International Journal of Advanced Nursing Education and Research
Vol.5, No.3 (2020), pp.1-14
http://dx.doi.org/10.21742/ijaner.2020.5.3.01

Predicting Hospital Re-admissions from Clinical Narratives

Pankti Joshi[1] and Sabah Mohammed[2]

[1,2]Department of Computer Science, Lakehead University, Thunder Bay, Ontario, Canada
[1]pjoshi@lakeheadu.ca, [2]mohammed@lakeheadu.ca

Abstract

In this era, hospital re-admissions have been a significant concern as the numbers of re-admissions are increasing at an alarming rate worldwide. The central idea of this paper is to predict unplanned patient re-admissions within 30 days of discharge. When a patient is admitted to a healthcare center, there are high chances of re-admissions based on many healthcare parameters. This paper proposes a Machine Learning-based K-Nearest Neighbor model to predict 30-day unplanned hospital re-admission using clinical notes. The extracted dataset will undergo various text pre-processing stages to improve the model's overall accuracy. To validate our proposed model, we have implemented many other Machine Learning models to compare different parameters obtained from each model. Hyperparameter tuning techniques and feature extraction techniques have been implemented to study the prediction results. According to our observations, the K-Nearest Neighbor model got the best accuracy of 85 percent, while logistic regression did not provide high accuracy. In this way, clinicians can intervene in patients' conditions beforehand, predict possible re-admission chances, and take various precautionary treatment steps to avoid unplanned re-admissions.

Keywords: *Hospital re-admission, Machine learning, Prediction system, Text cleaning*

1. Introduction

Digitizing personal medical history aims to improve healthcare quality by analyzing medical data and providing patients with the best possible treatment. Healthcare data is so enormous and complex today that it can be used in predictive models to improve healthcare delivery. Unfortunately, most information is inappropriate or suitable for our model predicting analysis. This is because the data is usually mishandled, or clinicians are restricted from sharing data due to the health sector's privacy or integrity norms. If the information is utilized efficiently, scientists predict that it would effectively help improve health care and provide patient safety and quality, reducing the cost of health care all over the globe. Predicting a patient's re-admission in the hospital is a significant healthcare data application in predictive analytical modeling. Hospital re-admission is a patient's admission within 30 days of initial discharge, irrespective of any hospital. Multiple factors are essential in hospital re-admissions, like premature discharge, insufficient care during initial admission, or unsuccessful treatment. The sole purpose of this paper is to build a predictive classification model from clinical narratives to classify which patients are likely to be re-admitted within 30

Article history:
Received (June 29, 2020), Review Result (September 4, 2020), Accepted (October 13, 2020)

days. Predicting unplanned re-admissions of a hospitalized patient can greatly help as it saves many medical and financial resources. It is advantageous to patients and medical staff; hence, it reduces the patient's exposure to the risk and well-being of their physical and mental status. The amalgamation of Natural Language Processing and Machine Learning techniques provides vast solutions to identify patterns in complex, enormous, multi-dimensional datasets. Thus, it offers opportunities to develop an efficient decision-making support system for doctors while discharging patients. According to the Canadian Institute of Health Information, re-admission to hospitals costs more than 2.1 billion dollars annually in Canada. As statistics illustrate, 1 in 11 patients has been re-admittedCtoada in the [1]. To combat and reduce the patient’s unplanned re-admission issue, the Hospital Re-admissions Reduction Program was created under the Affordable Care Act (ACA) to penalize the hospitals whose 30-day re-admission rates are higher than expected. According to the Centers for Medicare Services (CMS) statistics, after the program began, penalties worth 2.5 billion dollars were imposed on hospitals for re-admissions, including 564 million dollars in 2018. Machine Learning has been tremendously used in multiple application areas, like education, health, finance, communications, etc. This is a rapidly growing research area, and cross-validation techniques like Grid Search and Random Search have been implemented to optimize the performance of our model.

This study aims to analyze the model's predictive performance using different Machine Learning algorithms with the MIMIC-III (Medical Information Mart for Intensive Care III) database. MIMIC-III is a massive dataset widely used to derive predictive models for hospital re-admissions that hospitals can effectively implement. Section II describes the MIMIC-III dataset. Section III narrates the related work. Section IV discusses the design and methodology for developing the models and various Machine Learning algorithms. Section V discusses the results and implications. Section VI concludes the work.

2. The dataset

The predictive model is built using the MIMIC-III (Medical Information Mart for Intensive Care III) database [2], which is an enormous and accessible database for hospital records containing identified data from about 50,000 patients admitted to Beth Israel Deaconess Medical Center in Boston, Massachusetts, from 2001 to 2012. This is an open-access, relational database of patients who stayed in the ICU at the hospital. This database contains necessary information such as laboratory test results, medications, demographics, procedures, caregiver notes, imaging reports, fluid balance, procedure codes, diagnostic codes, imaging reports, hospital length of stay, and mortality (including post-hospital discharge). The work is done on two MIMIC datasets to build the predictive model. The ADMISSIONS table contains the necessary admission and discharge dates for prediction. Secondly, the note events table enlisting discharge summaries and notes specifying essential information regarding a patient's stay in the hospital. Both tables have HADMID as the unique primary key for each admission. To access this project's data, you must request access at the Physio net access link [3]. Due to the sensitive nature of medical data and privacy permission, this paper has not publicly included raw data.

