Experimental Validation and Development of Decision Tree - based System for Prediction of Service Management of Perfusors / Syringe Pump

Becir Isakovic [1], Zerina Masetic [1], Jasmin Kevric [1], Lejla Gurbeta [1,2], Enis Gegic [1]

[1] *International Burch University, Faculty of Engineering and Natural Sciences, 71000, Sarajevo, Bosnia and Herzegovina*
[2] *Medical Device Inspection Laboratory Verlab Ltd., 71000, Sarajevo, Bosnia and Herzegovina*

Abstract **– Despite the fact that technology is improving day by day and that the medical devices (MDs) are being constantly upgraded, their malfunction is not a rare occurrence. The aim of this research is to develop an expert system that can predict whether the device will satisfy functional and safety requirements during a regular inspection. This expert system can be seen as part of Industry 4.0 that is revolutionizing medical device management. In order to develop the system, five machine learning algorithms that are representative of each classifier group, were used: (1) Random Forest, (2) Decision Tree, (3) Support Vector Machine, (4) Naive Bayes, (5) k-Nearest Neighbour. The Decision Tree outperformed other classifiers achieving the classification accuracy of 100% with and without attribute selection applied on the dataset.**

This study showed that machine learning algorithms can be used in order to predict MDs performance and potential failures in order to make the process of maintenance of medical devices more convenient and sophisticated and it is one step in modernizing medical device management systems by utilizing artificial intelligence.

Keywords **– decision Tree, perfusor, medical device, maintenance, Industry 4.0., artificial intelligence**

DOI: 10.18421/TEM113-33
https://doi.org/10.18421/TEM113-33

Corresponding author: Becir Isakovic,
International Burch University, Faculty of Engineering and Natural Sciences, 71000, Sarajevo, Bosnia and Herzegovina.
Email: becir.isakovic@ibu.edu.ba

Received: 06 June 2022.
Revised: 04 August 2022.
Accepted: 10 August 2022.
Published: 29 August 2022.

1. Introduction

Medical devices (MDs) are one of the essential factors in modern medicine. Over the years these devices were improved and developed in order to perform tasks in a highly accurate and timely manner. Today, medical staff can perform their tasks focusing on the problem itself and not bother with the device being hard to use or out of service for most of the time. Although there are many directives, regulations, and international standards that are required to be followed by manufacturers of MDs [1], [2], their malfunction is a huge problem that can lead to wrong diagnosis and misguided patient treatment. There are many cases of patient injuries and some of the injuries end up lethally. One of the most popular databases containing reports about malfunction of medical devices is the Food and Drug Administration (FDA) Manufacturer and User Facility Device Experience (MAUDE) [3] database for the United States and European database on medical devices (EUDAMED) [4].

Even though most MDs have built mechanisms that can detect the potential failure of a device, the medical staff often fails to recognize the malfunction of the device, which results in wrong patient diagnosis, hence misguided patient treatment. The number of these unfortunate events is rising, and it suggests that the procedures for surveillance and

maintenance of MDs after they have been delivered to institutions are not efficient. For example, if a glucose meter (device for measuring the level of sugar in the blood) is reporting measurements that are not correct, the patient can be treated with higher or lower insulin dosage. In the same fashion if perfusor syringe is broken or if the flow is not as set by the medical staff, a higher or lower dose of the drug will be given to the patient that can lead to patient injury (overdose, cardiac arrest...) or even to patient death [5], [6]. Although post market surveillance is obligatory for all MDs, its implementation varies from country to country [7]. For example, in Bosnia and Herzegovina, this process is defined and implemented by following the established medical devices framework [8] according to which all perfusors are regularly tested by accredited laboratories according to ISO 17020 standard. Information about flow, visual inspections, and other necessary information is stored in a developed database.

Utilizing machine learning techniques in order to improve the process of maintenance and monitoring of medical devices is becoming more and more popular. Badnjevic et al. [9] developed an automated system in order to detect the potential failure of the defibrillator. They have used five different machine learning algorithms: (1) Decision Tree, (2) Random Forest, (3) k-Nearest Neighbour, (4) Support Vector Machine, (5) Naive Bayes on a dataset with all attributes included and with applied attribute selection on a dataset. They were able to develop a system that can detect hardware deviation on the defibrillator with an average accuracy of approximately ≈98.5%. Hrvat et al. [10] have used the Artificial Neural Network (ANN) machine learning algorithm in order to predict the performance of syringe pumps (infusomats and perfusor). After testing different ANN architectures, the one with one hidden layer with ten neurons had the best accuracy with 98.06% for perfusor pumps and 98.83% accuracy for infusion pumps. An accuracy of 98.41% was obtained on both syringe pumps with a recurrent neural network. Hadzic et al. [11] have used ANN and FL as well in their research where they proposed a system for performance prediction of Anesthesia Machine. They developed a two-layer feed-forward backpropagation neural network with 23 neurons in a hidden layer. With 197 data samples for both training and testing, they obtained an accuracy of 97.44%. According to related literature, it is noticeable that the prediction of MDs failure has not been researched much and that utilizing machine learning algorithms and techniques in surveillance of MDs is a novel approach in tackling this problem.

The objective of this paper is to develop an expert system to predict, based on collected data, the performance of the perfusor medical device and potential failures in order to enhance current strategies for the management of medical devices. Moreover, another objective is to lay the foundation for increasing safety standards and performance of medical devices during their usage in healthcare institutions through the improvement of the way of work of clinical engineering in the field of post-market surveillance of medical devices.

This work will serve as a basis for future work where we plan to create an algorithm that will predict, based on historical data, whether the device will pass the next regular measurement. The data that was used for this experiment will be rearranged in order to develop the expert system that will be able to notify healthcare institutions that the medical device may not pass the test prior to the test itself.

