UNIVERSIDADE FEDERAL DE JUIZ DE FORA
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Felipe Neves Braz

Health-PRIOR: An Intelligent Recommender System Architecture for Case Prioritization in Healthcare

Juiz de Fora

2020

Felipe Neves Braz

Health-PRIOR: An Intelligent Recommender System Architecture for Case Prioritization in Healthcare

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação, do Instituto de Ciências Exatas da Universidade Federal de Juiz de Fora como requisito parcial para obtenção do título de Mestre em Ciência da Computação.

Advisor: Fernanda Cláudia Alves Campos

Coadvisor: Victor Ströele de Andrade Menezes

Juiz de Fora

2020

Ficha catalográfica elaborada através do Modelo Latex do CDC da UFJF
com os dados fornecidos pelo(a) autor(a)

Braz, Felipe Neves.
Health-PRIOR: An Intelligent Recommender System Architecture for Case Prioritization in Healthcare / Felipe Neves Braz. – 2020.
79 f. : il.

Advisor: Fernanda Cláudia Alves Campos
Coadvisor: Victor Ströele de Andrade Menezes
Dissertação (Mestrado) – Universidade Federal de Juiz de Fora, Instituto de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação, 2020.

1. Palavra-chave. 2. Palavra-chave. 3. Palavra-chave. I. Sobrenome, Nome do orientador, orient. II. Título.

Programa de Pós-Graduação em
Ciência da Computação
www.pgcc.ufjf.br
Universidade Federal de Juiz de Fora
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Felipe Neves Braz

"Health-PRIOR: An Intelligent Recommender System Architecture for Case Priorization in Healthcare"

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação da Universidade Federal de Juiz de Fora como requisito parcial à obtenção do grau de Mestre em Ciência da Computação.

Aprovada em 16 de março de 2020.

BANCA EXAMINADORA

Profª. Dra. Fernanda Cláudia Alves Campos – Orientadora
Universidade Federal de Juiz de Fora

Prof. Dr. Victor Ströele de Andrade Menezes - Coorientador
Universidade Federal de Juiz de Fora

Prof. Dr. Mario Antonio Ribeiro Dantas
Universidade Federal de Juiz de Fora

Profª. Dra. Kele Teixeira Belloze
Centro Federal de Educação Tecnológica Celso Suckow da Fonseca (Cefet/RJ)

ACKNOWLEDGEMENTS

To my mom, for always encouraging me and supporting me to study and follow my goals, which is my friendly shoulder in the most difficult moments.

Special thanks to my advisors, Fernanda and Victor, for believing in my potential and for encouraging and supporting the taking on of new research and life challenges. For the friendship, partnership, and for always be present to decide, together, the research directions to take, my thanks to them.

To CAPES, for financial assistance through a scholarship, in which it was of paramount importance to conclude my research.

To Global Affairs Canada, for financial assistance through a scholarship, in which it provided me with new experiences and learning in a new country and culture.

To the PGCC professors, especially the professors Mário, Regina and José Maria, for believing in the power of scientific research and who helped me and always were available during this journey.

I am grateful to my friends and colleagues at PGCC and Engineering lab at Western, for their encouragement and partnership.

I am also grateful for the opportunities I had, which provided me with growth, both academic and personal. Although the forecast of events is seductive, adapting to difficulties provides us with greater experiences and learning than we can imagine.

"Never forget what you are, for surely the world will not. Make it your strength. Then it can never be your weakness. Armour yourself in it, and it will never be used to hurt you."

(George R. R. Martin)

RESUMO

Cidades inteligentes oferecem um ambiente de fluxo de dados e de monitoramento constante visando o bem-estar da população. Quando aplicados à saúde melhoram a qualidade de vida das pessoas possibilitando, por exemplo, a predição de doenças e acompanhamento de tratamentos. A priorização de casos em centros médicos é de grande importância, tanto para a saúde dos pacientes quanto para o dia-a-dia dos profissionais da área. Sistemas de recomendação são uma alternativa para integrar automaticamente os dados gerados nesses ambientes à modelos preditivos e recomendar ações, conteúdos ou serviços que possam beneficiar pacientes em seu contexto. O objetivo desse trabalho de pesquisa é auxiliar pacientes e médicos no diagnóstico precoce de doenças ou na detecção do agravamento de casos pós-operatórios através de um monitoramento constante. Visando atingir esse objetivo, esse trabalho propõe uma arquitetura para sistemas de recomendação aplicada à saúde, na qual é capaz de priorizar casos médicos emergenciais. A arquitetura traz uma abordagem conjunta para predição, onde permite a adoção de múltiplos algoritmos de aprendizagem de máquina. A metodologia para a realização do trabalho se desenvolveu em três passos: (I) realização de um mapeamento sistemático, buscando identificar as lacunas presentes no estado da arte no contexto de sistemas de recomendação para saúde; (II) construção e desenvolvimento da arquitetura; (III) avaliação por meio de estudos de caso, visando avaliar partes específicas da proposta, sua completude e aderência em contextos médicos. Sua adoção é justificada uma vez que os dados produzidos por dispositivos inteligentes são precisos e confiáveis para contextos preditivos e de tomada de decisão. Os resultados obtidos em cada estudo de caso mostraram a viabilidade da proposta, onde, para conjuntos de dados acurados e com pouco ruído ou valores faltantes, as predições se apresentam promissoras e aderentes ao contexto de aplicação.

Palavras-chave: Internet das coisas. Big Data. Assistência Médica. Sistemas de Recomendação. Modelos Preditivos.

ABSTRACT

Smart city environments offer a data flow and constant monitoring, aiming for the well-being of the population. When applied to healthcare, they improve the quality of people's lives, making it possible, for example, the prediction of diseases and treatment monitoring. The case prioritization in medical centres is of great importance, which favours patients' health and physicians in their daily job. Recommender systems are an alternative to automatically integrate the data generated in these environments with predictive models and recommend actions, content or services that can benefit patients in their context. The main goal of this research work is to assist patients and doctors in the early diagnosis of diseases or in the detection of worsening in postoperative cases through constant monitoring. To achieve this goal, this research work proposes an architecture for recommender systems applied to healthcare, that can prioritize emergency cases. The architecture brings an ensemble approach for prediction, which adopts multiple Machine Learning algorithms. The methodology for carrying out the work follows three steps: (I) conducting a systematic mapping, seeking to identify the gaps present in the state of the art in the context of health recommender systems; (II) construction and development of the architecture; (III) evaluation through case studies, aiming to evaluate specific parts of the proposal, its completeness and adherence in medical contexts. The adoption of the architecture is justified since the data produced by smart devices are accurate and reliable for predictive and decision-making contexts. The results obtained in each case study showed the feasibility of the proposal, where, for accurate data sets with a low amount of noise or missing values, the predictions are promising and adherent to the application context.

Keywords: IoT. Big data. Healthcare. Recommender systems. Predictive Models.

LIST OF FIGURES

LIST OF TABLES

SUMMARY

1 INTRODUCTION

Internet of Things (IoT) is a technology that allows smart devices to communicate with each other throughout the internet (ATZORI; IERA; MORABITO, 2010). A typical IoT system consists of sensors, communication interfaces, advanced algorithms, and the cloud interface. Their development and adoption in several contexts generate a significant volume of data. Those devices are components of a Smart City context and are responsible for capturing information in an automated way (RAHMANI et al., 2015). Many researchers seek to use this data to produce useful information, applying it in a real context.

Smart Cities are an example of a context that produces and uses a significant volume of data. Researches aim to improve urban environments to achieve greater service efficiency for their citizens. This goal is reached through monitoring and optimizing infrastructures, increasing collaboration between different economic agents, encouraging innovative business models in the private and public sectors (MARSAL-LLACUNA; COLOMER-LLINÀS; MELÉNDEZ-FRIGOLA, 2015).

Several benefits and possibilities can be added to human's life, providing information and improving the efficiency of health processes. This trend can be verified with the increasing number of discovered cases of diseases, which diagnosis is difficult to perform with a small number of data (BASTOS; KIRSZTAJN, 2011). An example of tracking human behaviour can be seen in hospitals, where through the help of medical devices and professionals to monitor patients, an early understanding of disease development or even the worsening in treatments can be successfully reached.

Different types of medical data are in use today and they are available for doctors to pursue treatments and diagnosis. X-ray and magnetic resonance images, medical notes and results of previous cases, are examples of medical data available at hospitals. These data help physicians in their daily work, allowing them to understand the related factors to the patients' conditions. Disease symptoms are directly linked to diagnoses and can be easily monitored by specific devices.

In this context, the patient's constant monitoring enables to get sufficient data to pursue early diagnosis and predictions of a possible disease case and its development. Considering the level of emergency, it is mandatory to inform doctors, patients and their families, as it is also necessary to correctly prioritize the cases. By monitoring patients, the data produced by IoT devices can be integrated (FARAHANI et al., 2018) with other systems where it would be used as problem identification parameter.

Recommender Systems (RS) are an alternative to automatically integrate data with predictive models, which may predict diseases or worsening in patients' treatments. These systems can be defined as "any system that produces individualized recommendations or that has the effect of guiding the user in a personalized way to relevant objects or that are useful to them from among the several possible options" (BURKE, 2002). In this way, the RS performs the filtering of information, analyzing the user's profile and interests, for later, to recommend content and actions, to him/herself or to the group which the patient belongs to.

Based on the available data produced at hospitals, predictions can be made by recommender system models, aiming to help physicians to understand the factors related to a specific disease or case. Several works adopted predictive approaches in diagnosis and forms of treatment (ALI et al., 2018), applied in recommender systems that detect chronic diseases (MUSTAQEEM et al., 2017; AOUEDI; TOBJI; ABRAHAM, 2018; RALLAPALLI; GONDKAR, 2018), detect the possibility of heart disease (LAFTA et al., 2015; KURIAN; LAKSHMI, 2018; TULI et al., 2020), the use of medical notes to make predictions in health centres (SANTOS et al., 2018) and to estimate the likelihood that an adverse event was present in postoperative cases (ZHANG et al., 2019; JEFFERY, 2017).

Predictive models aim to understand the factors around a context and convert a scenario into a class (classification models) or a number (regression models)(DREISEITL; OHNO-MACHADO, 2002). Those models are designed to solve specific problems, and depending on the application context and data, the accuracy is impaired. Problems such as noise and missing data are factors that directly influence the performance of a model. Besides that, each model can find a different result for the same dataset as they can detect different patterns on the data. Some approaches try to minimize the error and over-fitting possibility by using *ensemble learning approaches* (ARAYA et al., 2017). Ensemble approaches combine multiple learning algorithms to get a more adherent and precise result (ZHOU, 2015).

With that in mind and aiming to assist patients and doctors in the early diagnosis of diseases and worsening in postoperative cases by constant monitoring, this research proposes an architecture for a recommender system, based on Machine Learning ensemble model, where the analysis of the patient's data, captured periodically by IoT devices, define a prioritization case.

1.1 RESEARCH QUESTIONS

Based on the motivations and the problems discussed, the research question is:

How can data produced by IoT devices be integrated into Intelligent Recommender Systems for case prioritization in healthcare to improve assistance to doctors and patients?

From the proposed research question, we can derive the following questions:

- Q1: How predictive models are being used to recommend actions in the healthcare sector for patients and doctors?

- Q2: How to combine different machine learning models and approaches to achieve greater performance and a higher level of certainty in the predictions of early diseases diagnosis?

- Q3: How accurate is the proposed architecture based on predictive approaches?

- Q4: How the architecture can assist in other domain?

1.2 GOALS

In the context of smart cities, the use of recommender systems together with IoT devices can improve the patient's care in hospitals or residential environments. The use of IoT data to pursue predictions in these environments makes possible the early diagnosis of diseases, which favours patients' lives and doctors' daily job.

The goal is to make personalized notifications that are adherent to the patient's context and requirements. As the core of the system, prediction techniques (YANG et al., 2017) can be applied, by identifying the most opportune moment to provide a recommendation, based on the severity and level of emergency.

The main goal of this research is to propose a predictive architecture, based on a Machine Learning ensemble model, that can make personalized recommendations focused on the needs of patients in hospitals or even in home environments. According to the research questions, this main goal can be divided into four secondary ones, as follows:

- G1: identify the most used predictive approaches in the literature that can accurately predict classification problems and evaluate the flaws in the system. This goal can be reached with a systematic mapping of the literature, where the most valuable works are studied to find the gaps in the area.

- G2: develop an autonomous solution capable of synchronizing and managing various machine learning techniques. To achieve this goal, we propose an ensemble machine learning approach to combine different learners and to achieve the final result with high accuracy.

- G3: develop a preprocessing layer capable of cleaning and structuring the data coming from IoT devices. To achieve this goal, we preprocess the data coming from IoT devices by removing noise values and replacing the missing ones. It also normalizes the data to prepare the dataset.

- G4: evaluate the designed architecture with a real health data. We use case studies to perform the evaluation. We developed three case studies, evaluating different aspects of the architecture. The first case study focused on the ensemble model and its accuracy in an educational context as a feasibility study, the second focused on the models training time and its adherence in a healthcare environment. Finally, the third case study focused on the full aspects of the architecture in healthcare.

1.3 RESEARCH METHODOLOGY

The methodological process followed three main stages:

(i) **Systematic Literature Mapping:** stage to identify the state of the art of health recommender system architectures and the most applied predictive techniques to those systems. Systematic Literature Mapping consisted of an important step to identify the state of the art of predictive models, to form the structural outline of the proposed architecture. From the results, gaps were identified and a first architectural version was drafted. Models, algorithms, techniques, stages and flow to be followed were defined for the recommendation process.

(ii) **Development:** stage to design and build an architecture that supports multiple learning algorithms and can prioritize cases by severity level.

(iii) **Evaluation:** stage to measure the proposed architecture efficiency, in terms of adherence in the health domain and accuracy in predictions. We chose to use case studies to evaluate the architecture and its approach.

Case Study is defined by Wohlin et al. (2012) and Runeson et al. (2012) as an empirical investigation, which is based on different sources of evidence, used when the object of the study is a contemporary phenomenon difficult to be studied in isolation.

For this reason, each architecture version was evaluated to obtain its predictive potential, and the gains from the approach adoption, in comparison with other individual approaches present in the literature. The evaluation run by applying the architecture predictive models first in the educational context, as an experimental study, and finally in two health contexts. The proposed architecture faced an incremental process of evaluation and refinement when Health-PRIOR architecture versions were generated.

1.4 OUTLINE

This chapter presented the research motivations and justifications as well as the problems and the context that this dissertation seeks to address. This chapter also presented the objectives and the methodology followed in the development of the present work. Chapter 2 will address the theoretical assumptions and the main concepts about smart cities, recommender systems and machine learning models. Chapter 3 presents the systematic mapping, characterizing the state of the art of research topics. Chapter 4 presents the related works. Chapter 5 presents the proposal architecture, its modules, and steps followed for its structuring. Chapter 6 shows the proposal evaluating process, bringing the case studies, detailing their domain, data collection, architecture adequacy and main results. Finally, chapter 7 brings the final remarks and future works.

2 BACKGROUND

In this chapter, the main concepts related to the theoretical assumptions are presented. First, those related to the smart cities context and the use of IoT devices and then the concepts for prediction in Recommender Systems.

2.1 SMART CITIES

Due to technological advancement in urban environments, there is a requirement for cities to find ways to manage new challenges (ALBINO; BERARDI; DANGELICO, 2015). To offer high-quality urban services, with long positive effects on the economy, it is important to maintain and improve the economic needs for the growth of cities (ALBINO; BERARDI; DANGELICO, 2015).

Smart cities have a lot of definitions and all are related to the application factors. In our context, the definition that we are going to use is proposed by (KITCHIN, 2014): Smart city is "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 in our context because shows the connection between public and private agents, working together to get a higher quality of services.

The information gathering can be made throughout smart devices capable of capture information in real time in an automated way. The next section will describe some concepts around the topic.

2.1.1 **Internet of Things (IoT)**

IoT devices are capable of capturing actions, temperature, location among other information and transmit this data to systems that consume them (RAHMANI et al., 2015). Primarily used in monitoring contexts, it is anticipated that these systems will be widely used for healthcare in citizens' residences (KOOP et al., 2008).

The sensors in an IoT system are used to collect data from different devices and may be interconnected. Radio-Frequency IDentification (RFID) technology and Wireless Sensor Networks (WSN) technologies provide the necessary media of communications and network infrastructure (ATZORI; IERA; MORABITO, 2010). Several types of algorithms, proprietary or not, are used to process data and analyze everything that is enough significant, through APIs or applications.

The most typical communications, which are also widespread in industry development, are client-server requests. They exchange information between the devices in a standard format, usually in JavaScript Object Notation (JSON) format (WEHNER; PIBERGER; GÖHRINGER, 2014). These devices can be both local and cloud-based, being in charge of specialized companies.

The data production through these devices is quite fast and specific. That is, these devices have few or a single focus, such as measuring the temperature, triggering alarms when catching movement, among other functions, depending on the device. This diversity of information promotes systems that depend on sensitive data to act correctly. Among these systems are Recommender Systems. In eHealth, recommender systems gain much importance, being, according to (FARAHANI et al., 2018):

- Comprehensive: if people use IoT for health, exercise, safety or beauty reasons, it has a holistic solution for all needs.
- Integration with different technologies: IoT in eHealth allows different technologies to be used without problems or compromising the complexity of technological integration.
- Big Data processing and analysis: can effectively process, analyze and manipulate the grid volume of multi-scale, distributed and heterogeneous data produced by connected sensors. It allows extracting useful and reliable information from health data.
- Ability to customize content or service: IoT data analysis can greatly expand the possibilities of meeting the need for personalized health care and treatment. Therefore, it plays a remarkable role in well-being.
- Lifetime monitoring: patients can receive additional data about their past, present and future health, through the prediction of diseases or worsening in their condition.

These characteristics aggregated in a single database allows their analysis by intelligent techniques, to work as a predictive basis of smart decision support systems.

Wearable Intelligent Systems, according to Chan et al. (2012), are already used to monitor patients 24 hours a day, at home and outdoors, according to the protocols of preventive medicine. The monitoring system is connected to a technical service because the measured parameters are transmitted continuously or intermittently. For example, during extreme weather conditions, the elderly are more prone to toxic infections, dehydration or other diseases. Predictions, that use IoT data, can help to prevent infections and changes in patient therapies. Based on the prediction result, an appropriate therapy can be initiated before the situation becomes dangerous to the patient (CHAN et al., 2012).

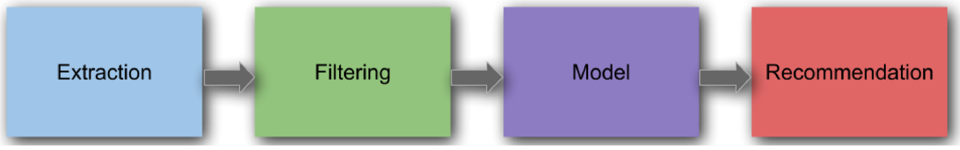
Recommender systems are intelligent systems that, by default, analyze users' data, preferences and needs to proceed with recommendations. In that way, the use of these devices is justified, having a more accurate database, because they generate data for understanding the real world.

2.2 RECOMMENDER SYSTEMS

Recommender systems are systems widely used mainly in the e-commerce sector and industry. With the development of streaming services, it has also been used on these platforms. These systems are intended to recommend content to users based on their profile and context, considering their preferences, needs and interests (BURKE, 2002). Recommendation techniques are used to help characterize the user's profile and context, allocate them into groups with similar needs, locate resources that meet user's needs and design a strategy to recommend in the most effective way.

Figure 1 shows the structure of a recommender system architecture by having four main parts to pursue the recommendation (ABDALLA et al., 2018). First, we have the **Extraction Layer** responsible for extracting data from the user's profile and/or context that the user belongs to. This layer is the first layer activated in the recommendation process. It is important to extract the relevant data from the user's profile to characterize his/her preferences correctly.

Figure 1 – Structure of a recommender system (ABDALLA et al., 2018)



The **Filtering Layer** is the second one, which is responsible for filtering the information, correlating users' preferences. The following section brings the types and definitions of each main filtering type present in the literature.

The **Model Layer** represents the system model, which is applied to the recommendation strategy. It is responsible for predicting the resources that are more adherent to the user context. They can be defined as model-based, memory-based or hybrid models.

The **Recommendation Layer** is responsible to present the resource to the user.

This process is continuous, that is, when a recommendation occurs, which means the recommendation is correct to the user, the positive result is used by the system to aggregate the user's preference to pursue further recommendations.

2.2.1 **Filtering Types**

There are different perspectives to recommend a resource in RS. The system can choose the most preferred item in a database or highlighted items. However, there are specific and well-defined strategies to pursue recommendations. Some of them will be described in this section.

