

Received 17 June 2024, accepted 1 July 2024, date of publication 11 July 2024, date of current version 15 October 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3426919

RESEARCH ARTICLE

Enhancing ICU Management and Addressing Challenges in Türkiye Through AI-Powered Patient Classification and Increased Usability With ICU Placement Software

YİĞİT HAKVERDİ^{1,2}, MELİH URVE GÜMÜŞ^{1,2}, AYHAN TAŞTEKİN³, KADIR IDİN³,
MURAT KANĖİN³, TANSEL ÖZYER⁴, AND REDA ALHAJJ^{1,5,6}

¹Department of Computer Engineering, Istanbul Medipol University, 34815 Istanbul, Türkiye

²Laurea Magistrale in Cybersecurity, University of Rome "La Sapienza," 00185 Rome, Italy

³School of Medicine, Istanbul Medipol University, 06050 Istanbul, Türkiye

⁴Department of Computer Engineering, Ankara Medipol University, 06050 Ankara, Türkiye

⁵Department of Computer Science, University of Calgary, Alberta, AB T2N 1N4, Canada

⁶Department of Health Informatics, University of Southern Denmark, 5000 Odense, Denmark

Corresponding author: Reda Alhajj (alhajj@ucalgary.ca)

ABSTRACT The increasing demand for intensive care unit (ICU) admissions, and the associated rising costs have urged the need for effective management strategies. In this research, we focus on the challenges faced by (1) hospitals in report generation and in their effort to properly allocate ICU patients, and (2) insurance organizations responsible for payments. We address the issues of misclassification and financial burden on hospitals and insurance organizations that arise from inefficient and subjective application of regulations while also considering the impact on medical personnel. Through existing literature analysis, as well as extensive discussions with critical care professionals and insights gained from university hospitals, we identified the need for a supportive machine learning model for ICU level classification of patients, and furthermore, we propose an easily deployable and highly interoperability software system specifically for placement of patients in various ICU levels. We aim to support healthcare professionals in their decision-making process with the supportive machine learning model and the software system that we named "heartbeat". This research aims to bridge the gap between hospitals and insurance institutions to ensure fair and objective patient classification and to improve the overall ICU management. The process has been tested using MIMIC-III version 1.4 dataset as a proof of concept to demonstrate the applicability and effectiveness of the developed system. Further testing using real data after official deployment and usage by various stakeholders.

INDEX TERMS ICU, MIMIC-III, supportive, machine learning, ICU management, ICU level, clustering, classification.

I. INTRODUCTION

Over the years, advancements in artificial intelligence (AI), machine learning (ML), intensive care unit (ICU) technologies, and automation systems development have significantly

The associate editor coordinating the review of this manuscript and approving it for publication was Valerio Freschi^{1b}.

enhanced the efficiency of healthcare professionals, including doctors, caregivers, and nurses. These technological breakthroughs have undeniably improved the quality of the healthcare services provided by hospitals. Despite these positive impacts, in some countries like Turkey, bureaucratic complexities have emerged as a major hurdle, hindering the full potential of these advancements. The cumbersome

bureaucratic processes counteract the efficiencies introduced by the technologies, causing delays and inefficiencies in the healthcare system. This issue becomes even more pronounced during critical times, such as the COVID-19 pandemic, when streamlined and agile healthcare practices were crucial. Non-compliance with regulations due to convoluted processes may lead to additional costs for patients, the government, insurance institutions and hospitals alike.

Moreover, the fact that the structure of critical care units is organized into tiers based on patient fatality rates, necessitates different equipment packages and medical personnel, including nurses and doctors, for each level of care. While this tiered approach aims to provide tailored treatment, it can lead to significant cost variations between levels. Consequently, the cost paying entity may face higher charges based on the level of care they require, which may turn into a financial burden for many. Addressing these bureaucratic complexities and streamlining healthcare processes is essential to fully harness the potential benefits of AI, ML, emerging ICU technologies, and automation in healthcare. By reducing administrative burdens and fostering compliance, it will be possible to ensure that advances in technology will continue to drive efficiency and improve patient satisfaction without imposing undue financial strain on paying entities, the government, or hospitals.

In general, ICU is mostly regulated and managed by institutional and government policies in various parts of the world. For instance, in the UK, there are several works reported to manage ICU as a combined effort by institutions and academicians. One of the most significant works was performed by the intensive care society in the UK [1]. The work described in [1] provides enough details about ICU logistics and management; most importantly, it includes detailed information on patient-level classification, starting from admitting a patient to the most appropriate ICU level which matches the encountered health case of the patient, and how to decide whether a patient is to be moved to another ICU level. On the other hand, regulations related to these ICU aspects in the United States are subject to rigorous reviews and frequent updates trying to address the concerns of the parties involved.

One notable ICU related work within the United States is described in the research of Eddleston et al. (2009) [2]. It offers a comprehensive exploration of ICU concepts and patient classification. This work delves into the intricacies of categorizing ICU patients and determining the necessary resource allocation for each patient and ICU level. The guidelines put forth in the study described in [2] encompass a wealth of illustrative examples to enhance clarity. Specifically, within the context of ICU-level description, the guidelines layout the criteria for patient admission to the ICU ward. For example, the guidelines provide symptomatology and diagnostic indicators for different patients as well as the presence of comorbidities, thereby enhance the precision of the classification process. As far as Turkey is concerned,

these types of guidelines were prepared in a similar way. Indeed, the regulations applied in Turkey heavily reference United States research on this matter; they are periodically updated to catch up with the international standard. However, one main difference specific to Turkey is that the employed regulations provide more comprehensive explanation about the type of wards that hospitals may utilize for ICU patients. The regulations also identify hospitals which are allowed to treat patients in each category of the ICU ward [3].

To be more specific, in Turkey, the government organization (SGK - Sosyal Güvenlik Kurumu -a.k.a., Social Security Institution) partially covers ICU costs for Turkish citizens. The amount of coverage is determined based on the patient's level of treatment and the specific clinic/hospital they have visited, resulting in varying levels of insurance payment from the government to either hospitals or patients. Unfortunately, the current regulations and categorization system for the ICU structure introduce subjectivity in determining the severity level of patients. This creates challenges for doctors and SGK in efficiently managing the ICU ward and covering the expenses, especially when hospitals provide non-compliant reports. This is because hospitals may subjectively show a patient in the highest cost ICU level which could be associated with the case, and accordingly, SGK is expected to pay the cost. Consequently, doctors and officials responsible for validating refunding may engage in subjective debates about the ICU admission criteria, leading to potential rejections and inadequate reimbursements for hospitals. The high cost associated with ICU awards and the increasing demand for ICU admissions have given rise to significant price discrepancies caused by misclassification. These discrepancies affect both hospitals and doctors, posing financial challenges on the healthcare system.

After carefully analyzed the current situation of the ICU awards in Turkey, we identified the discrepancies in the system leading to major conflicts due to subjective decisions where hospitals try to increase their income, while SGK tries to minimize the payments. This motivated the development of a complete transparent solution which implements the current ICU mechanism in Turkey enriched by regulations well defined in other systems in developed countries, mainly the United States and UK.

With the help of ICU doctors specialized in adult, paediatric and neonatal units at Medipol University Hospital, we identified issues having dual or multiple implications/interpretations within the tiered structure of the ICU system. The drawbacks of the current system not only impact the medical personnel, but also burden hospitals financially. With escalating equipment and personnel costs, hospitals tend to charge patients in a manner that surpasses the SGK-covered fees. Moreover, there is a noticeable discrepancy in expenses between hospitals offering high-quality critical care services and those providing more standard care. This difference is primarily observed in private and government hospitals. During discussions with domain experts, the

discrepancy in quality and the lack of government coverage for expenses due to these regulations being subjective and non-compliant for both SGK and hospitals were frequently highlighted.

In light of the aforementioned circumstances, the objective of this research project is to develop a supportive machine learning model as part of an easily deployable automated system with increased interoperability. The system aims to provide doctors with a modern Web interface for streamlined report generation specifically tailored for SGK. By implementing some predefined templates for each ICU unit, the system seeks to enhance and enforce standardization and ensure consistency, thereby further improve the quality of ICU management. Hence, the primary problem that is intended to be resolved is the conflict between hospitals and the refunding institution SGK. This will be achieved by introducing objective and non-debatable admission criteria by enforcing new regulations devised by domain experts. Classification models have been employed to decide on the most appropriate ICU level for a given patient based on his/her various body indicators. The classifiers have been trained using data extracted and adapted from MIMIC-III due to the unavailability of ICU specific data which could be used in the training. Additionally, this project seeks to provide AI-powered information tool to further assist doctors in their decision-making process; it introduces explainable and fact-based data analysis for supporting streamlined report generation. The results reported based on MIMIC-III dataset are encouraging; they demonstrate the effectiveness and applicability of the developed system.

The rest of this paper is organized as follows. Section II is a brief overview of the related literature. Section III cover the necessary background. The methodology is presented in Section IV. The results obtained using the MIMIC-III dataset are reported in Section V. Section VI includes the discussion. Section VII is conclusions.

II. LITERATURE REVIEW

The literature review in this project is mainly divided into two parts. Papers related to ICU admission criteria, ICU levels and general information about ICU wards were collected, followed by the application of machine learning and deep learning for better handling the ICU ward. A literature review related to ICU information shows that there are certain admission criteria before transferring a patient to ICU. These criteria can be enumerated as (1) when to admit patients, (2) initial treatment, and (3) basic monitoring equipment provided in the ICU ward. Determining patient admission time is one of the crucial steps when it is intended to consider ICU in medical treatment. Minimizing the time for admission improves the chance of survival and improves patients' recovery duration, thus reduces the ICU length of stay. Although time is crucial, unnecessary admission needs to be avoided since ICU is a scarce and expensive resource maintained in hospitals, and only eligible patients need to be admitted. Although patient eligibility can be measured using

several analysis techniques, it should have less priority below vital function support [4].

