A Machine Learning Approach for Real Time Prediction of Last Minute Medical Appointments No-Shows

Ricardo Almeida[1], Nuno André Silva[2] and André Vasconcelos[3]
Instituto Superior Técnico, Lisbon, Portugal[1]
Grupo Luz Saúde, Learning Health, Lisbon, Portugal[2]
Instituto de Engenharia de Sistemas e Computadores, Investigac ̧ao e Desenvolvimento (INESC-ID), Lisbon, Portugal[3]
[1]ricardo.oliveira.almeida@ist.utl.pt, [2]nuno.asilva@hospitaldaluz.pt, [3]andre.vasconcelos@tecnico.ulisboa.pt

Keywords: No-show; Healthcare; Prediction Algorithms; Pre-Processing; Machine Learning Techniques.

Abstract: A no-show is when a patient misses a previous scheduled appointment. No-shows cause an impact in the healthcare sector, decreasing their efficiency. When a patient misses an appointment it wastes the clinic resources, postpones his or her chance to get treated for a medical condition and denies medical service to another patient. In this research machine learning techniques are used to find patterns in healthcare data and make no-show predictions. A no-show prediction model is proposed to integrate machine learning techniques into a model that supports the testing of predictions on different datasets. The model is integrated into an online medical appointment booking platform to allow the models and predictions made, to be saved and integrated in a real time system. Machine learning techniques are tested using three datasets with different characteristics. Through these tests the model proposed is able to find the best features, which are similar in every dataset. The results obtained are still not ideal for the real world, but allow the comparison of prediction algorithms and techniques.

1 INTRODUCTION

A no-show is when a patient misses an appointment that was previously scheduled. This phenomenon happens in all sorts of areas, where there is the need to schedule patients or clients into a time slot. In this paper we focus the healthcare area. No-shows cause an impact in every hospital and clinic in the world. Attenuating the effects of no-shows in the healthcare area is something that can provide many economic and social benefits. When a no-show occurs there are two consequences, the first happens to the patient who misses the appointment who postpones his chance to be treated for a medical condition. The second one affects the hospital and other patients, because there are other patients who could have used that opportunity to be seen by the doctor. This lost opportunity also means a loss of revenue to the clinic and hospital.

Given the current high demand for healthcare, wasting available resources is unacceptable, contributing for the increase in the list of patients waiting to receive assistance. To attenuate some of these consequences it is important to figure out what makes patients miss their appointments and, whether or not there are identifiable patterns that allow us to know how likely are patients to miss their appointment.

In order to predict the no show probability, the appointment data stored by hospitals and clinics around the world can be used. Using this data and combining it with machine learning techniques it is possible to find some of these patterns and obtain a probability for how likely is a patient to no-show. If these probabilities are high then specific actions can be performed by the hospital, like scheduling another patient for that time slot or contact the patient to try to confirm the appointment.

1.1 Objectives

The objective of this research is to improve and keep developing a no-show prediction system based on existing state of art research. The goal of this system is to help clinics and hospitals mitigate the negative effects of noshows. There are three main features of this system: The first one is to notify the patients of the appointments and to confirm their presence. The second one is a prediction algorithm that uses machine learning techniques and will return the probability of no-show for any appointment. Finally if the system detects a high probability of no-show it will automatically try to reschedule another patient to that time slot. This research is focused on the second one by improving and testing the machine learning techniques. To achieve this goal this paper focus three main objectives:

- Create a prediction model that automatically returns the no-show probability for an appointment, and is able to efficiently test data from many clinics and hospitals without the need of an advanced user to tweak the system.
- To test many machine learning techniques and find out which provide the best and most consistent results across many datasets.
- To find out which features are more important to obtain better results, which prediction algorithm works best and how accurate can the predictions be in large datasets from real world clinics and hospitals.

This work is expected to give a solid foundation to allow the continuation of tests in the no-show prediction system, and give some answers to what results are possible to obtain from these machine learning methods to tackle the no-show problem.

This work is structured as follows. Section 2 includes a review on related work including previous work developed with similar goals and its limitations. Section 3 presents the prediction model developed. Section 4 describes the results that were obtained while comparing different techniques. Finally section 5 concludes the work.

2 RELATED WORK

No-shows are estimated to have a big financial impact on hospitals and clinics and as such many studies can already be found on analysing the impact of no-shows (Neal et al., 2005), how to best deal with them and using new ways to try to predict them. All of this in an effort to reduce the level of impact they have in hospitals and clinics worldwide.

Many of these studies try to pinpoint what are the major causes of a patient no-show (George and Rubin, 2003), whether they are involuntary or not. They also focus on looking at what are the best practices to reduce the impact of no-shows, this is normally done by overbooking (LaGanga and Lawrence, 2007), but has discussed in this articles this practice may have some impact on waiting time and client satisfaction. Machine learning is a technology that has been emerging and being used on several fields and, as such, there are some articles that studied how to take advantage of this to reduce no-shows. This technology uses the appointments data, in a clinic or hospital, and using an algorithm it tries to find patterns in the data that can then be used to make predictions on no-shows. These predictions can be in form of a probability, which translates to how sure the algorithm is of that outcome. Using these probabilities a clinic or hospital could then decide whether or not it would be better to schedule a patient for that time slot.

