Imbalanced Learning for Hospital Readmission Prediction using National Readmission Database

Shuwen Wang, Magdalyn E. Elkin, and Xingquan Zhu
Department of Computer& Electrical Engineering and Computer Science, Florida Atlantic University
Boca Raton, FL33431, USA
{swang2020, melkin2017, xzhu3}@fau.edu

***Abstract*—In this paper, we propose to use imbalanced learning for hospital readmission prediction. The goal is to predict whether a patient, based on his/her current hospital visit records, is likely going to be re-admitted or not within 30-days after being discharged from the current hospital visit. The main challenge of hospital readmission prediction is twofold: (1) the readmission visits (i.e., the positive class) are a small portion of the total hospital visits, representing a severe class imbalance problem for learning; (2) due to privacy and health regulation, the information available for patient characterization is limited; and is often only limited to the payment level information. However, there are over 80,000 procedures code, representing a high dimensionality and high sparsity problem for learning. Motivated by the above challenges, in this paper, we design an imbalanced learning strategy to create features from patient hospital visit, by combining patient demographic information, ICD-10 clinical modification (CM) and procedure codes (PCS), and Clinical Classification Software Refined (CCSR) conversion. Instead of directly using ICD-10- CM/PCS code to characterize patients, we convert each patient's visit to CCSR code space with a smaller feature space. By using random sampling approach to balance the sample distributions in the training set, our method achieves good performance to predict patient readmission.**

***Keywords*—Hospital readmission, national readmission database (NRD), clinical classification software refined (CCSR), classification, imbalanced learning.**

I. Introduction

A hospital readmission is defined as an admission where a patient previously discharged from a hospital is being admitted to the same or a different hospital, within a specific time interval such as 30 days or 90 days. The reasons behind a hospital readmission often differ from patient to patient [1] and the readmission rates between different medical institutions also vary significantly [2]. A readmission implies extra costs to the stakeholders, adds financial burden to the patients and deteriorates their life quality [3], [4]. Hospital readmissions are also related to unsatisfying patient outcomes and heavy financial burden to the healthcare system [5]–[7]. Preventable readmissions can lead to almost $17 billion annual cost reduction [8]. Therefore, in 2012, a national Hospital Readmissions Reduction Program (HRRP) initiative started to link the health care payment to the quality of hospital care, by reducing payments to hospitals with excess readmissions and providing hospitals an incentive to improve their care coordination in post-discharge planning. HRRP is established to penalize hospitals with readmission rates exceeding the national average by a drop in their payments. It is expected that by implementing such a penalization, an improvement in post-discharge communication and care to patients can be implemented by hospitals and a reduction in readmissions can be expected [10].

Since 2012, many efforts have been taken, by hospitals, caregivers, and academics [21], [22], to reduce readmission. But unfortunately, after eight years, it is observed that the "needle has not moved very far" [9]. In 2019, Medicare, under the HRRP plan, cut payments to 2,853 hospitals. Among the 3,129 general hospitals which were evaluated in the HRRP program, 83% of them received a penalty [9].

The reduction of hospital readmission rate is of great significance to Medicare system and the effective usage of health care resources. It is meaningful and important to predict preventable hospital readmission earlier than it really happens. Intuitively, this problem is equivalent to predicting the likelihood of a patient being readmitted again in the defined time-frame, using patient's current information, including demographics, diagnose, treatment, *etc.*

Machine learning is a method for data analysis and recently there have been many predictive models built through machine learning methods to perform hospital readmissions prediction and provide corresponding results and suggestions [11]–[13]. Logistic Regression is a popular model in medical prediction fields [14]. In addition, many researches also proposed to use a variety of prediction models, such as support vector machines [15] and neural networks [16], for readmission analysis. Despite of the growing number of models built for this area, many of them cannot be applied to clinical practice, because of special coding and payment protocol/compliance requirements in the medical domains. On one hand, majority of existing models are focused on specific diseases [17], [18] and databases are obtained from regional hospitals, which may be biased towards the local populations and disease types. Meanwhile, due to HIPAA regulations, these methods often cannot share their data. As a result, when trying to apply these models to complex and more comprehensive real-world settings, the prediction results are often unsatisfactory.