After vital support is established for critically ill patients, resources that will be provided for the patient need to be determined using several analysis techniques. These resources vary greatly in different sections of the ICU ward; these sections are referred to as the ICU levels, and they are mainly divided into priority levels of patients: (1) critically ill patients who are characterized as severe cases should be admitted to priority 1, (2) less critical patients are classified into priority 2, etc. These prioritization levels are generally determined by the Society of Critical Care Medicine (SCCM). Since the ICU is an expensive system for maintaining and providing its services to patients, ICU resources should be allocated in the most possible efficient way. Although there are several analyses techniques provided for specific situations, misclassification of patients could be encountered in some hospitals. This is sometimes done intentionally for the hospital to charge higher rates. Unfortunately, misclassification of patients may lead to misuse of resources and mostly not having the appropriate resources accessible for patients who need them. Consequently, mortality rate may be negatively affected [5].

In addition to inefficient use of ICU resources and prioritization of patients into ICU levels, when difficulty of patient transfer from the emergency ward to ICU occurs, certain risks may appear and affect the mortality rate. For instance, Dacosta et al. [6] classified patients transfer to ICU into either slow or rapid transfer based on a threshold. The authors indicated that from the 80% slow transferred patients a larger percentage died in the hospital.

There exist a large number of utilities and resources that need to be efficiently allocated among ICU levels, and prioritized according to patients' criticality. These resources need to be managed by considering certain standards [7], [8]. The usage and type of utility needed also vary based on ICU levels, and more support is needed for critically ill patients; accordingly, the distribution of resources and utilities should be categorized according to ICU levels. These standards may change depending on the state of emergency. Certain categories are determined and reflected as color coded of patients' assessment scores [9]. Furthermore, a considerable amount of research has been conducted on mortality prediction and sepsis prediction. Most information provided on these works is mostly collected in systematic review papers, e.g., [10] and [11]. Mortality rate prediction may be guided by an essential study which compared some machine learning algorithms for predicting COVID-19 patients mortality rate [12].

Generally, traditional machine learning algorithms like K-nearest neighbor (KNN), support vector machine (SVM), Gaussian Process, etc. have been mainly used in this domain. Although some of these algorithms do not provide a probability rate easily, algorithms that transform classification results to a probabilistic distribution exist. The latter algorithms have been used by Sonu Subudhi et al. [13], who mentioned

that the Brier score could be used for determining the classification error. Further, comprehensive research has been conducted for predicting the mortality rate from medical EHR data [13]. Yilmazcan Ozyurt et al. introduced an approach for early warning and mortality prediction models by implementing a “deep Markov model”. The authors described the provision of a real-time mortality risk using AttDMM, which serves as a warning system. This real-time approach offers more comprehensive monitoring than other machine learning models which only estimate a single point in time for ICU mortality.

In addition to traditional method, intermediate machine learning algorithms have been observed to perform better, e.g., ensemble models [14]. For instance, Ramin Ghorbani et al. introduced an ensemble hybrid model to forecast the fatality ratio among critical patients by solving the unbalanced data problem with various methods. The dataset used to train the model was generated according to admissions to Shadid Beheshti University of Medical Sciences and Health Services in Iran between 2013 and 2019. The dataset was assembled from ICU patients within 24 hours of admission. The dataset consists of 1,999 records and 21 attributes. The researchers handled the unbalanced data using oversampling methods (SVM-SMOTE). They also used validation techniques like shuffled 5-fold cross validation and random hold-out.

Another study investigated the Cox proportional hazard (CPH) model for ICU mortality prediction [15]. The authors checked how the CPH model can be applied for the selection of features in critical care unit fatality prediction models. The data used in this research was used for the Physio net 2012 challenge; it is a subset of the MIMIC-2 dataset that contains information about ICU patients admitted to Boston’s Beth Israel Deaconess Medical Center from 2001 to 2008. It is available on the Physio net 2012 challenge website. A positive aspect of the research is that it introduced new methods for feature selection and showed their importance to produce the results.

Most of the current research utilizes MIMIC-III dataset which contains structured data of approximately 57,000 patients with ICU first-day measurements and additional features such as caregiver notes. The data was collected as part of regular routine hospital care, alleviating the burden on caregivers to manually collect this information. Additionally, data was obtained from various sources, including “Critical Care Information System,” the “Hospital Electronic Health Record” database, and the “Social Security Administration Death Main File” to capture data pertaining to the care process [16]. A tool called MIMIC-Extract tries to address the problem of a lack of processing frameworks for healthcare research for collecting and transforming data from electronic health records (EHR) and from MIMIC-III database [17].

In most countries the ICU ward is regulated and managed by institutions and government policies, e.g., [1]. The intensive care society in the United Kingdom prepared a report

which describes the regulations of the ICU ward. The report covers various aspects related to ICU, including staffing and resource allocation at each level of the ICU, patient-level classification, etc. Furthermore, ICU regulations have been comprehensively studied and are frequently updated in the United States [2]. These demonstrate the global interest in managing and handling the ICU ward better to the benefit of all parties, including hospitals, patients and parties covering the cost associated.

ICU guidelines have been prepared in a similar way in Turkey. They are set by the government in consultation with various healthcare institutions. They heavily reference United States research on this matter, and they are periodically updated. Additionally, in Turkey the government classifies hospitals and grants to each hospital the class of ICU patients who can be treated at the specific hospital. Accordingly, hospitals can treat patients at levels up to the one granted by the government [18].

Researchers also investigated bed occupation prediction and patient flow prediction. For instance, Chia et al. [19] focused on building a neural network model as a simple and generalized model for patient flow in the emergency department. They also predicted the number of patients arriving to the emergency department. However, they did not provide a comprehensive analysis of the structure of the data which flows through the ICU and the useful features that could be more beneficial. This helped in developing patient flow prediction leading to a comprehensive report to nursing staff.

The work of Johnson et al. [20] utilizes Bayesian networks by bundling domain knowledge and time series data into a single encapsulated structure for predicting hourly flow through the emergency department. They utilized the results for better allocation of the resources. They built a web app for managing the flow and the model. We developed a more advanced system compared to what they have built for the emergency department. Compared to PatientFlowNet [19], the system has been tested at Aalborg University Hospital; it had a direct impact on the emergency department. The tech stack that they utilized is simpler than what we have built. We provided richer selections regarding the choice of the platform. Further our work surpasses the research of Johnson et al. [20] on use cases. They reported some use cases in the front-end for Aalborg University Hospital emergency department.

We benefited from the work of Wang et al. [21] which, among other aspects, concentrated on occupancy prediction. They provide a framework and a data warehouse for building different kinds of models according to hospital needs. Their approach for predicting patient flow is different from the other approaches described in the literature. They predict the overall hospital occupancy by predicting the input and output of patients into the emergency department. They predict patient flow into the emergency department. Then, according to the predicted length of stay, they individually calculate the overall stay of the patient at the hospital and in the

ICU ward in particular. For patient types, they use the most common illnesses and treatments instead of focusing on every symptom that may be encountered in patients. This makes data exploration much easier and provides a more generalized and efficient algorithm.

We also inspired from the work of Wang et al. [21] the way they produce and communicate the various types of reports. We also inspired from the work of Eddleston et al. [22] which provided a model for predicting bed occupancy in Singapore. Compared to other related works, they predict the occupancy which is divided into different ward types from Class C to Class A. For the framework, they used fewer and simpler models compared to previous works, e.g., PatientFlowNet. They utilized the regression model for bed demand forecasting, the Poisson model for obtaining predictions with a probabilistic distribution, etc.

III. BACKGROUND

For the reader to further understand the research conducted in this study, we briefly cover some of the basic resources utilized; these include the MIMIC-III dataset, as well as important algorithms and models that have been used in the clustering and classification phase of the study.

A. MIMIC-III

MIMIC-III, stands for “Medical Information Mart for Intensive Care III,” is a comprehensive and publicly available database designed to support research in the field of critical care medicine. It was produced by MIT Laboratory for Computational Physiology. It is widely utilized by researchers, healthcare professionals, and data scientists to gain insights into various aspects of patient care and outcomes within ICU. We particularly used MIMIC-III Version 1.4 in this research

MIMIC-III contains de-identified data related to over 40,000 patients who were admitted to ICU at the Beth Israel Deaconess Medical Center in Boston, Massachusetts, between 2001 and 2012. The dataset is rich in clinical information, including demographics, vital signs, laboratory results, medications, diagnoses, procedures, and more. This extensive collection of data offers a valuable resource for studying patterns, developing predictive models, and conducting retrospective analyses to better understand critical care practices, patient trajectories, and treatment outcomes.

Researchers and healthcare professionals use MIMIC-III to address various research questions, such as evaluating the efficacy of interventions, predicting patient deterioration, studying disease progression, and identifying factors that contribute to patient outcomes. The dataset’s large size and diversity provide an opportunity to explore a wide range of clinical scenarios and conduct studies that can potentially lead to improvements in patient care and healthcare delivery. However, it is important to note that due to the sensitive nature of medical data, MIMIC-III has been de-identified to protect patient privacy. Researchers and users accessing the dataset are required to adhere to ethical and legal guidelines to ensure

the responsible usage of the data while preserving patient confidentiality.

B. CLUSTERING

Clustering is the distribution of a given set of objects into groups such that similar objects end up in the same group. The compactness of each group together with the separation of the groups confirms the quality of the clustering outcome. A large number of clustering techniques could be found in the literature. In this section, we briefly cover clustering techniques which we have used in this study.

A Gaussian Mixture Model (GMM) is a probabilistic model used in statistics and machine learning for representing complex data distributions as a combination of multiple Gaussian (normal) distributions. It is particularly useful when dealing with data that may come from a mixture of different processes. GMM assumes that the data is generated by a weighted sum of several Gaussian distributions, where each Gaussian component represents a distinct cluster or mode within the data.

Consider a dataset $X = \{x_1, x_2, \dots, x_n\}$, where x_i is a data point in a d -dimensional space. GMM represents the data as a mixture of K Gaussian components:

$$p(x) = \sum_{k=1}^K \pi_k \mathcal{N}(x | \mu_k, \Sigma_k) \quad (1)$$

Here, $p(x)$ is the probability density function of GMM, π_k is the mixing coefficient of the k th component (satisfying $\sum_{k=1}^K \pi_k = 1$), $\mathcal{N}(x | \mu_k, \Sigma_k)$ is the Gaussian distribution with mean μ_k and covariance matrix Σ_k for the k th component.