2.1 Causes of No-Shows

Finding causes for no-shows is a good starting point to check if these causes can be prevented from happening and, whether or not, they can be used as ways of predicting no-shows.

There are many studies that give a lot of emphasis on finding out what are the causes for a patient to not show to an appointment. Missing an appointment can be a voluntary or an involuntary act, this last one being, when the patient did not intend to miss the appointment. There are many reasons for not showing to an appointment these include forgetting the appointment, other competing priorities or conflicts, and the patient’s health status.

The most common reason is when a patient forgets the appointment (Neal et al., 2005), for this, many clinics have already implemented a phone or e-mail reminder, which is reported to reduce no-shows (Leong et al., 2006; Liew et al., 2009). Other reported reasons for no-shows are the health of the patient which may feel better and not need the appointment anymore, other priorities like a work schedule change or having to take care of another family member and some scheduling problems due to bad quality of the service can lead to wrong appointment information and to problems in cancelling the appointment. The weather can also be a factor if it is raining or snowing people prefer to stay home and if the health problem is not serious they can no-show (Norris et al., 2014). Financial problems and lack of transportation were also some of the reported reasons.

2.2 Features for No-show Prediction

To be able to predict whether a patient is going to no-show to an appointment, it is required to have access to many factors about the appointment and the patient, which in conjunction leads to a prediction that can be stronger by having access to many factors and many similar cases. Many studies tackle this problem in an attempt to make their prediction algorithms better (Turkcan et al., 2013; Alaeddini et al., 2015; Elvira et al., 2017; Daggy et al., 2010; Huang and Hanauer, 2003). So there is already some information to help figure out which features in the appointment data of a clinic or hospital have more relevance to predict a no-show.

These features can be divided in two categories: some are relevant to the patient like gender, age, marital status and insurance status. The others are relevant to the appointment like day of the scheduled appointment, the amount of time between the day the appointment was scheduled and the actual appointment day and the type of clinic.

The feature found, relevant to the patient, which most articles conclude as having the most predictive power is age, where younger patients seem to no show to more appointments than other age groups (Lee et al., 2005). In the patient category there are other features that have some impact. These are being unmarried, not having health insurance, the severity of the illness and the scholarship level. The gender of the patient was considered by most articles as having very little impact, in other words, there are no differences between men and women regarding their attendance to previously arranged appointments (Turkcan et al., 2013). In the appointment features it is found that the waiting time has a larger impact (Dantas et al., 2018; Norris et al., 2014). Other characteristics that also have an impact are the hour of the day, whether it is the patient first appointment (Bennet and Baxley, 2009), the medical specialty chosen, the hospital centre, whether it is a weekday or weekend, the type of appointment and the distance to the clinic. There are other features that are predicted to have an impact, but it is required to use other datasets than the ones that exist in most clinics like, for example, the weather.

Beyond all this features there is another one that has a huge impact in predicting accurately if a patient will no-show, which is the prior no-show history, whether the patient has missed the last scheduled appointments (Dantas et al., 2018; Norris et al., 2014). This feature is the one that has the most predictive power but it has an issue, we never know if a patient is going to have a sudden change of behaviour.

To get some of these features the given datasets by the clinics need to be pre-processed so it is possible to make new features out of the existing ones. This allows the use of more specific information that can work as a better predictor, for example, having a feature that tells you if the previous appointment was a no-show and having the hours of the day and the age divided in categories. Creating these new features allows for an improvement of the results to as much as 10% better predictions (Elvira et al., 2017).

2.3 No-show Prediction

Before starting this work it is important to look at the works that already exist for this no-show problem, what are the techniques used and the results obtained. The most important articles found on no-show prediction, that clearly stated which prediction algorithm was used, the features used, and the results were analysed and compared. (Turkcan et al., 2013; Alaeddini et al., 2015; Rinder, 2012; Ferreira, 2019).

After comparing the articles, four prediction algorithms stand out with the best results. These algorithms are the Hybrid Model, the Logistic Regression, the Neural Network and the Gradient Boosting Machine. It is still hard, based on these articles, to say which algorithm performed the best since they were not tested with the same datasets, neither on the same conditions.

Some promising results were already obtained, but the algorithms were tested for just one dataset. This is one of the issues this work will tackle by using different datasets, different features and different metrics to try to find out if there is an algorithm that actually has a general better performance.

2.4 Last Minute Medical Appointments No-Show Management Previous Research

There is already previous research that addressed the goal of predicting last minute noshows in healthcare. One of the first research to address this specific problem was developed by Daniel Sousa (Sousa and Vasconcelos, 2020), which focused on developing the algorithm to predict the no-shows, and creating a model to replace patients that have a higher chance of not showing. There is another reseach by Inês Ferreira (Ferreira and Vasconcelos, 2019), which focuses on testing other prediction algorithms and also updated the model for replacing patients.

