Using Machine Learning Techniques to Support the Emergency Department

Computing and Informatics, Vol. 41, 2022, 154–171, doi: 10.31577/cai_2022_1_154

USING MACHINE LEARNING TECHNIQUES TO SUPPORT THE EMERGENCY DEPARTMENT

Roeland A. J. J. van Delft, Renata M. de Carvalho

Eindhoven University of Technology
Eindhoven, Netherlands
e-mail: r.a.j.j.v.delft@student.tue.nl, r.carvalho@tue.nl

Abstract. This research lays down foundations for a stronger presence of machine learning in the emergency department. Using machine learning to make predictions on a patient's situation can increase patient's health and decrease the waiting time. This paper explores to what extent it is possible to accurately predict ER outcome. These predictions will be based on routinely available ER data from a Dutch hospital. The data set used is representative for any Dutch Hospital. Prediction performance is compared between ML predictors. Using random forest and stacked ensemble gathered the best results. This research found that for more than half of the adult patients, the algorithm can very accurately predict hospitalization, with similar results for children and during the COVID-19. Moreover, it is investigated which characteristics and events contribute to the direction of the patient. Finally, several plans are introduced to substantially improve the ER process, for example by quickly reviewing patients selected by the algorithms. These might lead to an ER process that is significantly quicker, with more accurate diagnosis.

Keywords: Machine learning, emergency room, acute healthcare, random forest, hospitalization, diagnosis prediction

1 INTRODUCTION

Each year, there are about 2 million visits to an Emergency Room (ER) in the Netherlands [1]. During these visits, patients receive acute healthcare and are diagnosed. All knowledge about these patients is stored inside a database. The ER is often met with patients that stay there too long. A stay of more than 4 hours is rela-

tively common. This is undesirable, since the ER has a limited capacity, knowledge and facilities for enduring treatment.

Therefore, we investigate possibilities to optimize the diagnostic intake process by finding patterns in data for which patients are very likely to be submitted. The hospital is currently doing limited analysis on their data, while it has a huge potential due to the level of detail.

We will explore if, with a combination of Machine Learning (ML) techniques and an extensive ER data set, we are able to predict diagnosis and risk factors in patient health or ER-time. The information we gather using data mining can be used to optimize the process.

A straightforward aspiration is to minimize the time patients stay at the ER. This improves the patient's health and well-being while reducing costs. Moreover, the algorithms might be able to assist in the diagnosis, enhancing accuracy. And we can better schedule patients and resources, while anticipating on a patient influx. The ER data is spread over 2019 and the first months of 2020. This includes the rise of COVID-19, which had a large impact on hospitals.

The goal of this research is to explore how new ML techniques can be applied to data acquired on the patient during an ER visit. With this, the hospital will better estimate if the patient needs to be submitted. This improves patient service and staff scheduling, ensuring no time or cost is spent unnecessarily. To reach this aim, we focus on answering the following question: *To what extent is it possible to accurately predict ER outcome based on patient and event data?*.

The remainder of the paper is organized as follows. Section 2 discusses how other works approach this problem, while our view is introduced in Section 3. Section 4 shows the results achieved in this research. Finally, we conclude the paper with further discussions in Section 5 and conclusions in Section 6.

2 RELATED WORK

Similar work regarding the use of ER data to derive indications for patients has, compared to reference models, consistently outperformed. However, this research took place in countries where patient influx is different than in the Netherlands.

Multiple approaches all over the world can be found. Riata et al. [2] used data from the US CDC to predict hospitalization using ML, outperforming their reference model. Sun et al. [3] investigated the use of regression models at a hospital in Singapore. Choi et al. [4] (predicting the KTAS level) and Kwon et al. [5] (focusing on deep learning with a massive data set) both took place in South Korea.

These articles are relevant, as for example their triage processes are quite close to a Dutch Triage process. An interesting point made in [4] is that they used both logistic regression in a way familiar to [3] and the Random Forest and XGBoost similar to [2]. They found that these last techniques consistently outperformed the logistic regression. This supports our choice to focus on Random Forest. In [3],

they identify the overcrowding issue and see the potential of the technique to find patients that can be admitted to the hospital earlier.

A similar approach should be viable for the Dutch hospital we cooperate with. We aim to extend and contribute by exploring the value of additional data, like from events at the ER. Moreover, we investigate categorical predictions on the direction of the patient. We also introduce ways to implement the findings in the ER workflow.

3 METHODOLOGY

3.1 Data Set

Variables used. This section describes the data used for the analysis. No personal identifiable information was visible for this research, anonymization was carried out by the hospital in advance. The data is then sorted and coupled in a way that each row represents a single trace, or unique visit, of a patient to the ER. In this context, we considered the following features, among others: *admission ID*, *age*, *gender*, *specialism*, *arrival time*, *triage color*, *transport*, *weight*, *complaint*, *temperature*, *dismissal to*.