2. Materials and Methods

For the purpose of the development of this expert system for the performance prediction of perfusors, five different machine learning algorithms were used. These algorithms were selected as they are the representatives of each classifier group and they were proven to work well and used in various research papers [9], [12], [13]. Those algorithms are: (1) Random Forest, (2) Decision Tree, (3) Support Vector Machine, (4) Naive Bayes, (5) k-Nearest Neighbour. A more detailed description of these algorithms is given in the further text.

The expert system for the prediction of perfusion performance was developed using two different approaches. In the first approach, all attributes are given to the algorithms without previous analysis of the attributes. In the second approach, attributes were filtered by using attribute selection algorithms. Algorithms that were used are as follows: (1) InfoGain, (2) CfsSubsetEval, (3) WrapperSubsetEval. This approach was used in order to get information about the impact of the attribute on the classifier performance. In this way, the system was optimized to get the same or even better results by using fewer attributes and less time.

These algorithms were selected because they are representative of two groups of attribute selection algorithms and CfsSubsetEval is used as a correlation-based attribute selection algorithm. Information Gain is representative of filter methods. Filter methods select an attribute by calculating the relevance score where attributes with the lowest score are removed. The advantages of these methods are a) computationally fast and simple, b) handy with high-dimensional datasets, c) performance is independent of the algorithm used.

WrapperSubsetEval is representative of wrapper methods that embeds the attribute subset search with the model hypothesis search. A various subset of attributes is obtained from the search in space of all possible attributes, where the evaluation of the attribute subset is obtained by training and testing a particular model for a specific classification algorithm. The drawback of the wrapper method is that there is a higher risk of overfitting than by using the filter technique and the overall method is computationally intensive. The advantage of the method is that the attribute subset is interacting with the classification algorithm in order to build the best possible model [14]. The whole workflow is shown in

Figure 1.

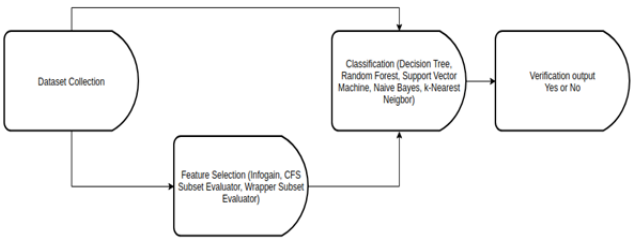

Figure 1. Workflow of proposed system

2.1. Data Collection

The system development was done based on 1168 inspection reports that were provided to this research group by an appointed inspection body accredited by ISO 17020. This inspection body conducts yearly inspections of medical devices, including perfusors / syringe pumps in healthcare institutions in Bosnia and Herzegovina according to the Legal metrology framework that was adopted in 2014 [8].

The reports used for system development contained perfusor performance measurements collected during the 2015-2019 period. The 1168 inspection reports correspond to 574 devices inspected during this period. This means that in the dataset, one device can have up to 5 records (the same device was examined every year) or it can be just seen once. All performance measurements were taken using the Fluke Biomedical IDA 5 analyser [15].

The abovementioned inspection records were used to construct a database. Each sample in this database consists of 39 input attributes and an output class. Attributes are grouped into eight categories which are as follows: (1) device manufacturer, (2) type of device, (3) visual inspection of device cleanliness, (4) visual inspection of device completeness, functionality, and prescribed casing, (5) visual inspection of inscriptions and markings, (6) visual inspection of power chords and accessories, (7) measured flow value of the device, and (8) inspection decision (conformity assessment) as output class. Attributes (1) - (6) were obtained by visual inspection of the device conducted by the technician and attribute (7) was measured by the abovementioned etalon IDA 5. This etalon was calibrated according to the ISO 17025 to ensure measurement traceability.

Flow safety inspection was performed in six measurement points and the following attributes were examined: (1) expected value (in ml), (2) measured value (in ml), (3) relative measurement error (in percentage), (4) allowed deviation (in percentage). As it can be seen, both visual inspection and measurement are assessed by defined criteria which are stated in the Rulebook on metrological and technical requirements for infusion and perfusor pumps, *Official Gazette of Bosnia and Herzegovina*, No. 75/14. Attribute (8) inspection decision (conformity assessment) is set to YES or NO based on this criteria, whether or not the legal requirements on device performance are met. All attributes of the dataset are listed and explained in Table 1.

For the purpose of constructing an expert system, the dataset was split into two subsets: training and testing. Because of the fact that the dataset was created incrementally over the years, instead of having the random split of our data that can lead to a skewed class problem, the dataset was divided by the year of inspection in the following manner: (1) TRAINING - measurements taken in 2015, 2016, and 2017 and (2) TESTING - measurements taken in 2018 and 2019. As a result, our training set contains 608 samples whereas the test set has 560 samples. Instead of the traditional 80-20 train-test split ratio, we used an approximately 50-50 split ratio that makes the model itself more convenient and realistic. The positive and negative classes were also evenly distributed in both training and test sets: the training set contains 553 positive and 55 negative samples whereas the test set contains 508 positive and 52 negative samples.

Table 1. Attributes description

Category	Descriptor Name	Descriptor Type	Values	Range
Device Manufacturer	Manufacturer	Categorical		
	Year	Discrete		
Type of Device	Type	Categorical		
Visual Inspection of Device Completeness, Functionality, and Prescribed Casing	Visual inspection 1 - Clean device?	Categorical	YES / NO	
	Visual inspection 2 - complete, functional, with prescribed casing	Categorical	YES / NO	
Visual Inspection of Inscriptions and Markings	Visual inspection 3 - inscriptions and markings	Categorical	YES / NO	
Visual Inspection of Power Chords and Accessories	Visual inspection 4 - power cords and other accessories necessary for normal operation	Categorical	YES / NO	
Measured Flow Value of Device (There were six measurements conducted and available in the dataset so there are 1-6 sets of these attributes)	Parameter of inspection	Categorical	FLOW	
	1: Set value [ml]	Continuous		0 - 100
	1: Measured value [ml]	Continuous		0 - 100
	1: Error [%]	Continuous		0 - 100
	1: Allowed deviation [± %]	Categorical		2 - 20
	1: Pass measurement	Categorical	YES / NO	
Inspection Decision (Conformity Assessment)	Conformity Assessment FINAL	Categorical	YES / NO	

2.2. Attribute Selection Algorithms

Having a lot of attributes in the dataset may lead to a very complex model that can have lower accuracy and a longer time to be trained. In order to simplify the model, increase the accuracy and training time, attributes that are irrelevant or redundant should be removed. Attribute selection is a method of selecting a subset of relevant attributes that will be used in model creation. As is representative of different types of attribute selection algorithms three attribute selection methods were used: 1) Info Gain, 2) Cfs Subset Evaluator, 3) Wrapper Subset Evaluator. These algorithms were used as well in previous studies [9], [16].