2.4.1 Limitations

Although important contributions were done in these two researches, there is still much to be done in order to improve the prediction system.

In Daniel Sousa the prediction algorithm used was a simpler version of the Hybrid model in article (Alaeddini et al., 2015). The model was tested with only four features and only the accuracy was used to describe the results. This is not enough to infer the quality of the prediction algorithm.

In Inês Ferreira research four prediction algorithms were tested, which were Gradient Boosting, Logistic Regression, Random Forest and K-Nearest Neighbours. The results achieved seem good but there were some limitations in the tests done. One limitation was because the SMOTE technique was used in the test data at every split made by the cross validation. By doing this the approach generated fake samples and then predicted on those, which do not translate to the real world, since the real data will come imbalanced. The only way to validate these results would be to use the trained model on a dataset that has not yet been seen by the training model and no sampling technique was used on the test data. Other limitation on the tests is that some categorical features were used as numeric which makes the algorithm get wrong conclusion when performing calculations on those numeric values.

There are still other limitations found, one is in the fact that most datasets are different from each other with different features. This requires the code for pre-processing the datasets to have to be changed each time the prediction algorithm is trained. This is not scalable to the needs of the prediction system that MedClick wants to develop, as it will have to work with many clinics and hospitals from all over Portugal. Prediction algorithms can also work better with some datasets than others, being able to know how to get the best no-show predictions out of any dataset is really important for the scalability of the prediction system and the main focus of the solution that will be developed.

3 PREDICTION MODEL

This research proposes a creation of a Prediction Model to automate the process of constructing prediction models and obtaining predictions from different datasets, without requiring an advanced user to tweak the model every time a different dataset is used. Another advantage of this model, is that the predict model is running in short periods of time, allowing that any new appointments that arrives on the system API will be given a no-show probability value on a short period of time.

The prediction model has two main models and one model that supports the other two and provides a configuration file for the user. These models are discussed in the following subsections.

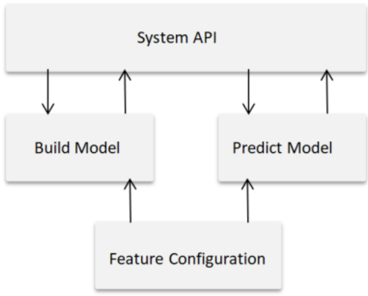

Figure 1: Models and their interaction in the Prediction model.

3.1 Feature Configuration

Feature configuration is a support file to be used in the build model and predict model phases. The goal of this configuration file is to make it easier to use any type of dataset, without having to change the way it was programmed every time a different dataset is used. This configuration file is basically a simple python file with a set of variables, which need to be filled with the names of the features of the specific dataset we are using. These variables are then used to by the Build Model to automatically pre-process the dataset features into the ones that will be used to train the dataset. This list of variables contains one to specify whether the data comes from the MedClick API or from a CSV file. It has also two variables that are specific to the CSV file, where the path to the CSV should be placed. The remaining features must be matched with the corresponding feature name of the dataset that will be used. The variables in the configuration file are:

- **Type of Input:** whether the data comes from a CSV or API.
- **CSV Path:** the path to the appointments dataset in CSV format.
- **CSV Path2:** the path to the patient's dataset in CSV format (only required if they come in separate).
- **ID:** patient's identifier.
- **Age:** patient's age.
- **Birthdate:** patient's birthdate (only required if the dataset has no feature age).
- **Gender:** patient's gender.
- **District:** patient's living district.
- **Postal-code:** patient's postal code.
- **Insurance:** type of insurance used.
- **Specialty:** medical specialty of the appointment

- **ID Medic:** identifier of the medic responsible for that appointment.
- **Appointment date:** date the appointment was scheduled for.
- **Appointment hour:** hour of the appointment was scheduled for.
- **Scheduled date:** date when the appointment was scheduled
- **No-Show:** whether the appointment was a no-show.
- **No-Show positive value:** the value for no-show in the no-show feature (for example "YES").
- **Other features:** the remaining features of the dataset that will not be used.

The variables chosen for the configuration file, correspond to the most common features present in most healthcare provider's datasets and to the features that possess stronger predictive power. Other features that prove to have a strong predictive power can be added in a future stage so that they can be easily pre-processed.

A dataset does not need to have correspondence to all the variables that are in the configuration file, variables that do not have a correspondent feature should be left empty. This way any dataset, as long as it has the essential features, can be tested. The variables that need to be filled in order for the dataset to be tested are patient id, appointment date and no-show. Of course the more features that are filled, the more information the prediction algorithms will have to learn from, which will lead to better predictions.

The pre-processing done will not look at every possible scenario of input, as this would be almost impossible to do. This means that some features will be required to have a certain format in order to work. One of these cases is postal code, which requires the values to come in the format ####-###. Other features are more intuitive, age will require numeric values and appointment date will require a date, as expected. If the datasets do not have these formats they should be manually pre-processed beforehand.