Content-based Filtering is intended to recommend resources that match the preferred ones by the user in the past. This means the chosen resource will be the same or similar to the user's content preferences. According to Burke (2002), the content-based approach learns a profile of the user's interests based on the features present in objects the user has rated.

Collaborative Filtering is, according to Burke (2002), the most popular and most widely used in the literature. They aggregate classifications or recommendations of objects, recognize similarities among users, based on these classifications. The user profile in these systems is characterized by an array of items and their classifications, whose dimension increases as the user interacts with the system over time. These systems can be memory-based, comparing users with each other using correlation or other measures, or model-based, where it analyzes historical data and it uses to make forecasts (BREESE; HECKERMAN; KADIE, 1998 apud BURKE, 2002).

Demographic Filtering aims to categorize the user, based on his/her demographic class. Recommend a resource based on demography is very common in marketing, which recommends products, services and items that are near to the user and adherent to his/her profile (BURKE, 2002).

Utility-based Filtering tries to suggest items to users that have a utility to them. That utility is mostly calculated by computational functions to match the user's preferences with the characteristics of the items (BURKE, 2002). This can be pursued by popular items, promotional items, and items that could fit into their current context.

Knowledge-based Filtering makes inferences about the user's needs and preferences. Those inferences intend to understand implicit preferences in the user profile to pursue a recommendation (BURKE, 2002). Understand the user's characteristics to make recommendations is a trick form to pursue the suggestions because this approach can find out how a particular item meets a particular user need, which the user did not tell or mentioned explicitly.

Each filtering type has pros and cons. Content-based filtering is able to recommend an item even when it does not receive a rating by a user. But, it presents the *cold-start* problem, when the user does not have an item rated in the database, being difficult to recommend content which could not be of interest to the user. Similar to it, collaborative filtering has this problem; new users, that did not have rated any item in the database, cannot receive a recommendation with precision because the system does not have enough information about their preferences and interests (SON, 2016).

Other types of filtering do not present this problem since they do not need historical ratings to make recommendations. Some alternatives are available to treat this issue. Son (2016) presents a comparative review of forms of treatment to new user *cold-start* problems showing that among all algorithms evaluated, the *NHSM* algorithm, which is a heuristic similarity measure, performs better in this issue.

2.2.2 Recommendation Models

According to Bobadilla et al. (2013), there are two main model categories: (I) Memory-based models, that usually use similarity metrics to obtain the distance between two users or items. This distance represents the level of similarities between two users. And, (II) Model-based models, that use information captured in the extraction layer, to create a model that generates the recommendations. These two main categories have a vast number of algorithms, such as K nearest neighbours, Euclidean distance, Pearson correlation and cosine similarity. In memory-based methods are Bayesian classifiers, neural networks, fuzzy systems, genetic algorithms, among others. In model-based methods are decision trees, deep learning algorithms, among others.

The adoption of artificial intelligence predictive models favours the classification and grouping contexts that may be useful to model-based methods. This class of algorithms can provide predictions of user ratings and preferences to items available for recommendation. Through the combination of their results, it is possible to unify and enhance the prediction accuracy of the proposed models (BOBADILLA et al., 2013).

Among the models present in the literature, there are variations in their specification, where hybrid models emerge. Neves, Ströele and Campos (2019) proposes an architecture for a recommender system that uses a hybrid model to recommend resources to users. The use of social networks to design a model for recommendation is based on the combination of models, such as euclidean distance and the graph theory, which measures distance in a different way, as centrality measures and short paths between nodes. That combination intends to mitigate problems and limitations that each model presents individually (ABDALLA et al., 2018).

Limitations are noticed in these models, where the results may be unsatisfactory when the number of resources is scarce or even present the *cold-start* problem, which is a recurrent problem in the area of recommender systems (ABDALLA et al., 2018), especially those that use collaborative and content-based filtering.

2.3 FINAL CONSIDERATIONS

This chapter presented the main subjects about the research background. The aspects and definitions involved in a smart city context, and the recommender system field were described and detailed. Although many research works approach the use of RS to various contexts, this research focus in to combine the benefits of a smart city context with the power of recommender systems, aiming to improve assistance to doctors and patients in the healthcare sector. To achieve that, we use predictive approaches as a recommender system predictive model to select patients who need attention, based on their conditions.

3 SYSTEMATIC MAPPING OF THE LITERATURE

Aiming to identify predictive approaches applied to Recommender Systems in the eHealth context, we conducted a systematic mapping following Kitchenham (2004) and Wohlin et al. (2012) guidelines. We were interested in investigating the contexts of using the predictive approaches of these systems in eHealth context.

A systematic mapping seeks to evaluate and interpret the relevant knowledge of a given area using rigorous methods with a well-defined methodology (COSTA; MURTA, 2013). The systematic mapping developed in this research work differs from previous studies, such as Zaidan and Zaidan (2018), since it seeks to identify methodologically and systematically the application of approaches in the eHealth context, seeking to understand the considered criteria and its applicability in this context.

The study was performed by the following questions:

- **MQ1:** Which are the conferences and journals more adherent to Recommender Systems on eHealth context today?
- **MQ2:** Which are the prediction challenges in Recommender Systems on eHealth context today?
- **MQ3:** Which methods are the most used to predict preferences and needs in Recommender Systems on eHealth context?
- **MQ4:** Which methods are more popular to evaluate the predictions accuracy in Recommender Systems on eHealth context?
- **MQ5:** Which filtering type is the most used by Recommender Systems in this context?
- **MQ6:** In which application contexts the prediction in Recommender Systems is present?

To make this section more well-organized and objective, we will describe the main mapping results. A total of 279 papers were analyzed. After all steps of the selection process, 26 papers remained, which were considered to answer the systematic mapping questions. The complete study is at *https://github.com/FelipeNb/SystematicMapping.*

3.1 SYSTEMATIC MAPPING RESULTS

MQ1 searches for the most common conferences and journals in the health recommender systems area. Table 1 shows a list of conferences and journals and the number of papers that belong to each category. MQ2 is concerned to answer which challenges are most common in RS contexts. Table 2 presents the challenges drawn from the selected papers as well as the number of occurrences of each of them. The first column identifies the challenge and the second column the papers that cite that challenge.

Table 1 – Conferences and journals where the papers were published

Type	**Conferences/Journals**	**Quantity**	**%**
Conference	ACM Conference on Recommender Systems, ACM International Health Informatics Symposium, Conference on Biomedical Engineering and Sciences, International Conference on Dependable, Autonomic and Secure Computing, International Conference on Pervasive Intelligence and Computing, International Conference on Big Data Intelligence and Computing, Cyber Science and Technology Congress, BHI 2014, IEEE International Conference on Fuzzy Systems, International Joint Conference on Web Intelligence and Intelligent Agent Technology, International Conference on Wireless Mobile Communication and Healthcare	8	30%
Journal	Journal of Computational Biology, Knowledge-Based Systems, Expert Systems with Applications, Future Generation Computer Systems, BMC Medical Research Methodology, Future Computing and Informatics Journal, Procedia Computer Science, Lecture Notes in Computer Science,Computer Communications, Mobile Networks and Applications,Neural Computing and Applications, Web Intelligence, Lecture Notes in Computer Science, International Journal of Medical Informatics, IEEE Access, Neurocomputing, BMC Public Health, International Journal of Medical Informatics	18	70%

Table 2 – Identified challenges

Challenge	**Occurrences**
C1. Identify the semantic context when predicting preferences and needs.	17
C2. Achieve greater precision through increased accuracy.	6
C3. Amount of data to be processed.	4
C4. Preservation of privacy when processing individual's data.	1

More than half of the papers mention the C1 challenge, which highlights the difficulty of assertively identifying the semantic context of an individual's needs and preferences. This fact is important to notice since the individual's context identification is the prediction basis (GAO et al., 2017).

C2 is another challenge that is related to predictive models accuracy. According to the calculated accuracy, the certainty about the recommendation is proportional, where the probability of an individual to approve the recommendation and adhere to it, increases proportionally (MUSTAQEEM et al., 2017). The amount of data to be processed is shown as a challenge. Depending on the volume of data, the results can be good or bad; this happens because old data generates a less accurate result, and when the amount of data is too large, distortions and missing values can influence the predictions (CHEN et al., 2017).

The individual's privacy preservation was pointed out as a challenge by Kaur, Kumar and Batra (2018). This fact should be treated as critical since all analyses use the individual's data to provide recommendations. Additionally, to better predict the user's needs, the private data may be considered when performing the prediction to not break the individual's privacy.

Answering the MQ3, the majority of papers show Machine Learning as the predictive approach, with a total of ten studies, that are described in Table 3. Variations of several Machine Learning techniques have been addressed in the works, such as Deep Learning and SVM. Those techniques are based on data training to pursue predictions, which analyze the data pattern to classify them into a group.

Table 3 – Predictive approaches

Method	**Occurrences**
Machine Learning	10
Similarity Techniques	8
Fuzzy	2
Bayesian Ranking	2
Ontology	1
Neural Network	1
Heuristics	1
Fourier Transformation	1

Other predictive approaches are considered in eight different studies, and their predictive basis is the similarity. Classical model-based models, suit this predictive category, for example, KNN, algebraic similarity, logistic regression, and Euclidean distance (HASSAN; SYED, 2010; ADAMS et al., 2014; THONG et al., 2015a). The 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 Thong et al. (2015b), Thong et al. (2015a) and Dhanalakshmi, Ramani and Reddy (2017), Gao et al. (2017), two of each. The methodology characterization that uses Fuzzy is based on the subdivision of sets, which it treats groups of objects to be analyzed similarly.

Approaches such as ontology (ALI et al., 2018), neural networks (ABOAGYE; JAMES; KUMAR, 2018), and predictions based on heuristics (LAFTA et al., 2016) are also solutions present in literature. Those approaches focus on the model, using it as the system predictive core, trying to get new information through data (LAFTA et al., 2016; ALI et al., 2018).

Answering the MQ4, the most common method to evaluate accuracy is the precision metric. Its calculation is simple and considers the index of correctness made in the recommendation process. Recall and MAE (Mean Absolute Error) metrics appear in second place, their justification is similar to the previous one. Table 4 describes in descending order, what are the most popular methods to calculate accuracy on the selected studies.

Table 4 – Accuracy methods

Method	**Occurrences**
Precision	10
Recall	6
MAE	6
AUPR or AUROC or ROC	5
RMSE or MSE	4
F1-measure	3
Not defined	3
Precision coverage	2
MAP	2
NDCG	2
Proprietary	1
MRR	1
Kappa	1
KTAU-b	1

To predict error, RMSE and MSE are widely used. They are based on prediction error to measure the percentage of reached accuracy, like the MAE. Other metrics are used to measure accuracy, but the number of occurrences is not too expressive.

Following the three main types of filtering in Recommender Systems (MQ5), Table 5 describes which type of filtering is more commonly used in the literature.

Table 5 – Types of filtering found in the papers

Filtering	**Description**	**Occurrences**
Collaborative	Uses choices from similar individuals to recommend. It is based on the similarity between people.	14
Content-based	It uses similarity between content, that is, it uses similar recommendation objects to those preferred in the past, to proceed with the recommendation.	8
Hybrid	It addresses the junction of the above concepts so as to best fit the accuracy of the recommendations.	4

We can observe that the most commonly used filtering type is the collaborative filtering, followed by content-based filtering and the hybrid filtering, with fourteen, eight and four studies, respectively. We realize that the accuracy validation is an expression factor in the prediction area, and it is necessary to analyze related points for its improvement. We also note that most studies carry out validation of predictive approaches in real datasets applied to this research context.

According to the used techniques, Table 5 shows the most classic ones, that were modified in works, such as (GAO et al., 2017; DHANALAKSHMI; RAMANI; REDDY, 2017). Algorithms and new predictive models were also proposed, and are described by (LAFTA et al., 2016; LAFTA et al., 2015; ZHANG et al., 2017; HUNG et al., 2015; DHANALAKSHMI; RAMANI; REDDY, 2017), making a significant contribution on prediction approaches in Health Recommender Systems.

Finally, the MQ6 is answered by the analysis of the approaches contexts. Thong et al. (2015a) use a fuzzy-oriented predictive method to predict possible diseases based on the symptoms. This is analyzed to maximize the accuracy of the predictions concerning the patient's clinical history. Similarly, Hussein et al. (2012) use the predictions of a Recommender System for the prognosis of chronic diseases.

Another type of prediction application with the support of RS is proposed by Fan, Yang and Jiang (2018), which uses predictive techniques to predict the side effects of drugs on the central nervous system. Hung et al. (2015) proposed the use of prediction to identify depression in people, from the perception of emotions through their smartphones. This approach justifies the challenge pointed out by Kaur, Kumar and Batra (2018), regarding the preservation of user's privacy, however, in a Smart City context.

3.2 FINAL CONSIDERATIONS

The selected papers have applicability in the eHealth sector, to assist in the prediction of diseases and forms of treatment, where different types of predictive approaches are addressed, especially machine learning techniques.

Different recommendation objects are used in the works, where the purpose of the recommender system depends on the context and domain. Systematic mapping had an important role in the definition of this research, helping to define the scope and technologies to be used.

Aiming at the research completeness, surveying similar studies, the next chapter presents and describes the related works. From these studies, the current research is motivated to treat identified gaps both in the systematic mapping and in the related works, seeking a greater adherence to the solution in real application contexts.

4 RELATED WORKS

This chapter will discuss some related works, from the systematic mapping results, that implement models to predict user's preferences and needs in Recommender Systems, specifically in the context of healthcare and prediction of diseases.

The selected works present similarities with our proposal, which is focused on predicting diseases and/or worsening in postoperative cases. The aspects of data consumption as well as prediction in real-time, are present in the selected works. At the end, we make a comparative analysis.

Thong et al. (2015a) propose a hybrid model named HIFCF (Hybrid Intuitionistic Fuzzy Collaborative Filtering) using picture fuzzy clustering and intuitionistic fuzzy recommender system for medical diagnosis. The fuzzy logic provides a membership function that can classify uncertain values which represent the symptoms. The overall idea of this particular work 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.

The patients are classified according to their information relationship. The relationships are determined by the relations between patients and symptoms. After the defuzzification, which represents the process that transforms a fuzzy value into a normal value, named crisp value, the clustering algorithm is applied.

Figure 2 represents the proposal workflow. The patients are grouped based on their information and medical history, and then they are fuzzified. The patients' relationships show the similarity level among them and collaborative filtering can be applied to identify the predicted disease.

Another work that presents a fuzzy recommender system architecture is proposed by Ali et al. (2018). Their proposal focuses on Type-2 fuzzy logic and fuzzy ontology to automate the overall process of foods and drugs recommendation for IoT-healthcare systems.

The authors propose an architecture (Figure 3) that has two main layers: a security layer and a Type-2 fuzzy ontology-based decision-making knowledge layer. The first one, manages the access and prevents unauthorized access to a smart refrigerator and investigates the true patient condition before the recommendation. The second layer extracts the patient's risk factor values from wearable sensors and determines the patient's condition using Type-2 fuzzy logic. That layer is also responsible for retrieving drug and food information from the fuzzy ontology and recommends a prescription for a smart medicine box and foods for a smart refrigerator, according to the patient's condition (ALI et al., 2018).

Figure 2 – Proposal workflow (THONG et al., 2015a)

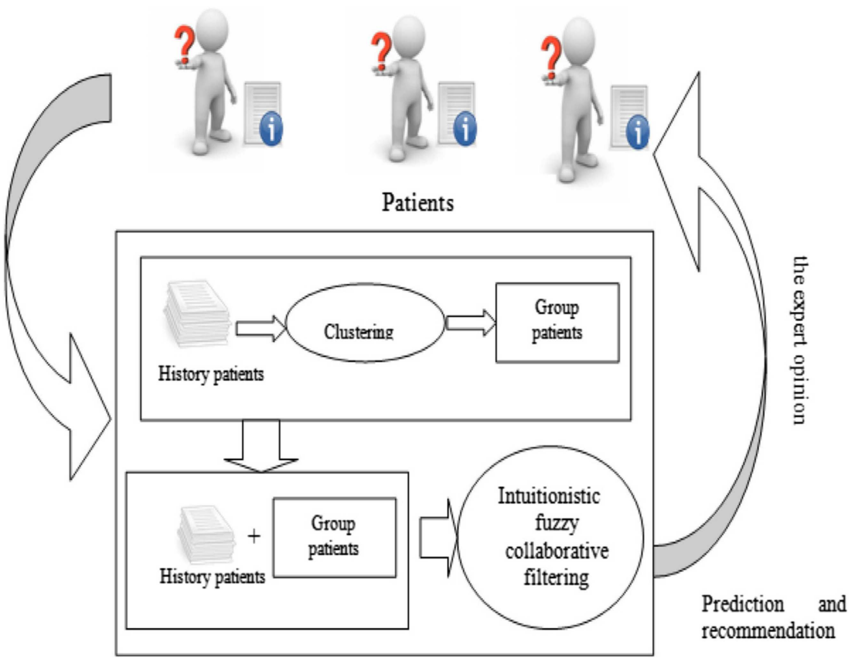
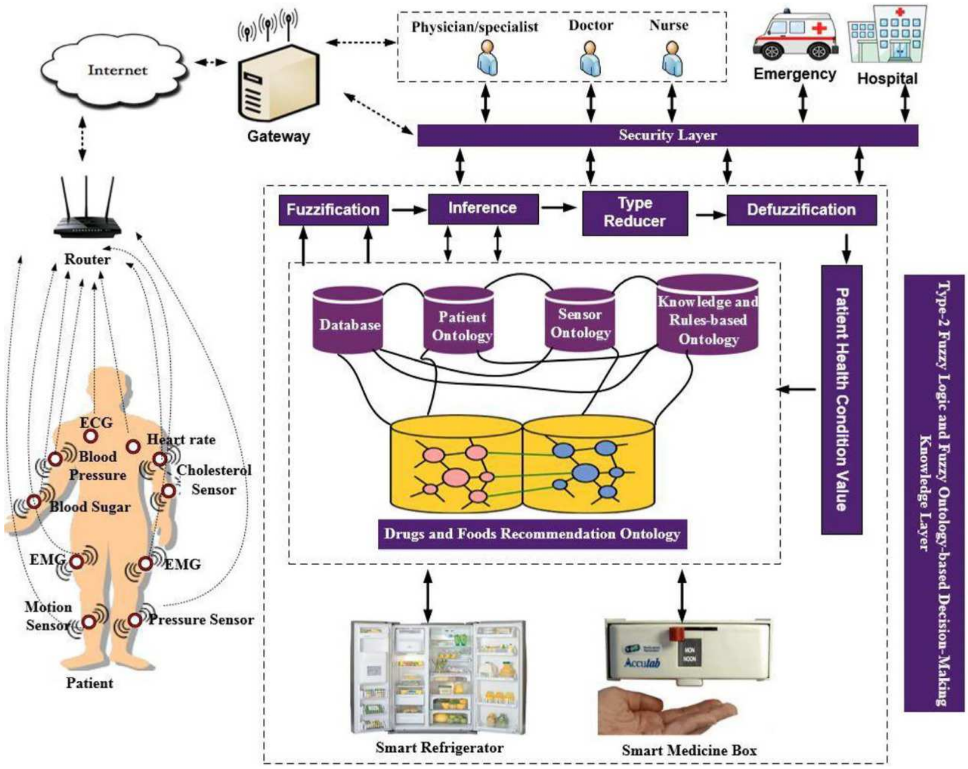


Figure 3 – Type-2 fuzzy architecture (ALI et al., 2018)



The overall idea of this study is to get the features from the patient's condition in real-time and predict the patient's needs by characterizing his/her condition with ontologies support. The inferences made by these ontologies are responsible to identify the necessary actions to a patient.

The ontologies are used to provide the right inference about the patient and the classifications made by the type-2 fuzzy logic. The ontologies represent decision-making knowledge for long-term care. They describe which are the foods and drugs more adherent to the patient's condition and retrieve the patient's personal information and disease history from the database.

The information collected by the sensors is stored in a sensor ontology and also is used as input for the fuzzification process, which maps these variables into a Type-2 fuzzy membership function. Type-2 fuzzy is a membership function that deals with uncertainty with two Type-1 fuzzy sets. This methodology works closer to the edges of a range that defines features classification. For example, in health systems, the features could be blood pressure, blood sugar, cholesterol, etc. And the classifications could be Low, Medium, and High. The boundaries of these ranges are uncertain and fuzzy approaches aim to solve it.

Similar to the previous work, Thanh, Ali et al. (2017) propose a hybrid recommender system for medical diagnosis named Neutrosophic Recommender System. In their proposal, the recommender system is based on a neutrosophic set and neutrosophic clustering.