3. Related research

In the study [4], "Analysis and prediction of unplanned intensive care unit re-admission using recurrent neural networks with long short-term memory," the author focuses on the unplanned re-admission prediction using chart events, demographics, and ICD-9 embedding features. This paper proposed an LSTM-CNN-based model to incorporate time-series data without losing any information. The proposed study successfully achieved an AUC score of 0.791 and a sensitivity of 0.742. These results were way better than the existing algorithms for predicting hospital re-admissions. However, since the model can have multiple operating points, its sensitivity and specificity can be tuned to match specific clinical setting requirements, such as high sensitivity for critical care. The proposed model mainly focused on input feature selection in the predictive model. The comparison was made with the logistic regression, where the author successfully demonstrated that the LSTM and CNN model proved to be accurate compared to that of the Logistic Regression model. Thus, the study proposes properly using supervised machine learning algorithms to predict hospital re-admissions. Various scenarios with varied combinations of Machine Learning algorithms have been implemented to achieve optimum results and better accuracy. The study depicts an exemplary implementation of the feature selection methodology. However, the model can be improved by cooperating with natural language processing tools for text. Clinical Narratives like TF-IDF, Sentence Tokenization, Word Tokenization, Text Lemmatization, Stemming, Stop Words, Regex, and Bag-of-Words.

In the paper [5], "Hospital re-admission is highly predictable from deep learning," the author uses a supervised learning approach to predict re-admission. The author uses ten-fold cross-validation techniques to check the accuracy of the predictive model. Five algorithms are compared- Logistic regression, Naive Bayes, Decision tree, Random forest, and Deep Learning. The output parameter used for accuracy comparison is the AUC Score. After the score comparison, the author concluded that logistic regression provided relevant results, but all the attributes affecting re-admission could not be used for score derivation. Machine learning algorithms gave better prediction results while using more information. Deep Learning and Random Forest provided the best prediction results.

However, a few loopholes were considered, which come with the dataset model. First, deaths outside the hospitals were not considered after discharge. Secondly, hospitals did not validate the Quebec dataset externally for future analysis. After that, the dataset did not include hospital-specific information, which might not give good analytical results. However, the author firmly states that it would be easier for clinicians to decide with such estimations while discharging them. Indeed, it would be beneficial to set up a system that will predict the patients' unplanned re-admission and provide an analytical report. Such a system would help improve health care as such information would help prepare patients prone to being re-admitted and help treat them accordingly. The doctors can also take aftercare of the patient's condition and ensure the patient is doing well outside the hospital. The main advantage is that this system can reduce many medical costs, and it is not expensive to deploy.

In the case study of CPD [6], “Predictive Modeling of the Hospital re-admission Risk from Patients' Claims Data Using Machine Learning," the author focuses on patients affected by Chronic Obstructive Pulmonary Disease (COPD) and conducts a systematic study on developing various types of predictive Machine Learning algorithms to predict the risk of unplanned re-admission of COPD patients. The author has considered a large real-world dataset from the Geisinger Health System containing medical claims of 111992 patients from January 2004 to September 2015. The machine learning models are built using the patient's features, from transfer learning, when features are extracted from clinical knowledge regarding unplanned COPD re-admission. This model is also called transfer learning. The author also tries to implement data-driven features extracted from the patient data. As a result of analysis based on one year of claims history before discharge, the AUC score by combining the feature-driven and data-driven features is 0.653 compared to 0.60 using knowledge-driven features. The article's prediction performance is based on the Area Under the receiver operating characteristic (ROC) Curve (AUC). The researcher also implemented deep learning models and got an AUC score 0.65; thus, deep learning barely improves performance. The author concludes by verifying the importance of medical knowledge in the predictive modeling process and demands better patient data.

The study [7], "Machine Learning-based Risk of Hospital Re-admissions: Predicting Acute Re-admissions within 30 Days of Discharge," proposed a predictive model using machine learning algorithms. The research's primary purpose was to predict the unplanned re-admission of patients to the hospitals within 30 days of discharge. Several base learner model types were used, such as Random Forest, XG boost, and Ad boost, to build the predictive model. This model was used after sufficient pre-processing and feature selection steps. LACE index and patient at-risk of hospital re-admission (PARR) models were used to validate the proposed predictive model. The significant attributes used for the predictive model include the length of stay, emergency department visits in the last six months, understanding of admission and age, social support, living situation, socioeconomic status, insurance, and behavioral and mental health data. The proposed model achieved an AUC score of 0.75. The author obtained a higher F1- score than the LACE index and PARR models by 12.5 percent and 30 percent, respectively. The proposed model's mean sensitivity was 6.0 percent higher than LACE and 42 percent higher than PARR. The proposed model obtained an F1-score of 0.386, a sensitivity of 0.598, a positive predictive value (PPV) of 0.285, and a negative predictive value (NPV) of 0.932.