In latter stages, these configuration files are not going to be required and the pre-processing should be optimized to the MedClick API, since here is where all the data will come from.

The main advantage of the feature configuration is that it permits many datasets to be worked without the constant need to change things in the main code. This is great for someone who is not familiar with the code to be able to easily test the prediction model on a dataset. Also in these early stages, Medclick will work with many datasets from different healthcare providers, which makes a configuration file like this essential.

3.2 Build Model

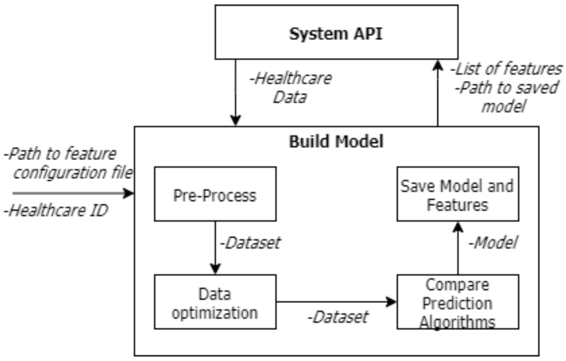

Figure 2: Build model components.

Build Model is the model that builds the prediction model, which is then used to obtain no-show predictions on the data. This data can come from a CSV file or directly from the system API. The reason to have the CSV option is because most datasets, from the majority of clinics and hospitals, come in this format. Also, it will be important to have a way to quickly test these datasets without integrating them in the API, since due to data protection measures this will not happen right away.

Build Model receives as arguments the path to the feature configuration file to be used and the healthcare Id of the associated healthcare provider. The model is then divided into four phases:

- **Pre-process:** Pre-processing is where the data from a specific healthcare provider is transformed into data that can be used by the prediction algorithms. This transformation adds new features from the existing ones to give the algorithms more information to learn from. It also removes or replaces missing values and transforms categorical variables with One-Hot encoding into dummy variables with values of 1 if true and 0 for false. Beyond this all the numeric features are normalized into values in the range of 1 to 0, since the rest of the features are all binary this puts all features in the same range of values. This allows the algorithms to learn better without giving too much weight to features with high numerical values.

- **Data optimization:** In this phase the data is ready to be used by the prediction algorithms, but first it should be optimized to give the best predictions possible. The first step is choosing only the features that possess the strongest predictive power and removing the others. First variance threshold is used to remove features that are almost constant, since these features will not contribute to the predictions. The next step is using a feature selection algorithm, the one chosen is Boruta. To validate the features chosen by this algorithm, it is used alongside a 10 fold cross-validation and for each cross-validation fold, the features chosen are registered and only the ones that appear more than 80% of the time are chosen. The final stage is to balance the data, since most of the data comes unbalanced with many more shows than no-shows. This can cause the prediction algorithms to prefer to classify appointments as shows in favour of having more accuracy. To mitigate this problem, SMOTE with Edited Nearest Neighbours is used which balances the data by generating data samples with SMOTE and then using k Nearest Neighbours it removes those samples that are misclassified by its neighbours. After the data is balanced, it is now ready to be fed to the prediction algorithms and be able to find which give better predictions.

- **Compare Prediction Algorithms:** It is impossible to find a prediction model that will be better for ever scenario and for every dataset, things like the size of the dataset and the number of features can affect the quality of the predictions for some algorithms. To solve this problem in this phase four different prediction algorithms are run on the dataset on a cross validation with 3 folds only, to prevent it from being very computationally expensive. The four prediction algorithms are Artificial Neural Network, Gradient Boosting, Logistic Regression and Random Forest. These algorithms were tweaked to provide better results, but the type of optimizations required for one dataset are not the same for other datasets. Because of this a general optimization was used in all datasets. After running the prediction algorithms, the one with the best overall score in the metric f1-score is the one chosen. This metric was chosen because, for these results, it will be more important to have the right balance between recall and precision than having good accuracy. At a later stage, this phase of comparing prediction algorithms can be removed once the data comes solely from the API, since at this stage there will be more control on the features used and the prediction algorithm that generally performs better can be chosen. This will save computation time which will be more important at that stage.

- **Save Model and Features:** After we have chosen the model it must be saved, this is done using a pickle which is a python module that allows us to save the model in a file .dat. This file can then be easily loaded to make predictions for that healthcare provider. The name of the saved file will be unique having the id of the healthcare provider, the name of the model used and the accuracy obtained. This name along with the features chosen in the data optimization phase will be saved in the MedClick API to be used in the prediction phase. The reason to save the features, as well, is that the predictions need to be made with the same features the model was trained on, otherwise it will not work.