In order to promote research and analysis of national readmission rates for all patients, a Federal-State-Industry partnership sponsored by the Agency for Healthcare Research and Quality (AHRQ) published National Readmission Database (NRD) [19]; including patient level admission information from 2010 to 2017, regardless of the expected payer for

978-1-7281-8156-1/20/$31.00 ©2020 IEEE
DOI 10.1109/ICBK50248.2020.00026

the hospital stay. The NRD database provides a powerful public data source for readmission analysis, using all cause national scale patient level data with demographics, hospital, and treatment/procedure information.

In this paper, we propose to tackle the data imbalance challenge in 30-day hospital readmission prediction, by using a sampling based approach. We validate our design using National Readmission Database (NRD). We use feature engineering to create 16 features from demographics, admission and discharge information, and also convert ICD (International Classification of Disease) code to clinical categories for reduced feature space for learning. Experiments and comparisons show that balanced under sampling using Random Forest classifier achieves the best AUC scores for readmission prediction.

II. Feature Engineering for Hospital Visit

A. National Readmission Database

National Readmission Database (NRD) was first created by the Agency for Healthcare Research and Quality (AHRQ) in 2015 to provide data support for analyses of national readmission rates and further promote the quality of health care [19]. AHRQ is in the family of Healthcare Cost and Utilization Project (HCUP); where a collection of longitudinal healthcare databases combined with professional data analysis tools are provided in order to facilitate healthcare-related policies improvement. The database contains both clinical and nonclinical elements and collects around 18 million discharges in a year. In order to protect patient privacy, no patient's is recorded in a NRD file. The actual admitted date, discharged date or any other content that may reveal personal information are coded in a special format for the derivation of the gap between two visits of the same patient. Both single and repeated visits for patients are captured in the NRD database, and patient revisits are linked through the "VLink" filed, as shown in Table II. In 2016, the NRD database replaced the International Classification of Diseases, Ninth Revision, Clinical Modification (ICD-9-CM) applied in version 2015 with the tenth revision (ICD-10-CM/PCS) codes to represent clinical diagnosed and inpatient procedures [19]. ICD-10-CM/PCS codes are an American adopted version modified by Centers for Medicare and Medical Services (CMS) and the National Center for Health Statistics (NCHS), based on ICD-10, the statistical classification of disease published by the World Health Organization (WHO). 'CM' in ICD-10-CM codes stands for 'Clinical Modification'. There are more than 72,000 ICD-10-CM codes in the 2016 NRD database. Each ICD-10-CM code consists of 3 to 7 characters and the main purpose is to enable healthcare institutions to have a better understanding on a patient's medical conditions so that a more comprehensive and efficient treatment can be provided to patients. ICD-10-PCS stands for an inpatient procedural system. The intention of ICD-10-PCS codes is to provide insurance companies, healthcare providers with specific and accurate patient medical records.

We chose to use the 2016 NRD database as the data resource for our research. There are three files in the database. The first file is a Core file, in which every patient is represented by a unique NRD-Visitlink. Each row encodes visit information for every single patient visit including patient demographics. The second file, severity file, contains supplementary data information for condition severity identification and hospital. The third file, the level file, represents the information about hospitals to which patients in Core file were admitted. For this paper, we mainly focus on data analysis using the Core file. There are total 17,197,683 number of visits recorded in the Core file, with each visit including 103 data elements recorded in 103 columns.

B. Feature Engineering for NRD Database

The most important steps for successful data analysis are pre-processing data and extracting critical features. In the clinical field, these steps are especially significant because medical data are inherently complex and contain a variety of data fields with different ranges. For this reason, we first removed patients visit records with outliers, which are marked as a special value in the database. After that, we normalized columns with large range, such as total charges, to a fixed range. This is helpful to improve the performance of the final result.