The Information Gain algorithm has been introduced by J.R. Quinlan [17] and it is based on the decision tree. Calculating the Information Gain (IG) helps in picking the attribute that will split at node N, where the attribute of N with the maximum IG is the one chosen for splitting. The attribute with the highest IG needs the least information in order to classify the objects in the partitions. The calculation formula is given as:

$$InfoGain(Class, Attribute) = H(Class) - H(Class|Attribute)$$

The InfoGain (Class, Attribute) measures the reduction in the entropy of a given class variable after the value for the given attribute is observed. Info Gain was used with the Ranker search technique that ranks attributes by their individual performance [18].

Cfs Subset evaluator determines the gain on a subset of attributes by analyzing their individual predictive ability and the redundancy degree between attributes. It was introduced in 1998 by Mark A. Hall [19]. The main idea is that the subset of attributes that have low intercorrelation while being highly correlated with the class is favored. This attribute selection algorithm has been used together with a Genetic search. The genetic search algorithm belongs to a larger class of evolutionary algorithms that are widely used for search problems and optimization.

The wrapper Subset evaluator is using the learning schema in order to evaluate the attribute set. Accuracy is determined by using cross-validation on a set of attributes of the learning schema [20]. This attribute selection algorithm has been used together with BestFirst search which is a graph search algorithm where expansion nodes are selected based on the evaluation function [21].

2.3. Classification Algorithms

In order to develop the expert system, a variety of machine learning algorithms and techniques were examined. Based on the previous efforts that aimed towards applying machine learning techniques and algorithms in building models in biomedicine, this paper presents the results of machine learning algorithms in predicting the performance of perfusors in order to optimize the management and predict potential failures of medical devices in healthcare institutions.

2.3.1. Random Forest (RF) algorithm

Random forest is a machine learning algorithm that combines several decision trees, and it belongs to the category of ensemble classifiers. The main characteristic of this algorithm is that randomness is provided to initial components in order to create the tree. The tree consists of multiple randomized decision trees where each tree is taking a random subset of attributes. As a result, the majority voting technique is performed in order to determine the output class [22]. The intuition has been captured in Figure 2.

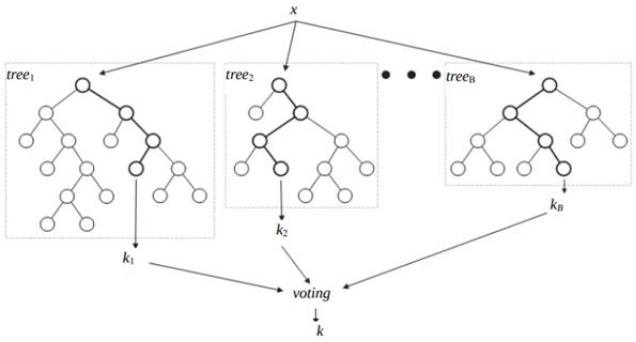

Figure 2. General structure of Decision Tree algorithm

In our algorithm, we have used 100 iterations in order to build our random forest model. The selection was made, considering the accuracy as the fitting function.

2.3.2. k-Nearest Neighbor (k-NN) algorithm

The k-NN algorithm tries to classify every sample by calculating the distance between that sample and its k-nearest neighbors. The class of the sample is then determined by using a majority voting algorithm. The performance of the algorithm heavily relies on the distance metric that is used to identify the nearest neighbor for that sample. In most cases, the distance is calculated by using the Euclidean metric that shows the similarity between neighbors that are given as vector inputs. Euclidean distance is represented by the following formula:

$$d(x_i, x_j) = \sqrt{\sum_{i=1}^{n} (x_{1i} - x_{2i})^2}$$

where our sample is represented as a vector x = (a1, a2,...an), n represents the number of sample attributes. ar is the rth attribute of the sample, wr is the weight of rth sample and r can take values from 1 to n. The smallest value of d(xi,yj) is the two examples that are most similar [23].

The final class is determined by the majority vote of sample k nearest neighbours, and it is defined by the following formula:

$$y(d_i) = \arg max_k \sum_{x_j \epsilon kNN} y(x_j, c_k)$$

where d_i is the sample, $y(x_j, c_k)$ shows whether the xj belongs to a class ck, and xj is one of sample k nearest neighbors in the set. To simplify what the equation does, let us run a short example. Let k = 7 (7-nearest neighbors algorithm) and let's say that 4 out of 7 neighbors say that it belongs to class A and three neighbors say that the sample belongs to class B. By using the majority voting the sample will be labeled as class A. This example is shown in Figure 3.

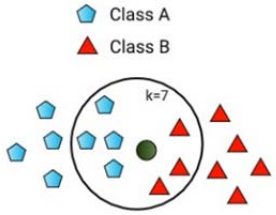

Figure 3. Representation of kNN algorithm

What is the optimal value of k in order to optimize the classifier for the highest possible accuracy can be determined experimentally. Initially, k is set to 1 and then the error rate of the classifier is observed. The k is then incrementally increased and the value of k for which error is lowest is chosen. In most cases, the value of k will be bigger for larger datasets. In our experiment, we tried odd values of k (1, 3, 5, 7, 9) and the accuracy remains the same for k being 1, 3, 5, and 7, and dropped when k was 9. Therefore, the selection was made to have k = 7. We have used LinearNNSearch as a search algorithm combined with Euclidean Distance distance function and batch size was set to 100.