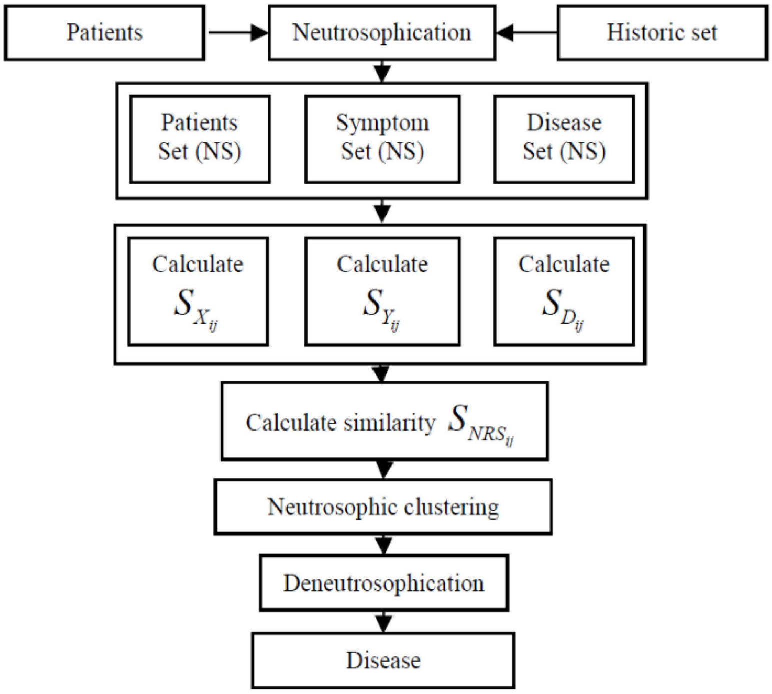
The neutrosophic set is proposed by (SMARANDACHE, 1999 apud THANH; ALI et al., 2017). A neutrosophic set has data that is independently characterized by a true, indeterminate, and false membership function. Their application in medical diagnosis based on correlation measures has been studied in various papers (THANH; ALI et al., 2017).

The authors propose a methodology with three main steps. First, the characteristics of a neutrosophic recommender system with neutrosophic similarity measures that use algebraic operations and their theoretic properties are investigated. Second, the neutrosophic clustering method is implemented to identify the neighbours of a considered patient who shares common characteristics. And, the prediction formula that uses the neutrosophic algebraic similarity measure and the neighbours resulted from the previous step is established. This formula is used to compute the membership value of new patients based on their neighbours.

Figure 4 represents the proposed Neutrosophic Recommender System. The historic data is used for training in the neutrosophication process and, the similarity through patient, symptom and disease data are calculated separately (THANH; ALI et al., 2017). The clustering process is used to identify all patients who share characteristics and symptoms with others to group them. Finally, the deneutrosophication is pursued to predict values to identify the final disease.

Focusing on the mixture of methods, Deng et al. (2018) propose a Neural Gaussian Mixture Model (NGMM). The proposal constructs two parallel neural networks with the reviewed items. One focuses on learning user preferences and the other learns item properties. So, they use a Gaussian mixture model to model rating information. The

Figure 4 – Neutrosophic recommender system (THANH; ALI et al., 2017)



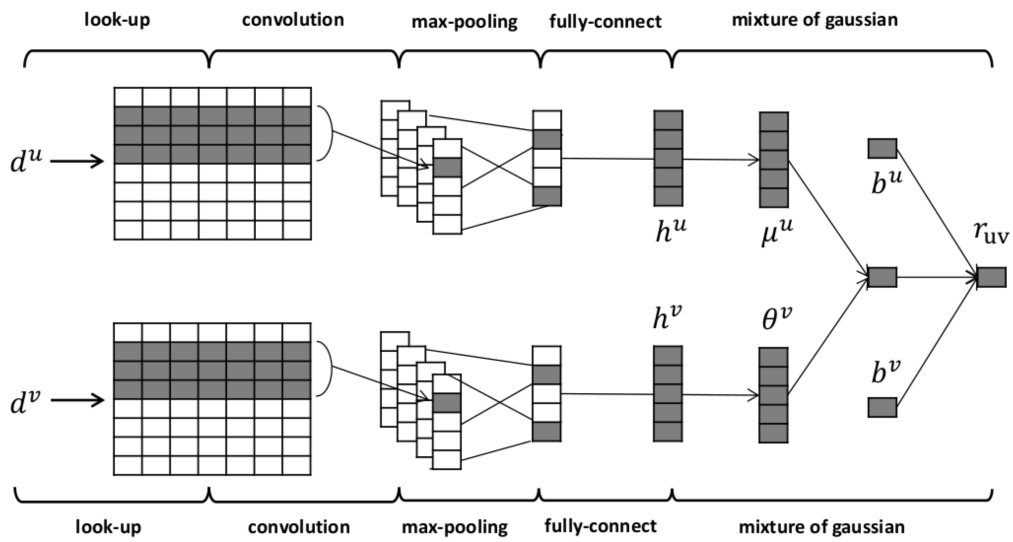
authors point out that user's rating tends to have a Gaussian distribution, which is investigated in their work. The proposal operates to review textual information by applying NGMM. The prediction is made by imitating the rating behaviour of users to items; that means, the items that present a better rating are recommended to the user.

The model assumes that each rating is generated from a Gaussian mixture model which models user preferences over factors of items (DENG et al., 2018). The authors designed a Gaussian layer on top of the neural network to simulate model parameters, mean and mixture proportion to incorporate the Gaussian mixture model into the neural network.

Figure 5 shows the proposed model. It is composed of five layers. The first one, called look-up layer, maps the user and item reviews to a uniform matrix of word embedding. This representation technique can capture more semantic and syntactic information among words that maps a word to a dense low dimensional real-valued vector (DENG et al., 2018).

The second layer, represented by convolution, contains a set of operations in Convolutional Neural Network (CNN) architecture based on models (KALCHBRENNER; GREFENSTETTE; BLUNSOM, 2014 apud DENG et al., 2018). This layer is responsible to make a filter capable of producing a feature map. In other words, the window of h words is converted into a feature in the neural network. After that, it is applied to a max-pooling operation, which is a non-linear subsampling function that returns the maximum of a set

Figure 5 – Proposed model (DENG et al., 2018)



of values (LECUN et al., 1998), over the feature map. Next, a fully connected weighted layer and an activation function are applied to get the final output.

The last layer is responsible to simulate the parameters of the Gaussian mixture model, which means μ and mixture proportions θ. The fully-connected layer is used to simulate $\Theta = (\theta, \mu)$ based on user and item embedding. The objective function of NGMM is given by:

$$\mathbb{L} = \sum_{u=1}^{|u|} \sum_{v=1}^{|v|} \mathbb{I}_{u,v} (\sum_{k=1}^{K} \mu_k^u * \theta_k^v + p^u + q^v - r_{u,v})^2 \tag{4.1}$$

Where $\mathbb{I}_{u,v}$ is an indicator function which is 1 when $r_{u,v}$, that is the relationship between user and item, is observed in training data and 0 otherwise. p^u is the bias of user and q^v the bias of item (DENG et al., 2018). The training process of the network is made by minimizing the equation 4.1. With the training dataset, the model is trained and uses the validation dataset to improve the hyper-parameters.

The work proposed by Mustaqeem et al. (2017) presents a hybrid prediction and recommendation model for diagnosis and treatment of heart disease patients. The main objective is to propose an intelligent and adaptive recommender system for patients that have different heart diseases. They pursue the research to identify any single heart disease and its subtypes. The recommendation objects are general medical recommendations about the disease that the system identified in the patient.

The authors designed the recommendation process as shown in Figure 6. From the clinical dataset, the features are extracted and treated their missing values to handle outliers. The prediction model is applied to these features to select the most relevant and significant ones, which aids the classification process. After that, the features selected are used to predict the classes of common heart diseases. The prediction information is used to recommend general medical advice to the patient, depending on the type of disease predicted, risk factor, probability of occurrence, and severity.

Figure 6 – Recommendation process (MUSTAQEEM et al., 2017)

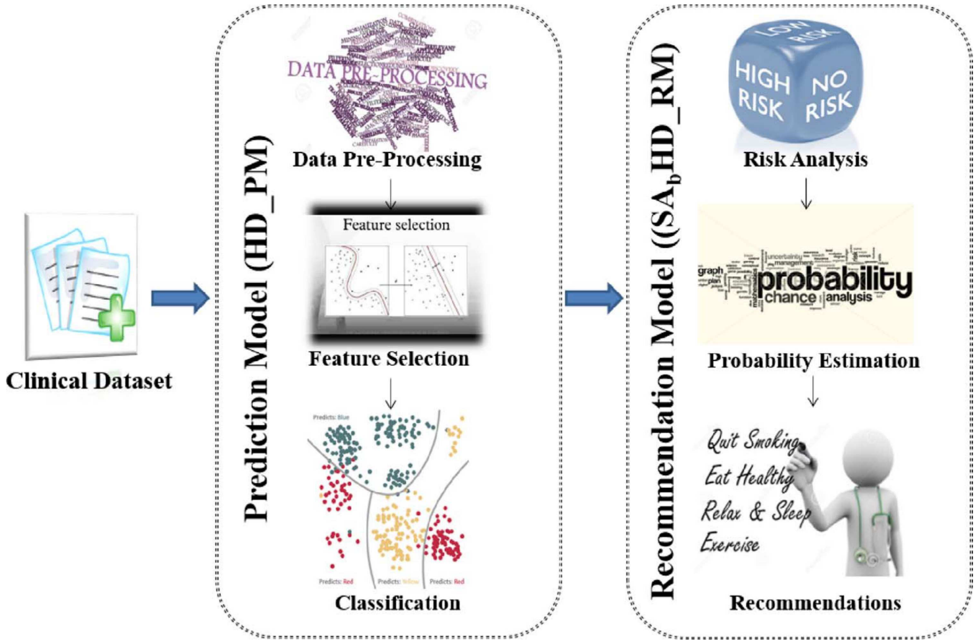
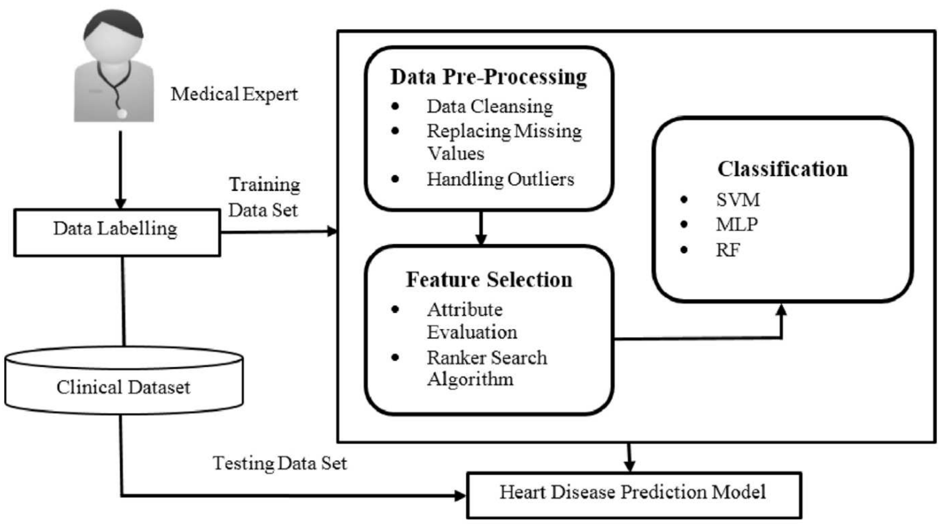


Figure 7 presents the architecture of the proposed model. The authors use three different types of machine learning techniques to perform the classification (Support Vector Machine, Multi-Layer Perceptron, Random Forest). These three techniques classify the features into the classes of diseases under study.

Figure 7 – Architecture model (MUSTAQEEM et al., 2017)



To make a recommendation, it is needed to formulate a knowledge base, that is addressed from the test dataset, identification of critical exposures, assigning weight and dataset labelling. To form the knowledge base the authors consulted a medical expert, which addressed the ranges of value features that represents severity in risk cases.

The risk of each heart disease against selected medical exposures is calculated as:

$$R = \frac{P(e)}{P(\bar{e})} \tag{4.2}$$

where R denotes the risk, $P(e)$ and $P(\bar{e})$ represents respectively the probability of a certain disease with and without an abnormality. $P(e)$ is calculated as,

$$P(e) = \frac{C_e}{C_p} \tag{4.3}$$

where C_e counts the exposure events within a class and C_P counts the total number of patients within a class. As well the preview equation, $P(\bar{e})$ is calculated as,

$$P(\bar{e}) = \frac{C_{\bar{e}}}{C_p} \tag{4.4}$$

where $C_{\bar{e}}$ shows the number of non-exposure events within a class.

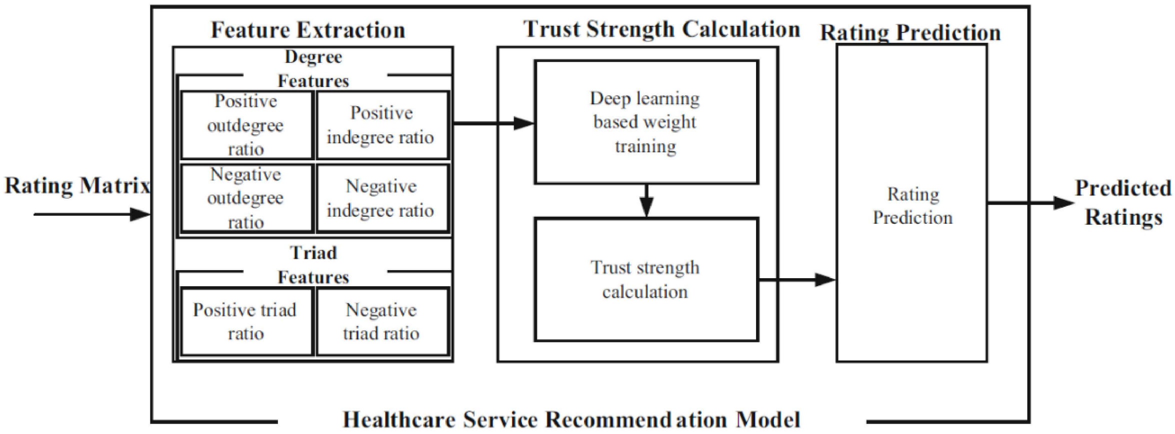
The recommendation process uses a combination of analysis results. A criterion is derived from the knowledge base, risk and probability estimations, which creates an inference based rule set for generating recommendations (MUSTAQEEM et al., 2017).

Yuan et al. (2018) presents an architecture for a socialized recommender system that uses the strength of the relationships between users to pursue recommendations. The proposal has the intention to recommend healthcare services that are more adherent to users using a deep learning approach.

The proposed architecture, presented in Figure 8, has three modules: (a) the feature extraction module, responsible for extracting the features from relationships between users and the structure of these relationships. Those relationships are represented by a graph, where the social theory is applied. The direction of the links is considered to extract the structure and trust/distrust information, where each link has a sign that represents if the relationship is positive or not. (b) The trust strength calculation is another module of the architecture, where is calculated the strength using the deep learning approach. (c) The last module is the rating prediction module, where the services are selected and recommended to users.

These three modules are recurrent in RS, as well as the flow of execution. The proposed model uses the extracted information, to estimate the level of strength between users.

Figure 8 – Socialized recommended system (YUAN et al., 2018)



After the extraction, it was pursued the information through a deep learning model which has six neurons in the input layer, representing the extracted features and, two neurons in 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, based on the recommendations given by each recommender to an item and the trust strength between each trusted recommender and the active user. With the calculated predicted rating scores, the proposed model recommends the healthcare services with the highest predicted rating score to the active user (YUAN et al., 2018).

4.1 COMPARATIVE ANALYSIS

The studies are compared in Table 6, aiming to identify the different solutions for the predictive problem in Health Recommender Systems.

The collaborative filtering method is present in four approaches, and only two studies adopted the Knowledge Filtering. Techniques such as Fuzzy and Neural Networks were adopted, as well as approaches that mix models of recommendation and prediction as in Mustaqeem et al. (2017) and Deng et al. (2018).

The use of IoT devices is increasing in the last years. Ali et al. (2018) propose an architecture which uses the information produced by those devices aiming to assist patients in home environments. The application of data produced in real-time health systems demonstrates the need to develop models capable of processing information in a more agile way. In critical contexts, like in the healthcare area, getting faster results can save lives and prevent the development of diseases. A factor that is related to this, is the information and symptoms classification. Approaches as fuzzy logic (ALI et al., 2018; THONG et al., 2015a), and neutrosophic sets (THANH; ALI et al., 2017) can correctly

Table 6 – Related works comparison

Paper	Filtering	Prediction	Object	Techniques
Thong et al. (2015a)	Collaborative	Prediction of diseases	Diseases	Intuitionistic fuzzy; picture fuzzy clustering
Yuan et al. (2018)	Collaborative	Trust/distrust	Healthcare services	Deep learning
Ali et al. (2018)	Knowledge	Patient's condition	Foods and drugs	Type-2 fuzzy logiz; Fuzzy ontology
Deng et al. (2018)	Collaborative	Review rating	-	Gaussian mixture model; Neural network
Mustaqeem et al. (2017)	Collaborative	Heart disease	Heart disease	SVM; Multi-layer perceptron; Random forest; Statistical probability risk score
Thanh, Ali et al. (2017)	Knowledge	Prediction of diseases	Diseases	Neutrosophication; Algebraic similarity; Clustering

categorize the values, in most cases. Correct classification of features can generate better predictions. Aiming to achieve that goal, in our solution we propose an architecture that is designed to automatically get information through IoT devices. Their use can benefit citizens and patients by monitoring them and making early predictions.

Mathematical approaches are addressed in the prediction methodology, as in (DENG et al., 2018), that use Gaussian distribution to compose the proposed model. The mixture of approaches and different models, usually perform better than individual ones, which can generate more accurate results, and in the health area, this is extremely important. The extraction and treatment of missing values and textual information to process in those approaches are presented as critical, where the data is directly associated with the final performance of the model.

Some filtering approaches perform better than others. Trust and distrust relations show the potential of collaborative filtering, by identifying the items that could be more adherent to users (YUAN et al., 2018). This information processed by intelligent methods, as deep learning algorithms can classify and identify diseases on a scalable level.

Different contexts of application are observed in the works. Mustaqeem et al. (2017) use statistical analysis and classical machine learning methods to predict the type of heart disease in a patient. They consider the treatment of clinical data, to later apply the classical machine learning algorithms. This data training is observed in most of the works that adopt those algorithms where it is necessary due to missing and noise values in the datasets. Another important thing is the size of the dataset to work with and how old is the data.

Thong et al. (2015a) conclude that most machine learning methods failed 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 reinforce, as Deng et al. (2018), that the combination of methods, especially fuzzy sets, and machine learning approaches is a good alternative to eliminate these disadvantages.

The evaluation is performed in most cases with information retrieval metrics, as precision, recall, ROC, RMSE, MAE, among others. These metrics can show how accurate is the adopted model and how close the recommendation is related to the reality of the user's interest and needs.

In general, the presented works have applicability in eHealth, to assist the prediction of diseases and forms of treatment, where different types of predictive approaches are addressed, especially machine learning ones. Its use in recommender systems is a viable alternative because it offers promising results for prediction, which allow a greater gain in recommendation accuracy.

Although it considered the classification and identification of diseases, none of the related works presented approaches that focus on case prioritization. Final remarks are made in the next section and the differences between the works presented in this chapter and the proposed research are discretized.

4.2 FINAL CONSIDERATIONS

The proposals analyze the efficiency in recommendations and how much they are affirmative for the users who receive them. Recommender systems are increasingly being used in the healthcare area, because of the large amount of data and content to be available on a daily basis. With the use of IoT devices in the health context, this amount of data are quickly generated, which reinforces its management, analysis, and processing. The works have focus, mainly, on collaborative filtering to obtain greater accuracy in their predictions.

Construct a model that fits into the work context is crucial to get good predictions with higher accuracy, which allows the system to perform well. Predictions are part of the recommender system architecture core. The combination of techniques is largely explored in the recommender system area, which aims to improve system efficiency. Thinking on that, our approach uses a strategy that combines different models to improve the prediction results.

Finding the gap in these works, the purpose of this research is focused on recommender systems for recommending actions to patients when requiring immediate attention based on their current condition. With the emerging development of IoT devices and smart devices, we benefit from data produced by these devices. However, we face new challenges to manage the produced data and their use of decision taking.

The proposed research makes use of data provided by those devices to investigate and design a recommender system capable of generating recommendations based on observed characteristics to efficiently respond to risk cases, to identify patients requiring vital attention.