4. Methodology

This section elaborates on various steps executed to get optimum prediction results. The figures below show all the significant measures implemented in the research [Figure 1]. It depicts how to prepare data by reducing dimension, removing noise and null values, changing data formats, and merging datasets for other pre-processing text. [Figure 1］. Significant steps were implemented in this research to develop a machine-learning model and prediction algorithms.

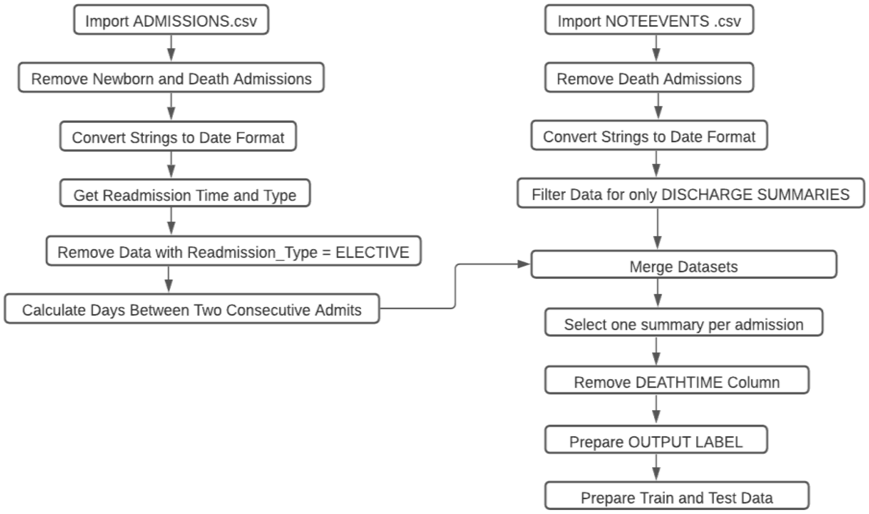
Import ADMISSIONS.csv
Remove Newborn and Death Admissions
Convert Strings to Date Format
Get Readmission Time and Type
Remove Data with Readmission_Type = ELECTIVE
Calculate Days Between Two Consecutive Admits
Import NOTEEVENTS .csv
Remove Death Admissions
Convert Strings to Date Format
Filter Data for only DISCHARGE SUMMARIES
Merge Datasets
Select one summary per admission
Remove DEATHTIME Column
Prepare OUTPUT LABEL
Prepare Train and Test Data

Figure 1. Prepare data

4.1. Prepare data

As mentioned above, we will use admissions and note events datasets from the MIMIC-III database [3]. Thus, both datasets must be pre-processed before they can be merged for predictive analysis. Primarily, we will explore the admissions table. The admissions table has six significant attributes: subjected, Hamid, admittee, disc time, deathtime, and admission type. Here, the subject represents a unique identifier for each subject, and Hamid represents each hospitalization's unique identifier. The admittee, disc time, and deathtime represent the admission date, discharge date, and death time (if it exists), respectively, with yyyy-mm-dd hh:mm: ss string format. The admission type is the primary type of admission, which can be elective, emergency, newborn, or urgent. Preprocessing stages were applied to the admissions dataset. Firstly, the entries with admission type as newborn and deathtime columns were removed. This is because our model predicts the chances of unplanned re-admissions within 30 days of discharge. There are obvious chances that patients with newborns and admissions with hospital deaths would never be re-admitted. Secondly, the date's string format is converted to a date-time format to pre-process the admissions dataset. For the empty dates, the errors = 'coerce' flag will depict the null values for missing dates. Lastly, we must filter out each subject separately to filter out the number of admissions of a particular SUBJECTED. This task is more comfortable if we sort the data using subject and admitted. We will use the "group by" and "shift" operators to get the next admission date and time (if it exists) of each subject and the subsequent admission type. After executing the group by and shift (-1) operator on the admittee and subjected, two new columns would be called the next admittee and the following admission type.

