number of neighbors to compare, as it is the most common value in literature. The type of distance was euclidean distance; for other parameters we maintained the default values.

SVM: We chose the sigmoid function as a kernel type because it showed better behavior in our previous tests.

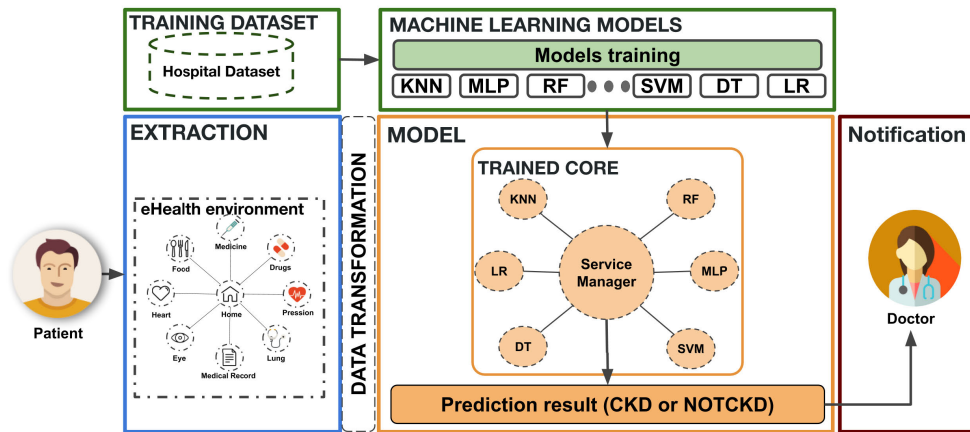


FIGURE 3. Chronic kidney disease architecture instantiated from the Health-PRIOR conceptual model.

Random Forest: The criterion of the function was entropy, as it is recommended in the framework documentation when one is interested in information gain [44].

Logistic Regression: We chose the liblinear solver to handle the data, which is recommended in the framework documentation [44].

Multilayer Perceptron: We used 3 hidden layers with 6 nodes each, the relu activation function, and one output layer for positive cases. The input layer has the number of features in the dataset, namely 24 features. This configuration showed better performance with the dataset used in our empirical tests.

3) DATA SOURCE

We used an open dataset [45], including the features that can characterize possible patients with CKD. The dataset enables the classification of a set of features that can be captured by IoT devices. We intended to simulate such behavior by analyzing the adherence of these features in the proposed model by sending them through HTTP requests.

Each file comprised all the attributes, a total of 25 features, such as “blood pressure”, “age”, “hemoglobin level”, “blood glucose”, and so on. These features were referred to as **ckd** (Chronic Kidney Disease) or **notckd** (not Chronic Kidney Disease).

The dataset contains a considerable amount of missing values. To treat them, we used *mean imputation*, the most common interpolation technique [46]. The missing values were replaced by the mean value of the entire feature column. The treatment of data was carried out by setting the noise values with the arbitrary value -1 and completing the missing ones with the mean of the column where it was located. The training process used all the features of the dataset, which were significant for the classification, as described in Table 1.

In Machine Learning solutions, it is necessary to define the training set and the test set to evaluate the model performance, i.e., a test sample is responsible for showing the results of predictions when the model is presented to unknown data. The

TABLE 1. Chronic kidney disease case study list of attributes.

| Attribute | Type | Granularity |
|-------------------------|-----------|---------------------------------|
| Age | numerical | year |
| Blood Pressure | numerical | mm/Hg |
| Specific Gravity | nominal | (1.005,1.010,1.015,1.020,1.025) |
| Albumin | nominal | (0,1,2,3,4,5) |
| Sugar | nominal | (0,1,2,3,4,5) |
| Red Blood Cells | nominal | (normal, abnormal) |
| Pus Cell | nominal | (normal, abnormal) |
| Pus Cell Clumps | nominal | (present notpresent) |
| Bacteria | nominal | (present notpresent) |
| Blood Glucose | numerical | mgs/dl |
| Blood Urea | numerical | mgs/dl |
| Sodium | numerical | mEq/L |
| Potassium | numerical | mEq/L |
| Hemoglobin | numerical | gms |
| Packed Cell Volume | numerical | - |
| White Blood Cell Count | numerical | cells/cumm |
| Red Blood Cell Count | numerical | millions/cmm |
| Hypertension | nominal | (yes, no) |
| Diabetes Mellitus | nominal | (yes, no) |
| Coronary Artery Disease | nominal | (yes, no) |
| Appetite | nominal | (good,poor) |
| Pedal Edema | nominal | (yes, no) |
| Anemia | nominal | (yes, no) |
| Class | nominal | (ckd,notckd) |

method consists of splitting the data from the open dataset into two files for training and testing purposes. As the original dataset has 400 instances, we split the data and selected 360 instances as our training set, and the other 40 instances were used as a test set, 20 for each classification. We built a python script that separated the instances randomly into two files. We did that 5,000 times.

As the Machine Learning models work with numerical data, the heterogeneous and categorical data were replaced by numerical ones. The algorithm was the Label Encoder from the *sklearn* framework [44], which gets all values and enumerates them, transforming them into numerical values.

The ensemble method used was the Voting Ensemble Method by averaging positive predictions. In that way, we evaluated the predictions that allow concluding that the patient can develop a given disease. This approach aims to

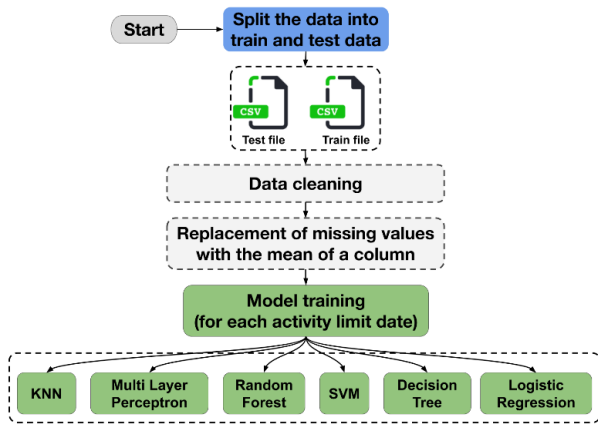


FIGURE 4. Chronic kidney disease case study training process.

minimize the difference between predictions and maximize assertiveness. The system model can handle data and offer a **Yes** or **No** output for case prioritization.

The ensemble averaging process considers only the prediction classes that are related to the recommendation trigger, which is the prior classification, in our case the *CKD* possibility. For a set of patient features, if the result indicates that the patient may develop a given disease, the system will recommend actions or services to the doctor in that particular context. This recommendation model was adopted since we are dealing with possible cases of chronic diseases, whose early diagnosis helps in treatment and prevention. Positive cases of recommendation can never be neglected.

To measure the proposed model’s assertiveness, we opted for classical statistical methods (RMSE and MAE) to evaluate the error and precision, recall, and F-measures to assess accuracy. We compared the predictive models together and separately.

The original weight for the F-measure equation is represented by $1 + \beta^2$, where β^2 configures the bias of each metric in the equation. The study by Zhao *et al.* in 2018 [47] defined $\beta^2 > 0$ as a balance factor between precision and recall, and, when $\beta^2 > 1$ the F-measure is biased in favor of the recall and, otherwise, the F-measure considers precision more than recall. In such a way, we chose value 2 in the F-measure metric to express an equal contribution of precision and recall [44].

The architecture model was implemented in Python by using Machine Learning frameworks such as Keras, TensorFlow, and Sklearn [44], [46].

4) EXECUTION

First, we clean the data and train the Machine Learning models that we use in our ensemble model (Figure 4).