3.3 Predict Model

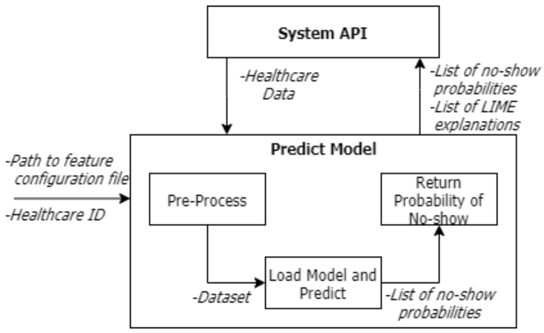

Figure 3: Predict model components.

The predict model is the model used to obtain the probability of a patient missing his or her appointment. This model has two functions, one to obtain the probability of no-show for all the appointments scheduled, in every healthcare, and another to obtain the probability of no-show for a specific healthcare provider. The first one does not receive any argument and only works for the appointments in the API. This function is scheduled to be executed every hour or less, so that the probabilities can be regularly updated and in the case new appointments are added, we can quickly figure out what is the probability of no-show.

This is especially important when appointments are scheduled for the same day or the next day. The other function is to predict for a single healthcare, this function receives as arguments the path to the feature configuration and the healthcare id. This function also works for data in CSV, and in this case it should receive an extra argument with the appointments we want to make predictions on. The predict model has three phases, which are:

- **Pre-process:** In the pre-processing phase the appointments which we want to predict are joined with the original dataset. This is done so that it is possible to pre-process the new appointments, in order for them to have all the features. Also it is required to be able to put the right values in features like the number of appointments and number of no-shows, since this needs to be calculated for the whole data. After this, the list of features saved in the MedClick API and associated with this healthcare provider is retrieved. The features that are not in this list of features are removed from the appointments to predict, this is done so they match the ones where the model was trained on. The numeric values are also normalized using MinMax Scaler so that everything is on the same scale.

- **Load model and predict:** In this phase the name of the model used to train the data is retrieved from the API and using python module pickle, the model is loaded. Using the loaded model the predictions for the probability of no-show are obtained for all of the appointments.

- **Return probability of no-show:** If the appointments come from the API, the list of probabilities must be uploaded to the API. This is done by using the appointment identifier and for each one uploading the respective probability of no-show to the API. Beyond this, an explanation of how the algorithm obtained that prediction is updated with the probability of no-show. This explanation is obtained using LIME, which gives a value for how relevant the features were to the prediction, and returns a list with the nine most relevant features for that prediction. An example of these explanations plotted can be seen in figure 4, where the left red bars are the features that contribute to being a show and the right green bars are the features that contribute to being a no-show. With this it is possible to have a better idea of how the prediction algorithms are obtaining that probability and, in these early stages, will allow a better understanding of which features have the most impact and which tweaks can be made to improve it. In case the appointments come from a CSV file, the main process is the same but the results are saved to a CSV file, instead of uploading them to system API. This file will have the appointments predicted with the original dataset features and an extra feature with the probability of no-show.

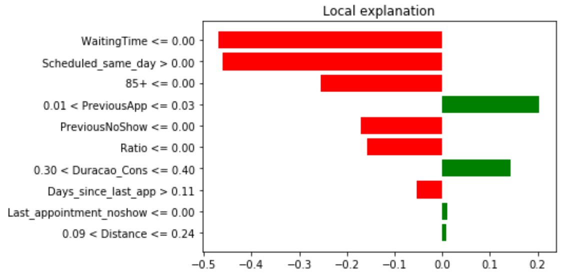

Figure 4: Plotted explanation obtained using LIME for the MD Clínica dataset.

4 RESULTS ANALYSIS

In this chapter the results obtained in each of the different datasets are analysed. The prediction algorithms and machine learning techniques were tested in datasets from three different healthcare providers. One from a dental clinic called MD Clínica, the second one was obtained online from Kaggle and comprises data from a Brazilian dataset and the last one is from Hospital da Luz, which is the largest dataset of the three.

With this analysis, it was possible to see which machine learning techniques and what conducts are more efficient at predicting no-shows.

4.1 Prediction Algorithms

This section compares four prediction algorithms to find out, which ones can provide more reliable predictions. The four algorithms used are Artificial Neural Network (ANN), Gradient Boosting (GB), Logistic Regression (LR) and Random Forest (RF). To compare the prediction algorithms, Boruta was chosen as the feature selection algorithm. Boruta was chosen because it uses the least amount of features and has a faster computation time, while achieving the same results has the other feature selection techniques tested. Since most of these datasets are imbalanced, a sampling technique for balancing the datasets was used. This will allow the prediction algorithms to find more no-shows and increase recall at the cost of some precision. The sampling technique chosen was SMOTE with Edited Nearest Neighbours because this technique achieved the best recall and f1-score in most datasets, which translates to more no-shows found. This sampling technique increases the number of samples from the minority class by generating similar samples to the existing ones and then uses k Nearest Neighbours to locate those samples in a dataset that are misclassified by its neighbours and then removes them.

This algorithm was executed in a 10 fold cross validation and the average scores for each one of the prediction algorithms was obtained. This was tested on the three datasets to find out if there is a prediction algorithm that performs well on all the datasets.