In the following admission, if the type is elective, then it is not considered an unplanned re-admission. Thus, we can drop the following admission type=elective entries. Check the values by sorting the data again based on subject and admittee and backfilling the weights so that the entries are not null for further steps. We can calculate the difference between re-admission dates, which can tell us exactly how many days the patient was re-admitted based on the pre-processed admissions dataset. Secondly, we can explore the note events dataset for pre-processing purposes. While working with the dataset, the significant rows worth considering include the subject, Hamid, and chart data depicting the date the note was charted. Chart date will always have a time value of 00:00:00. The category column represents the type of message recorded, such as nursing/other, radiology, nursing, e.g., physician, discharge summary, echo, respiratory, nutrition, general, rehab services, social work, case management, pharmacy and consult. The attribute text value contains the note text provided by Clinicians. Usually, there can be multiple notes per admission. It contains necessary patient information like name, doctor, location, and dates. This data has been successfully encrypted to maintain the confidentiality of the notes. For the data pre-processing in the note events table, we must remove the death entries by comparing Hamid's with those in the admissions table. If Hamid is not in the admissions table, we can disregard the entry for the note events table. We can now convert the chart date and time into the date-time format from the string. Thereby, we will sort the entries based on subject, chart date, and chart time. We will filter the data that are categorized as Discharge Summary. As mentioned, there can be more than one note per admission; we will select only one summary for entries. After this step, we can merge the datasets using the primary key "subjected" and "Hamid." The following section will briefly describe basic Natural Language Processing techniques used on the combined dataset.

4.2. Data Preprocessing

The problem of predicting re-admissions of patients from the clinical narratives includes studying the physicians' notes. The system must be provided with clinical narratives in a similar format. It is also essential that the unnecessary words or items in the sentences be removed to give the system a clear idea for easy and faster processing. This would also make the system learn quicker, and the prediction output is much more accurate and reliable. It is essential to carry out data pre-processing steps using NLP concepts in the text. This section consists of all the critical ideas implemented to prepare the NOTES field for the predictive model. We used the NLTK (Natural Language Toolkit) libraries in Python [8] to understand human language data. It is beneficial as it contains many text processing in-built libraries that help classify, tokenize, Lemmatization, POS tagging, parsing, and semantic reasoning.

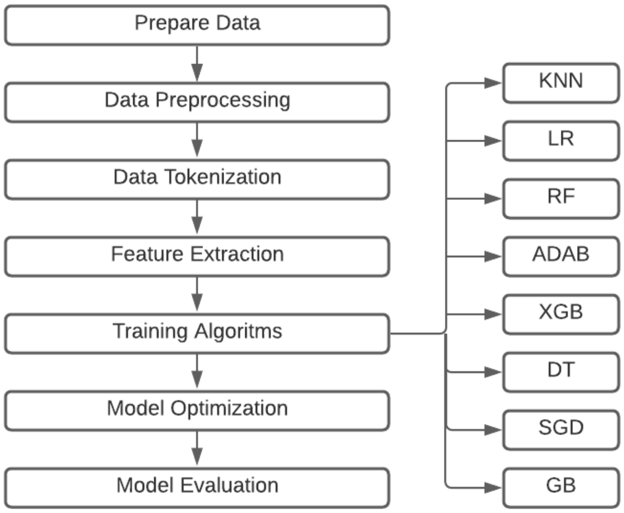

Figure 2. Flowchart of Methodology

In the pre-processing pipeline, the following steps are performed: Lowercase Text Conversion: This is done to solve the improper capitalization of texts. All the text in the notes field is converted to lowercase to ensure it does not misunderstand the input text. Punctuation removal: This step ensures that the data is free of punctuation and provides a simple input. Numerical Data Removal: Removing numbers would ensure that input text is a simple format that needs to be analyzed to predict the re-admission based on clinical notes. Spaces can replace the numbers while pre-processing.

Stop word removal: This is a crucial step when the stop words such as are identified and removed. These words are meaningless and add a lot of noise while processing the data in the system. There is no confirmed list of the stop words, and they must be removed as they do not modify or change the meaning of sentences.

4.3. Tokenization

Tokenization is splitting longer texts into small fragments in sentences or words for a better readability score. We have used Regex Tokenizer, which extracts tokens using the provided regex pattern to split the text (default) or repeatedly match the regex (if gaps are false). Regex Tokenization removes all the non-words characters in the Clinical Narratives. It can be efficiently removed using regex. Python provides a "re" module to replace regular expressions with the space character or replacement string. This is additional filtering and is optional to use. This step would make the input text more accessible to understand and evaluate.

4.4. Lemmatization

This normalization technique is when text is converted into a meaningful root format. Using lemmatization reduces the feature space, which makes the model more accurate.

4.5. Feature extraction

It is a dimensional reduction technique that reduces the text to a more understandable format. This reduces computation time, and the execution process is much faster and more efficient. We have used two primary feature extraction techniques - count vectorizer and Term Frequency- Inverse Document Frequency (TF-IDF). Both methods work similarly. These techniques quantify the words in the document, which helps to weigh the importance of words in the corpus. A count vectorizer counts the occurrence of words and depicts them in vector format. It provides the count in "int" format. In TF-IDF, Term Frequency summarizes how often a word is used within a document, and Inverse Document Frequency scales words with many repetitions. TF-IDF would return a score in float format. Once these features are extracted using a count vectorizer or TF-IDF from the dataset, we can only consider extracted features for further analysis. Thus, we can drop the text column from the dataset. This would efficiently reduce the complexity of the clinical notes and save computation time, power, and space.