The expectation-maximization (EM) algorithm is commonly used to estimate the parameters of GMM. The EM algorithm alternates between two steps: the E-step (Expectation step), where the probabilities of data points belonging to each component are computed, and the M-step (Maximization step), where the parameters (π_k , μ_k and Σ_k) are updated based on the computed probabilities.

GMMs find applications in various fields, such as clustering, density estimation, and data generation. They can model complex data distributions and identify the underlying patterns within the input data. The latter patterns might not be captured well by a single Gaussian distribution. However, it is essential to consider the appropriate number of components (clusters) and to handle challenges such as convergence to local optima during the optimization process.

K-Means is a popular clustering algorithm used in machine learning and data analysis. It aims to group a dataset into a predefined number of clusters. The algorithm works by iteratively assigning data points to the nearest cluster center and then updating the cluster centers based on the newly assigned points.

Mathematically, K-Means can be described as follows. Given a dataset $X = \{x_1, x_2, \dots, x_n\}$, where x_i is a data point in a d -dimensional space, and a user-defined number

of clusters K . For numerical attributes, the goal is to find K cluster centers $\{c_1, c_2, \dots, c_K\}$ that minimize the sum of squared distances between data points and their respective cluster centers:

$$\operatorname{argmin}_{c_1, c_2, \dots, c_K} \sum_{i=1}^n \min_{j=1}^K \|x_i - c_j\|^2 \quad (2)$$

The algorithm starts by randomly initializing the cluster centers. In each iteration, it performs two steps:

- 1) 1. Assignment Step (Expectation): Each data point is assigned to the nearest cluster center.
- 2) 2. Update Step (Maximization): The center of each cluster is updated as the mean of all data points assigned to the cluster.

These steps are repeated until convergence is achieved, usually when the assignments and cluster centers no longer change significantly. K-Means is efficient and works well when the clusters are spherical and have relatively uniform sizes. However, it is sensitive to the initial selection of cluster centers and might converge to local optima. Various techniques, such as trying different initializations and using more advanced clustering algorithms, can help mitigate these issues.

C. CLASSIFICATION

Classification is a technique which builds a model capable of learning from a given set of data instances, called the training set, the characteristics of each of a set of known classes. The success of the process is measured by checking its ability to correctly determine the class of a set of unseen data instances forming the test set. While classification is known as learning by example, clustering is accepted as learning by observation. The rest of this section covers the classification techniques used in our study.

XGBoost (Extreme Gradient Boosting) is a powerful machine learning algorithm known for its effectiveness in predictive modeling tasks. It falls under the category of boosting algorithms. It is particularly favored for its ability to handle complex relationships within data. XGBoost combines the predictions of multiple weak learners, typically decision trees, to create a strong predictive model. It does this by employing a process called boosting, where each subsequent model corrects the errors of the previous ones. In XGBoost, the algorithm builds decision trees sequentially, focusing on data points that were previously misclassified or had high prediction errors. Each new tree gives more weight to these exceptional cases, and hence gradually improving the overall model's performance. The key strengths of XGBoost include its regularized learning, which helps preventing overfitting, and its ability to effectively handle missing data and nonlinear relationships. It also provides insights into feature importance, aiding in feature selection and understanding the driving factors behind predictions. XGBoost offers a range of parameters that can be tuned for optimal performance; these include the number of trees, depth of trees, and learning rate. It is widely used in various machine

learning competitions and real-world applications due to its strong predictive capabilities and robustness.

Ensemble Model is a machine learning technique that combines the predictions of multiple individual models to create a more accurate and robust overall prediction. The idea behind ensembling is that by combining the strengths of different models, the weaknesses of one can be offset by the strengths of another, resulting in improved overall performance. Ensemble models work by training several base models on the same dataset and then combining their predictions in various ways. Two common types of ensemble methods are:

- 1) **Bagging (Bootstrap Aggregating)**: In bagging, multiple instances of the same base model are trained on different subsets of the training data, produced through random sampling with replacement. These models vote or average their predictions to produce the final prediction.
- 2) **Boosting**: Boosting focuses on sequentially training models, where each new model pays more attention to the data points that previous models misclassified. It assigns more weight to difficult cases. This helps in correcting errors produced by earlier models.

Ensemble models, such as random forest (a bagging technique) and gradient boosting (a boosting technique), often outperform individual models due to their ability to reduce overfitting, capture complex patterns, and generalize well on diverse datasets. They are widely used in machine learning for tasks such as classification, regression, and even more advanced tasks, such as object detection and natural language processing.

Voting Classifier is an ensemble machine learning technique that combines the predictions of multiple individual classifiers or models to make a final prediction. It operates by allowing each classifier to vote on the predicted class for a given input, and the class with the most votes becomes the overall prediction. Voting classifiers are particularly useful when different models excel in different aspects of a problem, as their combined decision can lead to better overall accuracy and generalization. There are two main types of voting classifiers:

- 1) **Hard Voting**: In hard voting, the prediction of each individual classifier is treated as a vote, and the class with the majority of votes is chosen as the final prediction. This is effective when the individual classifiers are diverse and bring different perspectives to the problem.
- 2) **Soft Voting**: In soft voting, the predicted probabilities for each class from the individual classifiers are averaged or weighted, and the class with the highest average probability becomes the final prediction. Soft voting takes into account the confidence level of each classifier's prediction; it can be more robust when the classifiers provide probability estimates.

Voting classifiers are simple to implement, and they can be used with various types of base classifiers, including decision

trees, support vector machines, and logistic regression. They form a practical choice for improving model performance and can be used in a variety of machine learning tasks, such as classification and sometimes regression.

IV. METHODOLOGY

The methodology of this research is organized into subcategories to comprehensively explain the different components of the proposed system. First, we introduce the development of a supportive machine learning model from initial data extraction to testing/evaluation. Next, we introduce “Heart-beat,” a platform-independent software system. Finally, we elucidate the steps and requirements outlined by domain experts in the development of an intuitive interface specifically designed for doctors to generate reports efficiently and effectively.

A. SUPPORTIVE MACHINE LEARNING MODEL

The initial development of the supportive machine learning model involved close collaboration by regular discussions with medical doctors at Istanbul Medipol University hospital. The steps of the model development can be summarized as follows: (1) data extraction, (2) preprocessing, (3) clustering, (4) classification, and (5) evaluation. For the purpose of building and testing a foundational model for future use in the hospital setting, the MIMIC-III dataset has been employed due to its suitability. However, as MIMIC-III database does not directly provide data that can be seamlessly integrated into our development process, a systematic approach to data extraction becomes essential. In MIMIC-III dataset, there is no direct target feature for ICU levels. Instead, it includes only the type of ward and whether the patient has been admitted to a different type of ward together with the treatment process. Therefore, we face to choose either create our own target variable with the help of domain experts or use an unsupervised learning approach. Both strategies were tested, and the critical decisions made are further detailed in the discussion section.

Figure 1 provides a big picture view of the overall process from data extraction to the evaluation phase; individual phases are later explained in more detail in their respective sections. In essence, the initial stage involves extracting data from MIMIC-III database. Given its vast array of information ranging from lab results and vital signs to handwritten caregiver reports, a careful selection and filtration process is required. After extracting the relevant data in .csv format, additional preprocessing steps are necessary to prepare the data for model training and testing phases.

Notably, MIMIC-III lacks ICU levels, which is our target variable, thus necessitating either manual labeling of samples or clustering. This aspect is further explored in the classification/clustering section, where hybrid solutions are proposed. These approaches require dimensionality reduction to visualize the dataset and execute clustering algorithms efficiently. Following clustering, the labels of each cluster generated from different clustering algorithms applied to the

dimensionally reduced dataset, are assigned to individual patient records. Subsequently, mortality rates are mapped to ICU levels, with detailed explanations provided in the classification section. Once the levels are mapped and cluster labels are allocated per patient, classification algorithms are applied.

1) DATA EXTRACTION PHASE

For the purpose of building and testing a foundational model for future use in the hospital setting, MIMIC-III v1.4 was utilized. However, as MIMIC-III does not directly provide data that can be seamlessly integrated into our development process, a systematic approach for data extraction becomes essential; MIMIC-extract [18] provides good insights for this process.

The complete flow is shown in Figure 2. We divided the data extraction process into two phases. The first phase is to extract the foundational dataset which will serve as the basis for constructing further datasets. Initially, three base tables were identified, these tables are `admissions`, `icu_details`, and `icustays`. These tables preserve the most basic and fundamental information about patients (e.g., demographics), ICU details (e.g., the admitted ward), admission date-time, the stay time, etc.

After these three tables were joined, a cohort filtering was done. Cohort filtering includes only patients who were first time admitted to ICU, (since MIMIC-III is a decade long database and it presents patients who were admitted to/discharged from ICU more than once over time). This approach allows us to modify a single phase over the whole pipeline. This will allow us to make sure only patients who pass through the cohort filter will be present in the main dataset produced at the end. After selecting the relevant patients, a preprocessing step was applied on the filtered datasets, resulting in a base/foundational dataset.