The data available for this research includes personal characteristics, independent of the current medical situation. Age, gender and weight describe certain risk factors the patient has. In past research (see Section 2) basic characteristics gave good results.

Triage takes the first measurements of the patient situation. Here, priority is assigned and vitals are observed. During the visit, the patient will undergo several tests and acute treatments, and might need an x-ray scan. Details for all activities are stored in the form of event-specific codes, time and free text fields. We treat radiology and activity data in the same way, which form the event data. Before leaving the ER, all patients will be diagnosed by a doctor. The information stored includes the direction of the patient and the diagnosis. In this paper, we assume the doctor is always right in his review and use the direction of the patient as a dependent variable.

We derive more knowledge from the data than directly available, such as calculating the time patients spent in the hospital, how many patients are at the ER at the same time, account for nights and weekends and if the patient visited the ER multiple times.

Generalisability. Our research includes data from 2019 and 2020, consisting of 76 000 ER visits in total. The average age of the patient was 42 years. The average time they were at the ER is 3 hours. 22 % stayed in the ER for more than 4 hours, and thus overstayed. The characteristics of the data from this ER are compared with the characteristics of general data sets in the Netherlands. In 2012, more than 75 % of the ERs got less than 30 000 visitors [1]. The ER of this research in 2019, with 46 000 visits, likely belongs to one of the more

visited ERs in the Netherlands. Moreover, the ER researched is a level-1 ER. According to Panneman et al. [1], only 20 % of the ERs in the Netherlands are level 1. A level 1 ER has the most advanced facilities and is able to care for all kinds of injuries 24/7 [6].

We can further explore generalisability by looking at certain descriptives. First of all, this hospital has significantly more infants at the ER than normal for the Netherlands. Not unexpected: the hospital is specialized in children care. Overall, the age distribution is pretty similar to the Dutch total of [7]. Slightly more males visited this ER in 2019 (51,7 %), compared to Dutch average of 50 % [1].

With regards to arrival modes the amount of references from the GP is relatively familiar, compared to [7]. However, we do see a difference between the ambulance usage in this hospital and the Dutch average. This does not exceed the confidence interval of this average in [7]. It seems likely that since this hospital is a level 1 trauma hospital, it receives more acute cases. The dismissal directions are numerous, but only a few directions together form the majority, as seen in Table 1. Overall, the dismissal directions in this ER are quite comparable to the directions set in other Dutch ER's. The analysis on the descriptives lead us to believe that the results of this research are viable for other hospitals in the Netherlands.

	This Hospital	The Netherlands
Hospitalized	31.1 %	32 %
Dismissed, no future checks	27.1 %	26 %
Dismissed, check by Clinic	28.2 %	24 %
Dismissed, check by GP	5.0 %	5 %

Table 1. Direction of patients compared to Dutch averages (RIVM 2016 [7])

3.2 Stages

To generate the results of Section 4, four stages are considered:

1. Loading – Performing the initial setup for further analysis;
2. Cleaning – Filtering anomalies in the dataset;
3. Encoding – Translating the dataset into a language understood by ML;
4. Predicting – Making estimates, and evaluating their performance.

The stages are constructed in separate Python scripts, in order to optimize running times. This makes loaded, cleaned and encoded data frames available for all kinds of predictors. The structure between these scripts is stated in Figure 1. Here, rectangles represent python scripts and trapezoids are one or multiple data

frames. We see that the encoder generates multiple data frames, which is due to the fact that each of the predictors need different variables in the dataset. These final data frames included little over 40 columns, with the exact amount depending on the predictors and what we are aiming to predict.

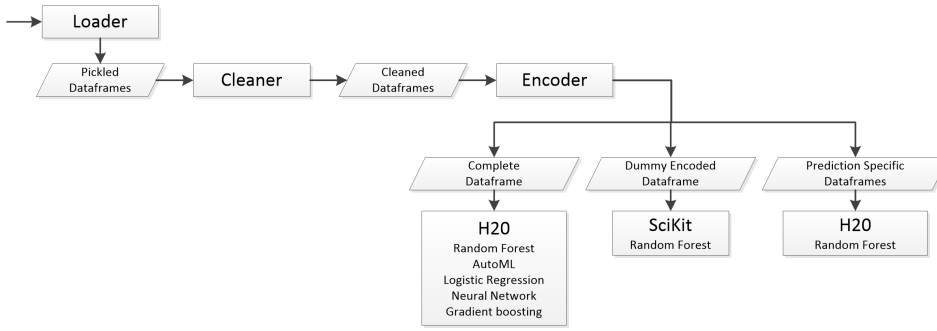

Figure 1. Flowchart program structure

Cleaning data. A sizable amount of our data is quantitative and clinical, which has the nice property that almost all correct entries are within a standard boundary. This is due to humans rarely being alive if their critical values are much higher or lower than usual. In our dataset, these kind of errors were present: someone with an oxygen level of just 4 % that was sent home. Together with staff from the ER, we sat boundaries to determine which data entries are realistic.