2.3.3. Naive Bayes (NB) algorithm

Naive Bayes (NB) belongs to the category of so-called probabilistic classifiers that is based on the Bayes theorem. The Bayesian theorem defines the probability based on the previous knowledge of circumstances that can be related to that event.

According to Bayes theorem, the probability is given as:

$$P(h|D) = \frac{P(D_h) * P(h)}{P(D)}$$

where P(h) is called prior probability and it represents the independent probability of the hypothesis h, P(d) is the independent probability of data D, P(D|h) is the conditional probability of data D given the hypothesis h and similarly, P(h|D) is the conditional probability of the hypothesis h given data D. The bayesian theorem assumes that attributes are independent and the number of parameters that are used for building a model is reduced. Because of that fact, the Naive Bayes is more convenient for large datasets because it doesn't involve any complicated parameter estimations and at the same time very sophisticated and complex classifiers can be built [24].

2.3.4. Decision Tree (DT) algorithm

A decision tree (DT) is a tree where nodes represent attributes and every branch is a decision point. If the node breaks the chain of the decision tree then that node is called a leaf node. One of the main challenges of the decision tree is to determine the root node at each level and there are multiple attribute selection methods (Information gain, Gini index...) that perform this task. In a nutshell, building the decision tree is mostly about finding the attribute that will result in the highest information gain. Our DT used a 0.25 confidence factor with 3 folds pruning. The algorithm is partitioning data into smaller subsets and building the decision tree incrementally [25]. In Figure 4. the visualization of the built decision tree has been shown.

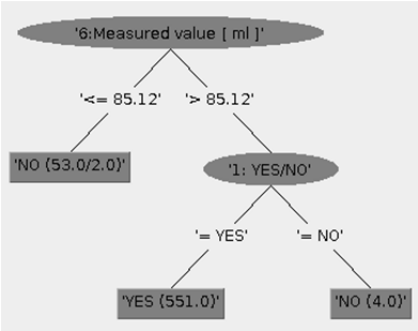

Figure 4. Decision tree architecture for given dataset

As shown in
Figure 4., the attribute on which branching happened is the attribute that represents the sixth measure of flow. This suggests that this attribute is the most important attribute when constructing the decision tree. After this, the next important attribute is the first conformity assessment and that is the last attribute where branching is needed. The next two nodes are leaf nodes in the decision tree where the final assessment (output class prediction) happened.

As the decision tree classifier is subject to overfitting of the training data, the pruning technique is applied to the learned model in order to remove any information that was acquired during the training process that can lead to the problem of overfitting.

2.3.5. Support Vector Machine (SVM) algorithm

Support Vector Machine (SVM) is a supervised machine learning algorithm and is widely used for regression and classification problems. The main objective of SVM is to find the optimal hyperplane in an N-dimensional space (where N is the number of attributes) that distinctly separates classes. The input samples for the SVM classifier are represented as vectors. It is obvious that there are many hyperplanes that can separate the classes but SVM tries to find the hyperplane with the largest possible margin (maximum distance between both classes) and that is why SVM is also called a large margin classifier. In this way, the next data points will be classified with more confidence. By using the One-vsAll approach, SVM can be used for multiclass classification problems where its primary purpose is to support binary classification and our dataset perfectly suits this algorithm [26], [27], [28]. Our SVM used a Polynomial Kernel with normalization where the C parameter was 1 in order to capture the intuition of SVM being a large margin classifier. The intuition behind the SVM classifier is captured in Figure 5. [29].

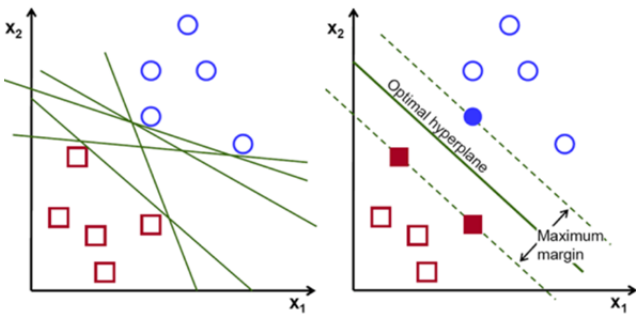

Figure 5. SVM classifier representation

3. Results and Discussion

3.1. Performance Evaluation Criteria

In order to evaluate the performance of machine learning algorithms, the whole dataset was divided into two subsets: training subset and testing subset. The training dataset was used to train data and the testing dataset was used for the evaluation of optimized classifiers. Testing and training datasets were chosen independently one from another and the split ratio was approximately 50-50 for both training

and testing datasets in order to have more realistic results. The positive and negative classes were also evenly distributed in order to have accurate results.

In order to evaluate algorithms that were used, four different criteria were taken into consideration: (1) accuracy, (2) precision, (3) receiver operating characteristic (ROC) curve, (4) F - measure. As being one of the most important criteria in evaluating the performance of algorithm classification, accuracy was taken as an important indicator and comparison criteria between classifiers. In order to have realistic accuracy, two new terms must be considered: sensitivity and specificity [18]. Specificity (also known as recall) defines the ability of the classifier to recognize positive data samples and it is defined by:

$$Sensitivity = \frac{TP}{TP + FN} * 100$$

where TP is the number of true positive data samples and FN is the number of false-negative samples. In this research, sensitivity defines the number of perfusors that successfully passed measurement and have positive test results. On the other hand, the specificity determines classifier performance in recognizing negative data samples and it is defined as:

$$Specificity = \frac{TN}{TN + FP} * 100$$

where TN is the number of true negative samples and FP represents the number of false-positive samples. Specificity in this research represents the number of perfusors that failed to pass the measurement test and that have negative test results. The accuracy is then calculated as follows:

$$Accuracy = \frac{Sensitivity + Specificity}{2}$$

Another important performance evaluation criteria is the ROC curve. The ROC curve is created by plotting the sensitivity on the y-axis, which is the percentage of the total number of positive samples, and specificity on the x-axis which represents the percentage of the total number of negative samples. Ideally, the ROC curve should be (0, 1) which means that all data samples with positive class are classified as positive and all data samples with negative class are classified as negative. Performance of algorithms is measured by calculating the mean area under the curve (AUC) where a better classifier model will have a bigger area [30].