In the second phase, a more dynamic approach was followed for creating the main/final dataset ready for model development. Compared to MIMIC-extract, instead of focusing on time-series data directly, we tried to include a variety of measurements contained in different tables such as comorbidity scores, medical severity scores, etc. based on the literature review or feedback from doctors at the University hospital. This processing pipeline provides flexibility to adjust the tables in our main dataset creation process. It allows us to incorporate contextual information, such as whether to include or exclude `Elixhauser`' scores based on their relevance or similarly. We can decide whether to include or omit height measurements of patients based on their significance to the methodology and the problem we aim to address. This process generates one dataset. Alternatively, we can maintain the `everyscore` table and only join the `elixhauser` table to create a separate dataset. This flexible approach allows us to control the dataset production with minimal adjustments to the table list. It particularly suits environments where research progresses with frequent feedback. This pipeline ensures that the joined table derived

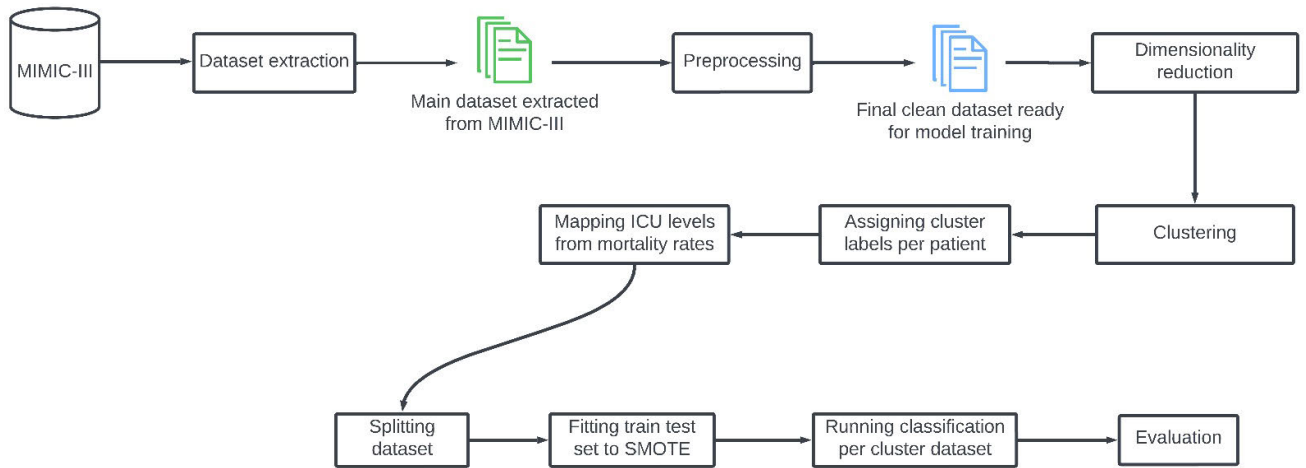


FIGURE 1. A summarized overview of the entire methodology. It begins with data extraction from MIMIC-III; it then constructs a base or foundational dataset which is used to form the main dataset. The latter dataset undergoes preprocessing steps, including encoding, aggregation, and imputation. After the dataset is prepared in the preprocessing step, the method proceeds to model operations, such as dimensionality reduction, clustering, and classification. Cluster labels are assigned, leading to the creation of separate models for each cluster. Ground truth ICU levels are derived from mortality rates and assigned to each patient. Finally, separate models are trained for each cluster using cluster labels, followed by an evaluation step.

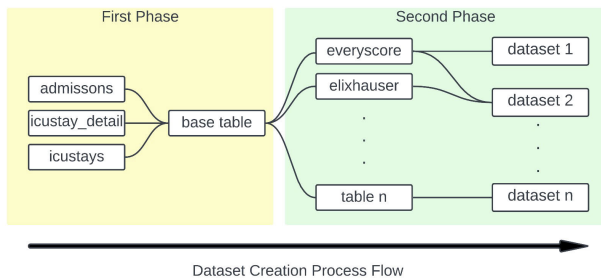


FIGURE 2. Dataset extraction process, the first phase shows generating foundational base dataset that will be fed to the second phase. The generated dataset is used for creating further datasets by joining them with tables which store specific information from comorbidity scores to medical severity scores, and so on. This type of approach provides more flexibility in terms of generating and producing a clean dataset.

from the base or foundational dataset will exclusively contain patients who have undergone the cohort selection processes and preprocessing steps. Consequently, we focused on solely obtaining the health records and measurements of these selected patients. With the final dataset in hand, it is forwarded through a preprocessing pipeline to obtain refined dataset ready for training.

2) PREPROCESSING PHASE

After the final dataset was obtained, a series of preprocessing and exclusion operations were performed. Preprocessing was performed using our customized preprocessing pipeline, where we were able to control intermediary steps in the pipeline, such as caching and turning verbose on and off in case needed; we can also add additional mediator operations, if needed. Figure 3 shows the detailed flow of the preprocessing pipeline with the number of records and shape/size of the dataset.

After obtaining the main dataset from the data extraction phase, it undergoes a series of minor and major operations. These operations remain consistent across different dataset

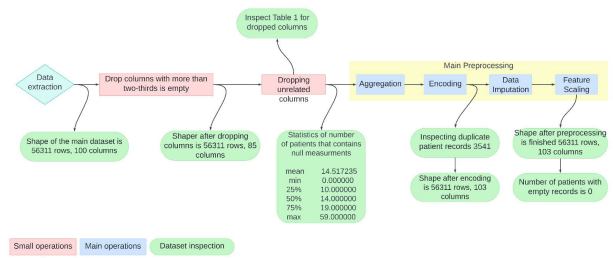


FIGURE 3. The whole preprocessing task with additional information provided in between the steps throughout the pipeline. Initially, the dataset extracted from MIMIC-III which is the main dataset built upon the foundational dataset is forwarded to the preprocessing pipeline. First, columns which have $\frac{2}{3}$ of the values missing as well as unrelated columns are dropped as specified under Table 1; second, aggregation is performed on the appropriate columns with the appropriate aggregation functions; third, separate encoding techniques are applied on the dataset; for values that have 1 or 3, one hot encoding is used, while for diagnosis frequency encoding is used since it includes over 15,000 unique values. Finally, feature scaling is done.

TABLE 1. Number of empty values for each column.

Column name	Number of empty values
avg_fio2	25009
avg_chloride	26629
avg_temperature	26284
avg_tidalvolume	30555
avg_albumin_min	22989
avg_albumin_max	22989
avg_peep	34784
avg_bicarbonate	52245
avg_o2flow	45942
avg_bands_min	41590
avg_bands_max	41590
avg_aado2	38324
avg_requiredo2	38323
avg_carboxyhemoglobin	54538
avg_methemoglobin	55357

versions by applying the exact same procedures to each version obtained from the dataset extraction phase. Initially,

columns containing a significant percentage of empty values were dropped. Subsequently, unrelated columns such as insurance type, timestamps, and stay IDs were also removed. During this stage, we observed that certain features which contain null values within the dataset exceed $\frac{2}{3}$ of the dataset's size. Consequently, these features were deemed insignificant and were subsequently removed. Table 1 contains all the dropped columns.

After further investigation, we discovered that certain measurements, particularly in the lab records, lack mean features due to having minimal and maximal records for specific patients. To streamline the dataset and reduce the number of repeated features along with varying ones that cause duplication of patients, we decided to aggregate these measurements and calculate their mean values. Additionally, to handle the remaining small number of duplicate records, we aggregated the mean measurements of patients recorded on the same day to obtain a single consolidated patient record. This approach does not only improve the dataset compactness, but also ensures meaningful and comprehensive analysis. All the aggregated columns are from the lab measurements, since those are features that vary while the rest of the patient data stays the same, hence resulting in high cardinal dataset.

After the aggregation was performed, we were left with a relatively refined dataset suitable for exploration; hence, we cached the dataset in this step, and further used it in the explosion phase.

After the aggregation and removal operations were completed on the dataset, different encoding techniques are performed on categorical features depending on their statistics. For instance, binary encoding may be applied on gender; for the rest of the categorical features, it is possible to apply one-hot encoding with `panda's pd.get_dummies()` function. For categorical features that have a large number of unique values, e.g., Diagnosis column, frequency encoding is applied instead. After completing the encoding process for each categorical feature, we proceed with data imputation using an iterative imputer.

Figure 3 shows the statistics for the number of empty values for each object, on average 14 features were identified as empty for a single patient record; all of these features are from lab measurements. Accordingly, a simple imputation method should be enough here. With data imputation and removal of unnecessary columns as well as columns with most empty values, no empty record is left in the dataset. The final operation is scaling the dataset and making it ready for model training. We employed standard scaling by removing the mean, and scaling to unit variance using the function `StandardScaler` from Skleran library.

3) CLUSTERING

Before running the employed clustering algorithms, some processing steps were performed on the dataset; these steps are different from those enumerated above for the preprocessing pipeline. For this phase of the development of the supportive machine learning model, scaling was

applied with the `sklearn.preprocessing` module `StandardScaler` on the final refined dataset which did not contain any repeated subject measurements (only unique patient records). After scaling was applied on the dataset, the three columns (`subject_id`, `hadm_id` and `icustay_id`) were removed since no further query operations or set operations were needed. As a result, we have the finalized, refined, and scaled dataset for clustering.

After the pre-clustering processing on the dataset was completed, dimensionality reduction was applied to the dataset. Initially, PCA was applied with a component number equal to 20 for further reduction and visualization of the dataset. Then two different algorithms, namely t-SNE and UMAP, were applied separately on PCA data. The results of both t-SNE and UMAP are shown in Figure 4.

After dimensionality reduction was completed with two different approaches, namely (PCA + t-SNE) and (PCA + UMAP), different clustering algorithms (e.g., Gaussian Mixture Models, K-Means, Spectral, Birch, DBSCAN, HDBSCAN) were applied on the reduced dataset. Figure 5 provides the results of the clustering applied on the reduced dataset.

In the last stage, silhouette scores were computed using the labeling outcomes derived from the clustering algorithms. However, it is worth noting that density-based clustering algorithms do not effectively capture the underlying clusters, as evident in Figure 5. As a result, the analysis excluded the density clustering algorithms. The outcomes of this comprehensive analysis are meticulously presented in the subsequent section. Among the clustering techniques considered, notable performance was observed for some algorithms, namely, K-Means, GMM, and BIRCH. Both K-Means and GMM exhibited slightly superior scores. In case it is important to have the model easy to understand and user-friendly, choosing K-Means or GMM becomes a viable choice. It is worth mentioning that for the sake of simplicity, GMM labels were adopted for the subsequent classification phase.