Some patients might not have a departure date, or have a departure date more than 24 hours after arrival. These anomalies are removed, as are patients that left against advice. The cleaning script drops all traces that did not have a *TriageTime* or *Triage_color*. Further choices on missing values are made by the predictors. Lastly, it does not make sense to include all possible choices. Therefore, the categories are narrowed down.

Sequential event data. We see two possible approaches to process event data. Either embed supervised learning in the process model (use process mining techniques with decision point analysis), or use the supervised learning and embed information gathered on the workflow in the input data. This project focuses on the latter. We will investigate how the index encode method [8] can help us retain as much information as possible. The encoder uses this index encode variation to pivot all rows in the Activity, Radiology and Orders data frames, so each row corresponds to exactly one trace. In the columns, the sequence in which the events took place is preserved. This technique also ensures predictors are able to use the data in a categorical manner, improving accuracy compared to an one hot encoded analysis. First, each event gets a number, by sorting all activities over time, after which they are grouped by patient. Then we add

the slug of the category to the numbers (in this case *activity_*). This makes the columns for each different event category uniquely identifiable.

The encoder also checks whether the activities took place within a predefined time after arrival. Almost all events occur within the first hour of the patient entering the ER. Therefore, a filter to ensure predictions are made using information that is available at the time of the prediction is not required as long as the predictions are used at least an hour after arrival. Only the first 3 activities and the first radiology event are included, patients with more events are quite rare. This makes the data compact: otherwise, the amount of columns for a specific event is equal to the maximum number of events that a patient might endure.

Natural language processing. Complaint information is a free-text field. Therefore, a list is made of the 100 most used terms, which are then manually narrowed down into a filter including some of the occurring synonyms and spelling mistakes. It is also possible to use the complaint information to correct measurements. An application of this in our research would be that if people list fever as a complaint, and their temperature is not measured, we list their temperature as 38.5°C. The reverse is also true: if people have a temperature above 38 degrees, we list them as having a fever.

Medication, unlike complaints, is a relative standard noted field. Therefore, it makes more sense to allow clustering. For medication, all medication that a single patient got during its ER visit is grouped and put into a variable Med_list. Using technique of the complaints, we can check if the Med_list includes one of the most used medicines, so this can become an indicator on its own. Important, since the ML algorithm cannot distinguish all individual medicine in the Med_list, instead, it groups each patient with all patients that received exactly the same cocktail of medication. Thus, we separately one-hot encode most used medicine, so the algorithm can conclude that getting a specific medication administered is of importance.

Predictors The data set is split into a training (70 %) and test set (30 %). From training set, the machine learning libraries used will make samples for tuning hyper parameters and performing cross-validation: the validation sets. Each run, another part of the training set is left out for validation. The main focus is Random Forest. We use both the H2O and the scikit approach on the estimators, highlighting strengths and weaknesses for each of them. Apart from that, we use the H2O AutoML algorithm, which finds what (combination of) machine learning approaches could give the best predictive performance.

An overview of the main differences between the used approaches can be seen in Table 2. The table shows the resemblance between Random Forest and Gradient Boosting, both based on decision trees. A major advantage of using Random Forest over other techniques is that we can distinguish important features better. Therefore, we make accurate predictions while retaining information about why a certain

Name	Concept	Categorical Variable Support	Variable Importance	Missing Values
H2O Random Forest	Random Forest algorithm	Yes, using the *enum* method	Visible	Separate category
SciKit Random Forest	Random Forest algorithm	No, separate one-hot encoding needed	Visible	Replaced with averages
AutoML	Comparing common ML algorithms	Yes, method depending on algorithm	Hidden	Either of above
Gradient boost	Decision tree ensemble	Yes, using the *enum* method	Visible	Separate category
Linear Model	Logistic regression	No, one-hot encoding used	Visible	Replaced with averages
Deep learning	Neural Network	Yes	Visible	Replaced with averages

Table 2. Comparison of prediction algorithms

prediction is made. A familiar argument can be made for gradient boosted decision trees. In neural networks, this process is harder to observe due to the use of hidden/deep nodes. Another benefit is that Random Forest is often faster and more accurate then techniques like AdaBoost [9]. In H2O Random Forest, several ways of implementing categorical variables are possible [10]. Using the H2O auto setting, which equals the *enum* method, tends to give the best results. This enumeration method maps the categories to integers. It then uses ordinal or perfect group splits. This leads to each category still being a separate category. The integers tied to the categories are then irrelevant [10]. Another advantage H20's Random Forest implementation has over other techniques is the way missing values are used. The documentation [10] describes that missing values are "interpreted as containing information, i.e. missing for a reason". This seems to be a closer representation to the reality at the ER. The values missing in our data set are not missing because of software failures, but because the nurse did not see any reason to measure them.