5 HEALTH-PRIOR ARCHITECTURE

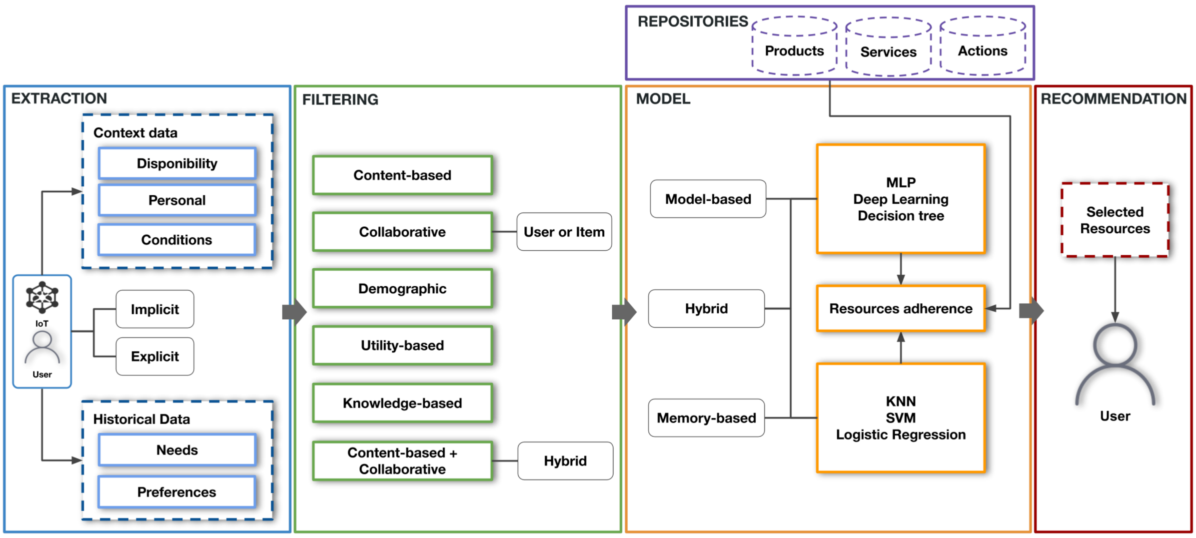
In this chapter, it is described the proposed research work, named Health-PRIOR architecture. Therefore, it is presented the entire architecture and adopted technologies.

5.1 RECOMMENDER SYSTEM PROPOSAL

Aiming to make recommendations based on the severity and level of emergency in the healthcare domain, we designed a recommender system architecture that is initially dedicated to alert health professionals to make decisions about patients' risk cases. The architecture focuses on indicating cases that need prioritization or attention and recommend information that could help in that case. The main goal is to use IoT data to compose the patients' profile and periodically monitor patients' health by predicting possible worsening of their overall condition.

Figure 9 presents the proposed conceptual architecture. The architecture follows the concepts of recommender systems architecture (LOPS; GEMMIS; SEMERARO, 2011) and it is defined in five layers.

Figure 9 – Health-PRIOR recommender system conceptual architecture



The first layer is responsible for data extraction. It is designed to collect data provided by IoT devices in real time and historical data of patient's conditions and needs. Thus, static sources as forms, applications, medical records can also be used to compose the patient's profile.

The second layer is responsible for filtering the information, applying to the working context the most adherent filtering types, which were presented in Section 2.2.1. Depending on the working context, one specific filtering could be more indicated than others. The utility-based filtering, for example, is more indicated to recommend objects based on their utility to users as well as knowledge-based filtering is to recommend more adherent inferred objects to users.

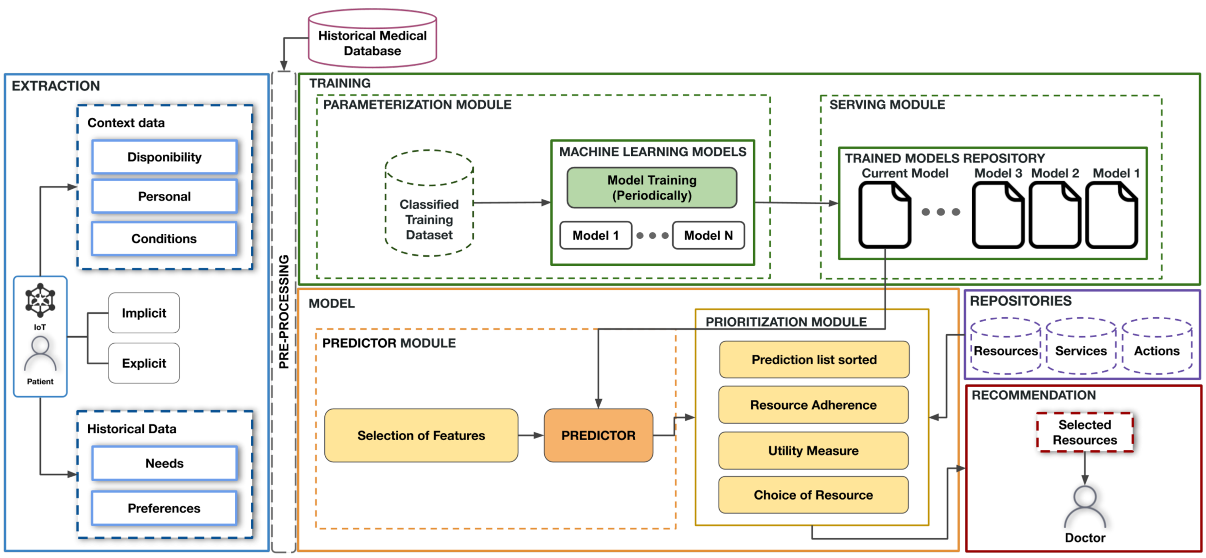
The third layer contains the predictive models, which are responsible to predict the resources and their adherence to users. It can be performed by a memory-based model, a model-based model or even by a hybrid model, representing a combination of both.

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, Figure 10 shows the proposed architecture of this research. It includes the capturing and processing data flow aspects as well as the need for prioritization in patients' cases, which is an important module for the context of the current research. We designed the Health-PRIOR architecture containing six main layers.

Figure 10 – Health-PRIOR architecture



The **Extraction layer** is responsible for capturing data and configure the user's profile. The data can be provided by IoT devices connected to people or facilities in various environments and from medical notes. The extraction can be made implicitly or explicitly. The use of smart devices allows the extraction in an automated way, much more precise and accurate with the data.

The **Pre-processing layer** is responsible for filtering the data. It is where the data is treated by replacing the missing values and dealing with outliers. It also is responsible for removing the noise to construct a cleaned version of the dataset. It converts the raw data received from smart devices into clean data. This layer preprocess the historical medical database for training purposes and the data came from IoT devices.

The model definition and training are made in the **Training layer**, which captures the historical medical data periodically and trains a new set of models with newer data, providing a closer representation of the real world, applied to predictions. Predictive models are based on data training to make future predictions, that is, they analyze which pattern the data follow to serve as a predictive basis. This layer has two modules, the **parameterization module**, which is responsible for finding and setting the best parameters, and also training the models, periodically. The other module is represented by the **serving module**, responsible for storing the trained models in a repository and making available the most recent trained model to the Model Layer.

According to Chen et al. (2017), large health data sets, relatively old to the prediction context, tend to present lower assertiveness compared to more recent data and in a smaller amount. Considering that, the architecture considers the most recent historical information to train the models.

The **Model layer** is responsible for the trained models execution with the received preprocessed data in the **predictor module**. The data are classified and the patients are sorted by criticalness in a prioritization list based on the result, in the **Prioritization module**.

The **Repositories layer** stores all recommendation objects that can be recommended, which includes resources, actions or services, depending on the final purpose of the recommender system. Those information are files, content and past analysis, that can be used as recommmendation to the system.

Finally, the **Recommendation layer** presents the recommendation object to the user, allowing them to make a decision or an action. Having the predictions and the actions made by physicians, the historical dataset is refueled throughout time allowing to train a new model with newer data. This tells us the correct classifications and actions for each set of features received from IoT devices in the past. This automated system can be applied in several contexts, especially in the health area, which by the constant monitoring can provide a better life for citizens and an early diagnosis for various diseases.

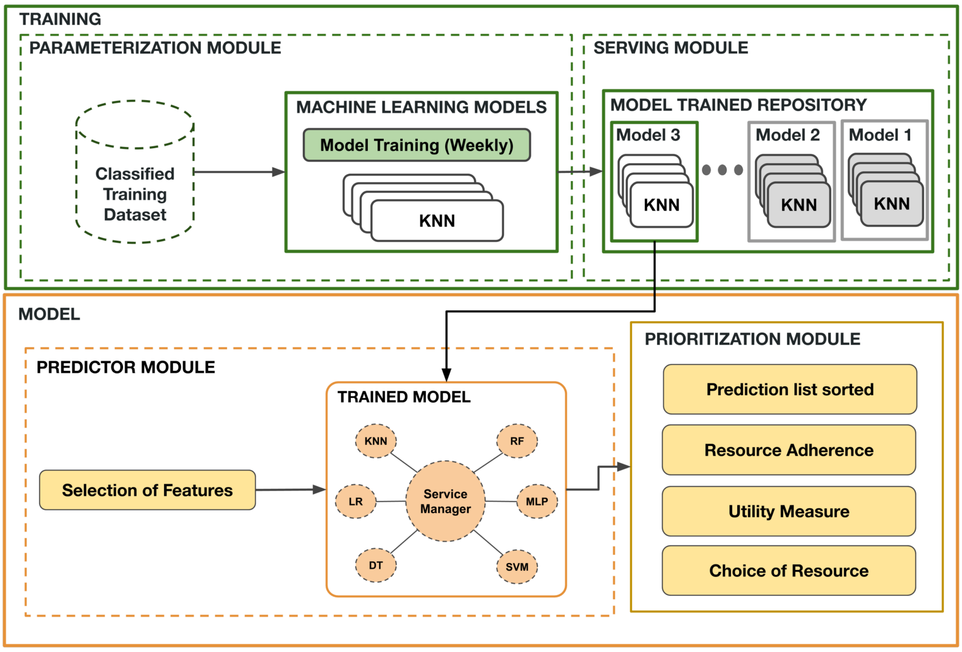
The following section will describe in more details the data flow and how the predictions are made by using the architecture.

5.2 RECOMMENDATION MODEL

To the proposed recommender system architecture, we have designed a prediction strategy combining classical Machine Learning models to work as an ensemble model capable of autonomously solving classification problems. The model is represented by the

Predictor module and it is responsible for making the predictions. Figure 11 shows the two main layers that contain the data flow.

Figure 11 – Health-PRIOR prediction process



The **Training Layer** is responsible for storing the classified data and train the individual models. Here, each model is intended to be trained separately with the same data, generating different predictors. Each model is tested with several parameters aiming to maximize its prediction. Once selected the best parameters and trained, all models are stored in the serving module to be available for the **Model Layer**.

The ensemble model was designed to combine different classic Machine Learning models, implemented as autonomous software services, with reactivity, intelligence, and social characteristics. They are suppose to deal with requests and to predict.

The **Model Layer** is responsible for making predictions in real-time. For instance, the data received from IoT devices are preprocessed and in the model layer its features are selected and processed by the ensemble model. The **Predictor Module** has a service manager that coordinates each model as a service. Aiming to synchronize all models, the service 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 (POLIKAR, 2012) is applied to evaluate the results that most models have in common to get the final prediction with higher certainty.

The **Prioritization Module** sorts the results by level of severity and gets the recommendation objects that are most adherent to each prioritizing case. After that, it can present the selected object to the user, as well as the list of predictions to the doctor. This module can be parameterized to work based on the doctor's concerns or by the patient's needs.

5.2.1 Models

To compose the set of models in the architecture predictor module with the most adherent models for the supervised classification problem, we have identified in the literature those that are classically used. The chosen techniques are showing as follows:

Decision Tree: is an abstract structure that is characterized by a tree, where each node denotes a test on an attribute value, each branch represents an outcome of the test, and tree leaves represent classes or class distributions (HAN; PEI; KAMBER, 2011).

KNN: K-Nearest Neighbors is a classical and lazy learner model. They are classifiers based on learning by analogy, which measures the distance between a given test tuple with training tuples and compares if they are similar (HAN; PEI; KAMBER, 2011).

SVM: is a classification algorithm that works with linear and nonlinear data (HAN; PEI; KAMBER, 2011). By using kernels, this model transforms the original data in a higher dimension, from where it can search and find a hyperplane for the linear optimal data using training tuples called support vectors.

Random Forest: is a classifier consisting of a collection of tree-structured classifiers which fits a number of classifying decision trees to various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting (BREIMAN, 2001).

Logistic Regression: is a generalized linear approach that models the probability of some event to occur and is modelled as a linear function of a set of predictor variables (HAN; PEI; KAMBER, 2011).

Multi-Layer Perceptron: This model is represented by a neural network that contains neurons to pass the data through it. The model can learn a nonlinear function approximator for either classification or regression.

5.3 FINAL CONSIDERATIONS

This chapter described the proposal for this research. The description was made aiming to detail the adopted technologies and the development aspects present in the architecture.

The models described in the previous section were the ones used in the evaluation of the proposal. However, the architecture can fit different types of models to work in an ensemble way, through autonomous services.

The evaluation process and the case studies pursued to evaluate the architecture will be described in the next chapter as well as the results found.

6 HEALTH-PRIOR EVALUATION

To improve and evaluate the proposal, we made three Case Studies, carried out in different domains and real scenarios. Each case study contributed to the final architecture design. We first evaluated the proposed model in the educational context focusing on the ensemble model and its accuracy. Based on the first results, we evaluated the model in a health context with the data flow that the architecture was designed for. Including the aspects of training and processing time to receive the prediction results. Finally, we evaluate the improved architecture in a third case study, which is in the health context with real patients' data focusing on the full aspects of the architecture.

According to Wohlin et al. (2012), the main advantage of a case study is the ease of planning it and also the characteristic of being more realistic, while at the disadvantage the authors present the difficulty of generalizing and interpreting the obtained results. The following sections will describe the three case studies.

6.1 EDUCATIONAL CONTEXT CASE STUDY

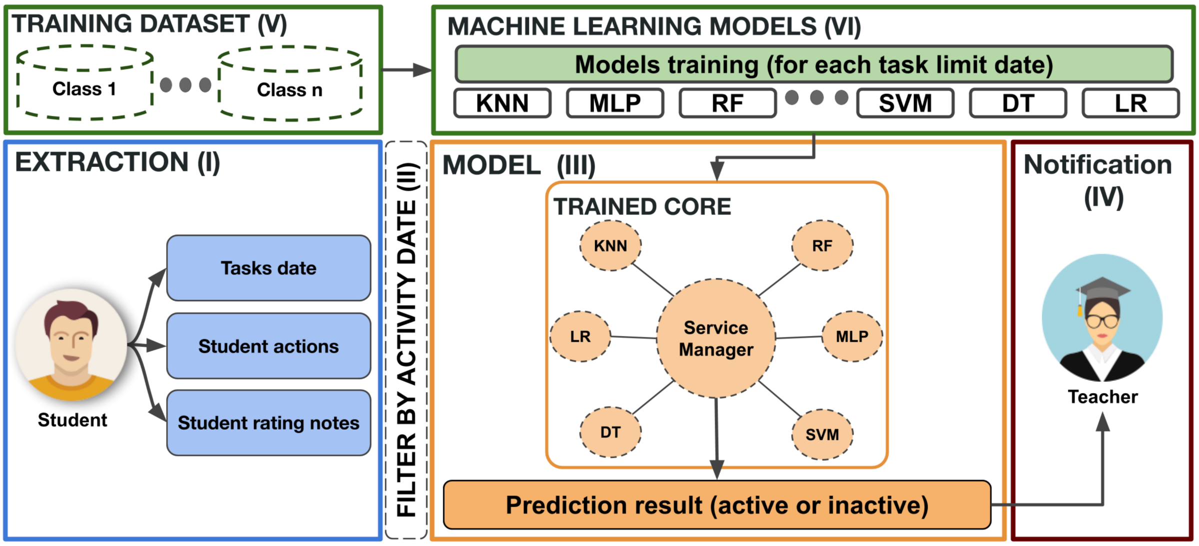
To understand the solution applicability in a real context, a case study was carried out primarily in an educational context where we could see how the prediction process worked and how we could improve it. The proposed ensemble model was instantiated and we used real data from a Computer Science curriculum class, with its past and current data. The Fundamentals of Information Systems class is offered to most Computer Science courses and its main goal is to prepare the students to recognize the importance of information systems in different organizations and identify different possibilities for their implementation.

For this Case Study, we selected two classes from the online Computer Science Teaching course, 2018 and 2019. The course uses the Learning Management System Moodle and the teacher and students' interactions are based on messages, forum, chat and wiki.

6.1.1 **Educational Architecture Model**

The main goal of this study is to evaluate the Ensemble Model used as a core by an Early Warning System (EWS) to predict student's dropout (BRAZ et al., 2019). The proposed model of this case study is illustrated in Figure 12. The model can predict if a student presents a risk to drop out of class and the system can notify teachers about this possibility. The model is dependent on a data set in the work context; it can be used in different classes, requiring only a configuration compatible with the class model, such as the total number of tasks, the assigned values and so on.

Figure 12 – Educational architecture



The Educational EWS is composed of six main layers:

(I) **Extraction Layer** is responsible to extract all student's information needed to compose his/her activity profile. In Educational context, information such as student actions and rating notes are important to understand the student activity as well as his/her performance in the class.

(II) As we want to predict the student dropout probability, it's necessary to filter his/her performance throughout the course. This filtering is made by the **Filter Layer** which filters all students' activities by limit date, pre-established by the teacher.

(III) The architecture core is composed of an **Ensemble Model** that combines different classic Machine Learning models, in view of offering a more accurate result with a higher certainty. Each model is intended to be an autonomous service capable to deal with requests and predict the student dropout probability. In the model core, we have defined six different machine learning models to compose the main autonomous services. To synchronize and combine them, we use a seventh service, that works as coordinator of the proposed model. All the models adopted in this case study are supervised because the issue is a classification problem, which consists of indicating if a student can drop out of the class. The chosen ensemble method was the Voting Ensemble Method by averaging of the positive predictions. This approach aims to minimize the difference between the models' prediction and maximize the ensemble model assertiveness (KUMARI; JAIN; PAMULA, 2018). The averaging process takes into account both classifications, and if the result express that the student is *inactive* a notification is triggered.

(IV) a notification is sent to the teacher responsible for the class by the **Notification Layer**, where the teacher can intervene to prevent the student from dropping out of the class.

The other two layers (V and VI) are responsible for the machine learning models training with past data from another previous class (2018), which shares the same structure of measuring student performance.

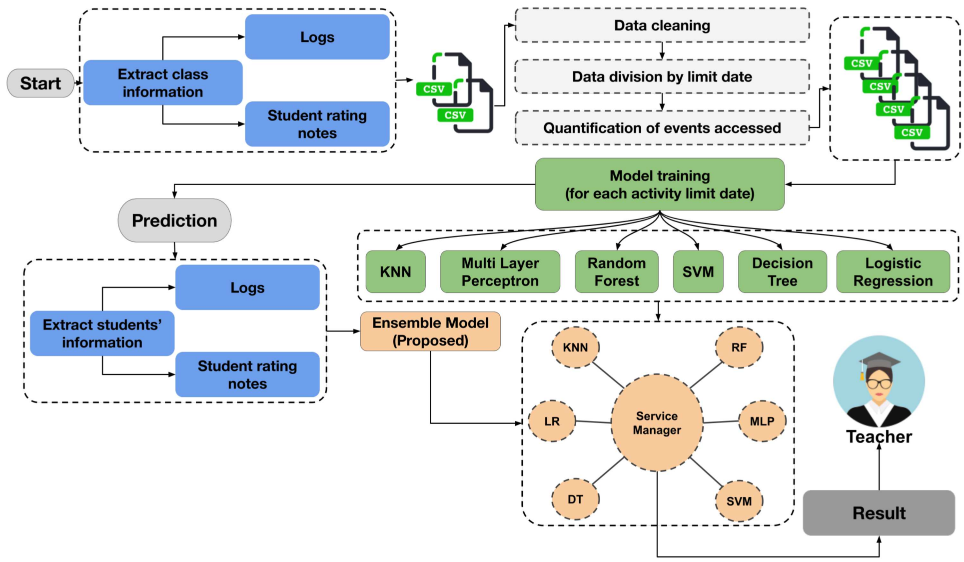
6.1.2 Data Description

To pursue the study, we've asked permission to a professor at the Federal University of Juiz de Fora to collect data produced by the Learning Management System (LMS) Moodle classes that are under her responsibility. Each class schedule includes various events triggered by the students and their rating notes. With the professor's permission, the data was collected and it was treated to offer a better environment to the experiment.