4) CLASSIFICATION

Once the clustering is applied and the best performing cluster labels are assigned to the dataset as a new column, additional pre-classification operations are performed on the dataset. Initially, as stated in the first part of this section and in the background section, MIMIC-III does not contain any patient ICU level; it only contains the type of ward. Further information could be obtained from medical scores, which can then be mapped to mortality rates using related logit functions of those specific severity scores. For simplicity and initial testing, we obtained mortality rates from the LODS scoring system of the patients, and this feature was calculated from the official logit function. Refer to [1] for more details regarding these scoring systems.

$$\text{logit} = -3.4043 + 0.4173 \cdot \text{score} \quad (3)$$

The value of the logit function score represents the LODS score of the patient. Once the logit score is calculated, the

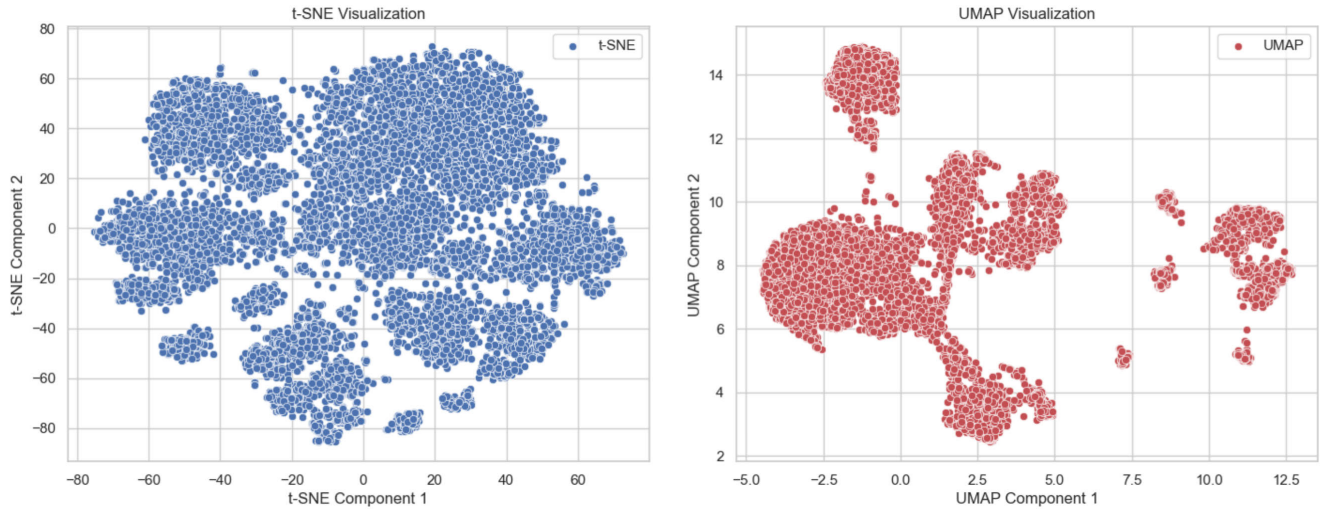


FIGURE 4. Visualization of UMAP (left) and t-SNE (right). For t-SNE and UMAP, the final dataset resulted from the preprocessing phase with 105 features has been reduced to two components and visualized.

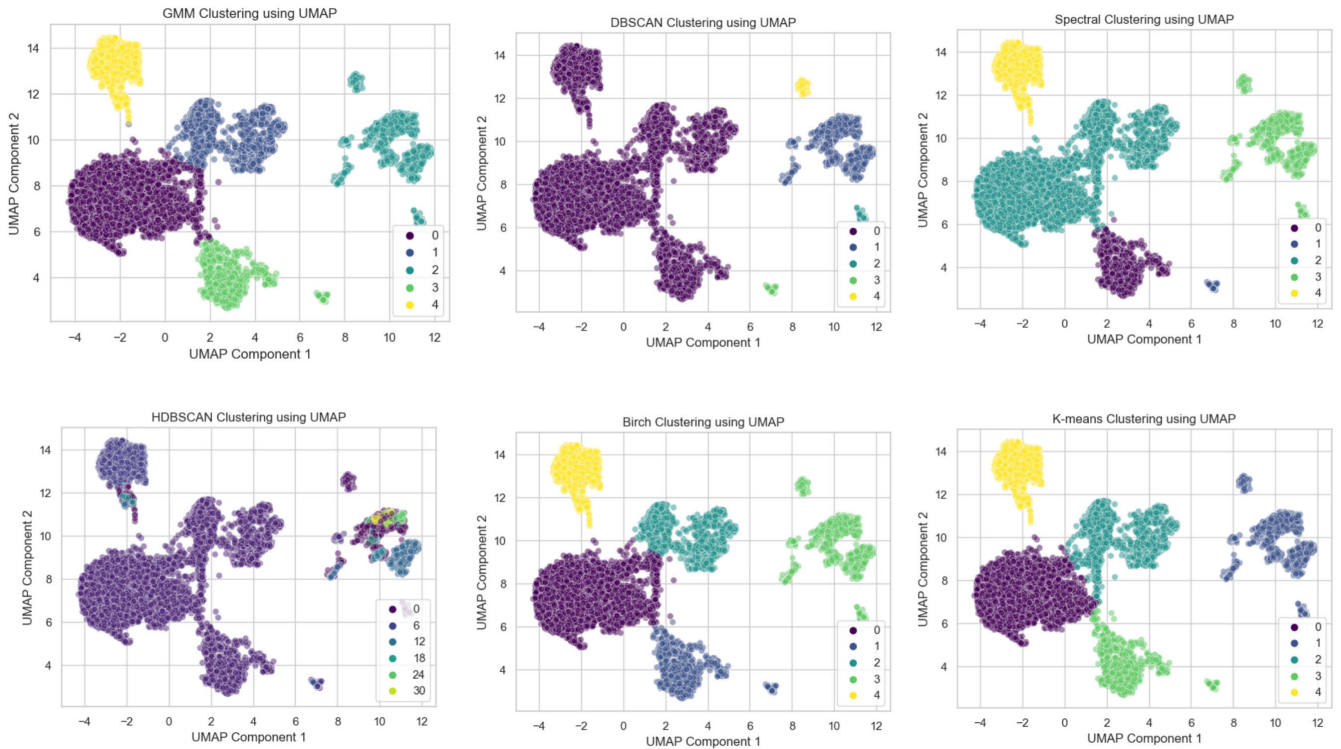


FIGURE 5. Clustering results. Clustering is applied on a dataset reduced with t-SNE which has more clear distinction between clusters and provides better visualization. From left to right, the following clustering algorithms were used, GMM, DBSCAN, Spectral, HDBSCAN, Birch and K-Means. For K-Means and similar algorithms the number of clusters was fixed to 5 clusters; for algorithms which require parameters like DBSCAN, the parameters are set to produce 5 clusters. This will allow for fair comparison.

mortality rate is found using Equation 4. After obtaining the mortality rate for each function by applying Equations 3 and 4 to the LODS score, ICU levels were obtained by mapping each mortality rate to an interval number, where the number of intervals equals to the number of ICU levels as shown in Equation 5. Figure 6 shows the labeled level distribution of

patients, 0 as level 1, 1 as level 2, and 2 as level 3.

$$\frac{e^{\text{logit}}}{1 + e^{\text{logit}}} \tag{4}$$

ICU level mapping is performed by dividing the mortality rate into three groups; the number of groups may increase or

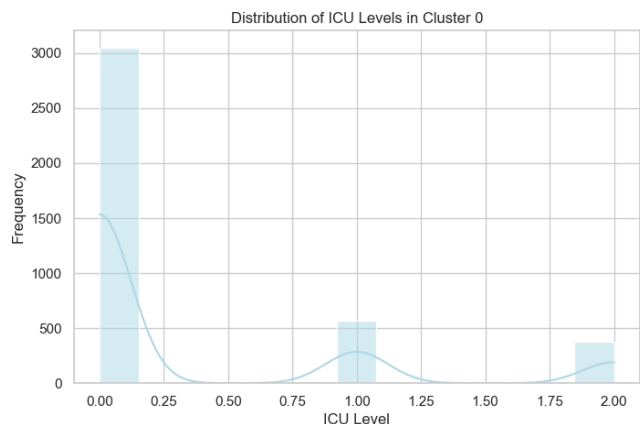


FIGURE 6. Histogram plot of the distribution of patients in ICU levels; these were calculated as described under section IV- A.4 Classification. Most patients are admitted to ICU level 1 which is the level for less critical patients; this is the natural distribution of the patients in ICU levels since critical patients occur less compared to more neutral patients.

decrease depending on the intended ICU levels. For our study, the number of ICU levels was set as 3 which is the common number of ICU levels used in most countries, including Turkey. The three levels used in this study may be interpreted as follows: Level 1 for mild cases, Level 2 for moderate cases, and Level 3 for severe cases. Equation 5 shows the interval distribution.

$$\begin{aligned}
 0 &< \text{mortality_rate} < 0.33 \\
 0.33 &\leq \text{mortality_rate} < 0.66 \\
 0.66 &\leq \text{mortality_rate} < 0.99
 \end{aligned} \tag{5}$$

After obtaining the ICU levels for patients, the features that were used for obtaining ICU levels were removed from the dataset. These features are LODS, mortality rate, and the rest of the medical scoring parameters present in the dataset because some of these scoring parameters have major similarities between them. After labelling patients with the appropriate ICU levels, the classification procedure follows a cluster-then-predict methodology. In this approach, the ensemble model was individually executed on each cluster, thereby accommodating distinct patient cohorts more effectively. The advantages and drawbacks of this approach are explained in Section VI. In essence, owing to the presence of ICU-level labels and cluster labels in the preclassification and post-clustering phases of the dataset, a grouping operation based on cluster labels was activated. Consequently, subsequent split, training, and testing procedures were carried out on these segmented datasets. Five fold cross-validation has been employed in the process. This enabled the execution of classification on distinct clusters in a segregated manner. Finally, it is obvious that there is an apparent class imbalance problem in our dataset when patients are labeled with ICU levels. To resolve this issue the oversampling methodology SMOTE [31] has been applied, other researches also used an under-sampling methodology as well; refer to Figure 1 for details [32].

B. HEARTBEAT

Some of the main challenges of our research problem are the design of the UI and the lack of standardization in the ICU reporting section of the already used programs in the hospital management environment.

We engaged in extensive consultation with domain experts, including medical professionals with different specializations and expertise in varying demographic structures. Their main concerns were identified as the lack of ICU report standards and the usability of the ICU reporting system. In light of this feedback, the following design choices were considered for Heartbeat. These design choices have been further backed by figures and explanations of the system architecture, UI design, and flow of data in the application.

- - **The system should be simple:** Many hospitals rely on third-party software solutions designed for general operational purposes. In our context, this was true, and a single software package was utilized for the whole hospital management system. This required us to develop a software system that would be smoothly integrated within an environment where different software systems are already running.
- - **Modern and compatible UI:** Medical doctors primarily expressed concerns about the lack of standardized reporting of patient status during their ICU engagement. Moreover, this particular use case for patient reporting relied on simple text file editing. Consequently, we decided to develop a user interface that not only standardized the reporting process, but also offered flexibility and modularity to domain experts, enabling them to adapt the standardization to regulatory requirements.