With the data separated, we ran the Machine Learning services. We started the six main autonomous ones in parallel, training them with the same treated dataset. The architecture model is then ready to be evaluated.

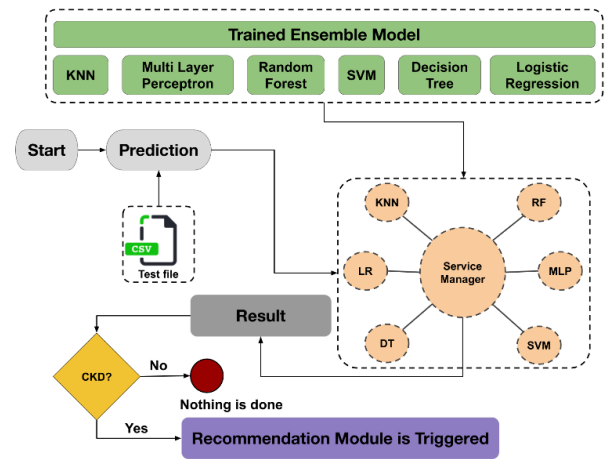


FIGURE 5. Chronic kidney disease case study test process.

As the data features were unbalanced, it was necessary to balance the number of instances to assure that the models behaved as expected. Aiming to reduce the disparity between the label classes, we chose the Synthetic Minority Over-sampling Technique (SMOTE), an over-sample technique to multiply the minority instances to balance the classes for training. This technique balances the number of instances for each classification.

After the balancing step, we sent the 40 test samples to the coordinator service, which is responsible for activating the Machine Learning models with the parameters. As the response time is impaired, it waits until all services are executed and combines the results by the voting ensemble method. If the prediction result is positive for the disease, the system triggers the recommendation module. Figure 5 presents the test process.

5) RESULTS

As we created 5,000 files to train the models with different sets of data, we evaluated the proposed model by averaging the results of those files.

We ran the experiments in a computer with Ubuntu 18.4 LTS; 8 GB RAM; Intel Core Processor i7-7500 and 220 GB SSD. After executing the experiment for each file, we obtained the average of each model result by going through all files. The results are presented in Table 2, and Figure 6 provides an illustrative chart.

In general, all models present a similar average, except for the SVM model. In that specific model, the average is lower in all metrics, if compared to the others. With high precision, none presents 100% accuracy, which is good, as this behavior configures our experiment with no over-fitting in any model.

The Random Forest model provided a smaller prediction error and higher accuracy in comparison with the others. However, if we analyze the general prediction context, there is a disproportion between the results for the same dataset. In other words, for the same instance there are distinct results,

TABLE 2. Chronic kidney disease case study ensemble and models results.

| | Precision | Recall | F-Measure | RMSE | MAE |
|---------------------|-------------|-------------|-------------|-------------|-------------|
| Ensemble | 0.981079543 | 0.937980000 | 0.958019757 | 0.175323319 | 0.040545000 |
| KNN | 0.932567667 | 0.899600000 | 0.913884369 | 0.278126508 | 0.084220000 |
| SVM | 0.794183031 | 0.185900000 | 0.292298362 | 0.656211922 | 0.432060000 |
| Decision Tree | 0.955836442 | 0.965720000 | 0.958913405 | 0.170613417 | 0.043010000 |
| Logistic Regression | 0.916640789 | 0.922890000 | 0.917935620 | 0.272555547 | 0.082520000 |
| Random Forest | 0.959745041 | 0.995760000 | 0.976501848 | 0.114523216 | 0.025185000 |
| MLP | 0.917695595 | 0.980430000 | 0.941969774 | 0.199103402 | 0.069630000 |

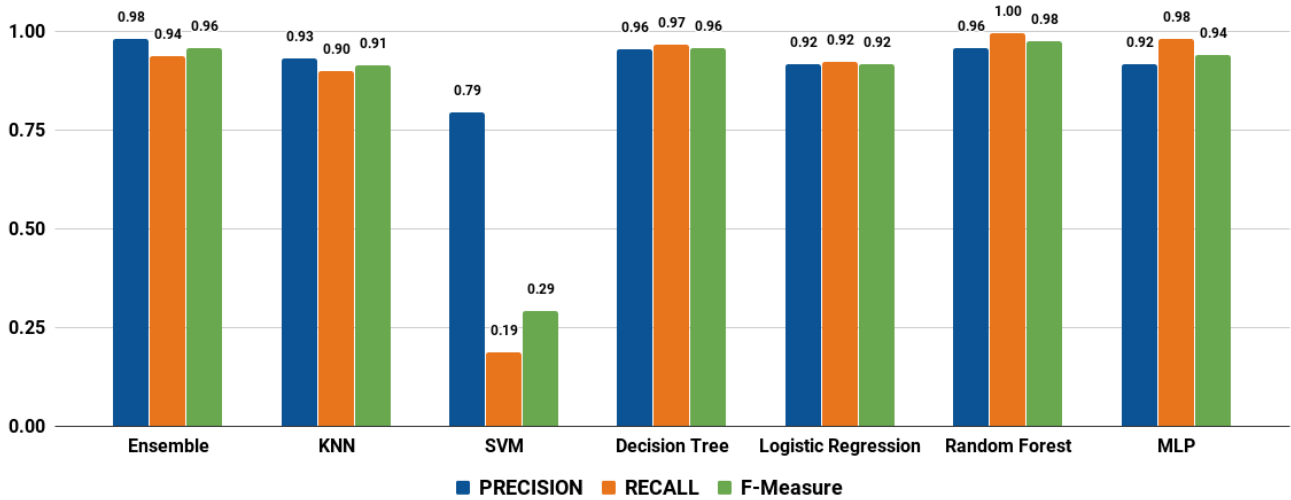


FIGURE 6. Chronic kidney disease case study ensemble and models graphic results.

leading to prediction errors, where one model correctly predicts one situation, and the other does not.

Figure 7 shows the error in each model and in our ensemble approach. Our model presents a low error rate of 4%, which is statistically accepted. Our ensemble model has an error slightly higher than that of Random Forest because it considers the results from all the others, and when a model predicts wrongly, our approach is penalized by that.

However, the ensemble model maintains high accuracy with minimal error and provides higher certainty by considering all the other models in its prediction. The results are promising when there is some disparity between predictive models, even though they are used in the same context and with the same data.

The way in which the models are trained and the configuration used for each one influence the processing time and the results. Figure 8 shows the training time for each model.

The Decision Tree model is the one that presents the fastest training for the training base, while the multi-layer perceptron (MLP) model presents the most time-consuming training. By comparing the training time and the result provided by the models, it can be seen that individually the model that presents the best cost-benefit ratio is the Decision Tree one. In general, the results are consistent with training time, except for the MLP model. As the training process of individual models is carried out in parallel, our proposal's training time

is equivalent to the training time of the slowest model, which is useful when a large number of samples is to be manipulated [46].

The proposal here allows that recommendation to take place by weighting the results of different models in order to inform doctors about case severity. This enables professionals to decide how to act, taking into account the information provided in the prediction and their expertise.

There is evidence that the proposed architecture and its model could be a viable alternative to bring higher levels of certainty in predictions results in health recommender systems, especially those that predict chronic diseases.

B. WOUNDS HEALING CONTROL CASE STUDY

Trying to minimize the flow of people at hospitals and clinics by prioritizing emergency cases, the e-Wounds Project (https://www.uwo.ca/fhs/pt/about/faculty/bryant_d.html) brings about the necessity of postoperative patients to receive notifications about their healing status. If any worsening is observed in their treatment, patients must be notified to go to the hospital/clinic to receive in-person care. The dataset is based on a patient-reported e-Visit questionnaire [12], a type of form to administer two- and six-week postoperative cases. Those patients have to come periodically to the clinic after surgery.