The treatment of data was made by setting the missing ones with zero value (0) and replace all non-English characters to English ones. After preprocessing the data, we split the past data by date, generating a file for each limit task date, classifying the features as *active*, which represents the student interaction with the system and *inactive* otherwise.

Each generated file is responsible for training the proposed model with the respective data for each task to be delivered. The whole notification process is presented in Figure 13.

Figure 13 – Complete notification process



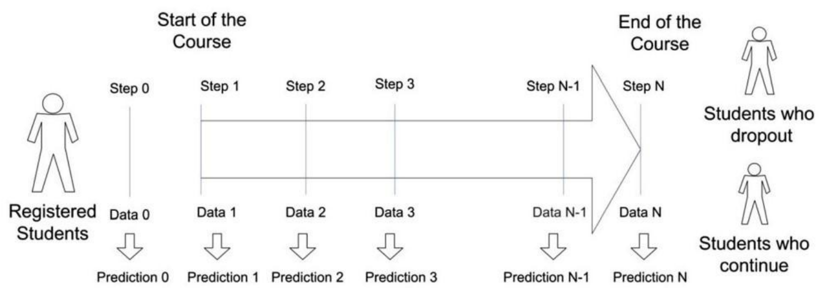
The process starts with the information extraction from logs and students' rating notes in past classes. The data are stored in CSV files which will pass by a preprocessing step. The data are divided by the limit date for each activity, aiming to generate different predictors for each activity throughout the class.

For each data set, the models are trained to be available to the ensemble model, and the predictions are made with data extracted from the current class. If the ensemble model predicts inactivity in the student profile, the teacher is notified, being able to take a decision related to the student's context.

6.1.3 **Ensemble Model**

We adopted the methodology of Dropout prediction followed by (MÁRQUEZ-VERA et al., 2016), showed in Figure 14. The process consists in to predict the student's dropout possibility throughout the course or class. Thus, for each delivered task, a prediction can be made and different instances of the proposed model can be trained to act at different times throughout the class; thus making predictions of potential student dropouts.

Figure 14 – Dropout prediction methodology (MÁRQUEZ-VERA et al., 2016)



Aiming to get the best configuration to our problem, we combined the different model's parameters by the Cartesian Product and generated a CSV file with a total of 36.000 combinations. After that, we trained the data for each combination until getting high accuracy in each one. This process generated a total of 37 different configurations. We set each model with the configuration that was more representative and more accurate for it. The techniques used are showing as follows:

Decision Tree: with the generated CSV file we used the *entropy* as the criterion and maintained the default values to the other parameters.

KNN: to this model, we set *K=3*, which represents the number of neighbours to compare. And the type of distance used was *euclidean* distance, the other parameters we maintained the default values.

SVM: to this model, we set the *linear* function as kernel type and *gamma* value with 100, as well as the *regularization* equals to 1.

Random Forest: as a parameter, we utilized *entropy* as the criterion. The number of estimators was set to 60.

Logistic Regression: to this model, the regularization was set with the value 100 and the *liblinear* solver to handle the data, which is recommended in the framework documentation as well (PEDREGOSA et al., 2011a).

Multi-Layer Perceptron: in this model, we used 3 hidden layers with 6 nodes each and the *relu* activation function, one output layer to positive cases with the *relu* activation function and the input layer has the number of features treated in the dataset.

Also we used the *accuracy* as the metric, the *binary cross-entropy* as loss function and the *Adam* optimizer.

The test sample is responsible to show the results of predictions and it was composed of the current class data. To evaluate the proposed model, we chose classical statistical methods RMSE, MAE to evaluate the error and F-measure, precision, and recall measures to evaluate accuracy. We compared the predictive models adopted together and separately.

In Equations 6.1, 6.2 and 6.3, the TP represents the *true positive* classifications, which means that the prediction has a positive result and the classification was positive (SOKOLOVA; LAPALME, 2009). The FP represents the *false positive* classifications, which means that the result was positive, but the correct classification was negative, and theFN represents the *false-negative* for otherwise. The RMSE and MAE evaluation uses the available functions in the Sklearn framework. These methods of evaluation are more adherent to supervised learning, which is the type used in this study.

$$Precision = \frac{TP}{TP + FP} \tag{6.1}$$

$$Recall = \frac{TP}{TP + FN} \tag{6.2}$$

$$F - measure = (1 - \beta^2)\frac{precision \cdot recall}{\beta^2 precision + recall} \tag{6.3}$$

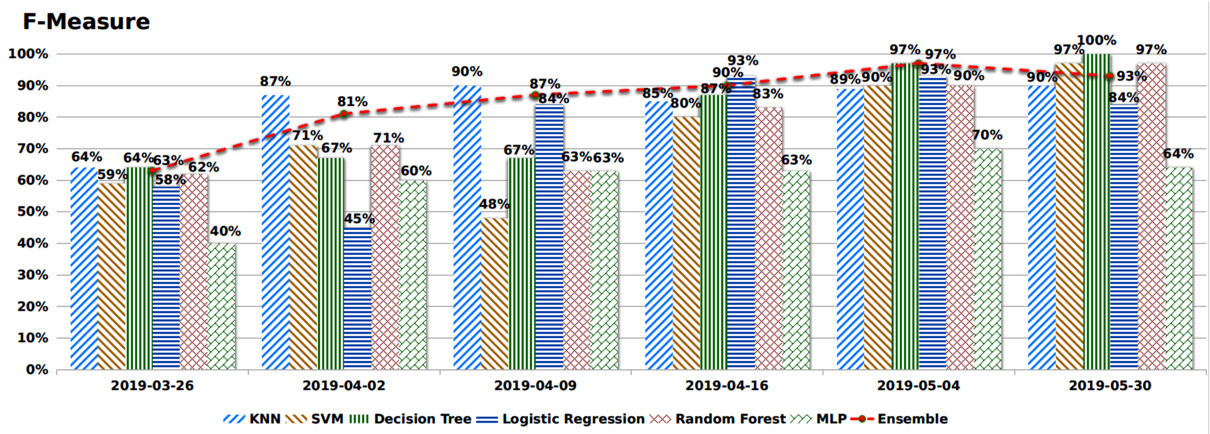
The original weight to the F-measure equation is represented by $1 + \beta^2$, where β^2 configures the bias of each metric in the equation. Zhao et al. (2018), define $\beta^2 > 0$ as a balance factor between precision and recall, and, when $\beta^2 > 1$ the F-measure is biased in favour of recall and otherwise the F-measure considers precision more than recall. In such a way, we choose the value *2* in the F-measure metric to express an equal contribution of precision and recall (PEDREGOSA et al., 2011b).

6.1.4 **Prediction Results and Analysis**

Class dropout is a real challenge, and considering the distance education, we have to deal with the student disengagement since the beginning, as most of the time, they even make any task. With the data separated and the models trained, we executed the experiment following the methodology presented previously. The student's data was collected to predict the performance on the course and the same data structure of the models was maintained.

We ran the experiment in a computer with Ubuntu 18.4 LTS; 8 GB of RAM; Intel Core Processor i7-7500 and 220 GB of SSD. For each task delivered in the class a prediction was done. When the results output the inactive class, a notification is triggered and it is sent to the teacher. We evaluated the obtained results using Equations 6.1, 6.2 and 6.3, and we compared them to visualize and understand the differences between models. The main accuracy results presented in Figure 15 show that individually, the models present a linear structure where the observed accuracy has not increased along time in ascending form, presenting variations. However, in our solution, which combines the models using the voting ensemble method, the accuracy increases gradually throughout time as the class goes on and students have more activities in their schedule.

Figure 15 – F-Measure metric



The main accuracy results presented in Figure 15 shows that individually, the models present a linear structure where the observed accuracy has not increased along time in ascending form. However, in our solution, which combines the models using the voting ensemble method, the accuracy increases gradually throughout time.

Individually, some models present a better performance in specific metrics, such as the Decision Tree model in F-Measure. However, the overall result shows a gain of performance and certainty in predictions through our ensemble model.

The error measures presented in Figures 16 and 17 shows that the MLP model was the only one to present the same average in error in both metrics through time. The other models, although they do not present a bad performance, also do not present a descending linear structure in error measure as ours.

The class had 37 students, from 12 different cities. The main evaluation activities included participation in two forums, two individual tasks, and a group final task. The second task includes a visit to a school to identify its Information System features and the group activity proposes a School Information System component. The students made the activities individually or in groups composed of two or three members. The evaluation process also includes a peer review of the final task proposal presentation.

Analyzing the results and the data, it was found that 29.72% of the students did not make any task. 16.20% made at least one task. 8.10% had made 3 of 7 tasks. 27.00% made 6 of seven tasks and failed in one of them. Finally, 18.91% made all the tasks.

Figure 16 – MAE measure

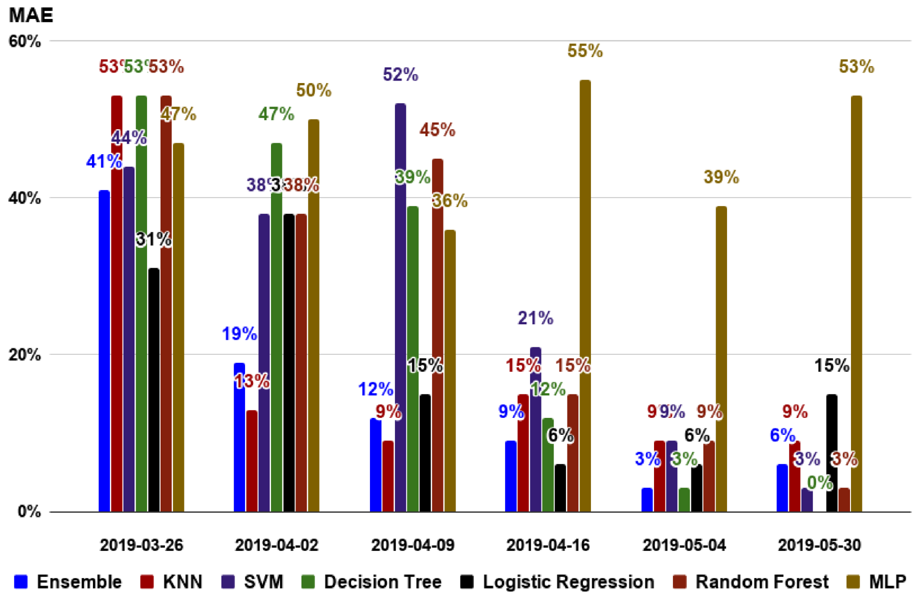
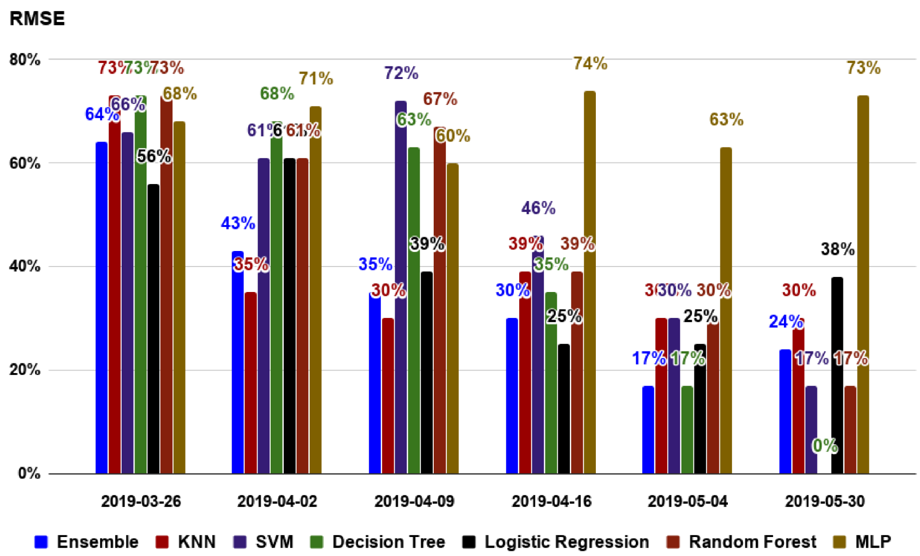


Figure 17 – RMSE measure



Considering the results we can say that at least 43.24% of students should have had special attention and motivation and at least 8.10% could be approved finishing one or two more tasks.

Considering the results of the proposed model we can infer that it would probably help to avoid so many dropouts in the class as:

- All the students that did not make any task would be notified after all tasks deadline;
- 80% of the students that made 1, 2 or 3 tasks would be notified after all tasks

deadline;

- All the students that made any task would be notified at least once;
- All active students would not be notified by the system.

6.2 CHRONIC KIDNEY DISEASE CASE STUDY

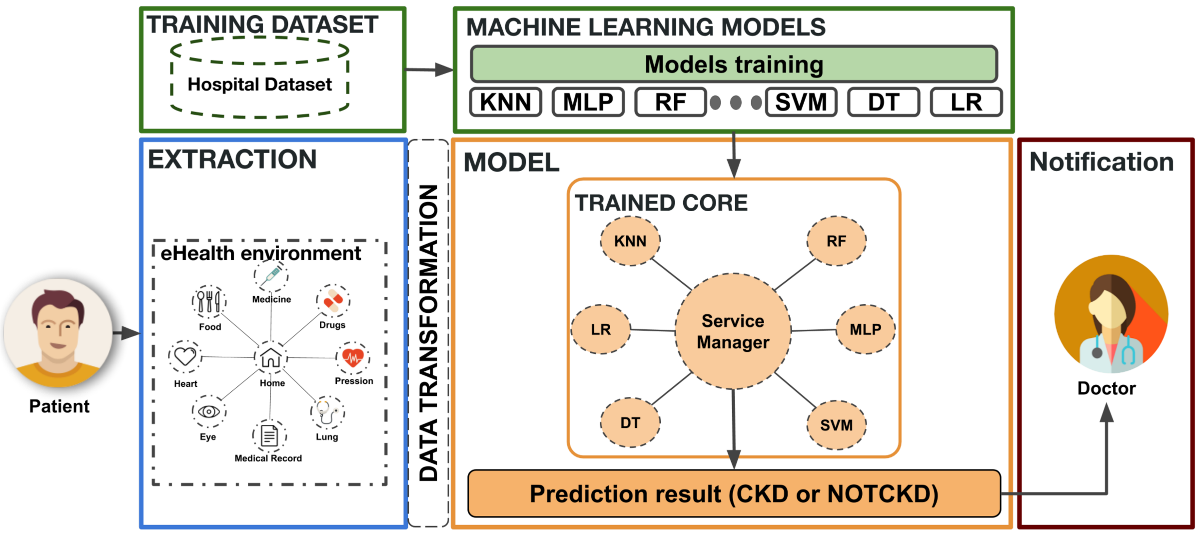
This Case Study evaluates the Ensemble Model in a Chronic Kidney Disease (CKD) context in a Smart City environment. The focus of this evaluation is to understand how recommender systems and prediction models can be merged to bring more health and quality of life to citizens in a smart environment.

Chronic Kidney Disease has received increased attention from the international scientific community after studies showed its high prevalence. The nonattendance of symptoms in patients in the early stages of CKD requires that physicians maintain an adequate index of suspicion in all patients, especially in those with medical or sociodemographic risk factors for CKD (BASTOS; KIRSZTAJN, 2011).

6.2.1 Architecture Model

The instantiated architecture is similar to the previous case study, but we evaluate aspects as automated notifications and model's training time as well. Figure 18 presents the CKD Health-PRIOR architecture and the strategy that was used in this specific context.

Figure 18 – Chronic Kidney Disease Health-PRIOR architecture



The **Extraction Layer** was altered to get patients' information such as blood information, historical data and all data that can configure the patient's profile. All those data are intended to be captured by IoT devices connected to patients. The **Training**

Layer considers only one historical dataset that has the training information as well as the classifications for each case. The training process is made once because we have only one dataset for training and all scenarios are based on the same configurations.

The main difference between both case studies is the fact that we can pursue predictions with different sets of data. Each set represents a prediction context, this includes a set of characteristics that represents that context, specifically.

The **Data Transformation Module** was introduced to translate the signals captured by IoT devices into raw values, remove noise, treat missing values and transfer the captured data to the system core. Wearable devices are examples of IoT devices capable of automatically capturing information. Once that is made, the data is recognizable by the model core. In this module, the data is formatted and organized in a way that the predictive model can use it.

The **Machine Learning Models** Layer is responsible for training the models used in the ensemble model and serves the Model Layer, which uses the trained models in the predictions. The **Model Layer** process the treated data from Data Transformation Module.

The last layer is the **Notification Layer**. It is responsible for notifying doctors about their patients' conditions.

The system model can handle data and offer an output **yes** or **no** for the prioritization necessity. This behavior is expected in a recommender system to pursue the recommendations. Knowing the necessity to receive care, the recommendation module can search for adherent resources in health repositories and recommend it to a doctor.

The averaging ensemble process takes into account only the predictions classes that are related to the recommendation trigger, which is the classification that intended to be recommended, in our case, the CKD classification. If for a set of patient's characteristics the result presents that the patient can develop the disease, the system must recommend related contents to the doctor, such as actions or services in that context.

This recommendation model was adopted since we are dealing with possible cases of chronic disease, whose early diagnosis helps in the treatment and prevention of its development. Positive cases of recommendation cannot be neglected.

6.2.2 Data Description

To validate the instantiated architecture model, we have used the open dataset (DUA; TANISKIDOU, 2017), including the features that can characterize possible patients with Chronic Kidney Disease. The dataset enables the classification of a set of features that can be captured by IoT devices. So, we intend to simulate that behaviour by analyzing the adherence of these features in the proposed model, by sending them through *HTTP* requests.

Each file has all the attributes, totalling 25 features, such as *"blood pressure"*, *"age"*, *"hemoglobin level"*, *"blood glucose"* and so on. These features are classified as **ckd** (Chronic Kidney Disease) or **notckd** (not Chronic Kidney Disease)[1].

The dataset contains a considerable amount of missing values, to treat them, we set the the missing values with the arbitrary value *-99999*. The training process uses all the features present in the dataset, which are significant to the classification. They are described in Table 7.

Table 7 – 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)

We separated 40 instances of the dataset for evaluation purposes, 20 for each classification, which was not used in the training stage. The rest of the instances (360 instances) was used to train the models present in the architecture. Each generated file is responsible for training the proposed model with the respective data for each task to be delivered. The whole prediction process will be presented in Section 6.2.4. Aiming not to get a skewed result, we made a python script that separated randomly the instances into two files, one for testing and the other for training. We did that 5,000 times.

The methodology followed in this study consists of split the data from the open dataset into two files for training and testing purposes. After that, data cleaning is applied

[1] https://github.com/FelipeNb/SBCARS-eHealthRecommenderSystem

to remove noise and treat missing values. The intention is to simulate a real context in the health area, which can demonstrate the behaviour of the system to recommend actions or services to the doctor viewing the assistance of a patient.

6.2.3 Ensemble Model

In this study, the model responsibility is to predict the patient's possibility of developing CKD and to process the transformed data. Like the previous study, each model is intended to be an autonomous service capable to deal with requests and pursue predictions. We have made empirical tests and we followed some recommendations from the Sklearn Framework documentation (PEDREGOSA et al., 2011a) about the model's parameterization, viewing a better model construction. The parameters used for each model in this study are described as follows:

Decision Tree: We built empirical tests and the maximum tree depth value adopted was 10 and the default values to the other parameters were maintained.

KNN: We set K=5, which represents the number of neighbours to compare. This number was chosen because it is the most common in literature. And the type of distance used was *euclidean* distance, the other parameters we maintained the default values.

SVM: We have chosen the *sigmoid* function as a kernel type because it had better behaviour in our previous tests.

Random Forest: We utilized *entropy* as the criterion of the function, as it is recommended in the framework documentation when you are interested in information gain (PEDREGOSA et al., 2011a).

Logistic Regression: We chose the *liblinear* solver to handle the data, which is recommended in the framework documentation (PEDREGOSA et al., 2011a).

Multi-Layer Perceptron: We used 3 hidden layers with 6 nodes each one and the *relu* activation function; one output layer to positive cases; the input layer has the number of features in the dataset, which is 24 features. This configuration had better performance with the dataset used in our empirical tests.

As the machine learning models work with numerical data, the heterogeneous and categorical data were replaced with numerical ones. The algorithm used to do this task was the Label Encoder from the sklearn framework (PEDREGOSA et al., 2011a). The algorithm gets all values and enumerates them, transforming them into a numerical value.

The chosen ensemble method was the Voting Ensemble Method by averaging of positive predictions. In that way, we evaluate the predictions that conclude that the patient can develop the disease. This approach aims to minimize the difference between the models prediction and maximize the proposal model assertiveness.