The system architecture is characterized by a relatively simple structure, consisting of a frontend and a backend components. Each hospital management system has its unique protocols for managing patient data and compliance obligations concerning privacy regulations. Rather than standardizing our data, we expose APIs that will be easily integrated with the rest of the hospital’s IT system. Therefore, the standardization and formatting of the data according to regulations could be further processed in the hospital IT system data pipeline. Given these design considerations, Flask and SQLite emerged as the natural choices perfectly suited for this task. Flask offers a streamlined single-file API solution with its dedicated server, while SQLite provides a straightforward file-based database system. Figure 7 provides a summarized view of the system architecture.

As shown in Figure 7, the frontend exists under the Hospital network along with the other IT solutions, including the patient management system, and other 3rd party medical software systems. This separation is emphasized here to show that with API exposed from Flask and SQLite behind, it is straightforward to consume the associated endpoints. This way, with few HTTP requests to the API, Heartbeat should be operational based on the set design considerations as depicted

TABLE 2. Silhouette Score.

Model	Silhouette Score
KMeans	0.614
GMM	0.495
BIRCH	0.614

in Figure 7. The backend source code and the frontend visual interface are, respectively, available under:

- 1) [yigithakverdi/mybs-heartbeat](https://github.com/yigithakverdi/mybs-heartbeat): A simulation/backend system for the ICU environment (github.com), [yigithakverdi/mybs-web-app](https://github.com/yigithakverdi/mybs-web-app): Medipol Yogun Bakim Sistemi web uygulaması (github.com)
- 2) [yigithakverdi/icu-flow-prediction](https://github.com/yigithakverdi/icu-flow-prediction): Series of ML models for ICU patient flow prediction (github.com)

UI Improvements: Significant enhancements have been made to the UI elements. Previous iterations required doctors to manually input their notes using predefined templates tailored to their specific use cases. This approach led to variations in ICU report templates across different wards and among individual doctors. In order to ensure compatibility and seamless integration with the existing hospital IT infrastructure, we decided to use JSON documents to represent regulations for each ICU ward. This further improved the modification and maintenance of the ICU level identification process for patients, since once a regulation is updated it will be straightforward to modify JSON documents and changes would immediately be reflected to relevant UI elements. Instead of visualizing the JSON format directly in the frontend and allowing editing from there on, checkbox trees are employed to represent the hierarchical structure of JSON documents.

Figure 8 illustrates an example of the reflection of the hierarchical structure of a JSON document onto a checkbox tree. This will allow standardization across different ICU wards and will provide consistency on reporting to government institutions, since simply checking or unchecking the boxes will determine the ICU levels of the patients as compared to the traditional method of inconsistent text writing. The algorithm which defines the ICU levels was developed by considering the regulations. It simply consists of multiple conditional statements which implement the applicable rules.

Data in Heartbeat: The data produced by doctors mainly describe patient conditions; it is distinct from the data collection directly from the ICU equipment. This data is presented in categorical format taken from the values in the JSON document that generates the UI elements. Doctor-generated reports primarily address the requirements of government insurance agencies; they do not consider the treatment relevant for individual patients. The latter necessitates more granular and time-sensitive data.

This paper underscores the significance of our software system in addressing critical issues within the realm of ICU management. Figure 78 provides the UI flow diagram and detailed view of the selection page. The source code for

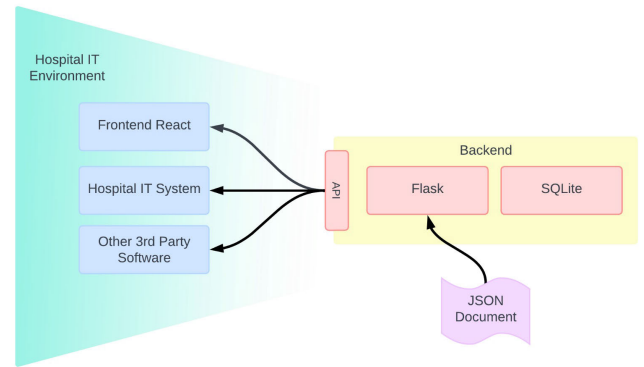


FIGURE 7. A simple system architecture diagram. By simply consuming the endpoints, our application should be operational. A JSON Document is stored in the backend; it is automatically loaded to the frontend after an authorized successful login.

Heartbeat and machine learning experiments are available as open source on the following links:

Backend: <https://github.com/yigithakverdi/mybs-heartbeat>,

Frontend: <https://github.com/yigithakverdi/mybs-web-app>,

Machine learning experiments with notebook: <https://github.com/yigithakverdi/icu-flow-prediction>

V. RESULTS

Before delving into the clustering metrics, we begin by visually inspecting the cumulative variance explained by the PCA components. This step helps us identify the components that capture the most appropriate variance in the data. The cumulative variance plot in Figure 6 displays the distribution of the levels.

For the evaluation of our clustering phase, we employed silhouette scores. These scores provide insights into the quality of our clustering results. Table 2 provides an overview of the silhouette scores obtained from various clustering algorithms. We excluded density-based methods from this comparison. Notably, the clustering plots align closely with the silhouette scores, clearly showcasing the superior performance of the three algorithms GMM, K-Means, and BIRCH.

During the classification phase, we provided a deeper understanding by generating and visualizing the confusion matrices. These matrices allow us to assess the classification performance of our models. The accuracy scores, shown in Table 3, quantify the performance further.

VI. DISCUSSION AND FURTHER IMPROVEMENTS

The clustering process begins with an initial reduction of variances using PCA. To further refine our dataset, we harnessed two distinct algorithms, namely, t-SNE and UMAP. These outcomes are visually depicted in Figure 4. Remarkably, UMAP exhibited a significant proficiency in crafting distinct groups. This pattern held true for t-SNE, yet UMAP consistently demonstrated superior performance, as exemplified in the illustrated instances.

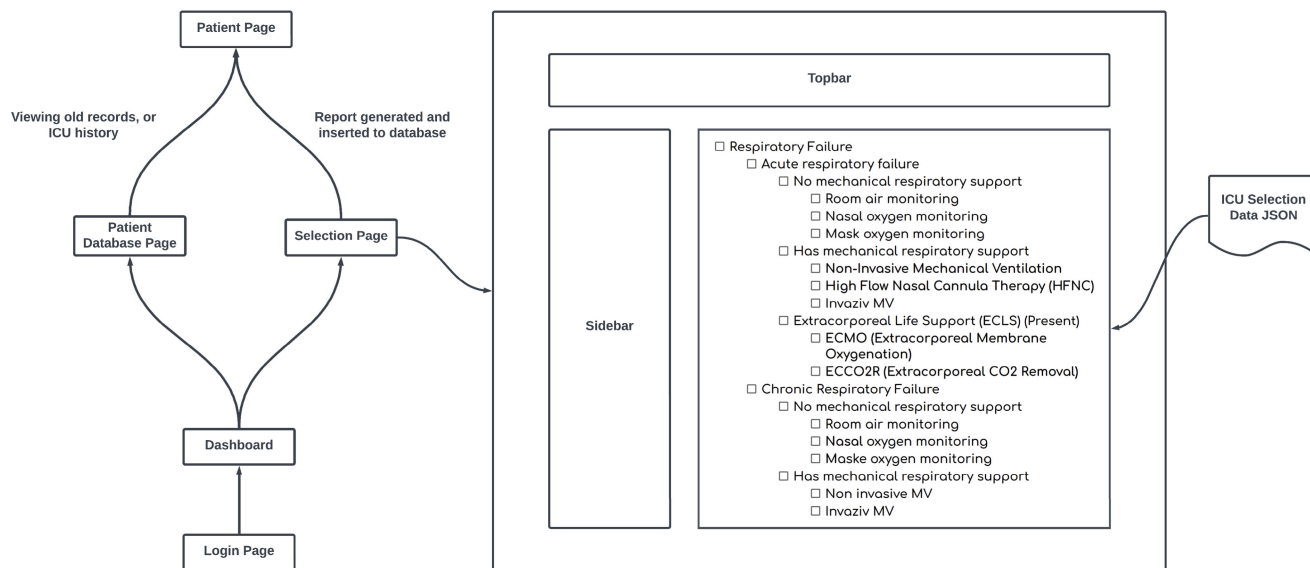


FIGURE 8. User Interface Flow; this UI was designed based on multiple meetings and feedback from medical doctors; UI flow is shown on the left side and UI elements are summarized on the right side. Initially doctors use the login page to access the ICU patient database page; then they select the ICU placement page which best fits the age category of the patient, and data entry is completed accordingly. The selections on the right side are provided as categories with associated checkboxes for seamless data entry. Patients are placed in the ICU levels which fit their case by considering doctor's choices and the related regulations. Categorized checkboxes are visualized using JSON document to facilitate smooth modifications when needed.

TABLE 3. Accuracy scores for each cluster per model.

Model	Cluster 0	Cluster 1	Cluster 2	Cluster 3
RandomForestClassifier	0.827	0.896	0.864	0.915
GradientBoostingClassifier	0.842	0.899	0.857	0.918
SVC	0.778	0.896	0.814	0.876
LogisticRegression	0.806	0.891	0.825	0.925
GaussianNB	0.727	0.704	0.729	0.363
XGBClassifier	0.855	0.904	0.864	0.928
VotingClassifier	0.837	0.899	0.850	0.918

Transitioning to the realm of classification outcomes, our results showcase an exceptional level of accuracy across a spectrum of clusters. It is vital to underscore that each training phase operates independently. To elucidate, upon concluding training for a specific cluster label (considering the example of GMM cluster label 0), the model undergoes a complete reinitialization, embarking on fresh training within the ensuing cluster. This strategic approach empowers us to capture the distinct variances exclusive to each cluster. This strategy aims to foster bespoke classification models for individual patients residing within their corresponding clusters, thereby enhancing our capacity to cater to the unique needs of a diverse patient population.

To streamline our subsequent analysis, we decided to utilize GMM for conducting the classification task based on distinct GMM labels or inclusion in clusters. After completing the clustering process, we examined the level distribution within the clusters. This distribution can be observed in Figure 9.