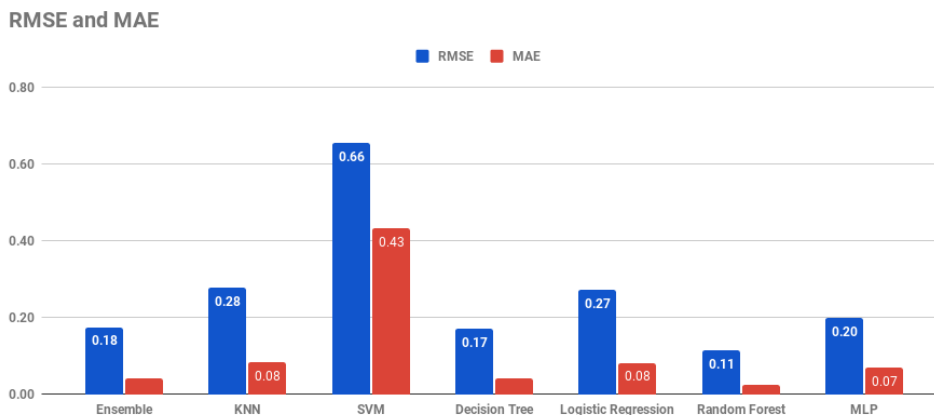


FIGURE 7. Chronic kidney disease case study ensemble and models errors measures.

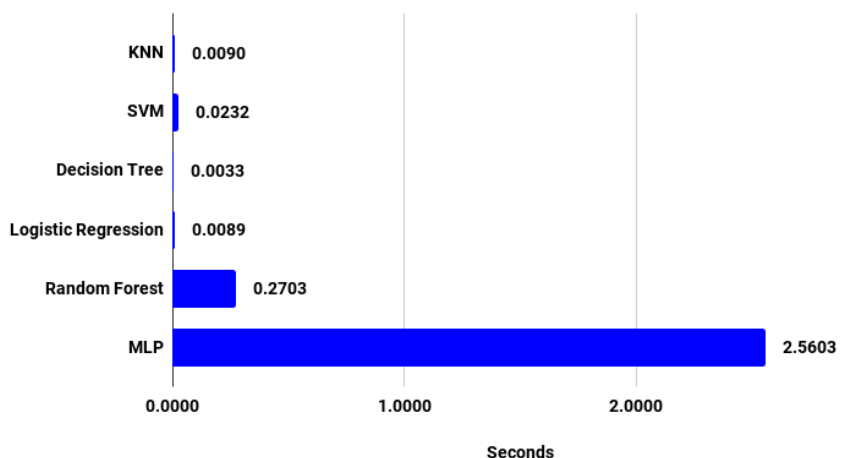


FIGURE 8. Chronic kidney disease case study models training time.

1) PLANNING

Our proposal can automatically provide notifications to the patients on whether they need to go to the clinic for their healing status. This particular case study focuses on testing our model’s capacity to generate accurate predictions and prioritize emergency cases correctly. With that, we aimed to evaluate the dataset, the parameterization of the Machine Learning models in question, and how we measured the case study. Figure 9 presents the E-Wound Architecture, named as the original project.

We captured information from all the IoT devices like a digital form with specific questions about patients’ post-operative statuses. If cases are prioritized, notifications are sent to patients and clinic/hospital as well. This process allows system self-feeding and provides real-time monitoring of patient notifications. The whole process is illustrated in Figure 10.

For this case study, we incorporated some techniques that reduce the dimensionality of the features. For this purpose,

we used an autoencoder to reduce the number of features in the dataset. An autoencoder algorithm [48] is part of a special family of dimensionality reduction methods, implemented using artificial neural networks. It aims to learn a compressed representation for input through minimizing its reconstruction error [14], [49]. The ability to learn from the “intrinsic data structure” is useful when the available data are noise and have too many missing values.

During training, the dataset is compressed by the encoder, and then the decoder reconstructs the data by minimizing reconstruction error. In this study, ReLU was used as the autoencoder activation function.

For this case study, we chose four classical Machine Learning models to combine in an ensemble model. We decreased the number of models for the predictions because two of them did not fit well into the available data. The set of models in the proposed architecture comprises those most adherent to the supervised classification problem. For our context and available datasets, the models that best performed models

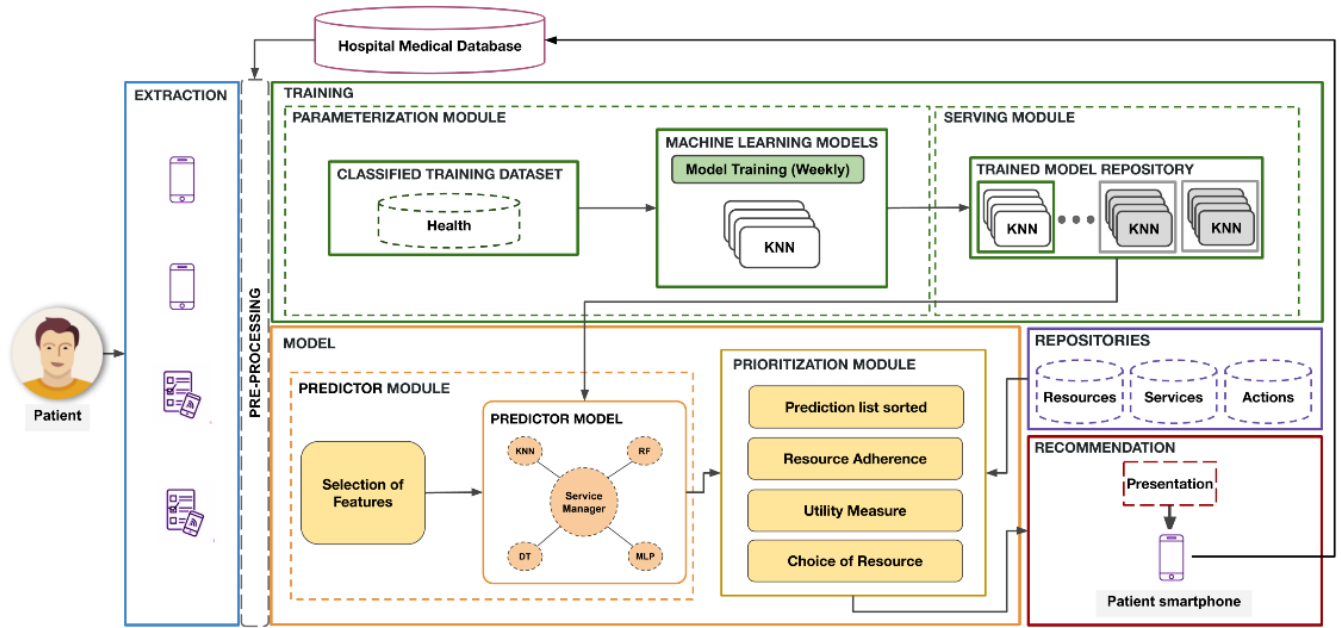


FIGURE 9. E-Wound architecture instantiated from the Health-PRIOR conceptual model.