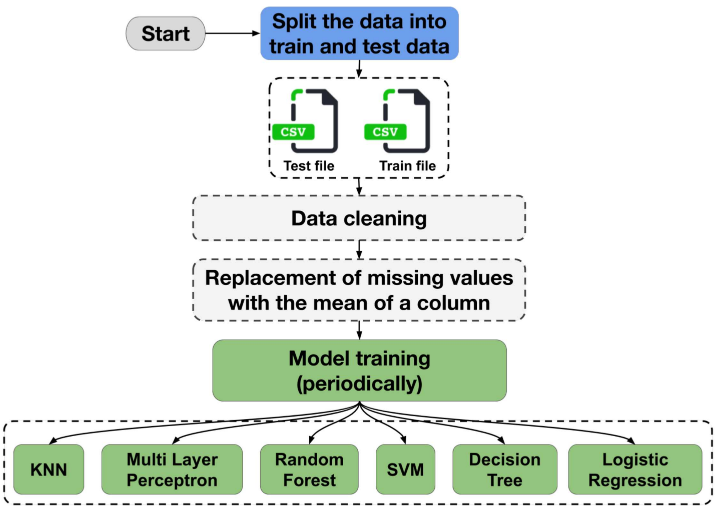
In Machine Learning it is necessary to define the training set used in the model, and the test set used to evaluate the model performance, i.e, a test sample is responsible for showing the results of predictions when the model is presented for unknown data. So, as the original dataset has 400 instances, we split the data and we selected 360 instances as our training set and the other 40 instances were used as a test set.

To measure the assertiveness of the proposed model, we have chosen classical statistical methods (RMSE and MAE) to evaluate the error and precision, recall, and F-measure measures to evaluate accuracy. We compared the predictive models adopted together and separately.

6.2.4 Execution

Aiming to understand the behaviour of our model in that context and validate it, we perform this case study in two steps. First, we clean the data and train the machine learning models that we use in our ensemble model, this process is represented in Figure 19.

Figure 19 – CKD training process

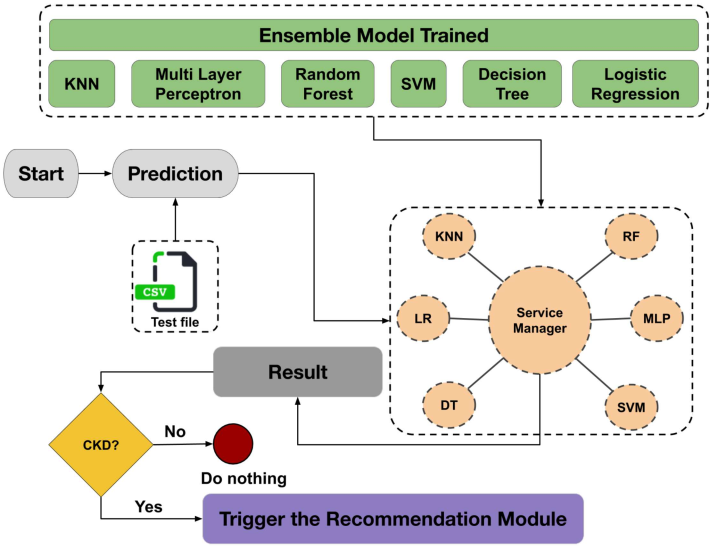


With the data separated, we ran the implemented services. First, we started the six main autonomous services in parallel, training them with the same treated dataset. With the models trained, the proposed architecture model is ready to be evaluated.

The features present on data are unbalanced, being necessary to balance the number of instances to assure that the models behave as expected. Aiming to reduce the disparity between the label classes in the database, we chose to use SMOTE (Synthetic Minority Over-sampling Technique), an over-sample technique to multiply the minority instances to balance the classes for training (CHAWLA et al., 2002). This technique balances the number of instances for each classification.

After balancing the classifications, we sent the 40 test samples to the coordinator service which is responsible for activating the others with the passed features values. The coordinator service waits until all the other services finish to execute the models and combines the results by the voting ensemble method. If the prediction result is positive to the disease, the system triggers the recommendation module. Figure 20 presents that process.

Figure 20 – Execution process



It is important to note that each request to the coordinator service is sent in parallel to the other services, this means that the response time is not impaired since the final result for the request has practically the same processing time of individual models.

The architecture model implementation was made in Python by using machine learning frameworks as Keras, TensorFlow, and Sklearn (RASCHKA, 2015; PEDREGOSA et al., 2011b).

6.2.5 Prediction Results and Analysis

According to the specified metrics and the proposed ensemble approach, the evaluation of the predictive models was performed. As we have 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 of RAM; Intel Core Processor i7-7500 and 220 GB of SSD. After executing the experiment for each created file, we made the average of each model by going through all files. The final results are presented in Table 8, and Figure 21 brings the result visualization.

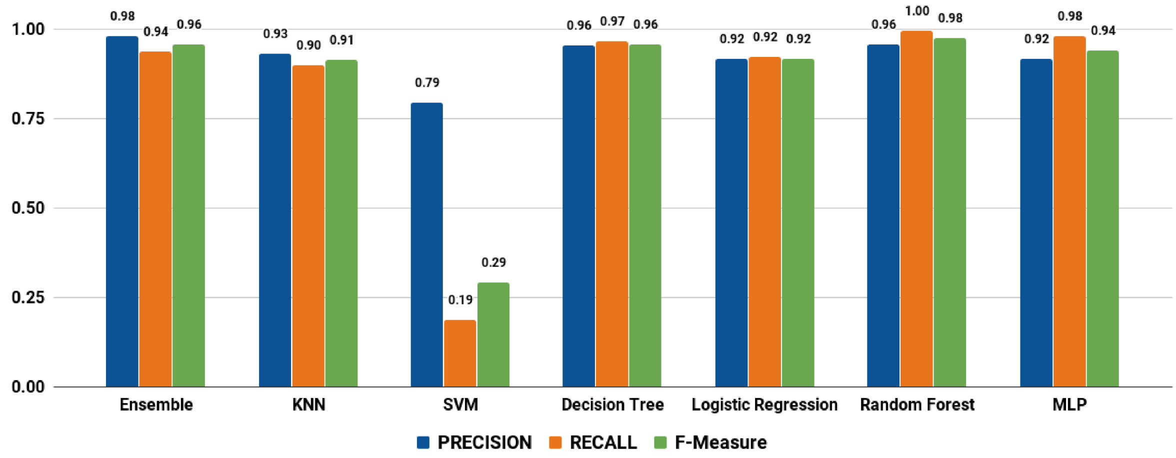
We can observe that in general, all models present a similar average except the SVM model. In that specific model, the accuracy is very low compared to the others, in all metrics except the error ones.

Table 8 – Prediction model 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

Table 8 shows, with high precision, that no method presents the accuracy of 100%, which is good, viewing that this behaviour configures our experiment with no over-fitting in any model.

Figure 21 – Model accuracy results

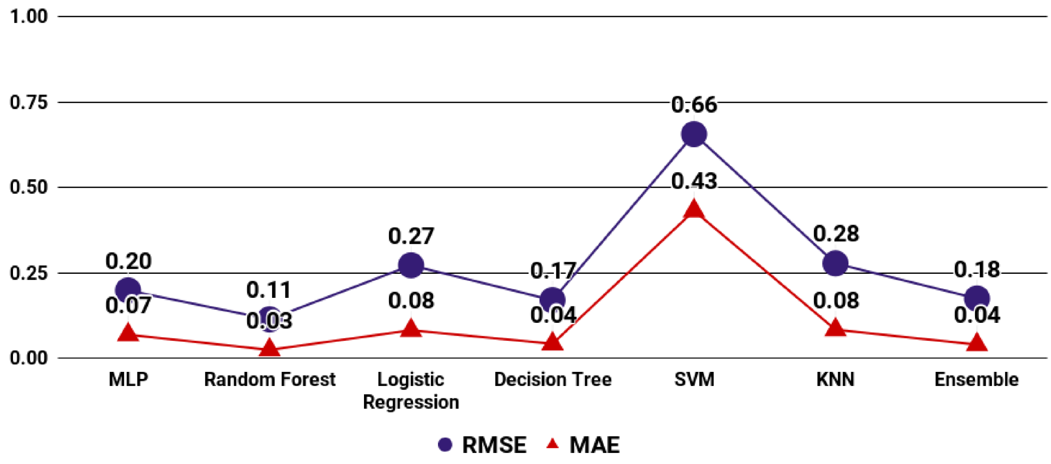


The final result shows the Random Forest model providing a smaller prediction error and high accuracy in comparison with the others. However, if we analyze the general prediction context, we can see that there is a disproportion between the results for the same data set. That is, for the same sample, there are distinct results, depending on the model. This leads to prediction errors, where one model correctly predicts one situation and another does not.

Figure 22 shows the error found in each model and in our approach. It is possible to see, that our model presents a low error rate of 4%. To have a margin of error, in this case, it is statistically well seen and accepted. In comparison, our model has a higher error than other models because it considers the results from all the others, and when a model predicts wrongly, our approach is penalized by that.

Our model, however, maintains a high accuracy with minimal error and provides a higher certainty by considering all the other models in its prediction. The results are promising when there is a certain disparity between predictive models, even though they are used in the same context and with the same data. It reinforces the need to guarantee

Figure 22 – Error measure results



certainty and assertiveness in the predictions through ensemble models, as proposed in this work.

The proposal allows the recommendation to occur, with the weighting of the results of different models, to inform the expert about the case severity and some actions to take. This allows the professionals to decide how to proceed, considering the information present in the prediction and their expertise.

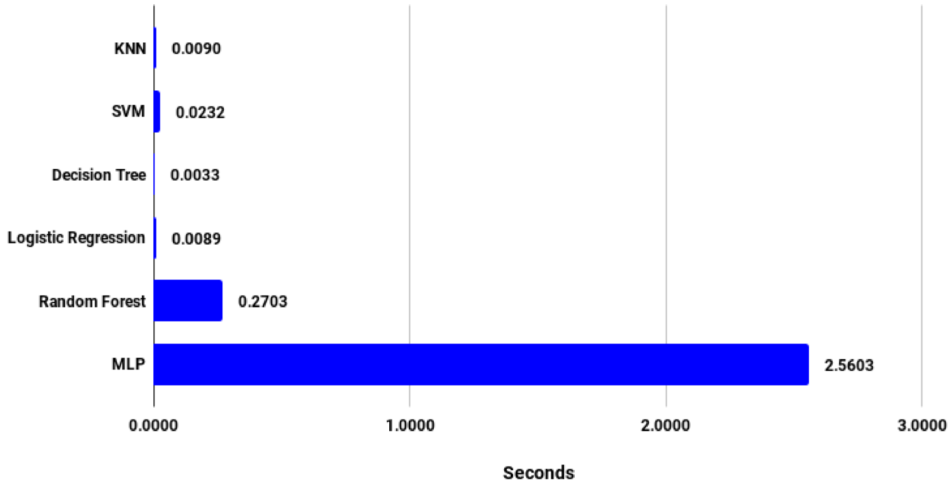
Due to the obtained results, there are indications that the proposed architecture and its model could be a viable alternative, to bring a higher level of certainty to the prediction result to health recommender systems, especially the ones that predict chronic diseases.

The way the models are trained and the configuration used for each one, influence the processing time and the result to be obtained. Figure 23 shows the training time for each model.

The Decision Tree model is the one that presents the fastest training for the used training base and the MLP model the most time-consuming. By making a comparison between the training time and the result provided by the models, we see that individually the model that presents a better cost-benefit is the Decision Tree. In general, the results are consistent with training time, except for the MLP model. As showed by Figure 18, the training process is made once, and the trained model is used in the architecture, which represents no congestion of processing time over time. As the training process of individual models is made in parallel, the training time presented by our proposal is equivalent to the training time of the slower model which is good when it is manipulating a large number of samples (RASCHKA, 2015).

Having a proposal that uses parallel processing represents a gain in time and provides a good way to retrain the models while pursuing predictions.

Figure 23 – Ensemble model training time



6.3 EWOUND CASE STUDY

To evaluate the proposed architecture in a real context, and trying to minimize the flow of people at hospitals and clinics by prioritizing emergency cases, we pursued a case study through a partnership with a Canadian project at The University of Western Ontario, named eWound Project. This project is under the responsibility of Professor Dr. Dianne Bryant in the physical therapy department [2]. The evaluation goal is to test the capacity of our model to generate accurate predictions and to correctly prioritize emergency cases in the healthcare context.

The project brings the necessity of postoperative patients to receive notifications about their healing condition. If a worsening is observed in their treatment, the patient should be notified to go to the hospital/clinic to receive in-person care. Those patients, periodically, have to come to the clinic, after surgery, to check if their wounds are healing properly or not.

Jeffery (2017) approached a new type of form to administer a patient-reported eVisit questionnaire, at two- and six-weeks orthopaedic postoperative cases. The author used the data collected to build a statistical model by using the logistic regression model to estimate the likelihood that an adverse event was present. A notification is made as to whether the patient should be seen by the surgeon in-person. In the study, among the two-weeks patients, only 24.3% of patients needed the appointment. For patients who returned for an in-person follow-up six-weeks postoperative, only 31.6% of patients needed the appointment.

Aiming to evaluate the designed architecture and based on the Jeffery (2017) work, we accessed, under the permission of The University of Western Ontario, its partial dataset.

[2] (https://www.uwo.ca/fhs/pt/about/faculty/bryant_d.html

6.3.1 Architecture Model

According to population growth and ageing around the world, the flow of people at hospitals and clinics is increasing. In the United States, approximately 100,000 ligament reconstruction visits are annually performed (JAMESON et al., 2012). This change in clinical and demographic factors has increased the risk of postoperative complications on patient population (MARTIN et al., 2012). Between 11% and 52% of emergency ambulance calls are to patients with non-serious problems (DALE et al., 2004), which impair the efficiency of care in higher priority cases.

The consultation process offers an online or in-person standardized collection of medical data, such as patients' symptoms and medications. These data create profile and historic treatments for each patient in health centres. In large hospitals or clinics, the amount of daily data generated is expressive.

In that context, our proposal fits very well and can provide notifications to these patients automatically, not requiring the patient to go to the clinic under normal healing conditions. This can decrease the flow of people at clinics or hospitals and can improve the efficiency of physicians' schedules.

Having the results from the previous case studies, and the requirements that this context presents, we made improvements in the Health-PRIOR architecture, incorporating factors that contribute for better prediction results and system self-feeding as well as the factors related to capture of information from IoT devices.

The designed architecture is intended to be composed of the proposed ensemble model, which focuses on the combination of models capabilities, aiming to maximize the prediction power of the entire system. Each service is composed of a machine learning model, in order to work individually or together, allowing the integration of partial or total reuse of its components. These systems can handle data and offer an output **yes** or **no** for prioritization of cases. Figure 24 presents the instantiated architecture.

We have incorporated the capture of information throughout IoT devices as a digital form with specific questions about postoperative patient's conditions. Those questions were extracted from the eWound form present in the eVisit questionnaires (JEFFERY, 2017). The notifications sent to patients, in case of prioritizing cases, are sent to the clinic/hospital as well. This process allows the system self-feeding, and provide real-time monitoring of patient notifications. The whole process is illustrated in Figure 25.

The **Extraction Layer** is responsible for capturing the information from patients' about their postoperative conditions. The answers about the eWound questionnaire are captured and sent to the **Pre-processing Module**.

This module pre-processes the patients' data and the Historical Medical Data (HMD), which is provided by the hospital database and is used for training purposes. The

Figure 24 – eWound-PRIOR architecture

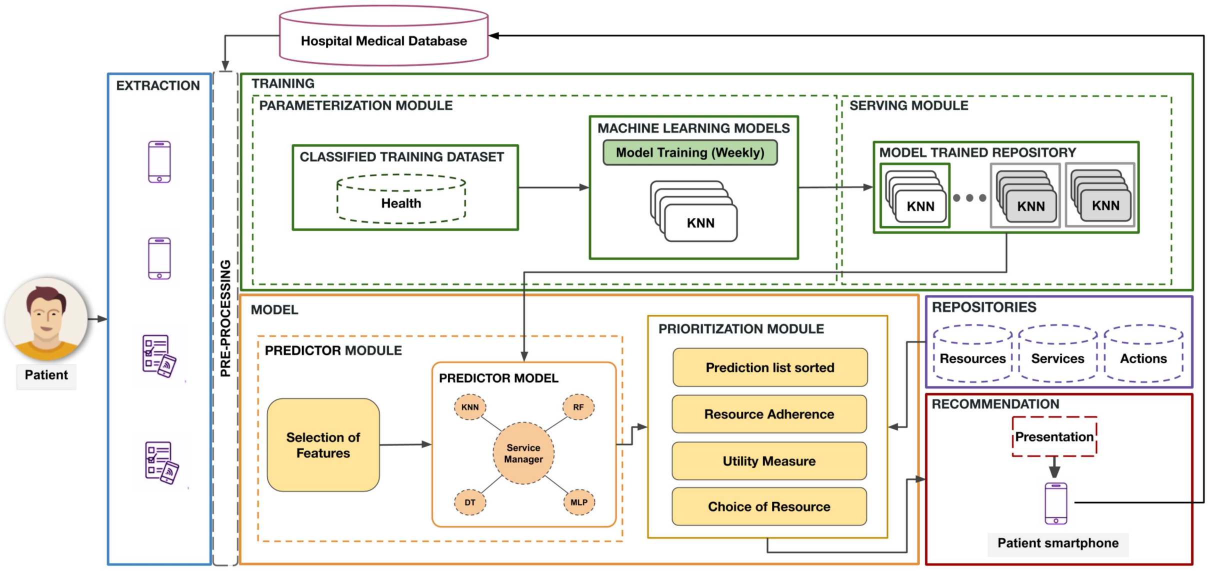
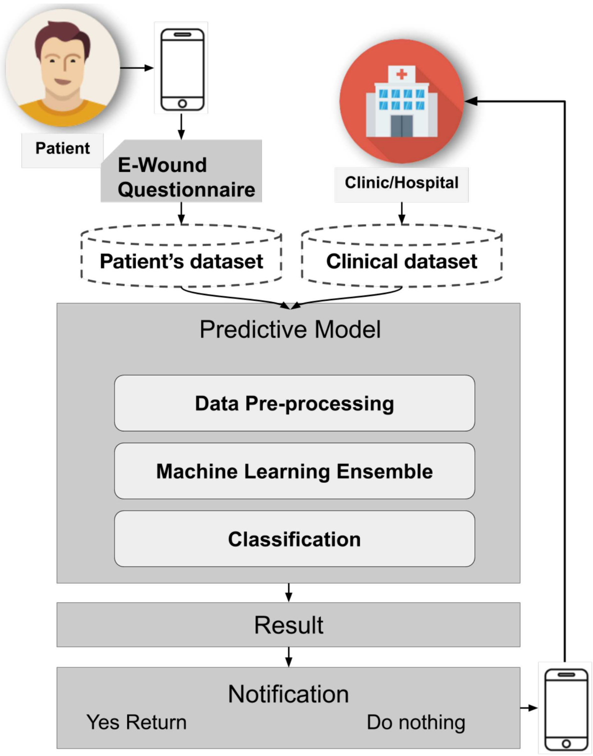


Figure 25 – eWound-PRIOR activities workflow



process removes noise values and replaces the missing ones. It also normalizes the data to prepare the dataset for training.

The **Training Layer** was divided into two modules. The **Parameterization Module** is responsible for storing the treated data and train the Machine Learning models. Once trained, the models are stored and made available to use in the **Serving Module**. The **Model Layer** executes the predictions by selecting the most adherent features to the context and makes the prediction using the trained models.

The **Prioritization Module** is responsible to, according to the classification made in the previous module, prioritize the case and select the most adherent resource, based on utility to the patient's context, to recommend. The recommendation resources are stored in the **Repositories Layer** and are accessed by the Prioritization Module.

Finally, the **Recommendation Layer** presents the prioritized case and the adherent resources to patient.

6.3.2 Data Description

The available dataset contains all information related to the eWound questionnaire (JEFFERY, 2017). The questions are about the patient's condition, which asks if the patient felt something after the surgery, the prescribed medication and the symptoms in their daily life. It is divided into two subsets of data considering the surgery date, one related to two-weeks patients and another related to six-weeks patients.

A total of 352 patients were asked to respond to the eVisit questionnaires for two- and six-weeks patients. Table 9 shows some questions from the datasets and their structure.