SciKit, the standard used package, needs one-hot encoded categorical variables on beforehand. Moreover, SciKit is unable to process missing values, which means that we need to fill them with averages. In our implementation, this is slightly tweaked. For example, when filling the averages for weight, the gender is taken into consideration and the average weight of Dutch males and females according to the Central Bureau of Statistics is used.

The last predictor which is interesting to point out is AutoML. This algorithm is able to automatically select the best performing predictor for a given dataset. It includes a stacked ensemble model: this is effectively a super-learner, that takes

the predictions made by all other models as input, and uses them as independent variables to be able to make a final prediction that maximises performance on a predefined metric. In [11], it is shown that this technique worked especially well in case of imbalanced binary outcomes. In this paper, since just 30 % of the patients are hospitalized, this imbalance is present.

After training the predictors, we use Python to generate predictions on the test set. We compare these estimates with the actual outcomes to determine the performance metrics.

3.3 Improvement Assessment

For a data-driven prediction to take place, stakeholders are to be convinced of its added value. So besides accurately predicting ER outcome, we need to review if the predictions are usable in the ER environment and if there is a drive towards this direction. The presented methodology is on its own not sufficient for long term change. Leavitt [12] introduces a model that describes the factors needed for sustainable organizational change. This model, Process, People and Technology, will be used to asses the proposed changes. Each of the factors needed for successfully implementing improvements in the ER will be described.

4 RESULTS

4.1 Estimating Hospitalization

Predictor performance. In Section 3.2, we identified possible predictors. In order to determine the best performing predictors, we gathered prediction outcomes based on all data of adults, excluding 3 months of the first corona wave: $(Age > 16) \cap ((ArrivalTime < \textit{20-3-2020}) \cup (ArrivalTime > \textit{20-6-2020}))$

The resulting metrics are displayed in Table 3. We compare the overall performance in terms of the array under the ROC curve (AUC). The H2O Random Forest and the Stacked ensemble Machine Learning methods seem to have a clear advantage, so we use H2O Random Forest for the remainder of this section. Logistic regression performs the worst, expected since logistic regression has more trouble handling data that is heterogeneous and possibly redundant.

In Table 3, we also included other metrics, like the array under the precision-recall curve (AUCPR). Since a majority of the traces are not hospitalized, the data is imbalanced (64 %–36 %). Therefore, AUCPR is more informative, as it respects the imbalance in the data [13]. With AUCPR, the ranking of predictors remain the same. For all predictors, running time was not an issue.

Important variables. A better understanding of how each variable influences the final decision makes it possible to develop better predictors in the future. The ten most important variables are listed in Table 4.

	AUC	AUCPR	Accuracy	MSE	Sensitivity	Specificity
[H2O] Random Forest	0.917	0.879	0.844	0.109	0.813	0.819
[SciKit] Random Forest	0.890	0.833	0.827	0.128	0.687	0.905
[H2O] AutoML (Stacked ensemble)	0.920	0.883	0.846	0.108	0.819	0.861
[H2O] Logistic Regression	0.849	0.792	0.774	0.152	0.756	0.784
[H2O] Neural Network	0.902	0.860	0.828	0.116	0.800	0.844
[H2O] Gradient Boosting	0.909	0.862	0.836	0.113	0.807	0.852

Table 3. Metrics of predictors

	Variable	Relative Importance	Scaled Importance
0	Weight	254 932	1.00
1	Respiratory rate	107 209	0.42
2	Medication list	80 382	0.32
3	Temperature	72 944	0.29
4	Radiology_1	70 963	0.28
5	Pulse rate	59 256	0.23
6	Specialismecode	58 684	0.23
7	Age	43 451	0.17
8	Systolic	38 887	0.15
9	Diastolic	31 477	0.12
10	Natriumchloride	29 381	0.12

Table 4. List of variable importance for hospitalization

Although most factors are expected, one factor jumps out: $Weight$. This major influence might be more visible due to the way the variables are handled. Not having weight measured is also an indicator the machine learner is allowed to use. Since weight is important to know when controlling the doses for anesthesia, it might be that there is an embedded bias in which people have their weight measured: If the nurse knows that someone might need to be anesthetized, their weight is measured in advance. We can observe that when weight is excluded as an independent variable for the predictor in this data set, the accuracy is affected as well: instead of 84.1 %, the accuracy is 81.6 %. Variable importance is observed differently among ML predictors. For example, when we check the five states that contribute the most to the Neural Network predictor, we see:

1. $weight = unmeasured$,
2. $POB = True$,
3. $Specialismcode = CHI$,
4. $Medication = Null$,
5. $Specialsimcode = CAR$.

We observe the presumed bias that makes weight not being measured indicative. Besides, it uses POB (an abbreviation for pain in the chest), looks if people used

medicine during their stay, and checks if people belong to *CHI* (Surgery) or *CAR* (Cardiology). These predictors seem intuitive: a person having heart problems is likely to be monitored overnight.