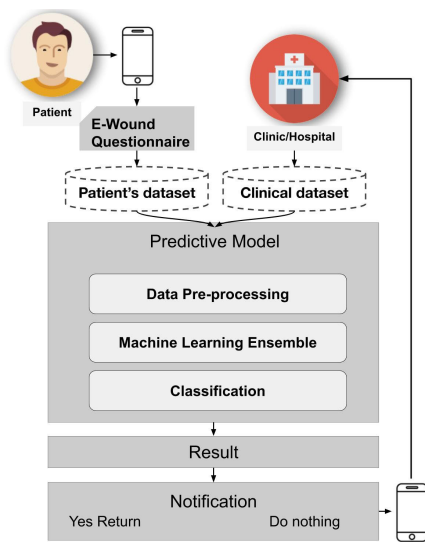


FIGURE 10. E-Wound case study predictive process.

were Decision Tree, KNN, Random Forest, and Multi-Layer Perceptron.

Based on the Voting Method [36], we used the weighted average of the model results to define the final classification for a patient’s context. Our goal was to reduce the flow of people at health centers, but without impairing the patients who need care/attention. To do so, we set the weights of the *not attention* classification to 1, and, in order to focus on the care needs of patients presenting some conditions, we set the weight for the *attention* classification to 2.

The strategy is described by Algorithm V-B1, where the input is a classification list containing the class result

Algorithm 1 Ensemble Strategy

Input: An array A of objects (Classification, intensity)

Output: An object (Classification, intensity)

```

1: NegSum = 0
   PosSum = 0
   WeightSum = 0
2: for  $i = 0$  to  $A.length$  do
3:   Prob =  $A[i].intensity$ 
   Class =  $A[i].classification$ 
4:   if ( $i = Classeqrequires\ attention$ ) then
5:     NegSum =  $NegSum + (1 - Prob)$ 
     PosSum =  $PosSum + 2 * Prob$ 
6:   end if
7: else
8:   NegSum =  $NegSum + Prob$ 
   PosSum =  $PosSum + 2 * (1 - Prob)$ 
   WeightSum =  $WeightSum + 1$ 
9: end for
  MediaPos =  $PosSum / 2 * WeightSum$ 
  MediaNeg =  $NegSum / WeightSum$ 
10: if ( $MediaNeg \leq MediaPos$ ) then
11:   return (requires attention, MediaPos)
12: end if
13: return (does not require attention, MediaNeg)

```

obtained by each classifier (negative - “do not require in-person attention,” positive - “need attention”) and the class intensity. The algorithm calculates both classes’ means and returns the final ensemble classification based on the positive and negative class means. As the input is the number of classifiers (n) involved in the process, the Algorithm 1 is $O(n)$.

The return value is an object with the final classification and its intensity. In case the **Attention** class is returned, patients are notified to go to the clinic/hospital where a doctor can assist them.

The evaluation metrics analyze model accuracy. In this regard, we highlight **sensitivity** or the true positive rate (TPR), which measures a model’s capacity to identify cases that need in-person care, and **specificity** or the true negative rate (TNR), which measures a model’s capacity to identify cases that do not need in-person care. The sensitivity and specificity were evaluated using Equations (1) and (2):

$$Sensitivity = TPR = \frac{TP}{P} \quad (1)$$

$$Specificity = TNR = \frac{TN}{N} \quad (2)$$

where **true positive** (TP) is the number of cases that need care, which is correctly identified as requiring in-person attention. **True negative** (TN) is the number of cases that do not need in-person care, which is correctly identified as not requiring attention. **P** is the total number of positive instances. **N** is the total number of negative instances.

2) EXECUTION

The dataset contains all the information related to the e-Visit questionnaire [12]. The questions are about the patients’ health condition, for instance, if they felt anything after the surgery, the prescribed medication, and symptoms that might have appeared or worsened in their daily lives. It is divided into two subsets, one before attending their two-week post-operative appointment and another before the six-week appointment.

In the present study, a total of 383 patients were asked to complete the e-Visit questionnaire with information of the abovementioned periods of two and six weeks. It is composed of yes/no questions as well as those that can be answered with “constant or intermittent” and “>3 days or <3 days”.

Each dataset has 49 features, which provide information about the patients and their answers to each question. In the first stage, we removed four features, two of them referring to patient identifiers, and the other two that had all the same values.

After the extraction process, we pre-processed the data by removing the duplicated answers and inconsistent cases. We also replaced the missing values by the mean of each column. It is crucial to pre-process the datasets because there are too many missing and noise values, which can impair the predictions if not treated. Figure 11 shows the training process.

Both datasets had the same number of instances (383); each of them corresponds to patient responses. In the Two Weeks dataset, 112 cases with inconsistencies unable to be fixed were removed. Among them, 31 had only the patient’s identifier. Seventy-five had only information about patients’ height and weight, but had no answers. And six were removed because, although we had the patient’s classification (‘need attention’ or ‘do not require in-person attention’), there was no information about the patients’ responses. At the end of this stage, the Two Weeks dataset had 271 instances.

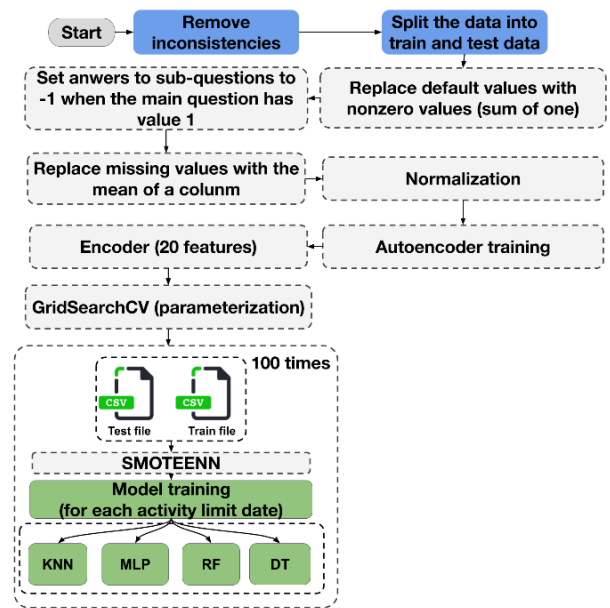


FIGURE 11. E-Wound case study pre-processing process.

We removed 82 instances from the Six Weeks dataset. Eighty had no information on the patients’ responses, and two were excluded because, although they were labeled, they had no information about the patient. This dataset had 301 instances at the end of this stage.

In order to better understand the datasets, some statistical measures were analyzed. Table 3 presents these analyzes for the numerical attributes: patient’s height and weight and pain intensity (before and after medication). As the patients are the same, demographic information has changed little. The difference in the values in the two forms is related to the difference in the number of instances. We can see that there is a reduction in the average pain value when comparing the responses of the second and sixth weeks. It is also possible to observe that there is a reduction in the average pain value after using the medication.

As the e-Visit questionnaire has a skip logic, we treated the sub-questions by setting their values to –1, where the main question has “no” as value. In this case, –1 indicates that the sub-questions were answered, meaning that the patient did not feel anything related to the main question.

The form has four main questions that, if filled with an affirmative answer, the patient needs to complete a set of sub-questions. Table 4 summarizes the number of answers given to the main questions, which are: Question 1 - Are you presently having pain in your knee or leg?; Question 2 - Have you experienced any issues with swelling in your knee or leg since your surgery?; Question 4 - Have you had any fluid leak from your incisions and/or arthroscopy portals in the last week?; Question 5 - Are you experiencing any illness (i.e., Fever or shaking chills), or have you experienced illness in the last three/four days ?. As a matter of space, we do not present the analysis of the derived questions.