In order to extra features for patient readmission prediction, we consider three types of features, including (1) patient demographics, (2) patient admission and discharge information, and (3) patient clinical information. Table I summarizes the features created for readmission prediction.

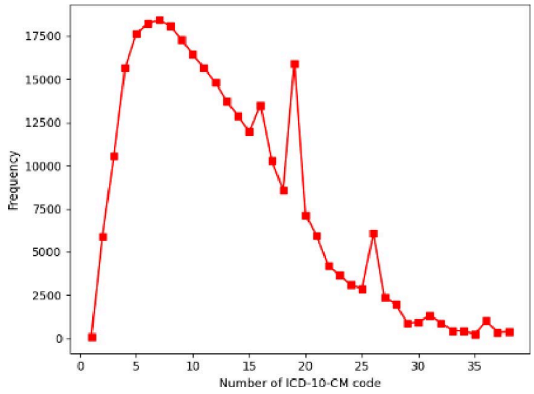


Fig. 1: Distributions of the number of ICD-10-CM codes in each visit. The $x-$axis denotes the number of ICD-10-CM codes in a patient visit. The $y-$axis denotes that for each $x-$axis value, the number of patient visits (frequency) with the specified number of ICD-10-CM codes.

1) Patient Demographics and Admission Information: For research purposes, the patient demographics and medical records during patients hospitalization and discharge information are key for readmission prediction. Data provided by demographics information about each participant, such as age and gender, are crucial in helping us determine whether

TABLE I: Features chosen for prediction

Feature Type	Feature	Description
Demographics	AGE	Patient's age
	FEMALE	Patient's gender (binary, '1' is female)
	PAY1	Payment method
	PL_NCHS	Patient's location (based on NCHS Urban-Rural Code
	ZIPINC_QRL	Estimated median house income in the patient's zip code
	RESIDENT	Patient's local (binary, '1' is the patient comes from same state as hospital)
Admission and Discharge Information	AWEEKEND	Patient's admission Day (binary, '1' means the admission day is a weekend)
	MONTH	Patient's discharge month
	QUARTER	Patient's discharge quarter
	DISPUNIFORM	Disposition of patients
	LOS	Length of the hospital stay
	ELECTIVE	Binary, '1' represents elective admission
	REHABTRANSFER	Binary, '1' is rehab transfer
	WEIGHT	Weight to discharges in AHA universe
	TOTAL CHARGES	Patient's inpatient total charges
	1^{st} HOSPITAL VISIT	Binary,'1' means the first hospital visit
Clinical Information	CCSR Code	Clinical categories

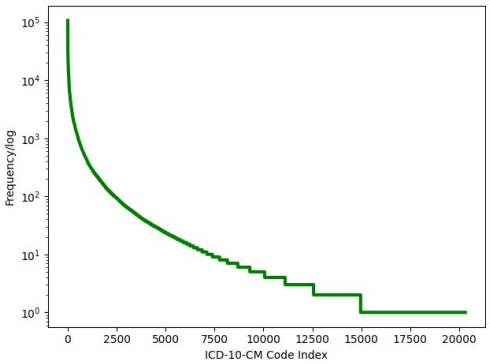


Fig. 2: Distributions of ICD-10-CM codes across all patient visits in log-scale. The $x-$axis denotes the ICD-10-CM codes ranked in a descending order according to their frequency. The $y-$axis denotes the frequency of each code in log-scale.

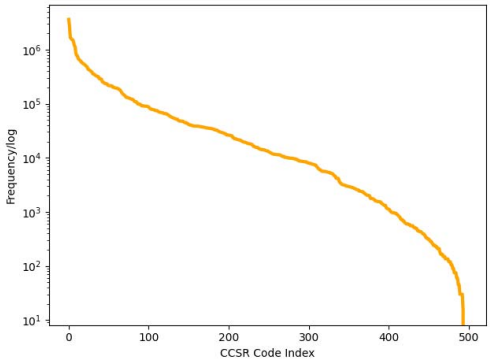


Fig. 3: Distributions of CCSR codes across all patient visits in log-scale. The $x-$axis denotes the CCSR code ranked in a descending order according to their frequency. The $y-$axis denotes the frequency of each CCSR code in log-scale.

individuals in this study are representative samples of the target population. Analyzing demographics characteristics is a catalyst for exhaustive medical policy enhancement.