Event data inclusion. Index encoded event data and the use of natural language processing decreased the error rate by 13 % compared to a model only including the basic triage information. Also, there was a difference between an AUC of 0.917 with, and an AUC of 0.887 without event data. Since we already had 80 % accuracy without the event data, all steps that extend and improve our model are important to consider for reaching an optimal accuracy. The use of event data and natural language processing is an interesting factor to improve the reliability of the prediction. Especially if the outcome of event measurements are available, they would be able to contribute even more.

Compared to related work. This section compares the performance of our approach with the performance of past research, as discussed in Section 2. The results should not be over interpreted: there is a vast difference between the used data sets. For instance, the data set reported by Raita et al. [2] contained a significantly lower hospitalization rate of 16.2 %. In this research, 36 % of the patients are hospitalized and in the other research discussed this amount is also close to 30 %. For all research, except for Sun et al. [3], the Random Forest metrics are included in Table 5.

Research	AUC	AUCPCR	Sensitivity	Specificity
Random Forest Model (H2O Default)	0.917	0.879	0.813	0.862
Riata et al. [2]	0.810	–	0.770	0.710
Kwon et al. [5]	0.738	0.557	–	–
Random Forest Model ($T = 0.7$)	0.917	0.879	0.532	0.974
Sun et al. [3]	0.849	–	0.334	0.968

Table 5. Comparison between research models

Raita et al. [2] did not specify the threshold used by the ML algorithms in their research. Since they seem to cope with issues regarding the sensitivity compared to their reference model, it is possible they did not change thresholds. We therefore used the standard measure of the maximising f1 threshold for calculating metrics. This is H2O default behaviour. Sun et al. [3] uses a classic logistic regression model. They used the threshold around 0.7 to ensure Specificity. In order to compare their results with ours, we used a similar technique. Since the amount of visits that result in hospitalization is quite similar to that of [3] (36.0 % in this research and 30.2 % in theirs), a similar threshold of 0.7 is used to provide the comparison.

Tuning for application. We can increase the prediction accuracy with two thresholds: one for the patients that should be admitted, and one for the patients that can be safely sent home. If these thresholds are not equal to each other, we are left with a third group of patients: those for which the situation is undeter-

mined. In the current application, this is perfectly acceptable: these patients walk through normal process at the ER and are admitted or sent home the usual way. Since the accuracy of the Random Forest predictor is 84.4 %, it is needed to fine-tune its behaviour for practical use. Using threshold adjustments lead to the confusion matrix displayed in Table 6. The total test data set contained 30 % or 14 640 of all patient visits. Only 7 668 traces are included in the confusion matrix, the predictor left 47.6 % undetermined when we tuned the thresholds.

	Actual Negative	Actual Positive	Error Rate
Predicted Negative	4 672	196	4.02 %
Predicted Positive	155	2 645	5.55 %
Total	–	–	4.58 %

Table 6. Confusion matrix at $T_{negative} = 0.13$ and $T_{positive} = 0.73$

This means that for a bit more than half of the patient visits, 52.4 %, we are able to predict the hospitalization with an 95.4 % accuracy. The sensitivity and specificity of this predictor are then 94.5 % and 96.0 % respectively.

COVID. We can predict hospitalization for specific complaints. In Zheng et al. [14], a ML predictor on deterioration of patients was highly effective in assessing which patients require hospitalization. Our predictor will be less accurate, since lab measurement results are not available. However, it can give an indication on how ML can help to quickly triage patients in case of an unexpected influx. In the data set of the first COVID-19 wave, 1 573 cases with comments of possible coronavirus infection are included. Unlike the regular scenario, a slight majority of the patients, 54 %, get hospitalized. The random forest model based on all patients during the wave has an AUC of 0.877, while the AUCPR is higher, at 0.898. Understandable, since there are more patients taken in, than sent home. Overall, the accuracy is slightly lower compared to when the whole data set is used.

Therefore, although the corona situation is unique, it does not seem to require a separate predictor model. When including the months of the corona wave, and including COVID-19 as a possible complaint, we can actually reach slightly better prediction results than without, which is shown in Table 7.

	AUC	AUCPR	Accuracy	MSE	Sensitivity	Specificity
RF incl. COVID	0.915	0.872	0.846	0.111	0.791	0.877
RF for children	0.953	0.863	0.917	0.061	0.778	0.952

Table 7. Random Forest model, with 1. COVID data added, 2. Children

Children. Children have their own set of acceptable measurements, with different problems than adults. Therefore, we have build a separate Random Forest model to predict their hospitalization. Of all patients with $age < 16$, 20 % will be hospitalized. The results can be seen in Table 7. Interestingly, children seem to

be even better predictable than adults. However, when inspecting the AUCPR, knowing that less children get hospitalized, we can conclude that the predictor does not perform significantly better depending on the age of the patients.