TABLE 3. E-Wound case study two and six datasets basic measures.

| | Week 2 | | | | Week 6 | | | |
|-------|-------------|-------------|-------------|-----------------------|-------------|-------------|-------------|-----------------------|
| | Height (cm) | Weight (kg) | Pain (1-10) | Pain after medication | Height (cm) | Weight (kg) | Pain (1-10) | Pain after medication |
| Avg | 172 | 86.68 | 4.95 | 3.15 | 171.11 | 86.47 | 4.46 | 2.90 |
| Stdev | 14.41 | 22.91 | 2.05 | 1.95 | 14.27 | 21.31 | 1.91 | 1.81 |
| Max | 206 | 210 | 9.87 | 9 | 206 | 189 | 9 | 8.04 |
| Min | 54 | 44 | 1 | 0 | 54 | 44 | 1 | 0 |

TABLE 4. E-Wound case study questionnaire numeric answers analysis, considering two and six datasets.

| | Week 2 | | | | Week 6 | | | |
|-------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|--------------|
| | Q1 | Q2 | Q4 | Q5 | Q1 | Q2 | Q4 | Q5 |
| Yes | 190 (70.11%) | 168 (61.99%) | 26 (9.59%) | 15 (5.54%) | 173 (57.48%) | 185 (61.46%) | 7 (2.33%) | 10 (3.32%) |
| No | 81 (29.89%) | 102 (37.64%) | 243 (89.67%) | 256 (94.46%) | 128 (42.52%) | 116 (38.54%) | 293 (97.34%) | 290 (96.35%) |
| Empty | 0 | 1 (0.37%) | 2 (0.74%) | 0 | 0 | 0 | 1 (0.33%) | 1 (0.33%) |

It is observed that there is a large number of patients who responded that they feel pain and/or had swelling after surgery. However, few patients claim to have had more severe problems such as those questioned in questions 4 and 5. Also, few patients have failed to answer any questions, as shown in the table's last column.

Finally, we analyze the distribution of individuals by class to see if the dataset is balanced or not. In two weeks, we have 76.01% (206) of the instances classified as NO ('do not require in-person attention') and 23.99% (65) as YES ('need attention'). At six weeks, we have 68.11% (205) NO and 31.89% (99) YES. Thus, given the imbalance between classes, there is a need to use balancing techniques for the training and testing model process.

We applied the SMOTEENN, a combination of Edited Nearest Neighbours with SMOTE (Synthetic Minority Over-sampling Technique), an over-sample technique to multiply the minority instances to balance the classes for training. This technique balances the number of cases for each classification.

With the cleaned data, we normalized them by using the Normalizer provided by the Sklearn framework, and we trained an autoencoder structured with 10 hidden layers. ReLU was used as the activation function for all layers. After the autoencoder training, the Mean Square Error found was 0.003 for both datasets, which may be considered a good result for an autoencoder [50].

We performed empirical tests to set the resulting number of features using the autoencoder, ranging from 10 to 46. In our experiments, 20 features showed better performance for the available datasets. Therefore, the cleaned data were encoded, reducing the number of features from 46 to 20. The classifications remained untouched after the process, with 1 and 0 values for the *attention* and *not attention* classifications. By encoding the features to extract the essence of information, we achieved greater accuracy in the classifications.

Aiming to obtain the average of predictions, instead of selecting the best set of training instances, we randomized the selection of cases from the pre-processed datasets, splitting the cases into train and test sets. We randomly selected 80%

TABLE 5. E-Wound case study model parameterization.

| Model | Parameters | |
|---------------|--|--|
| | Two-week dataset | Six-week dataset |
| KNN | Algorithm: auto Metric: <i>euclidean</i> K: 2 | Algorithm: auto Metric: <i>euclidean</i> K: 3 |
| Decision tree | Criterion: <i>entropy</i> Max features: 6 Splitter: <i>best</i> Max depth: 8 | Criterion: <i>gini</i> Max features: 10 Splitter: <i>random</i> Max depth: none |
| MLP | # Hidden layers: 50 Activation func.: <i>relu</i> # Input: 20 # Output: 1 Batch size: 29 Learning rate: <i>constant</i> Warm start: true Solver: <i>lbfgs</i> | # Hidden layers: 50 Activation func.: <i>identity</i> # Input: 20 # Output: 1 Batch size: 12 Learning rate: <i>adaptive</i> Warm start: true Solver: <i>lbfgs</i> |
| Random Forest | Criterion: <i>gini</i> # Estimators: 11 Warm start: true Max features: 6 | Criterion: <i>entropy</i> # Estimators: 76 Warm start: true Max features: 9 |

of each dataset for training and 20% for testing. The process was executed 100 times.

In order to identify and select the most adherent parameters based on the pre-processed dataset, we used the Grid Search CV function from the Sklearn framework [44], [46]. The parametrization process was carried out by setting cross-validation with 10 folds based on (Stratified) KFold [44]. We also used the sklearn framework as a provider for each model. Each model parameterization is presented in Table 5.

For the two- and six-week datasets, we have applied the same process for training and test, which generated two sets of models. Each set was used for predicting the patients' care needs, considering the number of weeks since the surgery.

After the parametrization and pre-processing process, the two sets of models were sent to the coordinator service. As in the previous study, the results were combined with the voting ensemble method. The experiments were done on a server with Ubuntu 18.4 LTS; 94 GB RAM; Intel Xeon CPU E5-2630. The results are discussed in the following section.

TABLE 6. E-Wound case study Two and Six-Week results of eWound-PRIOR and models.

| Model | Two-week dataset | | Six-week dataset | |
|---------------------|----------------------------|----------------------------|----------------------------|----------------------------|
| | Sensitivity(Standard Dev.) | Specificity(Standard Dev.) | Sensitivity(Standard Dev.) | Specificity(Standard Dev.) |
| eWound-PRIOR | 0.730 (0.12) | 0.428 (0.08) | 0.728 (0.11) | 0.339 (0.08) |
| KNN | 0.497 (0.14) | 0.650 (0.08) | 0.561 (0.14) | 0.465 (0.12) |
| DT | 0.529 (0.17) | 0.582 (0.09) | 0.524 (0.13) | 0.527 (0.11) |
| MLP | 0.741 (0.20) | 0.344 (0.23) | 0.666 (0.14) | 0.507 (0.15) |
| RF | 0.526 (0.16) | 0.591 (0.09) | 0.521 (0.13) | 0.533 (0.09) |

3) RESULT

Table 6 shows the average results for the sensitivity and specificity metrics of each model and our ensemble for the two-week dataset and the results for the six-week dataset.

Sensitivity and specificity exist in a state of equilibrium [51]. The ability to correctly identify people who need special attention (sensitivity) usually reduces specificity (meaning more false positives). Likewise, high specificity generally implies a lower sensitivity (more false negatives). Still, high sensitivity is clearly important where the test is used to identify a severe but treatable disease. The model can identify people who need special care and, also, helps about 42% (two-week) and 33% (six-week) of patients from not seeking face-to-face care because the model detects that they don't need special care.

To understand how precise the results were from the outliers, we calculated the standard deviation for each metric.

Although the ensemble model presents the lowest value, the standard deviations in Table V-A5 shows that there is a significant variation in the results. As the datasets were based on patients' answers, this generates many missing values in the features of each case. In both datasets, the models had difficulty in classifying the cases correctly. Several classifications achieved around 60% of certainty in all models' predictions, which impair the ensemble's final result.

The questionnaire did not require the patients to answer all the sub-questions thereof. In this way, patients often skip a question that could be important for the classifications, creating missing values, impairing the models' training.

Based on the wrong predictions that should have been notified and were not, we observed that the questions related to body temperature and wound conditions were answered by most patients stating that there were no problems. This caused several dataset columns to have a missing data replacement, which impaired dimensionality reduction with the use of the autoencoder. If patients answered all questions without the skip logic, the models would have performed better by differentiating one case from another.