4.6. Bag-Of-Words (BOW) technique

Machine Learning algorithms do not interpret raw text format directly. Thus, it is essential to convert the texts into vectors of numbers. This is also known as feature extraction. BOW is a simple feature extraction technique that works with a group of texts. It will count the occurrences of words in the document and represent text data in numerical form. Eventually, this data is fed into the respective Machine learning model. We need to design a vocabulary of known words or tokens to implement the bow approach. Then, choose the measure of the presence of new words. The bow does not bother about the order or structure in which the terms are arranged. Thus, the name Bag-Of-Words. This step’s main motive is to understand if any known words occur in a document and learn more about its content.

4.7. Machine learning algorithms

This research includes the implementation of several Predictive algorithms to evaluate the accuracy of the output model. The prediction of unplanned re-admissions is a classification problem. Thus, various classification algorithms that are implemented are briefed in the section below.

(1) K-Nearest Neighbor (KNN)

KNN is a simple, easy-to-implement, and efficient supervised classification machine learning algorithm. It can be used to solve classification as well as regression problems. The basic idea of KNN is that the similar items are nearer to each other. The KNN classifier intends to classify unlabeled observations by assigning them to the data class that is most similar to the labeled data. The characteristics of observations are collected for both training and test data. The appropriate choice of K significantly impacts the evaluation performance of the KNN algorithm. A large K decreases the effects of variance caused by random error but runs the risk of ignoring a small but essential pattern [9].

(2) Logistic Regression (LR)

This algorithm is used to predict the probability of a categorical dependent variable. A dependent variable is a binary variable that contains data coded as 1 (positive) or 0 (negative). LR uses the Sigmoid function, which is a complex cost function. The hypothesis used in LR limits the cost function between 0 and 1. Therefore, linear functions fail to represent it as they can have a value greater than one or less than 0, which is impossible per LR's hypothesis. It is classified into binomial, multinomial and ordinal categories [10].

(3) Random Forest classifier (RF)

RF classifier is an Ensemble learning technique that consists of a large number of individual decision trees. Each tree in the random forest spits out a class prediction, and the class with the most votes becomes the model's prediction. Many relatively uncorrelated models (trees) operating as the committee will outperform any of the individual constituents' models [11].

(4) Adaptive boosting algorithm (ADA)

The AdaBoost algorithm creates a set of poor learners by maintaining weights over training data and adaptively adjusting them after each weak learning cycle. The weights of the training samples, which the current weak learner misclassifies, will be increased while the weights of the correctly classified samples will be decreased. AdaBoost is best used to improve decision trees' performance based on binary classification problems [12].

(5) Decision Tree (DT)

DT algorithm is non-parametric and can efficiently deal with large, complicated datasets [13]. It classifies data items by posing questions about their associated features. Each question is contained in a node, and every internal node points to one child node for each possible answer to its problem. The questions form a hierarchy encoded as a tree structure [14]. When the sample size is large enough, study data can be divided into training and validation datasets.

(6) Gradient Boosting Classifier (GBC)

GBC works on the algorithm to construct new base learners to be maximally correlated with the loss function's negative gradient associated with the whole ensemble. The learning procedure consecutively fits new models to provide a more accurate estimate of the response variable. The choice of the loss function is up to the researcher, with various loss functions derived so far and the possibility of implementing one's task-specific loss [15].

(7) Stochastic Gradient Descent Classifier (SGDC)

SGDC implements regularized linear models with Stochastic Gradient Descent. Unlike gradient descent, SGDC considers only random points while changing weights, which assumes all training data. SGDC is much faster than gradient descent when dealing with large data sets. LR, by default, uses SGDC, but LR needs a dataset on RAM to work with it. In contrast, SGDC does not require a dataset on RAM to work with it.

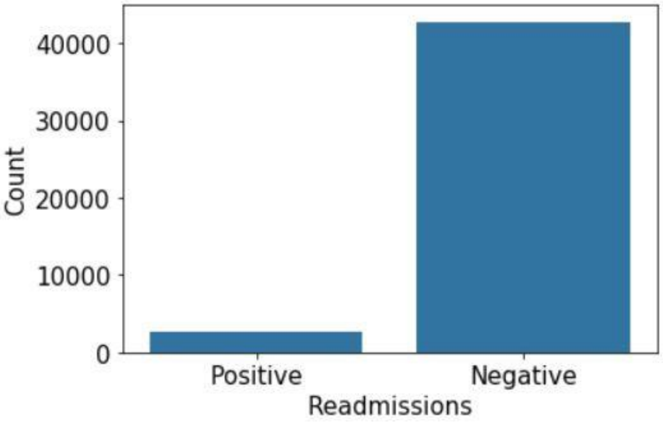

Figure 3. Data imbalance

(8) Extreme Gradient Boosting (XGB)

XGBoost is a decision-tree-based ensemble Machine Learning algorithm that uses a GB framework. Decision tree-based algorithms provide good results in prediction problems involving small-to-medium structured data. XGB is scalable, which drives fast learning through parallel and distributed computing and efficiently uses memory [16].