In addition, patient admission and discharge information also play important roles in determining the likelihood of a readmission visit in the future. For example, the length of stay (LOS) of the current visit may imply the degree of illness (or severity of the disease) with respect to the current visit. Take feature 'DISPUNIFORM' as another example. It refers to the place where a patient is discharged, such as a family with home care or a nursing center. This feature plays an important role in readmission prevention.

2) Patient Clinical Information: In addition to the patient demographics and admission information, we also consider patient clinical information which is encoded as the ICD-10-CM code in the NRD database. For each patient visit, the ICD-10-CM codes detail the diagnose and treatment carried out during the patient visit. One essential challenge is that because ICD-10-CM are used for payment purposes and include all disease types, the total number of unique ICD-10-CM is very large. There are over 72,000 unique ICD-10-CM codes in the 2016 NRD database, making it highly ineffective to directly use ICD-10-CM codes as features for learning.

In order to reduce the number of features reflecting the patient clinical information, we convert the ICD-10-CM codes into Clinical Classification Software Refined (CCSR) codes. CCSR is an aggregation version for ICD-10-CM and it can improve the specificity of ICD-10-CM codes. Its utilization greatly improves the analysis on health models including healthcare cost, efficiency, outcomes [20]. Figure 1 shows the distribution of total number of ICD-10-CM codes for each patient visit. The result indicates that the total amount of ICD-10-CM codes for per visit is concentrated between 5 and 20. Figure 2 and Figure 3 represent the frequency distributions of ICD-10-CM codes and CCSR codes respectively. Where the frequency of the codes in the dataset are sorted in a log 10 scale descending order and the x-axis stands for the rank order of the corresponding code. From these two figures, we can tell that the frequency of both kinds of codes follows a negative exponential function.

After converting ICD-10-CM codes to CCSR codes, the number of features used for patient clinical information is denoted by less than 500 unique CCSR codes, as we will soon explain in Section IV.

C. Readmission Labeling Protocol

In order to generate class label for each patient visit, we label each patient visit as a readmission or not a readmission, by using 30-day as the criterion. Because our objective is to predict the possibility of a readmission in the future, we employ the following labeling protocol. For two visits (V_a and V_b) of the same patient, if the admission of V_b happens within 30-day (inclusive) after the discharge of V_a, we label V_a as a readmission visit (denoted by 1). Otherwise, V_a is labeled as not a readmission (denoted by 0). If the patient only have two visits V_a and V_b, then V_b will be labeled as not a readmission, because there is no succeeding visit following V_b. Intuitively, if the prediction is accurate, for each current patient visit, we will be able to estimate his/her readmission possibility in the future, when discharging the patient from the current visit.

Because there is no exact date information for the admission and discharge date of patient admissions, we need to calculate the gap (time period) between two admitted dates before labelling. In the NRD database, they use NRD_VisitLink to represent patient, thus, privacy can be protected through this de-identified patient record. Another feature used for privacy protection is NRD_DaysToEvent, where the actual patient admission date is substituted to a randomly chosen number (the main purpose is to hide the actual admission/discharge date of each visit for privacy protection). LOS stands for time duration a patient stays in the hospital after admission. Using these three features we are able to label which visit is a readmission.