Following the analysis of each set of questions, most cases described redness/red streaks on the wound, and the incisions felt warm or hot to the touch. These symptoms would suggest that the patient needs to proceed to the clinic/hospital. However, the other answers configure the case with no priority due to the noise generated, allowing the models to classify the case wrongly.

Although we pre-processed the data, trying to reduce the noise and missing values, which also removes inconsistencies, all those described factors were limitations to this study.

As in the previous study case, we compared the results of the Machine Learning models separately and in the ensemble, according to our proposal. A skip logic questionnaire is not the ideal tool for prediction because it can generate a lot of missing data, as the individual is not required to answer all questions. In our experiments, the ensemble predictions were impaired by the number of missing values in the available dataset. We believe that more information about the patients is necessary, thus complementing the questionnaire. Perhaps the e-Wound Project will consider addressing these database constraints regarding this no linear sequence of answers.

We are concerned with the optimization of the dataset since each model could not predict several cases correctly. There is also the need to study and to improve the models' parameters, aiming at better performance for Project datasets.

C. CASE STUDIES FINAL CONSIDERATIONS

For evaluating the Health-PRIOR architecture, we used RMSE and MAE methods to evaluate the error and precision and recall and F-measures to assess accuracy. For the case study Chronic Kidney Disease, our ensemble results were: Precision 0.981079543, Recall 0.937980000, F-Measure 0.958019757, RMSE 0.175323319, and MAE 0.040545000. Bhardwaj and Hooda [16] used the k-fold cross-validation technique for the ten evaluation metrics and achieved an accuracy of 99.08%. In this way, we can say that Health-PRIOR reached a good result compared to the literature. Also, the use of a multi-criteria decision-making technique may favor the Health-PRIOR final result.

The results obtained in the e-Wounds project are incomparable because no other study was conducted in case prioritization or using the type of data used in this paper. According to Yong-HongKuo *et al.* [18], Machine Learning models are more effective than linear regression models for predicting patient waiting time. Our results also show that the great challenge is that each model must provide good accuracy in the predictions to ensure that the final ensemble result will be better than that of individual models. It takes time to test the models chosen and to compute strategies. We can reach certainty and assertiveness in the predictions through ensemble models, as proposed in this study. Having a proposal that uses parallel processing represents a gain in time and provides a good way to retrain models while pursuing predictions.

A medical appointment offers a standardized collection of data, which allows creating the profile and treatment history of each patient in health centers, with an expressive amount of daily data generated. If we obtain explicit information from IoT devices or even a questionnaire, the data can grow exponentially. This can be a constraint to update the training process if we use such data on a daily basis. Moreover, as the study deals with patient data, authorization is always required.

We had a chance to test large health datasets, and, as seen in the literature, recent data may contribute to higher assertiveness, even on a smaller scale. Having too many missing values in the dataset is unfavorable and may generate a wrong classification.

It is crucial to pre-process the datasets because they present too many missing and noise values, which can impair the predictions if not treated. It is important to assess the dataset size and its possible variations for training purposes.

VI. FINAL REMARKS

This study used IoT devices to design a recommender system capable of generating recommendations based on patients' characteristics to efficiently respond to risk cases and identify people requiring critical care.

This study main contribution is a recommender system architecture applied to healthcare, aimed at prioritizing emergency cases. It joins the data from IoT devices in assisted environments, with Machine Learning models' predictive power. An ensemble approach was adopted to maximize the accuracy of the proposal.

The results of each study showed that the dataset must have accurate information, where the classifications are well defined for a given context. A significant contribution was the assessment of the dataset size and its possible variations for training purposes. Aspects such as missing values or the lack of standards in the dataset can generate wrong predictions, thus causing wrong recommendations.

Smart devices in assisted environments are of great importance since they allow capturing information without disturbing the patient. So, this information is reliable for predictive and decision-making contexts. Although predictive approaches to health recommendation systems are not new in the literature, their use in smart city environments and emphasis on prioritizing medical cases is a contribution to the context.

A. LIMITATIONS

Regarding the use of predictive models, we realized that they should be parameterized to increase their predictive potential for each context. Considering the difficulty in the availability of healthcare data and regarding the use of predictive models for each context, one should be concerned with calibration in order to increase predictive potential. Classic models provide high performance, but approaches that combine these models stand out compared to individual models. Our proposal uses

ensemble methods, which allow greater assertiveness and confidence in each result [52].

Due to the context of the e-Wounds project and the development time, we could not evaluate the recommendation module with real recommendation objects. However, with the results, we believe that this module can provide relevant information for decision making by physicians in priority cases.

The dataset obtained from the e-Wounds project questionnaire brought us a challenge, given its structure and the number of missing data. We believe that, with a larger accurate dataset and relevant medical recommendation objects, the whole proposed architecture can fit properly in a physician's daily work. The proposed workflow of the architecture automatically helps to maintain the system self-sufficient.

B. FUTURE WORK

The recommendation module was not the focus of this paper, but we believe that this module can provide relevant information for physicians' decision-making in priority cases.

Our research's next steps also include real-time evaluations in predictions with patients under treatment by making recommendations of actions or services that could help in their healing process. We consider evaluating not only raw data but also images that could help in the diagnosis and predict worsening during treatment.

REFERENCES

- [1] R. Lo Gullo, S. Eskreis-Winkler, E. A. Morris, and K. Pinker, "Machine learning with multiparametric magnetic resonance imaging of the breast for early prediction of response to neoadjuvant chemotherapy," *Breast*, vol. 49, pp. 115–122, Feb. 2020, doi: [10.1016/j.breast.2019.11.009](https://doi.org/10.1016/j.breast.2019.11.009).
- [2] B. Farahani, F. Firouzi, V. Chang, M. Badaroglu, N. Constant, and K. Mankodiya, "Towards fog-driven IoT eHealth: Promises and challenges of IoT in medicine and healthcare," *Future Gener. Comput. Syst.*, vol. 78, pp. 659–676, Jan. 2018, doi: [10.1016/j.future.2017.04.036](https://doi.org/10.1016/j.future.2017.04.036).
- [3] F. Ali, S. M. R. Islam, D. Kwak, P. Khan, N. Ullah, S.-J. Yoo, and K. S. Kwak, "Type-2 fuzzy ontology-aided recommendation systems for IoT-based healthcare," *Comput. Commun.*, vol. 119, pp. 138–155, Apr. 2018, doi: [10.1016/j.comcom.2017.10.005](https://doi.org/10.1016/j.comcom.2017.10.005).
- [4] A. Mustaqeem, S. M. Anwar, A. R. Khan, and M. Majid, "A statistical analysis based recommender model for heart disease patients," *Int. J. Med. Informat.*, vol. 108, pp. 134–145, Dec. 2017, doi: [10.1016/j.ijmedinf.2017.10.008](https://doi.org/10.1016/j.ijmedinf.2017.10.008).
- [5] S. Rallapalli and R. R. Gondkar, "Big data ensemble clinical prediction for healthcare data by using deep learning model," *Int. J. Big Data Intell.*, vol. 5, no. 4, p. 258, 2018, doi: [10.1504/ijbdi.2018.094994](https://doi.org/10.1504/ijbdi.2018.094994).
- [6] O. Aouedi, M. A. B. Tobji, and A. Abraham, "An ensemble of deep auto-encoders for healthcare monitoring," in *Hybrid Intelligent Systems*. Cham, Switzerland: Springer, Mar. 2019, pp. 96–105, doi: [10.1007/978-3-030-14347-3_10](https://doi.org/10.1007/978-3-030-14347-3_10).
- [7] R. Lafta, J. Zhang, X. Tao, Y. Li, and V. S. Tseng, "An intelligent recommender system based on short-term risk prediction for heart disease patients," in *Proc. IEEE/WIC/ACM Int. Conf. Web Intell. Intell. Agent Technol. (WI-IAT)*, Dec. 2015, pp. 102–105, doi: [10.1109/wi-iat.2015.47](https://doi.org/10.1109/wi-iat.2015.47).
- [8] R. A. Kurian and K. Lakshmi, "An ensemble classifier for the prediction of heart disease," *Int. J. Sci. Res. Comput. Sci.*, vol. 3, no. 6, pp. 25–31, 2018.
- [9] S. Tuli, N. Basumatary, S. S. Gill, M. Kahani, R. C. Arya, G. S. Wander, and R. Buyya, "HealthFog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated IoT and fog computing environments," *Future Gener. Comput. Syst.*, vol. 104, pp. 187–200, Mar. 2020, doi: [10.1016/j.future.2019.10.043](https://doi.org/10.1016/j.future.2019.10.043).