The results obtained can be seen in figures 5, 6 and 7, each one corresponding to a different dataset. No prediction algorithm was found to be better in all the scenarios, the most consistent one is Gradient Boosting, but it will also depend on what is required. If a more conservative approach with more precision is required than Random Forest or Gradient Boosting are the best options. If the goal is to find as many no-shows as possible with the cost of many misclassifications then Logistic Regression or Artificial Neural Network are the best options.

No prediction algorithm will be discarded with the tests made, as larger datasets or different features can change the type of results, this is specially the case for Artificial Neural Network which needs many data to learn efficiently. More tests will be required to choose a prediction algorithm, but at the last stages of the prediction model, it will be important to have just one prediction algorithm. This would reduce computation time and resources spent. If a prediction algorithm is found to be constantly outperforming the others then it will be chosen.

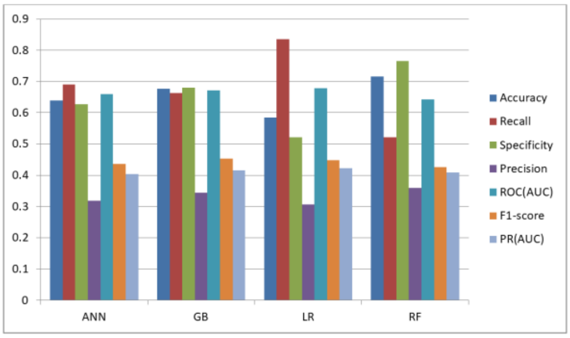

Figure 5: Results achieved by the prediction algorithms on the dataset from Brazil.

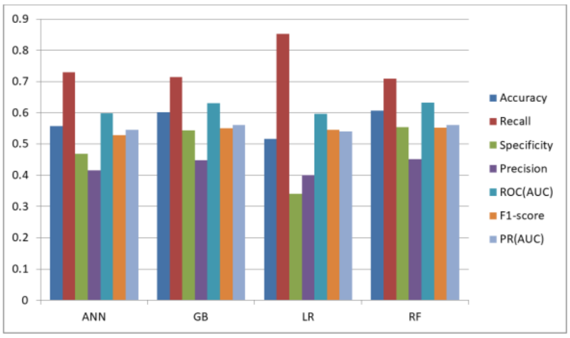

Figure 6: Results achieved by the prediction algorithms on the dataset from MD Clínica.

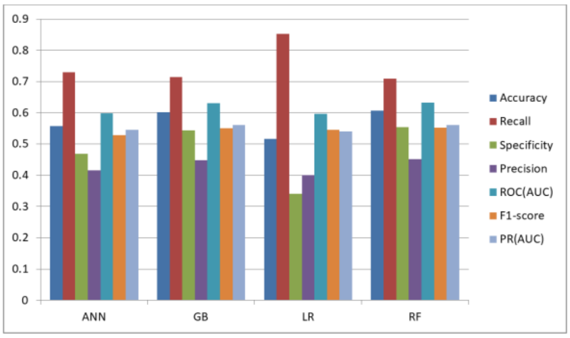

Figure 7: Results achieved by the prediction algorithms on the dataset from Hospital da Luz.

4.2 Predicting Last Week of Dataset

This comparison was done using the last week of each dataset for testing and the rest for training. A threshold of 70% was also used, what this means is that only no-shows with a probability of 70% or more are considered no-shows. This is an attempt to mimic a real life scenario and find out how many no-shows and misclassifications happen. The sampling technique used was SMOTE with Edited Nearest Neighbours and the feature selection was Boruta. In the next table 1, we can see the comparison between the confusion matrices for all datasets and prediction algorithms. The amount of appointments for that week varies for each dataset, Brazil is the one with more appointments compressed into that week with a much higher number than the others. In all of the datasets it is possible to see that on average 50% of no-shows are found by the prediction algorithms. This also depends on the prediction algorithm chosen. The prediction algorithm that finds more no-shows in all datasets is Artificial Neural Network but it also has the highest number of false positives. On the other hand, we have Random Forest with the least number of false positives but the least no-shows found, which translates to a more conservative and precise approach.

In MD Clínica dataset we can see that on average for every no-show found there is one false positive. The prediction algorithm with the best results here is Gradient Boosting, since it founds almost as many no-shows as Artificial Neural Network but at a much smaller cost of false positives. The Random Forest algorithm could also be used for a more conservative approach, as it has the least amount of misclassifications, making it the most precise of the four.

In Brazil dataset, for every no-show found there is slightly more than the double of false positives. The prediction algorithms with the best results here are Logistic Regression and Gradient Boosting with similar results and more balanced approaches than the other two algorithms.

In the dataset from Hospital da Luz, there is almost the triple of false positives compared to no-shows found. Here it is clear that the more the dataset is imbalanced the more false positives are to be expected. The best prediction algorithm here is Gradient Boosting, since it is even more precise than Random Forest and finds more no-shows. Also the number of no-shows found is not that distant from Artificial Neural Network but with less false positives.