	Variable	Relative Importance	Scaled Importance
1	Systolic	97 117	1
2	Diastolic	70 541	0.726
3	Medication list	52 037	0.535
4	Respiratory rate	26 725	0.275
5	Activity_1	21 484	0.221

Table 8. List of variable importance for children

Young patients have their own set of important variables, as seen in Table 8. For them, blood pressure has more importance than it has for adults. Also, their weight is of less influence. This might implicate that the inherent bias of measuring weight, proposed in Section 4.1, is not that influential after all. It can also be due to the variance the weights of children have: a weight of 40kg is very high for a 5 year old and quite low for a 15 year old, which we did not normalize.

4.2 Estimating Admission Department

These results are based on all traces that were hospitalized. We removed the 706 patients that were moved to another hospital and 168 patients that have no direction registered. The remaining dataset contains 16 552 patients. We are not able to look at the amount of people dismissed to the IC, as some of the past research in Section 2. This information was not available. We predict for the departments where the most ER patient arrive. In total, 44 % of the patients go to the AODA department. This is understandable: the department of Acute admission and Diagnostics might be viewed as a follow up on their stay in the ER. The H2O random forest predictor is used to estimate a binomial outcome on whether patients are admitted to this department. The resulting metrics are displayed in Table 9.

	AUC	AUCPR	Accuracy	MSE	Sensitivity	Specificity
Random Forest AODA	0.852	0.801	0.747	0.156	0.853	0.673
Random Forest CCU	0.981	0.805	0.953	0.031	0.855	0.962

Table 9. Random forest model predicting AODA and CCU admissions

With threshold tuning, one would be able to predict for 7.86 % of the total amount of hospitalized patients with 90 % accuracy that they would go to the AODA department. Since this department itself is also specialized in diagnostics, this result might be useful to explore in the future. The next department with high admission rates is the CCU, the heart monitoring department. 9.0 % of the hospitalized

patients will go to this department. At first glance, results in Table 9 seem pretty good. However, this is due to very imbalanced data. When looking at the AUCPR, we can see that this predictor is likely not good enough yet. The important variables for this predictor are interesting: apart from the specialism code and the weight, two new variables show up, which are not observed that close before. POB (pain on the chest), and all closely related complaint texts, forms an important part of this predictor. This seems intuitive, as is the inclusion of the medicine Ticagrelor, which is a heart function related medicine. Both are typical for heart patients. To make the predictor better for the CCU, it is possible to include more of these typical heart related characteristics.

4.3 Process Improvement

Several ML implementations in the ER process are assessed by their impact on the workflow, their current status and the potential benefits they could bring. This is displayed for all the plans in Table 10. As proposed in Section 3.3, for each of the plans, the new process idea is introduced and technological needs are described. Lastly, feedback from an ER doctor is included.

Improvements	**Workflow**	**Current Status**	**Potential Benefits**
Indication	*No impact*	Technology – Support –	Diagnose Accuracy ++
Quick review	*Small impact*	Technology + Support +	Diagnose Accuracy + Process Time ++ Service +
Auto. forwards	*Major impact*	Technology – Support –	Process Time +++ Service ++
Event priorities	*No impact*	Technology = Support +	Processing Time +
Var. Importance	*No impact*	Technology + Support +	Diagnose Accuracy ++
Inform overstay	*No impact*	Technology= Support+	Service +

Table 10. Assessments for the proposed improvements

Indication at diagnosis. A doctor can view the indication given by the algorithm on the direction of the patient. He can use this to improve diagnosis. In our data set, 3.0 % returned to the ER within 48hrs after being sent home. Assuming this as a rough proxy estimate on the error rate of doctors, we can see how the predictor is able to help out. In future, given enough measurements and diagnostics in the past, the predictor will be able to not only predict a global direction, but also which diseases and risks the patients have. It can even estimate specific progress of disease [14]. Technology-wise, the system would show the indication in the information system the doctor inspects when looking at a patient. This

indication needs to be sufficiently accurate. If the system is not good in the predictions, the indications given are either neglected or used in a way that makes the diagnosis less accurate then without the indication. Currently, the main objective for reaching the required accuracy should be researching the inclusion of more data. Initial feedback on this proposal was that it is very difficult to implement due to a lot of grey areas. For example, someone with stomach issues can eventually fall into any of URO, GYN, CAR or CHI specialisms.

Quick review. The doctor does not always have enough time to weigh in all factors or do additional tests. Therefore, patients with relative clear indications are forwarded as soon as possible. The algorithm then helps in making the decision which patients are able to be quick reviewed: that is, which patients are very likely to be submitted to a specific hospital department or to be sent home. This can slightly improve diagnose accuracy, since we know beforehand that these patients are likely to be submitted or not. More effect is noticeable for processing time: if a doctor is assigned to quick reviewing patients, it can handle a lot of cases in a short amount of time, while still being able to dedicate time to the more complex patients. This also increases the overall capacity of the ER: in case of a sudden spike in admissions, quick reviewing helps to make a distinction between patients. Currently, it is possible to predict hospitalization with an accuracy of 95 % for half of the patients. This means that, especially if carefully looked at additional improvements, like increasing the data available to the ML predictor, technology is present. Besides the predictor performance, the system would benefit from integration with current systems. It can also perform independent, relying only on the database connection: in this case a screen with patients that are ready for quick review is placed in a strategic location. Support on this plan is present as well, the specialist found it an ideal solution. It is quite often clear when someone could be submitted early on.