- [10] H. D. P. dos Santos, A. H. D. P. S. Ulbrich, V. Woloszyn, and R. Vieira, "An initial investigation of the charlson comorbidity index regression based on clinical notes," in *Proc. IEEE 31st Int. Symp. Comput.-Based Med. Syst. (CBMS)*, Jun. 2018, pp. 6–11, doi: [10.1109/cbms.2018.00009](https://doi.org/10.1109/cbms.2018.00009).
- [11] K. Zhang, X. Liu, J. Jiang, W. Li, S. Wang, L. Liu, X. Zhou, and L. Wang, "Prediction of postoperative complications of pediatric cataract patients using data mining," *J. Transl. Med.*, vol. 17, no. 1, p. 2, Jan. 2019, doi: [10.1186/s12967-018-1758-2](https://doi.org/10.1186/s12967-018-1758-2).
- [12] W. G. Jeffery, "e-visits for early post-operative visits following orthopaedic surgery can they add efficiency without sacrificing effectiveness," M.S. thesis, Dept. Phys. Therapy, Univ. Western Ontario, London, ON, Canada, 2017. [Online]. Available: <https://ir.lib.uwo.ca/etd/5053>
- [13] S. Dreiseitl and L. Ohno-Machado, "Logistic regression and artificial neural network classification models: A methodology review," *J. Biomed. Informat.*, vol. 35, nos. 5–6, pp. 352–359, Oct. 2002, doi: [10.1016/s1532-0464\(03\)00034-0](https://doi.org/10.1016/s1532-0464(03)00034-0).
- [14] D. B. Araya, K. Grolinger, H. F. ElYamany, M. A. M. Capretz, and G. Bitsuamlak, "An ensemble learning framework for anomaly detection in building energy consumption," *Energy Buildings*, vol. 144, pp. 191–206, Jun. 2017, doi: [10.1016/j.enbuild.2017.02.058](https://doi.org/10.1016/j.enbuild.2017.02.058).
- [15] Z.-H. Zhou, *Ensemble Learning*. Boston, MA, USA: Springer, 2015, pp. 411–416.
- [16] R. Bhardwaj and N. Hooda, "Prediction of pathological complete response after neoadjuvant chemotherapy for breast cancer using ensemble machine learning," *Informat. Med. Unlocked*, vol. 16, Jan. 2019, Art. no. 100219, doi: [10.1016/j.imu.2019.100219](https://doi.org/10.1016/j.imu.2019.100219).
- [17] F. Ali, S. El-Sappagh, S. M. R. Islam, A. Ali, M. Attique, M. Imran, and K.-S. Kwak, "An intelligent healthcare monitoring framework using wearable sensors and social networking data," *Future Gener. Comput. Syst.*, vol. 114, pp. 23–43, Jan. 2021, doi: [10.1016/j.future.2020.07.047](https://doi.org/10.1016/j.future.2020.07.047).
- [18] Y.-H. Kuo, N. B. Chan, J. M. Y. Leung, H. Meng, A. M.-C. So, K. K. F. Tsoi, and C. A. Graham, "An integrated approach of machine learning and systems thinking for waiting time prediction in an emergency department," *Int. J. Med. Informat.*, vol. 139, Jul. 2020, Art. no. 104143, doi: [10.1016/j.ijmedinf.2020.104143](https://doi.org/10.1016/j.ijmedinf.2020.104143).
- [19] S. Yang, P. Zhou, K. Duan, M. S. Hossain, and M. F. Alhamid, "EmHealth: Towards emotion health through depression prediction and intelligent health recommender system," *Mobile Netw. Appl.*, vol. 23, no. 2, pp. 216–226, Sep. 2017, doi: [10.1007/s11036-017-0929-3](https://doi.org/10.1007/s11036-017-0929-3).
- [20] R. Kitchin, "The real-time city? Big data and smart urbanism," *GeoJournal*, vol. 79, no. 1, pp. 1–14, Feb. 2014, doi: [10.1007/s10708-013-9516-8](https://doi.org/10.1007/s10708-013-9516-8).
- [21] A.-M. Rahmani, N. K. Thanigaivelan, T. N. Gia, J. Granados, B. Negash, P. Liljeberg, and H. Tenhunen, "Smart e-Health gateway: Bringing intelligence to Internet-of-Things based ubiquitous healthcare systems," in *Proc. 12th Annu. IEEE Consum. Commun. Netw. Conf. (CCNC)*, Jan. 2015, pp. 826–834, doi: [10.1109/ccnc.2015.7158084](https://doi.org/10.1109/ccnc.2015.7158084).
- [22] C. E. Koop, R. Mosher, L. Kun, J. Geiling, E. Grigg, S. Long, C. Macedonia, R. Merrell, R. Satava, and J. Rosen, "Future delivery of health care: Cybercare," *IEEE Eng. Med. Biol. Mag.*, vol. 27, no. 6, pp. 29–38, Nov. 2008.
- [23] M. Chan, D. Estève, J.-Y. Fourmiols, C. Escriba, and E. Campo, "Smart wearable systems: Current status and future challenges," *Artif. Intell. Med.*, vol. 56, no. 3, pp. 137–156, Nov. 2012.
- [24] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Model. User-Adapted Interact.*, vol. 12, no. 4, pp. 331–370, Nov. 2002.
- [25] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutiérrez, "Recommender systems survey," *Knowl.-Based Syst.*, vol. 46, pp. 109–132, Jul. 2013.
- [26] S. Gao, G. Guo, Y. Lin, X. Zhang, Y. Liu, and Z. Wang, "Pair-wise preference over mixed-type item-sets based Bayesian personalized ranking for collaborative filtering," in *Proc. IEEE 15th Int. Conf. Dependable, Autonomic Secure Comput., 15th Int. Conf. Pervasive Intell. Comput., 3rd Int. Conf. Big Data Intell. Comput. Cyber Sci. Technol. Congr. (DASC/PiCom/DataCom/CyberSciTech)*, Nov. 2017, pp. 30–37, doi: [10.1109/dasc-picom-datacom-cybersci.2017.22](https://doi.org/10.1109/dasc-picom-datacom-cybersci.2017.22).
- [27] J. H. Chen, M. Alagappan, M. K. Goldstein, S. M. Asch, and R. B. Altman, "Decaying relevance of clinical data towards future decisions in data-driven inpatient clinical order sets," *Int. J. Med. Informat.*, vol. 102, pp. 71–79, Jun. 2017, doi: [10.1016/j.ijmedinf.2017.03.006](https://doi.org/10.1016/j.ijmedinf.2017.03.006).
- [28] S. Hassan and Z. Syed, "From Netflix to heart attacks: Collaborative filtering in medical datasets," in *Proc. ACM Int. Conf. Health Informat. (IHI)*, 2010, pp. 128–134, doi: [10.1145/1882992.1883012](https://doi.org/10.1145/1882992.1883012).
- [29] R. J. Adams, R. S. Sadasivam, K. Balakrishnan, R. L. Kinney, T. K. Houston, and B. M. Marlin, "PERSPeCT: Collaborative filtering for tailored health communications," in *Proc. 8th ACM Conf. Rec. Syst. (RecSys)*, 2014, pp. 329–332, doi: [10.1145/2645710.2645768](https://doi.org/10.1145/2645710.2645768).