The fourth measure that was used is the F - measure. F - measure summarizes the performance of the model and it combines both precision and specificity in a single measurement and it is the most popular metric used to identify potential imbalanced classification problems. F - measure is calculated as follows:

$$F - measure = \frac{2 * Precision * Specificity}{Precision + Specificity}$$

3.2. Experimental Results

Table 2. shows the performance of 5 different machine learning algorithms on the dataset that contains all attributes. As shown in Table 2. all algorithms performed very well with an accuracy above 99%. The best performing algorithms were DT and Random Forest which correctly classified all data samples. The second-best performing algorithm was Naive Bayes with misclassifying the only one instance of the negative class. The k-NN and SVM obtained the lowest accuracy 99.11%. Both of these algorithms misclassified 5 of the instances of the negative class.

The results of the performance of algorithms using Info Gain attribute selection with Ranker search are presented in Table 3. The DT performed best with just 2 input attributes selected. The k-NN also performed better with fewer attributes (8 attributes selected by Info Gain) than previously with just one misclassified negative class sample. With 19 attributes, the SVM performance stayed the same. The NB classified had the same precision with all attributes with the attribute selection applied. The selected attributes (3 attributes selected) were the same for RF and NB.

With the Cfs Subset evaluator as attribute selection algorithm with Genetic search, the performance of the DT classifier dropped to 99.46 with 3 positive class samples being not classified as expected. The RF precision was also lower than in previous cases with 3 positive class and 6 negative class data samples being misclassified. The k-NN with k being 10 performance increased than in previous cases with all samples being correctly classified. The SVM and NB performed similarly with the Info Gain algorithm where SVM misclassified 5 negative class samples and NB only one. The Cfs Subset evaluator took 8 attributes for all machine learning algorithms as shown in Table 4.

The Wrapper Subset evaluator used with BestFirst search algorithm performance is presented in Table 5. The RF performance was increased with all samples being correctly classified with 3 attributes selected by the algorithm. With just 2 attributes selected, the kNN with 1 nearest neighbor had 100% accuracy. The SVM performed the same as in the previous case but now with just 3 attributes selected by the algorithm. The NB had 100% accuracy with 3 attributes selected and DT had the same accuracy with selecting 4 attributes.

The comparison of the performance of all machine learning algorithms has been summarized in Figure 6. As shown, DT had the best performance with all attributes and with Info Gain attribute selection applied with Ranker search with 100% accuracy. The accuracy dropped slightly with the Cfs Subset evaluator to 99.46 misclassifying 3 instances of the positive class. The Wrapper Subset evaluator chose four attributes and acquired 100% accuracy.

The Random Forest algorithm had 100% accuracy with Info Gain and Wrapper Subset attribute selection algorithm. With all attributes, it misclassified only one negative class instance and had 99.82% accuracy. With the Cfs Subset evaluator, RF ends up with 98.39% accuracy with having the wrong classification for 3 positive class and 6 negative class samples. The algorithm was used with 100 trees as a parameter.

For all attributes case, k-NN was used with k being 7 and that gave the best accuracy of 99.47 with misclassifying 5 samples of negative class. With modifying k to be 10 the accuracy increased for the Information Gain attribute selection algorithm to 99.82 with just one negative class instance being misclassified. With the same setup, the Cfs Subset and Wrapper Subset algorithm had 100% accuracy.

The SVM performed exactly the same with all attribute selection algorithms and also on all attributes with misclassifying 5 of negative class instances and 99.11% accuracy. With the Info Gain algorithm, the SVM took 19 attributes, eight attributes were selected with the Cfs Subset evaluator and just 3 attributes with the Wrapper Subset algorithm.

With correctly classifying all instances, NB had 100% accuracy with the Wrapper Subset attribute selection algorithm that picked 3 attributes. The Info Gain algorithm also selected 3 attributes but had lower accuracy of 99.82 with incorrectly classifying just one negative class sample. The accuracy was the same for all attributes and for the Info Gain evaluator that took 8 attributes.

The attributes that were selected by attribute selection algorithms were also examined. Among these attributes, in almost all cases attribute 12 (1st Measured value [ml]), 22 (3rd Measured value [ml]), and 37 (6th Measured value [ml]) were selected by the Info Gain algorithm where these attributes were ranked as being most relevant for the classifier. The Cfs Subset evaluator took 8 (7, 9, 12, 17, 20, 27, 32, 37) attributes were five of them were measurements taken, 2nd and 4th visual inspection, and 2nd measurement assessment. The Wrapper Subset evaluator took into consideration different subsets of attributes depending on the classifier that was used. In most cases, these attributes were: 1st allowed deviation (attribute number 14), 6th measurement value (attribute number 37), and 2nd and 4th visual inspection (attributes 7 and 9). The attributes that were selected by attribute selection algorithms vary depending on the classifier and search method that was used. However, we can see that both Info Gain and Cfs Subset evaluator took some same attributes.