An example to calculate gaps between hospital visits and the corresponding labels are shown in Table II and Figure 4. In Table II, a patient has three visit records in the dataset. The time interval between visit 2 and visit 1 equals to the second NRD_DaysToEvent minus the first NRD_DaysToEvent minus the first LOS. This is 2691 - 2679 - 2 = 10. For visit 3 and visit 2, the calculation is 2789 - 2691 - 5 = 93. For visit links that appear more than once, if the time interval between two visits is less than 30 days (inclusive), the earlier visit is label as '1', which represents a readmission. Therefore, we should label the first time visit as readmission and the second as well as the third visits are labeled as not a readmission as showed in Figure 4. The reason why we do not label the second time as readmission is that the purpose of our research is to predict whether there will be a possibility that a patient will return to hospital in 30 days or not after being discharged. For those visit links only appear once in the dataset meaning there exists no readmission for the patients, the time interval is infinite and they are labelled as '0'.

TABLE II: Example to calculate readmission days

Visit	Patient Visitlink	LOS	NRD_DaysToEvent
1	112233	2 days	2679
2	112233	5 days	2691
3	112233	3 days	2789

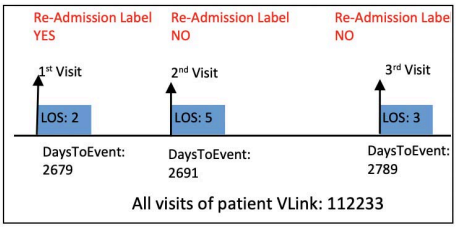


Fig. 4: Temporal arrangement of patient visits for re-admission labeling (Based on visits showing in Table II).

III. Imbalanced Learning for Readmission Predictions

Using feature engineering and labeling process, we are able to create a training dataset with both features and labels, where each instance in the dataset represents a hospital visit. This is a typical supervised learning task. Many leaning algorithms can be applied to learn classifiers for prediction.

A. Class Imbalance

The final dataset for our research includes 300,000 rows representing 300,000 patient visits, 498 columns of patient clinical features (CCSR code), 16 columns of patient admission features, and one additional column denotes the label of the visit. Although the number of features in this dataset is not particularly large, the data is actually severely imbalanced. There are only 2,926 patients who conducted multiple visits to hospitals, in which 2,851 patients were admitted into hospitals twice, 74 visited hospitals three times and only 1 patient visited 4 times. With respect to the label part, only 881 visits are labelled as readmission and the rest 299,119 visits are not readmission. Figure 5 and Figure 6 show the sample distributions of the dataset. As a result, the ratio between readmission visits *vs.* not readmission visits is around 1:340, meaning that positive samples (readmission visits) are less than 0.3% of the whole training samples. This represents a well-known imbalanced learning challenge, because majority learning algorithms prefer an equal percentage of positive *vs.* negative samples for learning accurate classifiers.

B. Imbalanced Learning

Severe class imbalance will deteriorate the performance of the learning algorithms, as a result, the learning tends to be biased to the majority (negative) class samples, and neglects the minority (positive) class. In our case, the positive samples (readmission visits) are less than 0.3% of the whole population, so a classifier can predict all instances to be negative and achieves 99.3% accuracy. This is, unfortunately, not useful for readmission prediction.

To tackle the class imbalance, we employ a random under sampling based approach to generate different versions of relatively balanced training set, where each training set contains a higher percentage of positive samples, compared to the positive/negative ratio in the original training set. More specifically, we applied a repeated k-fold cross-validation data frame in which re-sampling technique Random Under Sampling was used. Repeated k-fold cross-validation is a re-sampling method that repeatedly splits the dataset into k groups, and it is usually used to estimate the general performance of a model. In each fold, a bagging approach combined with three learning methods is implemented to combine results from multiple sampling. By doing so, the bias can be lowered and can demonstrate a better estimation in terms of statistics. The overall imbalanced learning algorithm is presented in Table III.

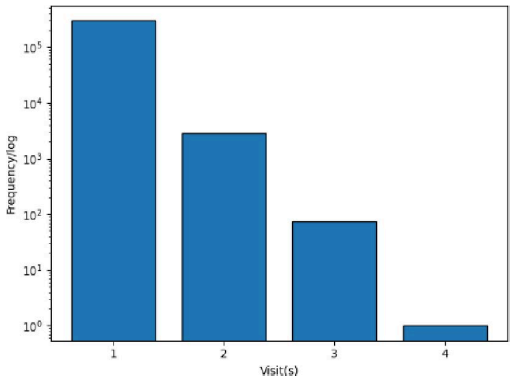


Fig. 5: Distributions of the number of hospital visit(s) of all patients. Out of all 300,000 hospital visits, only 2,851 patients have two more more visits. If a patient only has one visit, the visit will be labeled as "no a readmission" (0).