Table 1: Comparison of confusion matrices for all datasets and prediction algorithms.

	MD Clínica	Brazil	Hospital da Luz
ANN	[873 385] [292 373]	[12808 5109] [1785 2285]	[4335 704] [250 222]
GB	[931 327] [300 365]	[13651 4266] [2089 1981]	[4617 422] [289 183]
LR	[903 355] [359 309]	[13469 4469] [1982 2088]	[4342 697] [268 204]
RF	[993 265] [356 306]	[13866 4051] [2315 1755]	[4611 428] [329 143]

4.3 Feature Importance

After comparing the feature importance in each of the datasets we can see that most of the chosen features are similar. This means there are some constant features that are better at predicting no-shows. The most relevant feature is waiting time, it seems the time from when the appointment was scheduled to the time of the appointment is crucial to find no-shows.

Other feature that is very important is distance, which has even slightly more importance than waiting time in the dataset from Hospital da Luz. This feature is the distance between the place of residence and the hospital. This feature is especially important in Hospital da Luz, since this is an important private hospital in the centre of Lisbon and, as such, it attracts patients from all over Portugal and there are even examples of patients outside of Portugal. It was not possible to obtain this feature for Brazil dataset, because the available features did not allow this to be calculated.

Other relevant features that follow these two but with a large difference, with no specific order, are the days since the last appointment, number of previous appointments, number of previous no-shows, ratio and whether last appointment was a no-show. These features have different values of importance depending on the dataset but they are chosen in all the datasets, which means these features are also very important to accurately predict a no-show.

Other unique features of some datasets that got a considerable value of importance are appointment duration, which is specific to the appointments from MD Clínica and whether a message was received, which is specific to the appointments from Brazil. These features can lead to better results in the predictions and, as such, an attempt should be made to make this available on other obtained datasets.

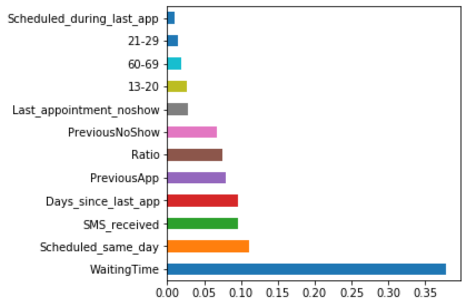

Figure 8: Feature importance graph for the dataset from Brazil.

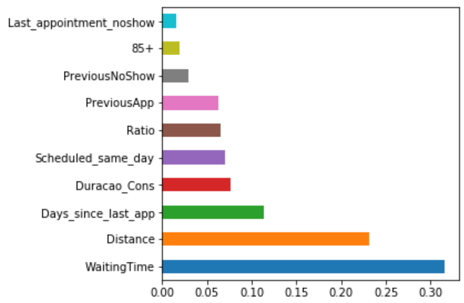

Figure 9: Feature importance graph for the dataset from MD Clínica.

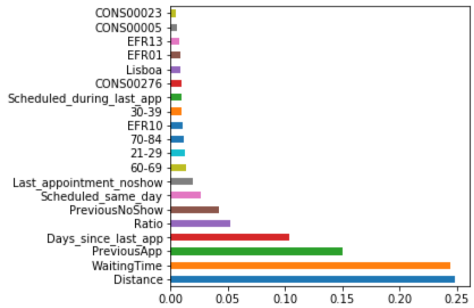

Figure 10: Feature importance graph for the dataset from Hospital da Luz.

5 CONCLUSIONS

This research was done in the healthcare area focusing on the no-show problem. It seeks to find and implement a solution capable of reducing no-shows and subsequently increase efficiency in the healthcare providers. The three major contributions of this research are next discussed.

The major contribution is the creation of a prediction model to optimize and automate testing. A prediction model was created to make the training of new models and obtaining of predictions from datasets easier and more efficient. Since all the datasets come with different characteristics and features, it would be required to change the code every time. This way the pre-processing phase and training phase were optimized, requiring a configuration file only to train the model and to make predictions. The prediction model was also integrated into an online medical appointment booking platform which is provided through an API. Functions had to be created for retrieving the data and saving the models and no-show probabilities.

With this work many new features were added and tested in an attempt to figure out which features are more relevant and improve prediction results.

Machine learning algorithms were tested on three different datasets, in an attempt to get better predictions for no-shows. Through these results it was possible to see which prediction algorithms and techniques are better for predicting no-shows.

The main conclusions that can be made are on the results obtained, while testing the prediction algorithms on three different datasets. The first thing that can be concluded is that the size of the dataset did not have a large impact on results. What impacted more was the type of features available and the how much imbalanced the data was.