4.8. Training and testing model

Before splitting data into tests and training, we must check if the dataset has a class imbalance. [Figure 3] provides data distribution for positive and negative samples. It is visible that the total number of unplanned re-admissions within 30 days, i.e., positive models, is less than negative samples. Thus, this is a clear case of class imbalance. This could create over-fitting while executing the model. This problem is solved by downsampling the negative possibilities, such that the number of positive and negative samples is almost equal in training data. Overall, after downsampling, the data, totally positive and negative models, were 1810 and 1805, respectively. After reshuffling and merging the dataset, it is broken down into training and testing data in the ratio of 70:30. Once the split is done and the algorithm is built, the data must undergo training to perform supervised learning. The algorithm uses trained data to learn features from the dataset. Later, testing is performed to evaluate our built model's performance using performance metrics like accuracy, precision, and f1 score. This is also called a confusion matrix. The evaluation metrics are also represented using a classification report.

4.9. Model optimization

Once the base model is executed in any Machine Learning algorithm, several ways exist to optimize the model and increase the base accuracy. I have used the Random Search Cross-Validation and Grid Search Cross-Validation Techniques for hyper-tuning estimators' parameters. The performance of both methods has been found to be consistent with all the models executed. Grid search CV evaluates the algorithm using all possible combinations of specified hyper-parameters. Only the best variety of parameters is retained, and optimum accuracy is provided. Random search CV is a technique where random combinations of the hyperparameters are used to find the built model's optimum result.

Copyright © 2020 Global Vision Press (GV Press)

4.10. Model of analysis

The predictive models can be successfully evaluated based on the parameters such as Accuracy, Precision, and F1-score. Accuracy is a critical metric for the evaluation classification model's performance. It represents the correctly predicted output while executing a model. True Positives and True negatives are crucial elements to evaluate the accuracy of a model. The precision represents the portions of the identifications that are correct. The F1-Score is the harmonic mean of precision and recall value. F1-score false positives and false negatives are crucial while evaluating the F1-score. For classification problems where data is imbalanced, the F1-score is a better metric to evaluate a model.

5. Results

We have analyzed the MIMIC III dataset to predict hospital re-admissions within 30 days of discharge. While executing several models, the results provide broad insights into how our clinical data interact with different Machine Learning Algorithms. Our research started by extracting the dataset from the Physio net and preparing data. After that, data was pre-processed. We have also performed the Feature Extraction Techniques - TF-IDF and Count Vectorizer for better comparative results. Both methods provided similar accuracy for almost all the Machine Learning algorithms. The features have importance as they contain valuable characteristics that define the dataset. As this is a supervised classification problem, labels were provided for data for further analysis.

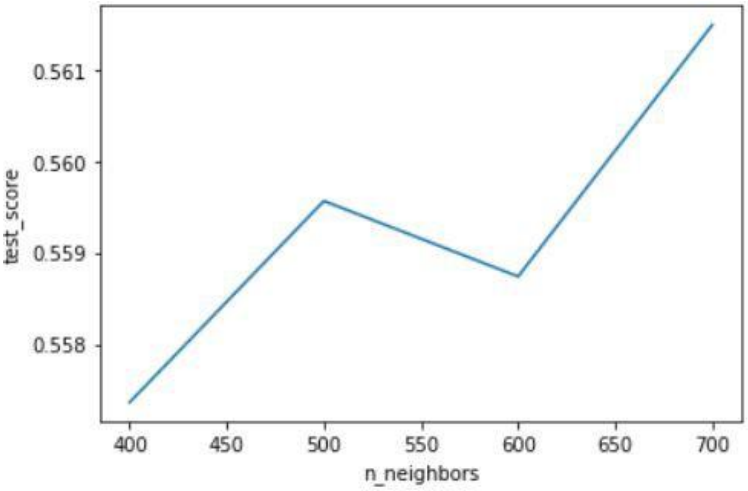

Figure 4. Graph plotting KNN score vs. Neighbors

The first algorithm performed on the dataset for analysis was Logistic Regression. All the previous researchers used LR as their base model to predict re-admission accuracy. It is considered a baseline model for any machine learning analysis as it works well with large datasets and reduces the chances of overfitting.