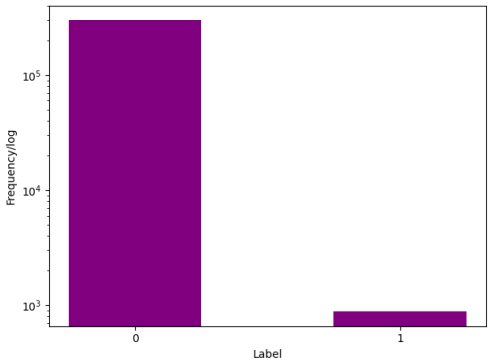


Fig. 6: Class distributions between 30-day re-admission visits (labeled as "1") $vs.$ non 30-day re-admission visits (labeled as "0". Overall, the re-admission visits are less than 0.3% of the total hospital visits.

TABLE III: Imbalanced Learning Algorithm

Algorithm: Imbalanced Learning for Hospital Readmission Prediction

Input NRD database;
Output Prediction of a new visit: *Test*
For features in NRD database:
  $\mathcal{F}$ ←Extract features as shown in Table I
For each visit ν in NRD database:
  Label v as first visit or not
  $\mathcal{F}_\nu$ ← Extract features from visit v using selected features $\mathcal{F}$
  Label v as Readmission(1) or Not(0)
D ← Created traning set of NRD database
For each sampling repetition *i*:
  $\mathcal{S}_i$ ← random under sampling to D to create a balanced training set
    $\mathcal{C}_j$ ← Train a classifier from $\mathcal{S}_i$
Test result [*j*]← Predict using classifier ($\mathcal{C}_j$, *Test*)
  End
Test Final prediction← Combine results from all sampling repetitions to make final prediction

IV. Experiments

A. Experimental Settings

We randomly extracted 300,000 patients visit records from the overall 17,197,683 patient visits and created 16 demographic and admission features, and 498 clinical features (CCSR codes) as shown in Table I to evaluate the algorithm performance for readmission prediction. In our experiments, the values in column AGE and TOTAL CHARGES are normalized through divided by the maximum value in the column to range [0,1]. Due to the large number of ICD-10- CM codes in 2016 NRD, instead of directly using them, we converted them into manageable number of clinical categories. The CCSR enables a way to identify specific clinical conditions using ICD-10-CM codes and this helps reduce the number to 498 but still keep the clinical information of each patient visit. In the experiments, we count the number of each CCSR code for each visit, and use the numerical values as features for learning. So in total, our training set contain 300,000 instances (visits), where each instance is represented by 516 features and a class label.

For all experiments, we used a 10 times 10-fold cross validation. Making multiple 10-fold cross validation repeatedly divided the data into 10 blocks for ten times where every block has equal size. As a result, it will generate 100 re-samples that with averaged data. For each fold in cross validation, we implemented Random Under Sampling with different sampling ratios, where the proportion of positive and negative classes are designed as 1:1, 1:2, 1:5, 1:10. Three learning algorithms are used in the experiments, including Decision Tree, Random Forest with 500 trees, and Random Forest with 1000 trees.

B. Experimental Results

The detailed performance including accuracy, F1_score and Area Under the ROC Curve (AUC) values for learning method Decision Tree (DT), Random Forest with 500 trees (RF-500), and Random Forest with 1000 trees (RF-1000) using four sampling ratios are reported in Table IV.