In the context of extensive consultations with medical specialists from various ICU units (adult, neonatal, pediatric), a consensus was reached regarding feature selection. Notably, the features integrated into our model do not directly mirror regulatory guidelines. To illustrate this, within the context of neonatal ICU placement regulations, patient's

weight emerges as a key determinant. Patients falling within specific weight ranges could experience significant shifts in ICU placement. However, these nuanced variations are absent from our dataset since MIMIC-III lacks explicit level information. Instead, level information is derived indirectly from medical scoring systems as described above. While these scoring systems might indirectly encapsulate features pertinent to regulatory processes (particularly those governing ICU patient placement in Turkey), feature importance results suggest that their influence is comparatively modest when adjacent to other features. Notwithstanding this limitation, our findings underscore the model efficacy by accurately predicting outcomes using a comprehensive array of pertinent features, measurements, and metrics sourced from the ICU environment within the first day of admission. Moreover, we pursued a basic mapping technique by simply dividing the mortality rate by the number of levels; instead, for further experiments, we can map ICU levels using simple clustering focused on them; it is also possible to use different statistical methods.

Considering the challenge at hand, a potential avenue for resolution lies in adopting a hybrid approach. Such an approach could integrate rule-based and statistical methods to map features within the dataset to corresponding levels derived from MIMIC-III dataset. Alternatively, supplemental anonymized data from hospital records could be leveraged to augment this endeavor.

For deeper specialization, a pragmatic path entails segmenting distinct ICU ward types (neonatal, pediatric, adult). By focusing on these ICU units individually, we could tailor models to meet their unique requirements. Additionally, the number of permissible levels might vary depending on the ward; for instance, considering the example of neonatal ICU,

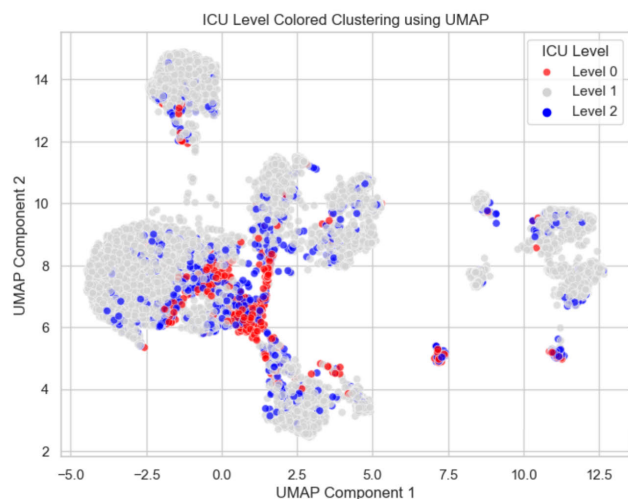


FIGURE 9. Visualization of patients with ICU levels over the clustered and dimensionally reduced t-SNE dataset. Blue colored dots are ICU level 2, grey dots are level 1, and red dots are level 3.

regulatory constraints impose a maximum of five levels, ranging from 1 to 4A, and an additional level, 4B, for severe cases. Given these constraints, level mapping could be divided into five intervals, which could then be dynamically adjusted based on the specific hybrid approach in use.

VII. CONCLUSION

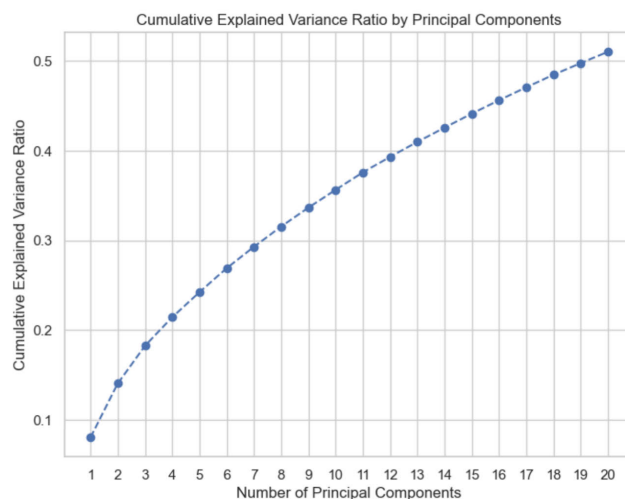
In conclusion, we proposed two solutions to answer the problems associated with patient classification to the ICU levels.

- 1) Supportive Machine Learning
- 2) ICU placement system/software that is easily deployable with increased interoperability

Each of the suggested solutions has received positive feedback from both domain experts, doctors and professors who have contributed to this paper. Their feedback was received in extensive meetings and has been realized in various parts of the project leading to the current version of the system. More feedback will be available from other domain experts who will use the actual system after it will be deployed at a number of hospitals. The latter feedback will be quantified and analyzed to help produce the next version of the system. The associated results will be shared in a future research outcome. We have also received positive responses for our proposed software system “heartbeat” from SGK in further meetings.

Our proposed model individually specializes in different clusterings of patients, and has been proven to accurately classify patients into the ICU levels. Doctors could utilize the proposed models to further support the regulatory process. Furthermore, our model could be utilized on the SGK insurance side to explain why patients are admitted to the predicted level. For our heartbeat system, the improved UI elements and the simple design provide easy deployment to the IT infrastructure/systems of hospitals.

The developed system is expected to be effective once deployed in healthcare providing institutions involved in



ICU admissions. Once set, it is planned to collect data and conduct a comprehensive analysis accordingly. In other words, we realized the need to conduct some tests using real-world data collected from local hospitals in Turkey. However, collecting the data will need deployment of the system at various hospitals. This will be our next step and will take considerable time, and hence has been left as future work. Further, though we used MIMIC-III in this study and reported interesting results, we plan to use MIMIC-IV in the next version of this study to investigate how the new version of the data may affect the results.

REFERENCES

- [1] J. R. Le Gall, J. Klar, S. Lemeshow, F. Saulnier, C. Alberti, A. Artigas, and D. Teres, “The logistic organ dysfunction system. a new way to assess organ dysfunction in the intensive care unit,” *JAMA*, vol. 276, no. 10, pp. 802–810, 1996.
- [2] G. Kour and R. Saabne, “Real-time segmentation of on-line handwritten Arabic script,” in *Proc. 14th Int. Conf. Frontiers Handwriting Recognit.*, Sep. 2014, pp. 417–422.
- [3] G. Kour and R. Saabne, “Fast classification of handwritten on-line Arabic characters,” in *Proc. 6th Int. Conf. Soft Comput. Pattern Recognit.*, Aug. 2014, pp. 312–318.
- [4] G. Hadash, E. Kermany, B. Carmeli, O. Lavi, G. Kour, and A. Jacovi, “Estimate and replace: A novel approach to integrating deep neural networks with existing applications,” 2018, *arXiv:1804.09028*.
- [5] G. Smith and M. Nielsen, “ABC of intensive care: Criteria for admission,” *BMJ*, vol. 318, no. 7197, pp. 1544–1547, Jun. 1999, doi: [10.1136/BMJ.318.7197.1544](https://doi.org/10.1136/BMJ.318.7197.1544).
- [6] D. W. Chang, D. Dacosta, and M. F. Shapiro, “Priority levels in medical intensive care at an academic public hospital,” *JAMA Internal Med.*, vol. 177, no. 2, p. 280, Feb. 2017, doi: [10.1001/JAMAINTERN-MED.2016.8060](https://doi.org/10.1001/JAMAINTERN-MED.2016.8060).
- [7] M. P. Young, V. J. Gooder, K. McBride, B. James, and E. S. Fisher, “Inpatient transfers to the intensive care unit: Delays are associated with increased mortality and morbidity,” *J. Gen. Internal Med.*, vol. 18, no. 2, pp. 77–83, Feb. 2003.
- [8] M. L. Barrett, M. W. Smith, A. Elixhauser, L. S. Honigman, and J. M. Pines, “Utilization of intensive care services,” in *Healthcare Cost and Utilization Project (HCUP) Statistical Briefs* (Statistical Brief #185). Rockville, MD, USA: Agency for Healthcare Research and Quality (US), Dec. 2014. [Online]. Available: <https://www.ncbi.nlm.nih.gov/books/NBK273991/>
- [9] (2013). *Minimum Standards for Intensive Care Units. College of Intensive Care Medicine of Australia and New Zealand ABN: 16 134 292 103*. Accessed: Jul. 2024. [Online]. Available: http://103.12.11.134/cms_files/IC-01%20Minimum%20Standards%20For%20Intensive%20Care%20Units%20-%20Current%20September%202011.pdf