- [30] N. T. Thong and L. H. Son, "HIFCF: An effective hybrid model between picture fuzzy clustering and intuitionistic fuzzy recommender systems for medical diagnosis," *Expert Syst. Appl.*, vol. 42, no. 7, pp. 3682–3701, May 2015, doi: [10.1016/j.eswa.2014.12.042](https://doi.org/10.1016/j.eswa.2014.12.042).
- [31] L. H. Son and N. T. Thong, "Intuitionistic fuzzy recommender systems: An effective tool for medical diagnosis," *Knowl.-Based Syst.*, vol. 74, pp. 133–150, Jan. 2015, doi: [10.1016/j.knsys.2014.11.012](https://doi.org/10.1016/j.knsys.2014.11.012).
- [32] P. Dhanalakshmi, K. Ramani, and B. E. Reddy, "An improved rank based disease prediction using Web navigation patterns on bio-medical databases," *Future Comput. Informat. J.*, vol. 2, no. 2, pp. 133–147, Dec. 2017, doi: [10.1016/j.fcij.2017.10.003](https://doi.org/10.1016/j.fcij.2017.10.003).
- [33] N. Mahendran, P. M. D. R. Vincent, K. Srinivasan, V. Sharma, and D. K. Jayakody, "Realizing a stacking generalization model to improve the prediction accuracy of major depressive disorder in adults," *IEEE Access*, vol. 8, pp. 49509–49522, 2020, doi: [10.1109/access.2020.2977887](https://doi.org/10.1109/access.2020.2977887).
- [34] G. Tuysuzoglu and D. Birant, "Enhanced bagging (eBagging): A novel approach for ensemble learning," *Int. Arab J. Inf. Technol.*, vol. 17, no. 4, pp. 515–528, Jul. 2020, doi: [10.34028/iajit/17/4/10](https://doi.org/10.34028/iajit/17/4/10).
- [35] H. R. Kadkhodaei, A. M. E. Moghadam, and M. Dehghan, "HBoost: A heterogeneous ensemble classifier based on the boosting method and entropy measurement," *Expert Syst. Appl.*, vol. 157, Nov. 2020, Art. no. 113482, doi: [10.1016/j.eswa.2020.113482](https://doi.org/10.1016/j.eswa.2020.113482).
- [36] R. Polikar, *Ensemble Machine Learning: Methods and Applications*. Boston, MA, USA: Springer, 2012, pp. 1–34.
- [37] E. O. Aboagye, G. C. James, and R. Kumar, "Evaluating the performance of deep neural networks for health decision making," *Procedia Comput. Sci.*, vol. 131, pp. 866–872, Jan. 2018.
- [38] R. Lafta, J. Zhang, X. Tao, Y. Li, V. S. Tseng, Y. Luo, and F. Chen, "An intelligent recommender system based on predictive analysis in telehealthcare environment," *Web Intell.*, vol. 14, no. 4, pp. 325–336, Nov. 2016.
- [39] N. Dang Thanh, L. H. Son, and M. Ali, "Neutrosophic recommender system for medical diagnosis based on algebraic similarity measure and clustering," in *Proc. IEEE Int. Conf. Fuzzy Syst. (FUZZ-IEEE)*, Jul. 2017, pp. 1–6, doi: [10.1109/fuzz-ieee.2017.8015387](https://doi.org/10.1109/fuzz-ieee.2017.8015387).
- [40] W. Yuan, C. Li, D. Guan, G. Han, and A. M. Khattak, "Socialized healthcare service recommendation using deep learning," *Neural Comput. Appl.*, vol. 30, no. 7, pp. 2071–2082, Oct. 2018.
- [41] D. Deng, L. Jing, J. Yu, S. Sun, and H. Zhou, "Neural Gaussian mixture model for review-based rating prediction," in *Proc. 12th ACM Conf. Rec. Syst.*, Sep. 2018, pp. 113–121.
- [42] C. Wohlin, P. Runeson, M. Höst, M. C. Ohlsson, B. Regnell, and A. Wesslén, *Experimentation in Software Engineering*. Berlin, Germany: Springer, 2012.
- [43] M. G. Bastos and G. M. Kirsztajn, "Chronic kidney disease: Importance of early diagnosis, immediate referral and structured interdisciplinary approach to improve outcomes in patients not yet on dialysis," *Brazilian J. Nephrol.*, vol. 33, no. 1, pp. 93–108, 2011.
- [44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, and D. Cournapeau, "Scikit-learn: Machine learning in Python," *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, Oct. 2011.
- [45] D. Dua and C. Graff. (2017). *UCI Machine Learning Repository*. [Online]. Available: <http://archive.ics.uci.edu/ml>
- [46] S. Raschka, *Python Machine Learning*. Birmingham, U.K.: Packt, 2015.
- [47] K. Zhao, S. Gao, W. Wang, and M.-M. Cheng, "Optimizing the F-measure for threshold-free salient object detection," 2018, *arXiv:1805.07567*. [Online]. Available: <http://arxiv.org/abs/1805.07567>
- [48] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning internal representations by error propagation," Inst. Cogn. Sci., California Univ. San Diego, La Jolla, CA, USA, Tech. Rep. ICS 8506, 1985.
- [49] W. Wang, Y. Huang, Y. Wang, and L. Wang, "Generalized autoencoder: A neural network framework for dimensionality reduction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshops*, Jun. 2014, pp. 490–497.
- [50] C. C. Tan and C. Eswaran, "Using autoencoders for mammogram compression," *J. Med. Syst.*, vol. 35, no. 1, pp. 49–58, Feb. 2011.
- [51] A. G. Lalkhen and A. McCluskey, "Clinical tests: Sensitivity and specificity," *Continuing Educ. Anaesthesia, Crit. Care Pain*, vol. 8, no. 6, pp. 221–223, Dec. 2008.
- [52] F. Braz, F. Campos, V. Stroele, and M. Dantas, "An early warning model for school dropout: A case study in e-learning class," in *Proc. Brazilian Symp. Comput. Educ., Simpósio Brasileiro de Informática na Educação-SBIE*, 2019, vol. 30, no. 1, p. 1441, doi: [10.5753/cbie.sbie.2019.1441](https://doi.org/10.5753/cbie.sbie.2019.1441).



include machine learning, social networks, and recommender systems.



member of the Master Program of Computer Science. She has experience in computer science, with emphasis on software engineering, specifically in the following topics: e-Learning, recommender systems, ecosystems, and e-Health.



working mainly on the following themes: clustering algorithm, social network analysis, recommender systems, and streaming data.



data, machine learning, service-oriented architecture, privacy, and security.



within healthcare and rehabilitation models.



development/evaluation of measurement instruments/quality of life.



and high-performance environments. He won 29 national and international awards related to teaching and researching.

• • •