Automatic forwarding. Use the indications as a strict guideline. The algorithm can be trained to have an acceptable small amount of false positives for certain directions. If the algorithm is assured of the direction of the patient, the patient can be forwarded by a nurse. Then, no more additional tests are carried out at the ER and the doctor will not review the situation of the patient, unless a patient and/or the nurse have doubts about the estimate. In that case a patient will be treated in the usual way. Theoretically, this plan would have a major impact on processing time. The main technological challenge for this plan is increase accuracy up to an acceptable level. If the predictions are improved, the implementation would not be extremely difficult, especially if the quick review mechanism is already present. However, as explained by the ER doctor, this plan is possible, but legally complex. For any decision made, someone should sign for responsibility according to the Dutch law.

Priority of events. Events, like lab tests or radiology, are executed in a specific sequence. The ones with most discriminating power can be scheduled first. By doing so, the accuracy of a diagnosis can be improved in a shorter amount

of time. Unfortunately, since results do not yet contribute to the prediction performed, this plan is harder to execute with the current data set. Apart from re-prioritising, problems in process time for highly informative events can be identified and measures to improve these specific event processing times can be taken. Since radiology seems to have the most impact on a patient's chance to overstay, one might consider starting here. For example, if there are scheduling problems at the x-ray, preregistering x-ray scans might help. Preregistration is already present, but an improvement of existing systems can be done. This makes this plan technically realizable in a short time. However, it should be noted that for major improvements, new variables need to be introduced to the model. The feedback on this proposal was that for the care paths a patient takes, some requests for lab research and radiology are already made in advance. Improvement of this process is very welcome.

Variable importance. The weight the Random Forest gives to a variable might uncover new diagnosis paths. Patients likely to be submitted to a certain direction possess specific characteristics. This can be used to improve diagnosis, by combining the weights with medical expertise. As data is better registered, and more patient traces are available, the ML predictor will be able to better distinguish which properties make a certain disease and its progress over time unique. This is already technically possible, either as part of a user interface or made into a report. It is again important to involve more data for this plan to yield beneficial results. The ER staff found it likely to be most usable in case of a clear diagnosis. For example, making this photo always results in hospitalization.

5 DISCUSSION

Researchers are encouraged to run this analysis on larger datasets: right now 48 000 cases are used to create the model. Increasing this number, for example by including data from the last 5 years, is projected to improve model prediction quality. Besides increasing the quantity of the data entries, increasing the quality is highly effective as well. For example, a research by [14] was effective on predicting which COVID-19 patients were prone to heavy deterioration and need hospitalization the most. To make these predictions, it included results of lab measurements, which were not available for us.

This research did not have access to ethnic and demographic information, which was a major factor in predicting hospitalization in other research [2, 3]. For outcome, a classifier indicating critical care was unavailable. Most of the research discussed in Section 2 got very interesting results predicting critical care as well. With regards to the support for ML techniques in the ER , this research did not measure support under patients. It could be that patients value the increased service, but perceive predictions about them made by computers as scary.

This research, and the related work, does not incorporate findings on which treatments are only acute care. For example, some patients have wounds that need

acute treatment to stop bleeding and infections, but are after this treatment at the ER safe to go home. The necessity and the frequency of these activities is hard to determine with the data set provided. For assessing the impact of implementing prediction technology in the ER in future, it is instrumental to keep this in mind.

Since all activities were performed within one hour after arrival according to the dataset, this research did not include a notice about the accuracy at different times in the ER process (e.g. at arrival, after X hours). It should be examined if the used information is available at the time predictions are made.

6 CONCLUSION

This research aimed at applying machine learning techniques to the ER process, to improve the diagnosis and decrease the time patients spent at the ER. We found that we are able to predict the hospitalization for more than half of all ER visits with an accuracy of 95 %. This leads us to conclude that it is definitely possible to predict the direction of a patient after ER using the available data. As seen in Section 3, this conclusion is likely applicable to all hospitals in the Netherlands. Using the Random Forest algorithm, we can identify the importance of variables for hospitalization. Based on which variable to predict, these indicators change. For example, when looking at the time patients spent at the ER, the amount of other patients at the ER becomes an important variable, both intuitively and in practice. We saw that event data improved the accuracy of the predictions.

With regards to which technique to use, Stacked ensemble and a specific Random Forest package yielded the best results out of the six techniques inspected. Lastly, we pivoted five ideas to improve the ER process using Machine Learning. Some, like the quick reviewing patients, seem very promising and will be investigated in follow-up research.