- [10] *Allocation of Critical Care Resources During a Public Health Emergency*, Univ. Pittsburgh, Pittsburgh, PA, USA, Apr. 15, 2020.
- [11] A. Rapsang and D. C. Shyam, "Scoring systems in the intensive care unit: A compendium," *Indian J. Crit. Care Med.*, vol. 18, no. 4, pp. 220–228, 2014, doi: 10.4103/0972-5229.130573.
- [12] D. S. McNair, F. M. Malila, J. E. Zimmerman, and A. A. Kramer, "Acute physiology and chronic health evaluation (APACHE) IV: Hospital mortality assessment for today's critically ill patients," *Crit. Care Med.*, vol. 34, no. 5, pp. 310–1297, 2006.
- [13] M. Scherpf, F. Gräßer, H. Malberg, and S. Zauneder, "Predicting sepsis with a recurrent neural network using the MIMIC III database," *Comput. Biol. Med.*, vol. 113, Oct. 2019, Art. no. 103395.
- [14] D. Shillan, J. A. C. Sterne, A. Champneys, and B. Gibbison, "Use of machine learning to analyse routinely collected intensive care unit data: A systematic review," *Crit. Care*, vol. 23, no. 1, p. 284, Dec. 2019.
- [15] H. Ravaghi, S. Alidoost, R. Mannion, and V. D. Bélorgeot, "Models and methods for determining the optimal number of beds in hospitals and regions: A systematic scoping review," *BMC Health Services Res.*, vol. 20, no. 1, p. 186, Dec. 2020.
- [16] S. Subudhi, A. Verma, A. B. Patel, C. C. Hardin, M. J. Khandekar, H. Lee, D. McEvoy, T. Stylianopoulos, L. L. Munn, S. Dutta, and R. K. Jain, "Comparing machine learning algorithms for predicting ICU admission and mortality in COVID-19," *Npj Digit. Med.*, vol. 4, no. 1, p. 87, May 2021.
- [17] Y. Özyurt, M. Kraus, T. Hatt, and S. Feuerriegel, "AttDMM: An attentive deep Markov model for risk scoring in intensive care units," 2021, *arXiv:2102.04702*.
- [18] R. Ghorbani, R. Ghousi, A. Makui, and A. Atashi, "A new hybrid predictive model to predict the early mortality risk in intensive care units on a highly imbalanced dataset," *IEEE Access*, vol. 8, pp. 141066–141079, 2020.
- [19] A. H. T. Chia, M. S. Khoo, A. Z. Lim, K. E. Ong, Y. Sun, B. P. Nguyen, M. C. H. Chua, and J. Pang, "Explainable machine learning prediction of ICU mortality," *Informat. Med. Unlocked*, vol. 25, Sep. 2021, Art. no. 100674.
- [20] A. E. W. Johnson, T. J. Pollard, L. Shen, L.-W.-H. Lehman, M. Feng, M. Ghassemi, B. Moody, P. Szolovits, L. Anthony Celi, and R. G. Mark, "MIMIC-III, a freely accessible critical care database," *Sci. Data*, vol. 3, no. 1, May 2016, Art. no. 160035.
- [21] S. Wang, M. B. A. McDermott, G. Chauhan, M. C. Hughes, T. Naumann, and M. Ghassemi, "MIMIC-extract: A data extraction, preprocessing, and representation pipeline for MIMIC-III," 2019, *arXiv:1907.08322*.
- [22] D. J. Eddleston, D. D. Goldhill, and D. J. Morris. (2009). *Levels Critical Care for Adult Patients*. [Online]. Available: <https://icmwc.com/wp-content/uploads/2014/02/Revised-Levels-of-Care-21-12-09.pdf> (May 2024). *Türkiye, Government Regulations on ICU*. Accessed: Jul. 2204. [Online]. Available: <https://mevzuat.gov.tr>
- [23] D. P. Dean, D. Y. M. Lau, D. A. Baldwin, and D. T. Wenham. (Mar. 2024). *Levels of Adult Critical Care Second Edition*. [Online]. Available: https://www.cc3n.org.uk/uploads/9/8/4/2/98425184/2021-03_levels_of_care_second_edition.pdf
- [24] Republic Turkey Ministry Justice. (2020). *Turkish Legislation*. Accessed: Jul. 2024. [Online]. Available: <https://justice.gov.tr>
- [25] A. R. Sharafat and M. Bayati, "PatientFlowNet: A deep learning approach to patient flow prediction in emergency departments," *IEEE Access*, vol. 9, pp. 45552–45561, 2021.
- [26] A. L. Madsen, K. G. Olesen, J. M. Mller, N. Sndberg-Jeppesen, F. Jensen, T. M. Larsen, P. Henriksen, M. Lindblad, and T. S. Christensen, "A software system for predicting patient flow at the emergency department of Aalborg university hospital," in *Proc. 10th Int. Conf. Probabilistic Graph. Models, Proc. Mach. Learn. Res.*, vol. 138, 2020, pp. 617–620.
- [27] S. J. Littig and M. W. Isken, "Short term hospital occupancy prediction," *Health Care Manage. Sci.*, vol. 10, no. 1, pp. 47–66, Jan. 2007.
- [28] A. Kumar and J. Mo, "Models for bed occupancy management of a hospital in Singapore," in *Proc. Int. Conf. Ind. Eng. Oper. Manag.*, 2010, pp. 1–20.
- [29] S. S. Lorenzen, M. Nielsen, E. Jimenez-Solem, T. S. Petersen, A. Perner, H.-C. Thorsen-Meyer, C. Igel, and M. Sillesen, "Using machine learning for predicting intensive care unit resource use during the COVID-19 pandemic in Denmark," *Sci. Rep.*, vol. 11, no. 1, p. 18959, Sep. 2021.
- [30] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, Jun. 2002.
- [31] E. J. Tsiklidi, T. Sinno, and S. L. Diamond, "Predicting risk for trauma patients using static and dynamic information from the MIMIC III database," *PLoS One*, vol. 17, no. 1, Jan. 2022, Art. no. e0262523, doi: 10.1371/JOURNAL.PONE.0262523.



YİĞİT HAKVERDİ received the bachelor's degree in computer science from Istanbul Medipol University. He is currently pursuing the M.Sc. degree in cybersecurity with the Sapienza University of Rome.

His coursework includes practical areas like network defense, ethical hacking, distributed systems, cyber-physical system security (critical infrastructure) and furthermore malware analysis focusing on cyber threat intelligence, forensics analysis, and reverse engineering. His professional experiences includes different roles at different companies, Airties, where he was an Automation Engineer, he streamlined testing procedures across access points and mobile devices, optimizing workflows. Additionally, his time with Turknet and SATKO GPS Tracking Systems included software and network engineering internships roles. During this period, he contributed to the development of CRM customer relation software, engaged in SOA and WCF projects, and simulated near real-life network topologies. Furthermore his involvement in research projects spans a wide spectrum of topics, including application of machine learning, deep learning on ICU systems, and cyber-security projects.



MELİH URVE GÜMÜŞ was born in Çanakkale, Türkiye. After completing primary school in his hometown, he pursued his high school education in Istanbul, in 2017. He received the degree in computer engineering from Istanbul Medipol University, in 2022. He is currently pursuing the master's degree with the Sapienza University of Rome, specializing in cybersecurity.



AYHAN TAŞTEKİN received the degree in medicine from Istanbul University, in 1991, the degree in pediatrics from Selçuk University, in 1996, and the degree in neonatology from Atatürk University, in 2004. He is currently a Distinguished Academician and a Researcher in the field of medicine, currently holding positions at various prestigious institutions. With over 60 publications, his contributions significantly impact the medical and academic communities, accumulating over 700 citations, including over 20 highly influential ones. He worked at various positions and institutions, including Selçuklu Medicine Faculty, Selçuk University; Medicine Faculty, Atatürk University; Medicine Faculty, Istanbul Medipol University; and Ministry of Health Çumra State Hospital. His research interests include neonatology and pediatrics, addressing critical areas within these domains.



KADIR İDİN received the M.D. degree from the Faculty of Medicine, Istanbul University-Cerrahpaşa, in 1989. He is currently an Intensivist and an Anesthesiologist. He specialized in anesthesia and reanimation with Haseki Educational Hospital, in 1997. He is currently an Associate Professor with Istanbul Medipol University, Türkiye. He is also the Chief of the Intensive Care Unit, Bahçelievler Medipol Hospital. He is a member of Turkish Anesthesiology and Reanimation Society, Turkish Intensive Care Society, Clinical Enteral-Parenteral Nutrition Society, and European Society of Intensive Care Medicine. His main research interests include acute respiratory distress syndrome, airway management, sepsis, continuous renal replacement therapy, fluid therapies, brain death, and organ donation.



MURAT KANĞIN received the degree in medicine from the Medical School, Ege University, in 1997. He has been a Professor with the Faculty of Medicine, Department of Pediatrics, Istanbul Medipol University, since 2021. Between 1998 and 2002, he completed his medical specialization with İzmir Tepecik Training and Research Hospital. In 2015, he completed his specialization in pediatric intensive care. He was the Chief Physician with Diyarbakir Training and

Research Hospital, from 2008 to 2012. He was the Secretary General of the Ministry of Health, from 2014 to 2016.



TANSEL ÖZYER received the B.Sc. degree from the Computer Engineering Department, Bilkent University, in 1996, the M.Sc. degree from the Computer Engineering Department, Middle East Technical University, in 1998, and the Ph.D. degree in computer science from the University of Calgary, in 2006. He is currently a Professor in computer engineering with Ankara Medipol University, Türkiye. Before that, he was a Professor in computer engineering with TOBB University,

Ankara, Türkiye. He is the Coordinator of the ACM/IEEE ASONAM Conference. He has published a large number of journal articles, edited books, refereed conference papers, and book chapters. His research interests include data science, machine learning, bioinformatics, health informatics, XML, mobile databases, image processing, and computer vision. He is an Associate Editor of SNAM journal and *Encyclopeida of Islam* (ESNAM).



REDA ALHAJJ is currently a Professor in computer science with the University of Calgary, Alberta, Canada. He is also affiliated with Istanbul Medipol University, Istanbul, Türkiye, and University of Southern Denmark. His primary work and research interests include data science, network science, machine learning, and big data management, with an emphasis on scalable techniques and structures for data management and mining, social network analysis with applications

in computational biology and bioinformatics, homeland security, disaster management, sequence analysis with emphasis on domains like financial, weather, traffic, energy, multi-data sources integration, and re-engineering. He is the Founding Editor-in-Chief of the premier *Social Networks Analysis and Mining* (SNAM) (Springer), Series “Lecture Notes on Social Networks” (LNSN) (Springer), and *Network Modeling Analysis in Health Informatics and Bioinformatics* (NHIB) (Springer); the Founding Co-Editor-in-Chief of *Encyclopedia on Social Networks Analysis and Mining* (Springer) (ranked top 3rd in most downloaded sources in computer science, in 2018); the Founding Steering Chair of the flagship conference “IEEE/ACM International Conference on Advances in Social Network Analysis and Mining” and three accompanying symposiums FAB (for big data analysis); and for homeland security and intelligence services (FOSINT-SI) and for health informatics and bioinformatics (HI-BI-BI). He is a member of the editorial board of a number of journals; he has been a guest editor of a number of special issues and edited a number of conference proceedings. He currently leads a large research group of Ph.D. and M.Sc. candidates. He received best graduate supervision award and community service award from the University of Calgary. He recently mentored a number of successful teams, including SANO who ranked first in the Microsoft Imagine Cup Competition in Canada and received KFC Innovation Award in the World Finals held in Russia, TRAK who ranked in the top 15 teams in the open data analysis competition in Canada, Go2There who ranked first in the Imagine Camp competition organized by Microsoft Canada, Funiverse who ranked first in Microsoft Imagine Cup Competition in Canada.

...