Table 2. Performance of algorithms on the dataset with all attributes

Classifier	Accuracy	ROC Area	F-Measure	True Positive (TP)			False Positive (FP)			Precision		
				Pass	Fail	Average	Pass	Fail	Average	Pass	Fail	Average
DT	100	1	1	100	100	100	0	0	0	100	100	100
RF	100	1	1	100	100	100	0	0	0	100	100	100
k-NN	99.11	0.970	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1
SVM	99.11	0.952	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1
NB	99.82	1	0.998	100	98.1	99.8	1.9	0	1.7	99.8	100	99.8

Table 3. Performance of algorithms on attributes selected by Info Gain algorithm with Ranker search

Classifier	Accuracy	ROC Area	F-Measure	True Positive (TP)			False Positive (FP)			Precision			Number of attributes
				Pass	Fail	Average	Pass	Fail	Average	Pass	Fail	Average	
DT	100	1	1	100	100	100	0	0	0	100	100	100	2
RF	100	1	1	100	100	100	0	0	0	100	100	100	3
k-NN	99.46	0.995	0.995	99.4	100	99.5	0	0.6	0.1	100	94.5	99.5	8
SVM	99.11	0.952	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1	19
NB	99.82	1	0.998	100	98.1	99.8	1.9	0	1.7	99.8	100	99.8	3

Table 4. Performance of algorithms on attribute selected by CfsSubsetEval algorithm with Genetic search

Classifier	Accuracy	ROC Area	F-Measure	True Positive (TP)			False Positive (FP)			Precision			Number of attributes
				Pass	Fail	Average	Pass	Fail	Average	Pass	Fail	Average	
DT	100	1	1	100	100	100	0	0	0	100	100	100	8
RF	99.11	0.999	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1	8
k-NN	99.11	0.961	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1	8
SVM	99.29	0.987	0.993	99.4	98.1	99.3	1.9	0.6	1.8	99.8	94.4	99.3	8
NB	99.82	1	0.998	100	98.1	99.8	1.9	0	1.7	99.8	100	99.8	8

Table 5. Performance of algorithms on attribute selected by Wrapper Subset algorithm with Best First search

Classifier	Accuracy	ROC Area	F-Measure	True Positive (TP)			False Positive (FP)			Precision			Number of attributes
				Pass	Fail	Average	Pass	Fail	Average	Pass	Fail	Average	
DT	100	1	1	100	100	100	0	0	0	100	100	100	4
RF	100	1	1	100	100	100	0	0	0	100	100	100	3
k-NN	100	1	1	100	100	100	0	0	0	100	100	100	2
SVM	99.11	0.952	0.991	100	90.4	99.1	9.6	0	8.7	99	100	99.1	3
NB	100	1	1	100	100	100	0	0	0	100	100	100	3

The proposed method is aimed to help predict inspection results for medical devices. The precision of measurements that are taken has to be taken into consideration because algorithm output heavily relies on it. As this is the area where any mistake can endanger human lives, healthcare institutions and professionals rely on artificial intelligence and machine learning techniques in order to perform their tasks with more confidence.

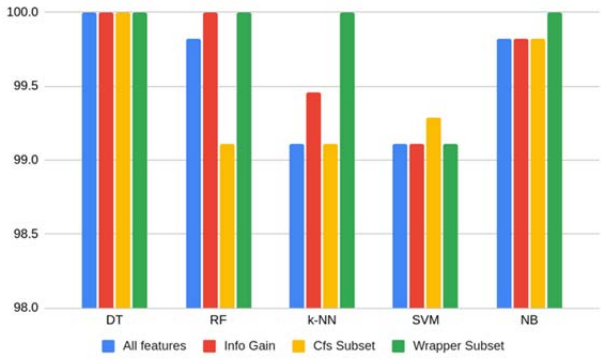

Figure 6. Accuracy of machine learning algorithms

The Decision Tree classifier, proposed in this work, achieved the accuracy of 100% for perfusor performance prediction, outperforming the previously achieved result of 98.06% using Artificial Neural Network, in the research done by Hrvat et al. [20].

The Artificial Neural Network is a more complex algorithm in terms of implementation and computation compared to Decision Tree, which is much simpler, faster, and more accurate in predicting the performance of perfusor. The Decision Tree algorithm has the potential to accurately predict the performance of other medical devices due to its speed and simple design. The limitation of this study is that the dataset that was used had only 1168 instances and it was further divided into training and testing subsets in a 50-50% ratio. Also, some important attributes like the year of device production, the total running time of the device in the institution per day were not available for the analysis. As for future work, we will add more measurement data into our database with more attributes that will make the classifier more convenient and realistic. Also, we plan to build a classifier that will inform institutions that the device, based on historical data, needs to be repaired before the next regular measurement.

Implementing this model in the real world project is straightforward and we have developed a piece of software by using Java programming language in which the user enters the measurements and then the final result (will device pass inspection) is returned to the user. For future work, we plan to feed the measurement results into the database directly and predict the device performance without user interaction.

Table 6. Performance of algorithms on feature selected by Wrapper Subset algorithm with BestFirst search

Author(s)	Method(s)	Attribute Selection	Accuracy	Device
Badnjevic et al. [15]	Random Forest	Genetic Algorithm	100%	Defibrillator
Hrvat et al. [16]	Artificial Neural Network	Not applied	98.06%	Infusomats and Perfusors
Hadzic et al. [17]	Artificial Neural Network and Fuzzy Logic	Not applied	97.44%	Anesthesia Machine
This study	**Decision Tree**	**Info Gain with Ranker**	**100%**	**Perfusors**

As shown in Table 6, the automatic medical device inspection and maintenance hasn't been the topic of research in many studies. The only work found, written for the Perfusor medical device, was done by Hrvat et al. [16] where they combined Infusomats and Perfusors together in one dataset. The authors have used the ANN with one hidden layer that consists of ten neurons. They used an 80-20 split ratio and the achieved accuracy was 98.06% for the perfusor pump. Compared to their study, we have used an attribute selection mechanism which resulted in higher accuracy. Moreover, as our focus was on Perfusors only, the algorithm performed better on the dataset that included only one type of device. In the work done by Hadzic et al. [17], ANN with 23 neurons in one hidden layer was combined with a fuzzy classifier. They have also used an 80-20 split ratio with a limited number of samples (only 197 samples in both training and testing set), again, implying the results not being reliable. Badnjevic et al. [15] used Random Forest with the Genetic Algorithm on defibrillators and achieved an accuracy of 100%. Defibrillators are much different from perfusor devices in terms of medical purpose and therefore are not comparable. However, the study served as a good reference in terms of methodology development. The split ratio that they used is 80-20.