Table 9 – Some eWound questions

Question	**Answer Structure**
Are you having pain in your knee or leg?	(Yes, No)
Is the swelling constant or intermittent?	(Constant, Intermittent)
How long have you felt unwell?	(>3 days, < 3 days)
Do you have nausea?	(Yes, No)

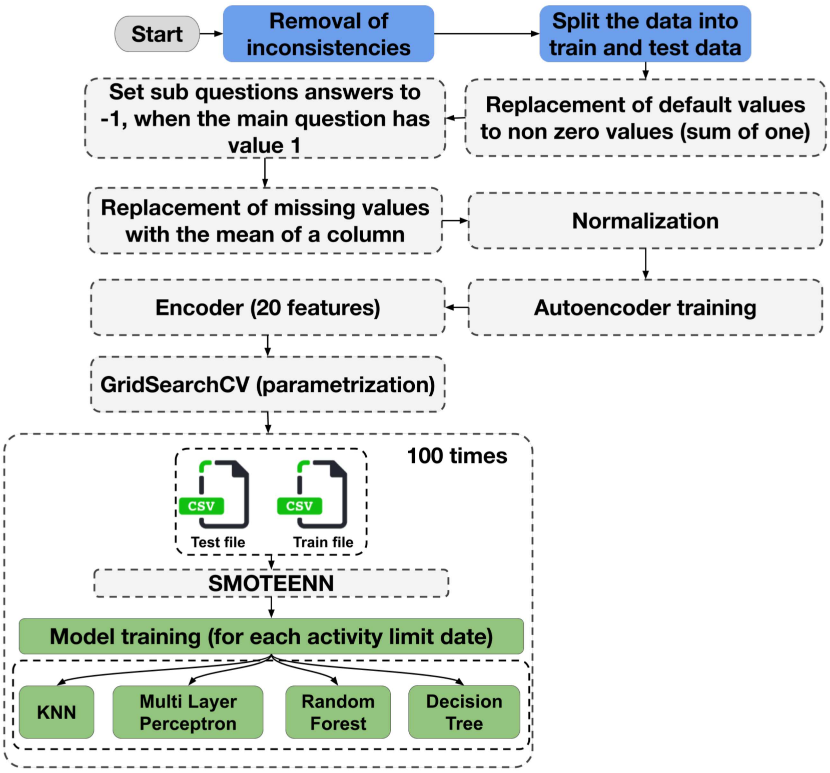
After the extraction process, we pre-processed the data by removing the duplicated answers and inconsistent cases. We also replace the missing values by the mean of each column. The datasets preprocessing is important because they present too many missing values. Figure 26 shows the training process.

As the eWound questionnaire has a skip logic, we treated the sub-questions by setting their values to *-1*, where the main question has the *no* as value. The questionnaire skip logic implies that if the sub-questions were answered that usually means the patients answered *yes* to the main question. Whereas if they did not answer them, it's because they answered *no* to the main question. In this case, *-1* represents that the sub-questions were answered if the main question got *no* as answer.

We normalized the data cleaned by using the *Normalizer* provided by the *Sklearn* framework, and we trained an autoencoder structured with 10 hidden layers. The *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. This value may be considered a good result for an autoencoder (TAN; ESWARAN,

Figure 26 – eWound-PRIOR Training Process



2011).

We have made empirical tests to set the resulting number of features using the autoencoder, ranging from 10 to 46. In our experiments, 20 features got a better performance for the available datasets. Therefore, the cleaned data was encoded, reducing the number of features from 46 to 20. The classifications remained untouched after the process, with *1* and *0* values for *requires attention* and *does not require attention* classification, respectively. By encoding the features to extract the intrinsic information, we improved the accuracy of classifications.

Instead of selecting the best set of training instances, we randomized the selection of cases from the preprocessed two- and six-week datasets. Once the models were parameterized with the best parameters to our datasets, we split the cases into training and testing sets. We randomly selected 80% of each dataset for training and 20% for testing.

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 in order to balance the classes for training. This technique balances the number of instances for each classification.

The process of splitting the dataset into training and testing sets was executed 100 times, and each set of models trained was used to compose the ensemble model for that splitted dataset. At the end, we got 100 ensemble models to evaluate the average accuracy of the architecture.

6.3.3 **Ensemble Model**

To compose the ensemble model, we decreased the number of models for the predictions because two models did not fit well into the available data. We discuss in more detail about that choice in the following sections.

For this case study specifically, we had to incorporate some techniques that reduce the features dimensionality because of the amount of missing values. For this purpose, we used an autoencoder to reduce the number of features in the dataset.

An autoencoder algorithm (RUMELHART; HINTON; WILLIAMS, 1985) 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 (WANG et al., 2014; ARAYA et al., 2017). The ability to learn from the "intrinsic data structure" is useful when the available data is noise and has too many missing values. Having the important information of a dataset available, it is useful to get better predictions in classification problems.

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

For this case study, we have chosen 4 classical Machine Learning models to combine in an ensemble form. The set of models in the instantiated architecture is composed of the most adherent ones to the supervised classification problem.

To build the ensemble model in this case study, we selected the same six models presented in the previous section. The models were trained, and the first results presented a discrepancy between the classifications, where the results presented for almost all cases the same class to the SVM and Logistic Regression models.

Viewing that, we removed both models from our ensemble, aiming to not impair the final classification. For our context and available datasets, the models that presented better performance were **Decision Tree**, **KNN**, **Random Forest** and **Multi-Layer Perceptron**.

6.3.3.1 *Parameterization of Models*

Aiming 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 (RASCHKA, 2015; PEDREGOSA et al., 2011b). The parametrization process was made by setting cross-validation with *10* folds based on the (Stratified) KFold (PEDREGOSA et al., 2011a). We also used the *sklearn* framework as a provider for each model. Each model parameterization is presented in Table 10:

Table 10 – eWound-PRIOR parameterization

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

6.3.3.2 *Ensemble Strategy*

Based on the Voting Method (POLIKAR, 2012), we use the weighted average of the model results to define the final classification for a patient's context. Our goal is to try to reduce the flow of people at health centres, but without impairing the patients that need attention. To do so, we have set the weights of *does not require attention* classification to 1, and to focus on the need of care in patients that present some conditions, we have set the weight for *requires attention* classification as 2. The strategy is described by Algorithm 1.

The return value is an object containing the final classification and its intensity. In case of class *requires attention* be returned, the patient is notified to go to the hospital/clinic where the doctor is able to assist the patient in its needs.

6.3.3.3 *Evaluation Metrics*

The evaluation metrics *precision*, *recall* and *f-measure* analyze the accuracy, and in addition, we highlight the **sensitivity** or true positive rate (TPR), which measures a model capacity to identify cases that need in-person attention, and the **specificity** or true negative rate (TNR), which measures a model capacity to identify cases that do not need

Algorithm 1: Ensemble strategy

Data: An array A of objects (Classification, intensity);
Result: An object (Classification, intensity)

```
NegSum ← 0
PosSum ← 0
WeightSum ← 0
for i ← 0 to A.length do
    Prob ← A[i].intensity
    Class ← A[i].classification
    if Class == requires attention then
        NegSum ← NegSum + (1 − Prob)
        PosSum ← PosSum + 2 ∗ Prob
    else
        PosSum ← PosSum + 2 ∗ (1 − Prob)
        NegSum ← NegSum + Prob
    end if
    WeightSum ← WeightSum + 1
end for
MediaPos ← PosSum / (2∗WeightSum)
MediaNeg ← NegSum / WeightSum
if MediaNeg ≤ MediaPos then
    return (requires attention, MediaPos);
else
    return (does not require attention, MediaNeg);
end if
```

in-person attention. The sensitivity and specificity were evaluated using Equations 6.4 and 6.5, respectively.

$$Sensitivity = \frac{TP}{P} \tag{6.4}$$

$$Specificity = \frac{TN}{N} \tag{6.5}$$

where true positive (TP) is the number of cases that need attention that is correctly identified as needing in-person consultation, true negative (TN) is the number of cases that do not need in-person attention that is correctly identified as not needing attention, P is the total number of positive instances and N is the total number of negative instances.

We use other two metrics because our goal is not only see the accuracy of predictions in general, but also to see if the architecture can classify correctly cases that do not need attention, without impairing the classification of cases that do.

6.3.4 Prediction Results and Analysis

Based on the eWound questionnaire, we developed a mobile app to simulate information gathering and to send the answers to the proposed framework. Figures 27 and 28 illustrate the application.

Figure 27 – Login screen

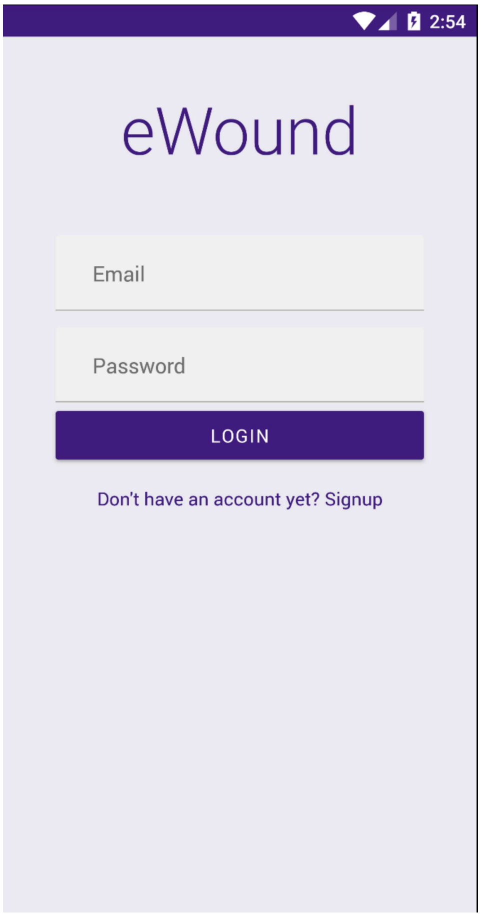
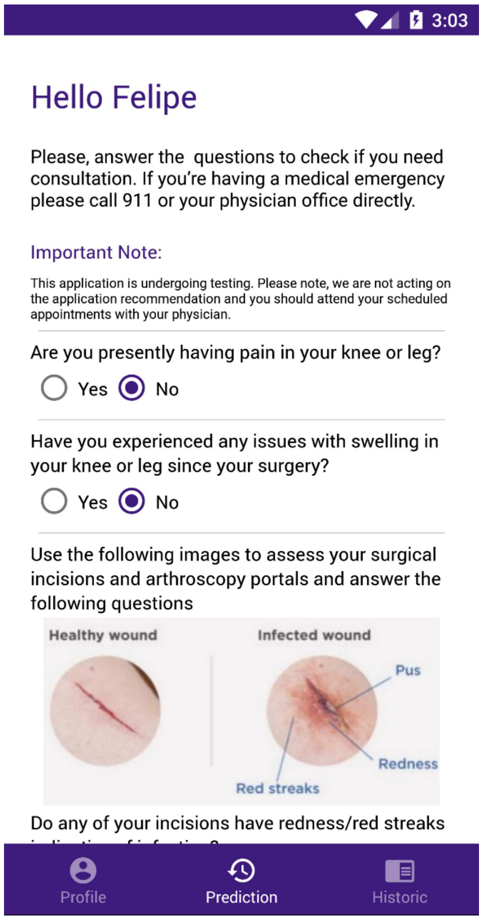


Figure 28 – eWound questionnaire



For the two- and six-weeks datasets, we have applied the same process for training and testing, which generated two sets of models. Each set was used for predicting the necessity of care for patients according to the number of weeks that have passed since surgery. The experiments were done in a server with Ubuntu 18.4 LTS; 94 GB of RAM; Intel Xeon CPU E5-2630.

Table 11 shows the average results for the sensitivity and specificity metrics for each model and our ensemble model for the two weeks dataset. Table 12 shows the results for the six-weeks dataset.

Table 11 – Two-weeks dataset results

Model	Precision	Recall	F-measure	Sensitivity	Specificity
Ensemble	0.289	0.730	0.413	0.730	0.428
KNN	0.313	0.498	0.380	0.497	0.650
Decision tree	0.286	0.731	0.368	0.529	0.582
MLP	0.272	0.742	0.389	0.741	0.344
Random Forest	0.290	0.731	0.369	0.526	0.591

Table 12 – Six-weeks dataset results

Model	Precision	Recall	F-measure	Sensitivity	Specificity
Ensemble	0.341	0.609	0.434	0.728	0.339
KNN	0.327	0.562	0.410	0.561	0.465
Decision tree	0.342	0.609	0.410	0.524	0.527
MLP	0.392	0.666	0.486	0.666	0.507
Random Forest	0.341	0.609	0.409	0.521	0.533

To understand how precise the results were from the outliers, we have calculated the standard deviation for each metric. Table 13 shows the values found.

Table 13 – Standard deviation for both datasets

Model	Precision	Recall	F-measure	Sensitivity	Specificity
Ensemble (2)	0.04	0.12	0.06	0.12	0.08
KNN (2)	0.08	0.14	0.09	0.14	0.08
Decision tree (2)	0.07	0.12	0.09	0.17	0.09
MLP (2)	0.05	0.20	0.06	0.20	0.23
Random Forest (2)	0.07	0.12	0.09	0.16	0.09
Ensemble (6)	0.05	0.12	0.07	0.11	0.08
KNN (6)	0.05	0.14	0.07	0.14	0.12
Decision tree (6)	0.06	0.12	0.07	0.13	0.11
MLP (6)	0.05	0.14	0.06	0.14	0.15
Random Forest (6)	0.06	0.12	0.08	0.13	0.09

Although Ensemble presents the lower value, the standard deviations in Table 13 show that there is a significant variation of the results. As the datasets were based on patients' answers, the patients did not answer the questions entirely and, in some cases, did not answer the questionnaire itself. This generated many missing values in the features of each case. Because of that, the results show that for both datasets, the models have difficulty in classifying the cases correctly. Several classifications got around 60% of certainty in the predictions for all models, which impair the final result of the ensemble.

The overall accuracy was severely impaired in this case study, since the models failed to correctly distinguish one case from the other. As the number of features used was reduced based on the original values of the 46 features present in the dataset, cases with different classifications had a similar configuration, which is harmful to the models.

For the incorrect predictions (missed patients requiring attention), we found that most patients answered the questions related to the body temperature and the wound conditions as not presenting any problem. This causes several columns in the dataset to have a replacement on missing data, which impairs the dimensionality reduction by using the autoencoder.

We understand that if patients would answer all questions without the skip logic, the models would have better performance by differentiating one case from another. The questionnaire was created with no obligation to answer all of the sub-questions on it. In this way, patients often skip a question that could be important for the classification, creating missing values, and impairing the models' training.

Following the analysis of each set of questions, the majority of cases presented redness on the wound and the incisions were hot to the touch. Those symptoms suggest that the patient needs to go to the clinic/hospital, however, the other answers configure the case with no priority, due to the noise generated, making the models classify wrongly these cases.

Although we have preprocessed the data, trying to reduce the noise and missing values, which also removes inconsistencies, all those described factors are limitations to this study. Having too many missing and similar (replacement by -1) values in the dataset is not ideal and may generate a wrong classification, as we can see by the results. We believe that more information about the patients is necessary, complementing the questionnaire.

In order to evaluate the instantiated architecture, we compare the results of the models adopted separately and together, according to our proposal. For instance, ensemble strategies are not prejudiced by different models, but there is a balance in the result. For that to happen, the majority of models adopted need to show good accuracy in the predictions, and the final ensemble result will be better than individual models (BRAZ et al., 2019).

The results show that with the adoption of the proposal there is a general gain in the classification, since it uses multiple classification results as a predictive basis. However, the ensemble predictions, in our experiments, was impaired by the number of missing values provided from the use of skip logic in the available dataset. Having a skip logic questionnaire is not ideal for prediction, because it can generate a lot of missing data, once the questionnaire has no obligation to answer all questions.

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 to get better performance for datasets like the current one.

Another concern is to evaluate the whole architecture by evaluating its predictions with patients in treatment in a real context 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 worsening treatment prediction.

6.4 FINAL CONSIDERATIONS

In this section, we described the development of three case studies pursued, focused on evaluating specific parts of the proposed architecture. In the first case study, we saw that the proposed ensemble model has a good performance where its prediction accuracy is improved. In this evaluation, we made an analysis of the impact that the notifications would cause if, throughout the course time, those responsible for the class could make decisions based on the pursued notifications and prevent the students to dropout. The results showed that, despite the small number of students present in the study, the triggered notifications would have a positive effect on the dropout control.

The second case study, analyzed the aspects of training time for each model used in the experiment. For a small dataset, the models performance with parallel training is promising. However, when thinking of a smart city environment, the volume of data generated is large and can cause some performance issues.

Based on the results of these two case studies, a third case study analyzed the general aspects of the proposed architecture. We improved the technical aspects that the architecture presented as limitations and performed an experiment with real data through a partnership project with The University of Western Ontario, in Canada. The eWound-PRIOR was proposed aiming to improve the efficiency of care in healthcare centres, consolidate knowledge by storing the captured data and make predictions in postoperative patients cases about their need to prioritized attention.

The three case studies, show the potential of the proposal and its adherence in different contexts of application. The amount of data for training and its completeness around the context should be considered when instantiate the architecture.

7 FINAL REMARKS AND FUTURE WORKS

This chapter describes the final remarks about the current research, the main results and contributions. It also describes the restrictions and further directions for the improvement of the proposal.

7.1 CONCLUSIONS

This research proposed a recommender system applied to healthcare, named Health-PRIOR, capable of predicting prioritization cases based on data captured from IoT devices. The proposal brings Machine Learning models combined in an ensemble form, aiming to improve the system prediction performance.

In the case studies development, we evaluated each part of the architecture to search for a complete and assertive proposal. Due to the difficulty in the availability of data for the focused context, we cannot generalize the results but perceive the benefits in adopting the proposal in assisted living environment.

The obtained results from the case studies brought evidences that the proposal is valid to help not only physicians, but also patients. Therefore, the Health-PRIOR can be applied in different contexts viewing the same propose, prioritizing cases or people in a given context.

The goals of this research work were complied through the Health-PRIOR architecture, which brings capturing and preprossessing of data to make predictions about a patient case. The implementation showed the proposal viability in larger environments to send recommendations about the cases prioritization and the case itself to specialized professionals, whom can take decisions around the case.

Some conclusions can be reached based on the obtained results:

- The capture and extraction of data from IoT devices can be managed and made in real-time as well as its use. Those data allow the prediction and decision taking in real-time by specialized professionals.
- The amount of data used in model training should be substantially complete and accurate about the context that they describe. Low amount of data can generate over-fitting in the models, and also, cannot describe completely the context in the training process. At the same time, large amount of old data can generate distortions in the training set and impair the classification results.
- With the proposal adoption, early diagnosis can be made and worsening of treatments can be detected. Through cases prioritization, patients can be helped by physicians in their treatment and case.

The proposal is broad, contemplating different contexts of application. The Health-PRIOR architecture can help with the case attendance and can be refined and improved in each stage, seeking to increase the accuracy and the adherence to the recommendations into patients' cases.

7.2 CONTRIBUTIONS

The use of smart devices in assisted environments is of great importance since it allows the capture of information without human alterations. This makes the data accurate and reliable for predictive and decision-making contexts. Although predictive approaches to health recommendation systems are not new in the literature, the use in smart city environments and aiming at prioritizing medical cases is a contribution to the context. The research potential that this theme brings allows the development and continuity of other researches about the same context.

This research work had as the main contribution the design of a recommender system architecture applied to healthcare, aiming at prioritizing emergency cases. The main goal was to aggregate the data captured by IoT devices in assisted environments, with the predictive power of Machine Learning models. An ensemble approach was adopted to maximize the accuracy of the proposal, to predict the need for prioritization of emergency cases.

Three case studies were carried out, where each one sought to evaluate specific architecture concepts. The results of each study show that the input dataset must have accurate information, where the classifications are well defined for a given context. According to our results, aspects such as missing values or the lack of standards in the data set can generate wrong predictions, thus causing a wrong action by the system.

7.3 LIMITATIONS

Regarding the use of predictive models, we realized that for each context they should be parameterized to increase its predictive potential. Classic models provide high performance in predictions, which enables ensemble approaches to combine those models, aiming to perform better compared to individual ones. Our proposal uses an ensemble approach, which allow greater assertiveness and confidence in each result (BRAZ et al., 2019).

Due to the context of the project in the third case study and the development spent time, we were unable to 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.

We believe that, with a larger accurate dataset and relevant recommendation medical objects, the whole architecture can fit properly in the physician's daily job. The proposed workflow to the architecture offers no human management of data, which helps to maintain the system self-sufficient. We understand that the proposal has the potential for implementation on a large scale, aiming to analyze its adherence in a greater environment.

7.4 FUTURE WORK

For further directions, we aim to improve the architecture by evaluating it in a real context by using the recommendation module. We also intend to improve the mobile application with images classifications to help in the diagnosis of diseases and worsening in treatments.

The recommendation module needs attention in its development, where analytical techniques should be considered to get the most adherent resource to a given case.