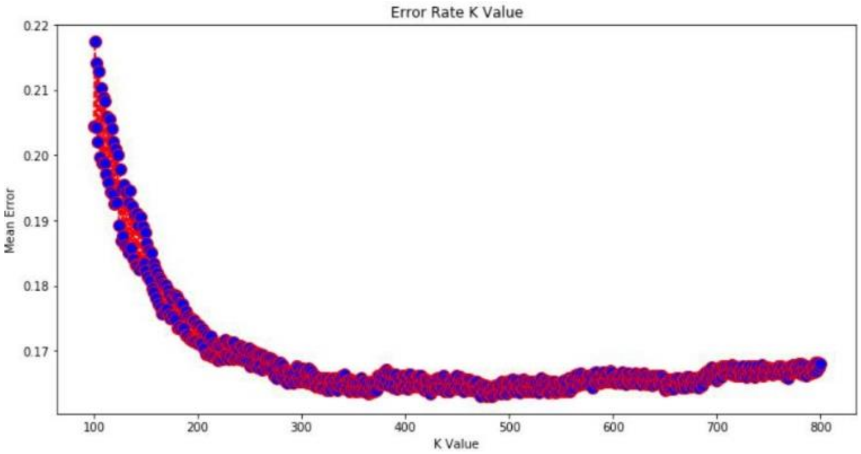

Figure 5. Graph plotting KNN Score vs Neighbors

However, LR could have performed better in accuracy with our dataset. LR algorithm achieved an accuracy of 66 percent on the base model and 69 percent accuracy on the optimized Grid Search Model. However, the KNN model gave an accuracy of 85 percent for both the base model and the Grid Search Optimized model when performed with Count Vectorizer. [Figure 4] depicts a graph showing the expected test score concerning the number of neighbors in the KNN model. It can be observed the test scores increase as the number of neighbors increases. In the graph, for n = 700, the maximum test score is attained. [Figure 5] shows the error rate vs. K value while executing the KNN model. [Table 1] illustrates the evaluation metrics of our research. It can be concluded from the diagram that for K values between 300 and 600, there is the most minor mean score error. Thus, these can be the most preferred K values for KNN model execution.

Table 1. Model evaluation (in the form of a percentage)

Algorithms	Accuracy	Precision	Fl-score
KNN	85	91	87
LR	68	92	74
RF	64	92	74
AD AB	68	92	74
XGB	64	92	74
DT	72	92	80
SGDC	73	92	80
GB	62	92	72

The accuracy of all the base models executed in the research work is shown in a bar chart in [Figure 6]. It can be observed that the GB model achieves the lowest accuracy of 62 percent, followed by XGB and RF models, which attain a similar accuracy of 64 percent. ADAB and LR showcase the same accuracy of 68 percent. DT and SGDC show a case with almost equal accuracy of 72 and 73, respectively. KNN provides the best accuracy among all the machine learning models at 85 percent. [Figure 7]. Provides a brief idea about the base as well as optimized accuracy scores of all the machine learning classification models implemented.

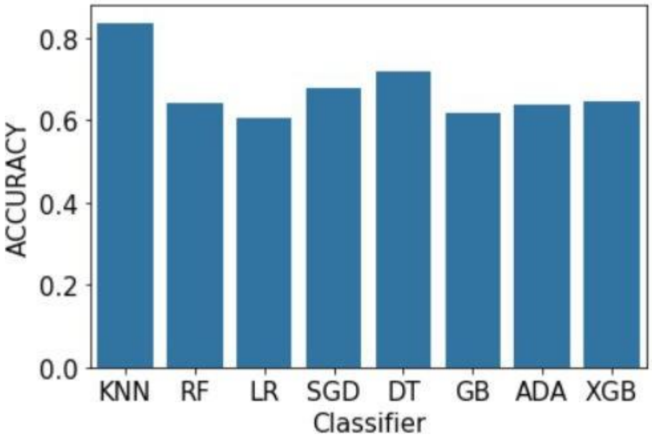

Figure 6. Comparison of Base Models

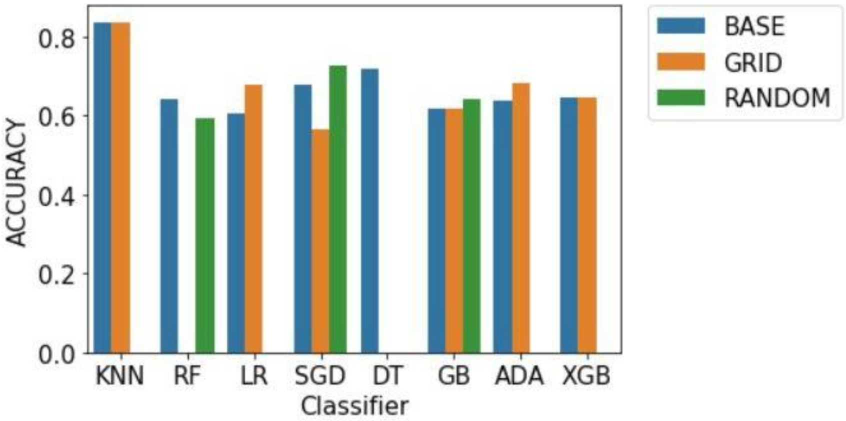

Figure 7. Comparison of Executed Models with the optimized version

6. Conclusions

Hospital re-admissions are expensive for both patients and the health care system. Many countries are sincerely working to reduce the chances of unplanned re-admissions. To tackle this issue, countries worldwide spend a lot of manpower on research and development in this sector. Governments are imposing strict fines and regulations on hospitals whose levels of re-admissions are higher than the permitted level. This increases the need to predict unplanned hospital re-admissions to multi-folds. Most of the research out there needs to provide better accuracy while predicting the re-admissions or only manual systems. This is tiresome and inefficient work in the modern world and the chances of error increase significantly. However, we are dealing with human beings and their medical welfare in the health industry, increasing the need for integrity and precision.