Unlike all the above-mentioned studies we have used a 50-50 split ratio, ensuring that the class distribution in both subsets is roughly the same because our dataset was highly unbalanced. If we applied an 80-20 split ratio to our dataset, there would only be a few negative instances in the test set which is not enough for the proper validation method. This implies that percentage split issues for such medical datasets need more attention in future research studies.

Our goal was to develop a methodology that can be applied to any other medical device. Among all algorithms that we have applied, the Decision Tree combined with Info Gain attribute selection achieved the best accuracy with just two attributes. The attributes that were selected are: attribute 37 (6th Measured value [ml]) and attribute 15 (1st conformity assessment). It is important to emphasize that the algorithm considers the entire attribute range in order to make the decision and not just the attributes that are tightly coupled. By using this approach the number of attributes that were examined is reduced to 2 and that decreased the computational cost and time to build the model while model accuracy remained the same. This will also reduce the inspection time by notified bodies because they do not need to collect all measured values of attributes. Accuracy of 100% is desirable due to the fact that we are examining real working devices that patients are using where any error can endanger patient life. We have used a 50-50 split ratio that made the model more convenient and reliable.

The decision tree algorithm is a good choice for our dataset because it contains both numerical and categorical attributes where numerical values can be either discrete or continuous. Therefore, intensive data pre-processing was not needed since the decision tree does not require data normalization or scaling and it can deal with missing values (missing inspection measurement) without any issues. Furthermore, the decision tree performs well on a linearly inseparable dataset and the outliers (measurement errors) are handled automatically.

This research is done in the context of the new Medical Device Regulation (MDR) which obliges stakeholders (not only manufacturers) to monitor the quality, performance, and safety of a device throughout the product lifecycle and to apply corrective or preventive actions when necessary, and to do so for every device. MDR also requires that the Periodic Safety Update Report is updated into EUDAMED (European Databank on Medical Devices) which implies the creation of "big data" structures in the domain of post-market surveillance of medical devices, therefore the possibility of usage of AI to increase the safety and performance.

4. Conclusion

In this paper an expert system for the prediction of perfusor performance was developed. The aim of the system is to detect potential deviations and failures of perfusor safety and performance that can lead to inadequate patient treatment. The system was developed based on big data formed based on 1168 perfusor inspection reports collected in the period from 2015-2019 by an appointed inspection body accredited by ISO 17020. For the development of an expert system, five different machine learning algorithms were used: (1) Decision tree, (2) Random Forest, (3) k-NN, (4) SVM, and (5) Naive Bayes. Although all classifiers had an accuracy of over 99% the best among them when considering all

performance evaluation parameters (accuracy, ROC area, F-measure), was DT with an accuracy of 100%, F-measure of value 1, and ROC area of value 1. This performance was obtained using different attribute selection algorithms and in a dataset with all attributes. Presented expert systems can be embedded on IoT devices and used standalone in daily practice in clinical engineering units, or they can be part of a centralized medical device management system. In both cases, it would use generated data to predict device performance and notify about possible performance and safety failures. As artificial intelligence has already been adopted in medical devices for predicting and managing diagnosis and treatments, this system offers to introduce artificial intelligence in medical device management routines based on evidence collected from a single device. We are aware that we need a larger dataset for better prediction and that is something that we intensively work on. We are also planning to enrich the dataset with more attributes (year of manufacture, working hours of the device) that will make the model more convenient and accurate.

As regulators are becoming much more focused on Post-market surveillance (PMS) and risk management in the MD industry, it is now the time to start research and investigation in this field. While researchers are focusing on using artificial intelligence as part of medical device software, its application for medical device management is under-developed. Motivated by this state-of-the-art, these research activities contribute to the investigation of the paradigm shift from "reactive" PMS to "predictive" PMS. The objective of this research was to develop elementary concepts for the application of artificial intelligence for predictive post-market surveillance strategies of perfusors.

References

[1]. European Parliament and the Council of the European Union. (2020). Regulation (EU) 2020/561 of the European Parliament and of the Council of 23 April 2020 amending Regulation (EU) 2017/745 on medical devices, as regards the dates of application of certain of its provisions. *Official Journal of the European Union*, *63*, 18-22.

[2]. Bos, G. (2018). ISO 13485: 2003/2016—medical devices—quality management systems—requirements for regulatory purposes. In *Handbook of Medical Device Regulatory Affairs in Asia* (pp. 153-174). Jenny Stanford Publishing. doi: 10.1201/9780429504396

[3]. *Food and Drug Administration. (2017). MAUDE—Manufacturer and User Facility Device Experience* Retrieved from: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfmaude/search.cfm [accessed: 06 May 2022].

[4]. European Commision. (2022). EUDAMED - European Database on Medical Devices. Retrieved from: https://ec.europa.eu/tools/eudamed [accessed: 26 May 2022].

[5]. FDA. (n.d.). *Maude Adverse Event Report: B. Braun Melsungen Ag Perfusor® Pump, Infusion, Pca*. Retrieved from: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfmaude/detail.cfm?mdrfoi__id=8335747&pc=MEA [accessed: 20 May 2022].

[6]. *FDA. (n.d.). Maude Adverse Event Report: B. Braun Melsungen Ag Perfusor Space Syringe Pump*. Retrieved from: https://www.accessdata.fda.gov/scripts/cdrh/cfdocs/cfmaude/detail.cfm?mdrfoi__id=8290393&pc=FRN [accessed 06 August 2022].

[7]. Kramer, D. B., Tan, Y. T., Sato, C., & Kesselheim, A. S. (2013). Postmarket surveillance of medical devices: a comparison of strategies in the US, EU, Japan, and China. *PLoS medicine*, *10*(9), e1001519. doi: 10.1371/journal.pmed.1001519.

[8]. Badnjević, A., Cifrek, M., Magjarević, R., & Džemić, Z. (Eds.). (2018). *Inspection of medical devices: for regulatory purposes*. Springer Singapore. doi: 10.1007/s10916-017-0783-7.