REFERENCES

[1] Panneman, M.—Blatter, B.: Letsel Informatie Systeem – Representatief voor Alle SEH's in Nederland? Technical Report No. 627, VeiligheidNL, Amsterdam, 2016 (in Dutch).

[2] Raita, Y.—Goto, T.—Faridi, M. K.—Brown, D. F. M.—Camargo, C. A.—Hasegawa, K.: Emergency Department Triage Prediction of Clinical Outcomes Using Machine Learning Models. Critical Care, Vol. 23, 2019, Art. No. 64, doi: 10.1186/s13054-019-2351-7.

[3] Sun, Y.—Heng, B. H.—Tay, S. Y.—Seow, E.: Predicting Hospital Admissions at Emergency Department Triage Using Routine Administrative Data. Academic Emergency Medicine, Vol. 18, 2011, No. 8, pp. 844–850, doi: 10.1111/j.1553-2712.2011.01125.x.

[4] Choi, S. W.—Ko, T.—Hong, K. J.—Kim, K. H.: Machine Learning-Based Prediction of Korean Triage and Acuity Scale Level in Emergency Department Pa-

tients. Healthcare Informatics Research, Vol. 25, 2019, No. 4, pp. 305–312, doi: 10.4258/hir.2019.25.4.305.

[5] Kwon, J. M.—Lee, Y.—Lee, Y.—Lee, S.—Park, H.—Park, J.: Validation of Deep-Learning-Based Triage and Acuity Score Using a Large National Dataset. PLoS ONE, Vol. 13, 2018, No. 10, Art. No. e0205836, doi: 10.1371/journal.pone.0205836.

[6] NVT: Levelcriteria NVT 2020–2024. Nederlandse Vereniging voor Traumachirurgie, 2020, https://www.trauma.nl/levelcriteria-nvt-2020-2024 (in Dutch).

[7] RIVM: Acute Zorg. RIVM, 2016, https://www.volksgezondheidenzorg.info/onderwerp/acute-zorg/cijfers-context/gebruik-acute-zorg (in Dutch).

[8] Leontjeva, A.—Conforti, R.—Di Francescomarino, C.—Dumas, M.—Maggi, F.: Complex Symbolic Sequence Encodings for Predictive Monitoring of Business Processes. In: Motahari-Nezhad, H., Recker, J., Weidlich, M. (Eds.): Business Process Management (BPM 2016). Springer, Cham, Lecture Notes in Computer Science, Vol. 9253, 2015, pp. 297–313, doi: 10.1007/978-3-319-23063-4_21.

[9] Breiman, L.: Random Forests. Machine Learning, Vol. 45, 2001, No. 1, pp. 5–32, doi: 10.1023/A:1010933404324.

[10] H2O.ai: H2O Documentation. 2017, http://docs.h2o.ai/.

[11] LeDell, E.: Scalable Ensemble Learning and Computationally Efficient Variance Estimation. Ph.D. Thesis. UC Berkeley, 2015.

[12] Leavitt, H.—March, J.: Applied Organizational Change in Industry: Structural, Technological and Humanistic Approaches. Carnegie Institute of Technology, Graduate School of Industrial Administration, 1962.

[13] Saito, T.—Rehmsmeier, M.: The Precision-Recall Plot Is More Informative Than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. PLoS ONE, Vol. 10, 2015, No. 3, Art. No. e0118432, doi: 10.1371/journal.pone.0118432.

[14] Zheng, Y.—Zhu, Y.—Ji, M.—Wang, R.—Liu, X.—Zhang, M.—Liu, J.—Zhang, X.—Qin, C. H.—Fang, L.—Ma, S.: A Learning-Based Model to Evaluate Hospitalization Priority in COVID-19 Pandemics. Patterns, Vol. 1, 2020, No. 6, Art. No. 100092, doi: 10.1016/j.patter.2020.100092.

Roeland A. J. J. van Delft is an enthusiastic data scientist with a passion for new and creative applications of machine learning to increase business process efficiency. He graduated with the Master degrees in business information systems from the Eindhoven University of Technology, and in finance from Tilburg University.

Renata M. de Carvalho is Assistant Professor (Universitair Docent) at the Eindhoven University of Technology (TU/e) since 2016. She received the B.Sc. and M.Sc. degrees in computer engineering from the University of Pernambuco (UPE), Brazil, in 2007 and 2010, respectively, and the Ph.D. degree in computer science from the Center of Informatics (CIn), Federal University of Pernambuco (UFPE), Brazil, in 2015. She was Postdoctoral Researcher with the Laboratory for Research on Technology for Ecommerce (LATECE), University of Quebec at Montreal (UQAM), for more than one year. Her research interests are focused on business process management, discover and improvement. She focuses the work on the domain of flexible/dynamic/adaptable business processes, and how it can enrich business process models with domain-specific knowledge. Currently, her research is focused mainly on the domain of healthcare.