We intend to use the architecture with a data workflow to evaluate the proposal with a high volume of data flow. In this case, we will evaluate whether the proposal can make decisions and rebuild the models properly.

The eWound application interface should be evaluated to ensure patient satisfaction. We recommend the eWound application be tested using a diagnostic validity study design to compare its predictions to the current gold standard of in-person patient assessment.

REFERENCES

ABDALLA, André Luiz Campos Esqueff et al. R. ecos-plataforma para suporte de um ecossistema de software para sistemas de recomendação. Universidade Federal de Juiz de Fora (UFJF), 2018.

ABOAGYE, Emelia Opoku; JAMES, Gee C; KUMAR, Rajesh. Evaluating the performance of deep neural networks for health decision making. **Procedia computer science**, Elsevier, v. 131, p. 866–872, 2018.

ADAMS, Roy J; SADASIVAM, Rajani S; BALAKRISHNAN, Kavitha; KINNEY, Rebecca L; HOUSTON, Thomas K; MARLIN, Benjamin M. Perspect: collaborative filtering for tailored health communications. In: ACM. **Proceedings of the 8th ACM Conference on Recommender systems**. [S.l.], 2014. p. 329–332.

ALBINO, Vito; BERARDI, Umberto; DANGELICO, Rosa Maria. Smart cities: Definitions, dimensions, performance, and initiatives. **Journal of urban technology**, Taylor & Francis, v. 22, n. 1, p. 3–21, 2015.

ALI, Farman; ISLAM, SM Riazul; KWAK, Daehan; KHAN, Pervez; ULLAH, Niamat; YOO, Sang-jo; KWAK, Kyung Sup. Type-2 fuzzy ontology–aided recommendation systems for iot–based healthcare. **Computer Communications**, Elsevier, v. 119, p. 138–155, 2018.

AOUEDI, Ons; TOBJI, Mohamed Anis Bach; ABRAHAM, Ajith. An ensemble of deep auto-encoders for healthcare monitoring. In: SPRINGER. **International Conference on Hybrid Intelligent Systems**. [S.l.], 2018. p. 96–105.

ARAYA, Daniel B; GROLINGER, Katarina; ELYAMANY, Hany F; CAPRETZ, Miriam AM; BITSUAMLAK, Girma. An ensemble learning framework for anomaly detection in building energy consumption. **Energy and Buildings**, Elsevier, v. 144, p. 191–206, 2017.

ATZORI, Luigi; IERA, Antonio; MORABITO, Giacomo. The internet of things: A survey. **Computer networks**, Elsevier, v. 54, n. 15, p. 2787–2805, 2010.

BASTOS, Marcus Gomes; KIRSZTAJN, Gianna Mastroianni. Chronic kidney disease: importance of early diagnosis, immediate referral and structured interdisciplinary approach to improve outcomes in patients not yet on dialysis. **Brazilian Journal of Nephrology**, SciELO Brasil, v. 33, n. 1, p. 93–108, 2011.

BOBADILLA, Jesús; ORTEGA, Fernando; HERNANDO, Antonio; GUTIÉRREZ, Abraham. Recommender systems survey. **Knowledge-based systems**, Elsevier, v. 46, p. 109–132, 2013.

BRAZ, Felipe; CAMPOS, Fernanda; STROELE, Victor; DANTAS, Mário. An early warning model for school dropout: a case study in e-learning class. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2019. v. 30, n. 1, p. 1441.

BREESE, John S; HECKERMAN, David; KADIE, Carl. Empirical analysis of predictive algorithms for collaborative filtering. In: MORGAN KAUFMANN PUBLISHERS INC.

Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence. [S.l.], 1998. p. 43–52.

BREIMAN, Leo. Random forests. **Machine learning**, Springer, v. 45, n. 1, p. 5–32, 2001.

BURKE, Robin. Hybrid recommender systems: Survey and experiments. **User modeling and user-adapted interaction**, Springer, v. 12, n. 4, p. 331–370, 2002.

CHAN, Marie; ESTÈVE, Daniel; FOURNIOLS, Jean-Yves; ESCRIBA, Christophe; CAMPO, Eric. Smart wearable systems: Current status and future challenges. **Artificial intelligence in medicine**, Elsevier, v. 56, n. 3, p. 137–156, 2012.

CHAWLA, Nitesh V; BOWYER, Kevin W; HALL, Lawrence O; KEGELMEYER, W Philip. Smote: synthetic minority over-sampling technique. **Journal of artificial intelligence research**, v. 16, p. 321–357, 2002.

CHEN, Jonathan H; ALAGAPPAN, Muthuraman; GOLDSTEIN, Mary K; ASCH, Steven M; ALTMAN, Russ B. Decaying relevance of clinical data towards future decisions in data-driven inpatient clinical order sets. **International journal of medical informatics**, Elsevier, v. 102, p. 71–79, 2017.

COSTA, C.; MURTA, L. Version control in distributed software development: A systematic mapping study. In: **2013 IEEE 8th International Conference on Global Software Engineering**. [S.l.: s.n.], 2013. p. 90–99. ISSN 2329-6305.

DALE, J; WILLIAMS, S; FOSTER, T; HIGGINS, J; SNOOKS, H; CROUCH, R; HARTLEY-SHARPE, C; GLUCKSMAN, E; GEORGE, S. Safety of telephone consultation for "non-serious" emergency ambulance service patients. **BMJ Quality & Safety**, BMJ Publishing Group Ltd, v. 13, n. 5, p. 363–373, 2004.

DENG, Dong; JING, Liping; YU, Jian; SUN, Shaolong; ZHOU, Haofei. Neural gaussian mixture model for review-based rating prediction. In: ACM. **Proceedings of the 12th ACM Conference on Recommender Systems**. [S.l.], 2018. p. 113–121.

DHANALAKSHMI, P; RAMANI, K; REDDY, B Eswara. An improved rank based disease prediction using web navigation patterns on bio-medical databases. **Future Computing and Informatics Journal**, Elsevier, v. 2, n. 2, p. 133–147, 2017.

DREISEITL, Stephan; OHNO-MACHADO, Lucila. Logistic regression and artificial neural network classification models: a methodology review. **Journal of biomedical informatics**, Elsevier, v. 35, n. 5-6, p. 352–359, 2002.

DUA, Dheeru; TANISKIDOU, Efi Karra. **UCI Machine Learning Repository**. 2017. Available from Internet: <<http://archive.ics.uci.edu/ml>>.

FAN, Jun; YANG, Jing; JIANG, Zhenran. Prediction of central nervous system side effects through drug permeability to blood–brain barrier and recommendation algorithm. **Journal of Computational Biology**, Mary Ann Liebert, Inc. 140 Huguenot Street, 3rd Floor New Rochelle, NY 10801 USA, v. 25, n. 4, p. 435–443, 2018.

FARAHANI, Bahar; FIROUZI, Farshad; CHANG, Victor; BADAROGLU, Mustafa; CONSTANT, Nicholas; MANKODIYA, Kunal. Towards fog-driven iot ehealth: Promises and challenges of iot in medicine and healthcare. **Future Generation Computer Systems**, Elsevier, v. 78, p. 659–676, 2018.

GAO, Shan; GUO, Guibing; LIN, Yusong; ZHANG, Xingjin; LIU, Yongpeng; WANG, Zongmin. Pairwise preference over mixed-type item-sets based bayesian personalized ranking for collaborative filtering. In: IEEE. **Dependable, Autonomic and Secure Computing, 15th Intl Conf on Pervasive Intelligence & Computing, 3rd Intl Conf on Big Data Intelligence and Computing and Cyber Science and Technology Congress (DASC/PiCom/DataCom/CyberSciTech), 2017 IEEE 15th Intl**. [S.l.], 2017. p. 30–37.

HAN, Jiawei; PEI, Jian; KAMBER, Micheline. **Data mining: concepts and techniques**. [S.l.]: Elsevier, 2011.

HASSAN, Shahzaib; SYED, Zeeshan. From netflix to heart attacks: collaborative filtering in medical datasets. In: ACM. **Proceedings of the 1st ACM international health informatics symposium**. [S.l.], 2010. p. 128–134.

HUNG, Galen Chin-Lun; YANG, Pei-Ching; WANG, Chen-Yi; CHIANG, Jung-Hsien. A smartphone-based personalized activity recommender system for patients with depression. In: ICST (INSTITUTE FOR COMPUTER SCIENCES, SOCIAL-INFORMATICS AND TELECOMMUNICATIONS ENGINEERING). **Proceedings of the 5th EAI International Conference on Wireless Mobile Communication and Healthcare**. [S.l.], 2015. p. 253–257.

HUSSEIN, Asmaa S; OMAR, Wail M; LI, Xue; ATI, Modafar. Efficient chronic disease diagnosis prediction and recommendation system. In: IEEE. **Biomedical Engineering and Sciences (IECBES), 2012 IEEE EMBS Conference on**. [S.l.], 2012. p. 209–214.

JAMESON, Simon S; DOWEN, Daniel; JAMES, Philip; SERRANO-PEDRAZA, Ignacio; REED, Mike R; DEEHAN, David. Complications following anterior cruciate ligament reconstruction in the english nhs. **The Knee**, Elsevier, v. 19, n. 1, p. 14–19, 2012.

JEFFERY, William G. e-visits for early post-operative visits following orthopaedic surgery can they add efficiency without sacrificing effectiveness. **Electronic Thesis and Dissertation Repository. 5053**, 2017. Available from Internet: <<https://ir.lib.uwo.ca/etd/5053>>.

KALCHBRENNER, Nal; GREFENSTETTE, Edward; BLUNSOM, Phil. A convolutional neural network for modelling sentences. **arXiv preprint arXiv:1404.2188**, 2014.

KAUR, Harmanjeet; KUMAR, Neeraj; BATRA, Shalini. An efficient multi-party scheme for privacy preserving collaborative filtering for healthcare recommender system. **Future Generation Computer Systems**, Elsevier, 2018.

KITCHENHAM, Barbara. Procedures for performing systematic reviews. **Keele, UK, Keele University**, v. 33, n. 2004, p. 1–26, 2004.

KITCHIN, Rob. The real-time city? big data and smart urbanism. **GeoJournal**, Springer, v. 79, n. 1, p. 1–14, 2014.

KOOP, C Everett; MOSHER, Robyn; KUN, Luis; GEILING, Jim; GRIGG, Eliot; LONG, Sarah; MACEDONIA, Christian; MERRELL, Ronald C; SATAVA, Richard; ROSEN, Joseph M. Future delivery of health care: Cybercare. **IEEE Engineering in Medicine and Biology Magazine**, IEEE, v. 27, n. 6, 2008.

KUMARI, Pooja; JAIN, Praphula Kumar; PAMULA, Rajendra. An efficient use of ensemble methods to predict students academic performance. In: IEEE. **2018 4th International Conference on Recent Advances in Information Technology (RAIT)**. [S.l.], 2018. p. 1–6.

KURIAN, Ria A; LAKSHMI, KS. An ensemble classifier for the prediction of heart disease. **International Journal of Scientific Research in Computer Science**, v. 3, n. 6, p. 25–31, 2018.

LAFTA, Raid; ZHANG, Ji; TAO, Xiaohui; LI, Yan; TSENG, Vincent S. An intelligent recommender system based on short-term risk prediction for heart disease patients. In: IEEE. **Web Intelligence and Intelligent Agent Technology (WI-IAT), 2015 IEEE/WIC/ACM International Conference on**. [S.l.], 2015. v. 3, p. 102–105.

LAFTA, Raid; ZHANG, Ji; TAO, Xiaohui; LI, Yan; TSENG, Vincent S; LUO, Yonglong; CHEN, Fulong. An intelligent recommender system based on predictive analysis in telehealthcare environment. In: IOS PRESS. **Web Intelligence**. [S.l.], 2016. v. 14, n. 4, p. 325–336.

LECUN, Yann; BOTTOU, Léon; BENGIO, Yoshua; HAFFNER, Patrick et al. Gradient-based learning applied to document recognition. **Proceedings of the IEEE**, Taipei, Taiwan, v. 86, n. 11, p. 2278–2324, 1998.

LOPS, Pasquale; GEMMIS, Marco de; SEMERARO, Giovanni. Content-based recommender systems: State of the art and trends. In: _____. **Recommender Systems Handbook**. Boston, MA: Springer US, 2011. p. 73–105. ISBN 978-0-387-85820-3. Available from Internet: <<https://doi.org/10.1007/978-0-387-85820-3_3>>.

MÁRQUEZ-VERA, Carlos; CANO, Alberto; ROMERO, Cristobal; NOAMAN, Amin Yousef Mohammad; FARDOUN, Habib Mousa; VENTURA, Sebastian. Early dropout prediction using data mining: a case study with high school students. **Expert Systems**, Wiley Online Library, v. 33, n. 1, p. 107–124, 2016.

MARSAL-LLACUNA, Maria-Lluïsa; COLOMER-LLINÀS, Joan; MELÉNDEZ-FRIGOLA, Joaquim. Lessons in urban monitoring taken from sustainable and livable cities to better address the smart cities initiative. **Technological Forecasting and Social Change**, Elsevier, v. 90, p. 611–622, 2015.

MARTIN, Billie-Jean; BUTH, Karen J; ARORA, Rakesh C; BASKETT, Roger JF. Delirium: a cause for concern beyond the immediate postoperative period. **The Annals of thoracic surgery**, Elsevier, v. 93, n. 4, p. 1114–1120, 2012.

MUSTAQEEM, Anam; ANWAR, Syed Muhammad; KHAN, Abdul Rashid; MAJID, Muhammad. A statistical analysis based recommender model for heart disease patients. **International journal of medical informatics**, Elsevier, v. 108, p. 134–145, 2017.

NEVES, Felipe; STRÖELE, Victor; CAMPOS, Fernanda. Information diffusion in social networks: a recommendation model in the educational context. In: ACM. **Proceedings of the XV Brazilian Symposium on Information Systems**. [S.l.], 2019. p. 25.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.;

DUCHESNAY, E. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.

PEDREGOSA, Fabian; VAROQUAUX, Gaël; GRAMFORT, Alexandre; MICHEL, Vincent; THIRION, Bertrand; GRISEL, Olivier; BLONDEL, Mathieu; PRETTENHOFER, Peter; WEISS, Ron; DUBOURG, Vincent et al. Scikit-learn: Machine learning in python. **Journal of machine learning research**, v. 12, n. Oct, p. 2825–2830, 2011.

POLIKAR, Robi. Ensemble learning. In: ______. **Ensemble Machine Learning: Methods and Applications**. Boston, MA: Springer US, 2012. p. 1–34. ISBN 978-1-4419-9326-7.

RAHMANI, Amir-Mohammad; THANIGAIVELAN, Nanda Kumar; GIA, Tuan Nguyen; GRANADOS, Jose; NEGASH, Behailu; LILJEBERG, Pasi; TENHUNEN, Hannu. Smart e-health gateway: Bringing intelligence to internet-of-things based ubiquitous healthcare systems. In: IEEE. **Consumer Communications and Networking Conference (CCNC), 2015 12th Annual IEEE**. [S.l.], 2015. p. 826–834.

RALLAPALLI, Sreekanth; GONDKAR, RR. Big data ensemble clinical prediction for healthcare data by using deep learning model. **International Journal of Big Data Intelligence**, Inderscience Publishers (IEL), v. 5, n. 4, p. 258–269, 2018.

RASCHKA, Sebastian. **Python machine learning**. [S.l.]: Packt Publishing Ltd, 2015.

RUMELHART, David E; HINTON, Geoffrey E; WILLIAMS, Ronald J. **Learning internal representations by error propagation**. [S.l.], 1985.

RUNESON, Per; HOST, Martin; RAINER, Austen; REGNELL, Bjorn. **Case study research in software engineering: Guidelines and examples**. [S.l.]: John Wiley & Sons, 2012.

SANTOS, Henrique Dias Pereira dos; ULBRICH, Ana Helena DPS; WOLOSZYN, Vinícius; VIEIRA, Renata. An initial investigation of the charlson comorbidity index regression based on clinical notes. In: IEEE. **2018 IEEE 31st International Symposium on Computer-Based Medical Systems (CBMS)**. [S.l.], 2018. p. 6–11.

SMARANDACHE, Florentin. **A unifying field in logics. neutrosophy: Neutrosophic probability, set and logic**. [S.l.]: American Research Press, Rehoboth, 1999.

SOKOLOVA, Marina; LAPALME, Guy. A systematic analysis of performance measures for classification tasks. **Information Processing & Management**, Elsevier, v. 45, n. 4, p. 427–437, 2009.

SON, Le Hoang. Dealing with the new user cold-start problem in recommender systems: A comparative review. **Information Systems**, PERGAMON-ELSEVIER SCIENCE LTD THE BOULEVARD, LANGFORD LANE, KIDLINGTON . . . , v. 58, p. 87–104, 2016.

TAN, Chun Chet; ESWARAN, Chikkannan. Using autoencoders for mammogram compression. **Journal of Medical Systems**, v. 35, n. 1, p. 49–58, Feb 2011.

THANH, Nguyen Dang; ALI, Mumtaz et al. Neutrosophic recommender system for medical diagnosis based on algebraic similarity measure and clustering. In: IEEE. **2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)**. [S.l.], 2017. p. 1–6.

THONG, Nguyen Tho et al. Hifcf: An effective hybrid model between picture fuzzy clustering and intuitionistic fuzzy recommender systems for medical diagnosis. **Expert Systems with Applications**, Elsevier, v. 42, n. 7, p. 3682–3701, 2015.

THONG, Nguyen Tho et al. Intuitionistic fuzzy recommender systems: an effective tool for medical diagnosis. **Knowledge-Based Systems**, Elsevier, v. 74, p. 133–150, 2015.

TULI, Shreshth; BASUMATARY, Nipam; GILL, Sukhpal Singh; KAHANI, Mohsen; ARYA, Rajesh Chand; WANDER, Gurpreet Singh; BUYYA, Rajkumar. Healthfog: An ensemble deep learning based smart healthcare system for automatic diagnosis of heart diseases in integrated iot and fog computing environments. **Future Generation Computer Systems**, Elsevier, v. 104, p. 187–200, 2020.

WANG, Wei; HUANG, Yan; WANG, Yizhou; WANG, Liang. Generalized autoencoder: A neural network framework for dimensionality reduction. In: **Proceedings of the IEEE conference on computer vision and pattern recognition workshops**. [S.l.: s.n.], 2014. p. 490–497.

WEHNER, Philipp; PIBERGER, Christina; GÖHRINGER, Diana. Using json to manage communication between services in the internet of things. In: IEEE. **2014 9th International Symposium on Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC)**. [S.l.], 2014. p. 1–4.

WOHLIN, Claes; RUNESON, Per; HÖST, Martin; OHLSSON, Magnus C; REGNELL, Björn; WESSLÉN, Anders. **Experimentation in software engineering**. [S.l.]: Springer Science & Business Media, 2012.

YANG, Shiqi; ZHOU, Ping; DUAN, Kui; HOSSAIN, M Shamim; ALHAMID, Mohammed F. emhealth: towards emotion health through depression prediction and intelligent health recommender system. **Mobile Networks and Applications**, Springer, p. 1–11, 2017.

YUAN, Weiwei; LI, Chenliang; GUAN, Donghai; HAN, Guangjie; KHATTAK, Asad Masood. Socialized healthcare service recommendation using deep learning. **Neural Computing and Applications**, Springer, v. 30, n. 7, p. 2071–2082, 2018.

ZAIDAN, AA; ZAIDAN, BB. A review on intelligent process for smart home applications based on iot: coherent taxonomy, motivation, open challenges, and recommendations. **Artificial Intelligence Review**, Springer, p. 1–25, 2018.

ZHANG, Ji; LAFTA, Raid Luaibi; TAO, Xiaohui; LI, Yan; CHEN, Fulong; LUO, Yonglong; ZHU, Xiaodong. Coupling a fast fourier transformation with a machine learning ensemble model to support recommendations for heart disease patients in a telehealth environment. **IEEE Access**, IEEE, v. 5, p. 10674–10685, 2017.

ZHANG, Kai; LIU, Xiyang; JIANG, Jiewei; LI, Wangting; WANG, Shuai; LIU, Lin; ZHOU, Xiaojing; WANG, Liming. Prediction of postoperative complications of pediatric cataract patients using data mining. **Journal of translational medicine**, Springer, v. 17, n. 1, p. 2, 2019.

ZHAO, Kai; GAO, Shanghua; HOU, Qibin; LI, Dan-Dan; CHENG, Ming-Ming. Optimizing the f-measure for threshold-free salient object detection. **arXiv preprint arXiv:1805.07567**, 2018.

ZHOU, Zhi-Hua. Ensemble learning. In: ______. **Encyclopedia of Biometrics**. Boston, MA: Springer US, 2015. p. 411–416. ISBN 978-1-4899-7488-4.