[9]. Badnjević, A., Pokvić, L. G., Hasičić, M., Bandić, L., Mašetić, Z., Kovačević, Ž., ... & Pecchia, L. (2019). Evidence-based clinical engineering: machine learning algorithms for prediction of defibrillator performance. *Biomedical Signal Processing and Control*, *54*, 101629. doi: 10.1016/j.bspc.2019.101629

[10]. Hrvat, F., Spahić, L., Pokvić, L. G., & Badnjević, A. (2020, June). Artificial neural networks for prediction of medical device performance based on conformity assessment data: Infusion and perfusor pumps case study. In *2020 9th Mediterranean conference on embedded computing (MECO)* (pp. 1-4). IEEE. doi: 10.1109/MECO49872.2020.9134359

[11]. Hadžić, L., Fazlić, A., Hasanić, O., Kudić, N., & Spahić, L. (2019, May). Expert System for Performance Prediction of Anesthesia Machines. In *International Conference on Medical and Biological Engineering* (pp. 671-679). Springer, Cham. doi: 10.1007/978-3-030-17971-7_101.

[12]. Masetic, Z., & Subasi, A. (2016). Congestive heart failure detection using random forest classifier. *Computer methods and programs in biomedicine*, *130*, 54-64. doi: 10.1016/j.cmpb.2016.03.020.

[13]. Das, H., Naik, B., & Behera, H. S. (2020). An experimental analysis of machine learning classification algorithms on biomedical data. In *Proceedings of the 2nd international conference on communication, devices and computing* (pp. 525-539). Springer, Singapore. doi: 10.1007/978-981-15-0829-5_51.

[14]. Saeys, Y., Inza, I., & Larranaga, P. (2007). A review of feature selection techniques in bioinformatics. *bioinformatics*, *23*(19), 2507-2517. doi: 10.1093/bioinformatics/btm344

[15]. Fluke Biomedical. (n.d.). IDA-5 Infusion Device Analyzer. Retrieved from: https://www.flukebiomedical.com/products/biomedical-test-equipment/infusion-pump-analyzers/ida-5-infusion-device-analyzer [accessed: 05 June 2022].

[16]. Visalakshi, S., & Radha, V. (2014, December). A literature review of feature selection techniques and applications: Review of feature selection in data mining. In *2014 IEEE International Conference on Computational Intelligence and Computing Research* (pp. 1-6). IEEE. doi: 10.1109/ICCIC.2014.7238499.

[17]. Quinlan, J. R. (1986). Induction of decision trees. *Machine learning*, *1*(1), 81-106. doi: 10.1007/BF00116251

[18]. Padmaja, D. L., & Vishnuvardhan, B. (2016, February). Comparative study of feature subset selection methods for dimensionality reduction on scientific data. In *2016 IEEE 6th International Conference on Advanced Computing (IACC)* (pp. 31-34). IEEE. doi: 10.1109/IACC.2016.16

[19]. Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning.* (Doctoral dissertation, The University of Waikato).

[20]. Kohavi, R., & John, G. H. (1997). Wrappers for feature subset selection. *Artificial intelligence*, *97*(1-2), 273-324. doi: 10.1016/S0004-3702(97)00043-X

[21]. Webster, R. W. (1991). Useful AI tools-a review of heuristic search methods. *IEEE Potentials*, *10*(3), 51-54. doi: 10.1109/45.127648

[22]. Nguyen, C., Wang, Y., & Nguyen, H. N. (2013). Random forest classifier combined with feature selection for breast cancer diagnosis and prognostic. *Journal of Biomedical Science and Engineering*, *6*(5), 551-560. doi: 10.4236/jbise.2013.65070

[23]. Sun, S., & Huang, R. (2010, August). An adaptive k-nearest neighbor algorithm. In *2010 seventh international conference on fuzzy systems and knowledge discovery* (Vol. 1, pp. 91-94). IEEE. doi: 10.1109/FSKD.2010.5569740.

[24]. John, G. H., & Langley, P. (2013). Estimating continuous distributions in Bayesian classifiers. *arXiv preprint arXiv:1302.4964*. doi: 10.48550/arXiv.1302.4964

[25]. Jin, C., De-Lin, L., & Fen-Xiang, M. (2009, July). An improved ID3 decision tree algorithm. In *2009 4th international conference on computer science & Education* (pp. 127-130). IEEE. doi: 10.1109/ICCSE.2009.5228509.

[26]. Noble, W. S. (2006). What is a support vector machine?. *Nature biotechnology*, *24*(12), 1565-1567. doi: 10.1038/nbt1206-1565

[27]. Üstün, B., Melssen, W. J., & Buydens, L. M. (2006). Facilitating the application of support vector regression by using a universal Pearson VII function based kernel. *Chemometrics and Intelligent Laboratory Systems*, *81*(1), 29-40. doi: 10.1016/j.chemolab.2005.09.003.

[28]. Fung, G., & Mangasarian, O. L. (2002, April). Incremental support vector machine classification. In *Proceedings of the 2002 SIAM International Conference on Data Mining* (pp. 247-260). Society for Industrial and Applied Mathematics. doi: 10.1137/1.9781611972726.15

[29]. Lin, Y., Yu, H., Wan, F., & Xu, T. (2017, September). Research on classification of Chinese text data based on SVM. In *IOP Conference Series: Materials Science and Engineering* (Vol. 231, No. 1, p. 012067). IOP Publishing. doi: 10.1088/1757-899X/231/1/012067

[30]. Witten, I. H., Frank, E., Trigg, L. E., Hall, M. A., Holmes, G., & Cunningham, S. J. (1999). Weka: practical machine learning tools and techniques with Java implementations. *Proceedings of the ICONIP/ANZIIS/ANNES'99 Workshop on Emerging Knowledge Engineering and Connectionist-Based Information Systems* 192–196 doi: 10.1016/C2009-0-19715-5