Thus, research into analyzing clinical notes has become crucial to building a robust and highly efficient model. This research has attempted to study the clinical notes and prediction re-admissions rate on the MIMIC-III database. All the text pre-processing steps were followed thoroughly, and features were processed to extract essential characteristics. Eight types of Machine Learning algorithms were analyzed using multiple evaluation metrics.

Hyperparameter tuning helped to increase the prediction accuracy of our base models. For this predictive analysis, we can conclude that KNN outperformed all the other machine learning models, acquiring an accuracy of 85 percent. It has been concluded that KNN provides better accuracy than other classification models due to its smaller number of features. Predicting hospital re-admissions is an actual application of Natural Language Processing, Machine Learning, and Deep Learning techniques in the medical field. It is beneficial for both patients and medical staff. This would also cut down health expenses incurred by governments worldwide in the form of health insurance. With the availability of accurate information, computing power, memory, and high-technology devices, these kinds of predictions would be easier to implement in the future. Multiple government agencies, health care departments, and private companies intend to build stable, robust, more accurate, and reliable predictive algorithms for unplanned re-admissions predictions.

Acknowledgments

This research is part of the first author's MSc Project.

References

[1] Canadian Institute for Health Information, "All patients re-admitted-admitted to the hospital," Oct., **(2020)**

[2] Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R., Stanley, and H. E., “PhysioBank, physio toolkit, and physionet: Components of a new research resource for complex physiologic signals,” Circulation [Online], vol.101, no.23, pp.215-220, **(2000)**

[3] Machine Learning Crash Course, Available: https://developers.google.com/machine-learning/crash-course, Oct., **(2020)**

[4] Z. Yu and W. B. Rouse, "A deeper look at the causes of hospital re-admissions," IEEE International Conference on Industrial Engineering and Engineering Management (IEEM), Singapore, pp.919-923, (2017) DOI: 10.1109/IEEM.2017.8290026

[5] E' chevin, Damien, Qing Li, and Marc-Andre' Morin, "Hospital re-admission is highly predictable from deep learning," Chairede Recherche Industrials Alliances Licensees Economicus Changesets De'mographiques, **(2017)**

[6] Min X., Yu B., and Wang F., "Predictive modeling of the hospital re-admission risk from patients' claims data using machine learning: A case study on COPD," Sci Rep, vol.9, no.1, 20 Feb., (2019) DOI: 10.1038/s41598-019- 39071-y, PMID: 30787351, PMCID: PMC6382784

[7] M. M. Baig et al., "Machine learning-based risk of hospital re-admissions: Predicting acute re-admissions within 30 days of discharge," 2019 41st Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC), Berlin, Germany, pp.2178-2181, (2019) DOI: 10.1109/EMBC.2019.8856646

[8] Rajkomar, Alvin, Eyal Oren, Kai Chen, Andrew M. Dai, Nissan Hajaj, Michaela Hardt, and Peter J. Liu et al, “Scalable and accurate deep learning with electronic health records.” NPJ Digital Medicine, vol.1, no.1, pp.18, **(2018)**

[9] Zhang and Zhongheng, “Introduction to machine learning: K-nearest neighbors,” Annals of Translational Medicine, vol.4, no.11, **(2016)**

[10] O. Aborisade and M. Anwar, “Classification for authorship of tweets by comparing logistic regression and naive Bayes classifiers," 2018 IEEE International Conference on Information Reuse and Integration (IRI), Salt Lake City, UT, pp.269-276, **(2018)**

[11] Tin Kam Ho, “Random decision forests,” Proceedings of the International Conference on Document Analysis and Recognition, ICDAR, pp.278-282

[12] Wang, Ruihu, “AdaBoost for feature selection, classification and its relation with SVM: A review,” Physics Procedia, vol.25, pp.800-807, **(2012)**

[13] Kingsford, Carl, and Steven L. Salzberg, "What are decision trees?" Nature Biotechnology, vol.26, no.9, pp.1011-1013, **(2018)**

[14] Song, Yan-Yan, and L. U. Ying, “Decision tree methods: Applications for classification and prediction,” Shanghai Archives of Psychiatry, vol.27, no.2, pp.130, **(2015)**

[15] Natekin, Alexey, and Alois Knoll, “Gradient boosting machines: A tutorial,” Frontiers in Neurorobotics, vol.7, no.21, (2013) DOI: 10.3389/fnbot.2013.00021

[16] Chen, Tianqi, and Carlos Guestrin, “Xgboost: A scalable tree boosting system.” In Proceedings of the 22nd Acmsigkdd International Conference on Knowledge Discovery and Data Mining, pp.785-794, **(2016)**