The most important features are similar in every dataset and the features that were considered more important to identify no-shows are waiting time and distance. Since all these datasets were imbalanced, sampling techniques were used to counter this problem. These techniques balance the data by generating new data until the number of no-shows matches the number of shows. Using a sampling technique allowed the prediction algorithms to find a much larger number of no-shows but at the cost of being less precise. There is always a trade-off between recall and precision, the higher recall there is the more no-shows are found, but also the number of false positives increase and consequently the precision decreases. Whether more precision is required or more recall will depend upon the clinic or hospital strategy. Some hospitals and clinics will want to keep waiting time to a minimum and favour precision, while others with less volume of patients might prefer higher recall. Having a confirmation strategy will also be very important to reduce many of these false positives.

In the case of the prediction algorithms all of them achieved different results some favouring more precision, like Gradient Boosting and Random Forest and the other two more recall. There was no prediction algorithm that can be considered better at this stage, more tests will be required, but the prediction algorithm that achieved better results consistently was Gradient Boosting.

The results obtained are far from ideal and more features will be required to make these predictions better. Many no-shows can already be found but at a cost of some mistakes. We conclude that these predictions can help but are not still strong enough as a standalone strategy and should be combined with other scheduling strategies like patient confirmation.

ACKNOWLEDGMENTS

I would like to acknowledge MedClick for all the productive discussions and insights given that shaped this work.

I would also like to express my gratitude to Grupo Luz Saúde for giving me access to Hospital da Luz data, even after all the complications due to the current pandemic, it was a crucial part of this work.

Last, I want to acknowledge MD Clínica for giving me access to their data which contributed greatly to this research.

REFERENCES

Sousa, D., Vasconcelos, A., 2020. 'Last minute medical appointments no-show management'. International Journal of Healthcare Information Systems and Informatics, Volume 15, Issue 4, ISSN: 1555-3396, DOI: 10.4018/IJHISI.

Ferreira, I., Vasconcelos, A., 2019. "MedClick: Last Minute Medical Appointments No-Show Management", 12th International Joint Conference on Biomedical Engineering Systems and Technologies, HEALTHINF (HEALTHINF 2019), Prague, Czech Republic-"

Turkcan, A., Nuti, L., DeLaurentis, P.-C, Tian, Z., Daggy, J., Zhang, L., Lawley, M., and Sands, L., 2013. "Noshow modeling for adult ambulatory clinics", in Handbook of Healthcare Operations Management, pp. 251–288.

Alaeddini, A., Yang, K., Reeves, P., and Reddy, C. K., 2015. "A hybrid prediction model for no-shows and cancellations of outpatient appointments," IIE Transactions on Healthcare Systems Engineering, vol. 5, no. 1, pp. 14–32.

Elvira, C., Ochoa, A., Gonzálvez, J. C., and Mochón, F., 2017. "Machine-Learning-Based No Show Prediction in Outpatient Visits", International Journal of Interactive Multimedia and Artificial Intelligence, Vol. 4, Nº7.

Rinder, M. M., 2012. "An Integrated Decision-Support Tool to Forecast and Schedule No-Show Appointments in Healthcare", Dissertation, Russ College of engineering and Technology.

Neal, R. D., Hussain-Gambles, M., Allgar, V. L., Lawlor, D. A., and Dempsey, O., 2005. "Reasons for and consequences of missed appointments in general practice in the UK: questionnaire survey and prospective review of medical records.".

Leong, K., Chen, W., Leong, K., Mastura, I., Mimi, O., Sheikh, M., Zailinawati, A., Ng, C., Phua, K. and Teng, C., 2006. "The use of text messaging to improve attendance in primary care: a randomized controlled trial".

Liew, S.-M., Tong, S., Lee, V., Ng, C., Leong, K. and Teng, C., 2009. "Text messaging reminders to reduce nonattendance in chronic disease follow-up: a clinical trial".

Daggy, J., Lawley, M., Willis, D., Thayer, D., Suelzer, C., DeLaurentis, P.-C., Turkcan, A., Chakraborty, S. and Sands, L., 2010. "Using no-show modeling to improve clinic performance".

George, A. and Rubin, G., 2003: "Non-attendance in general practice: a systematic review and its implications for access to primary health care".

Huang, Y. and Hanauer, D., 2014. "Patient no-show predictive model development using multiple data sources for an effective overbooking approach", Applied clinical informatics, vol. 5, pp. 836–60.

Dantas, L., Hamacher, S., Oliveira, F., Barbosa, S., Viegas, F., 2018. "Predicting Patient No-show Behavior: a Study in a Bariatric Clinic," Obesity Surgery.

LaGanga LR, Lawrence SR, 2007. "Clinic overbooking to improve patient access and increase provider productivity," Decis Sci. 38(2): 251–76.

Lee, V.J., Earnest, A., Chen, M.I., et al, 2005. "Predictors of failed attendance in a multi-specialty outpatient centre using electronic databases," BMC Health Serv Res. 5(1):51.

Bennett, K.J., Baxley, E.G, 2009. "The effect of a carve-out advanced access scheduling system on no-show rates," Fam Med. ;41(1):51–6.

Norris, J.B., Kumar, C., Chand, S., et al, 2014. "An empirical investigation into factors affecting patient cancellations and no-shows at outpatient clinics," Decis Support Syst. 57:428–43.