TABLE IV: Performance of imbalanced learning algorithm

Learning Method	Performance	Positive:Negative Sampling Ration			
		1:1	1:2	1:5	1:10
DT	Accuracy	0.8491	0.9429	0.9859	0.9933
	F1_score	0.4688	0.5003	0.5174	0.5161
	AUC	0.6789	0.6236	0.5466	0.5191
RF-500	Accuracy	0.858	0.9824	0.9955	0.9961
	F1_score	0.4751	0.5322	0.6106	0.5066
	AUC	0.7538	0.6114	0.5085	0.5046
RF-1000	Accuracy	0.8585	0.9824	0.9955	0.9961
	F1_score	0.4749	0.5322	0.5060	0.5065
	AUC	0.7535	0.6109	0.5080	0.5046

The three line graphs in Figure 7 indicate the change trend of three performance values with respect to different sampling ratios. For accuracy performance, as showed in Figure 7 (a), the results of RF- 500 and RF-1000 are almost the same except the value under sampling ration 1:1. All of the three methods show improved accuracy using 1:5 or more balanced sampling ratios (such as 1:1 or 1:2). When using more imbalanced sampling ratios (such as 1:5 or higher), the accuracy will remain stable. This is possible because that when data are imbalanced in the sampled set, using 1:5 or 1:10 sampling ratios, all positive samples will be misclassified as negative samples. Therefore, the accuracy will become stable (approaching to the percentage of negative samples in the test set).

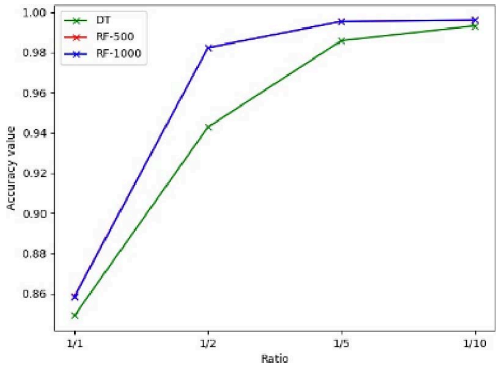
As for the F1_scores, shown in Figure 7 (b), the change shows two opposite trends at the point of ratio 1:5 for three methods. Overall, RF-500 and RF-1000 demonstrate a more significant rate of descent than DT. This is, in fact, consistent with the accuracy showing in Figure 7 (a), where the accuracy remain stable when using 1:10 sampling ratio.

Figure 7 (c) reports the AUC scores of all three methods with respect to different sampling ratios. Comparing to the accuracy and F1_score, AUC is much more accurate in evaluating the performance of the classifier with respect to both positive and negative samples. The results in Figure 7 (c) show that as the sampling ratio is becoming more imbalanced (from 1:1 to 1:5), the performance of all methods deteriorate in their AUC scores. After the sampling ratio reach 1:5, using more imbalanced sampling, such as 1:10, does not deteriorate the algorithm performance further, because all positive samples are classified as negative samples, resulting in 0.5 AUC values.

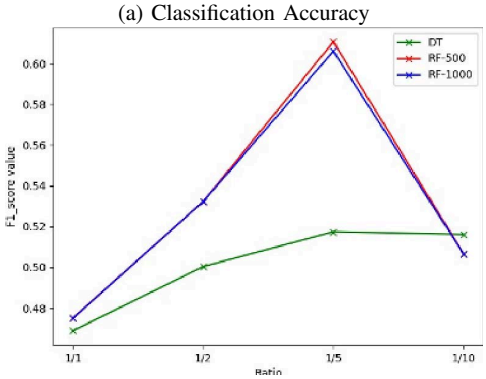
Figure 8 reports performance of three learning methods using different sampling ratios. For DT, Figure 8 (a), its accuracy and f1_score keep climbing before ratio 1:5 and after it the ascent scope becomes smooth. However, the AUC score decreases for all the four ratios. RF-500, Figure 8(b), is consistent with RF-1000, Figure 8 (c), in respect to accuracy and AUC, which is also the same as DT. The peak for RF-500 is the point at ratio 1:5 whereas it reaches the maximum at ration 1:2 for RF-1000.

CONCLUSIONS

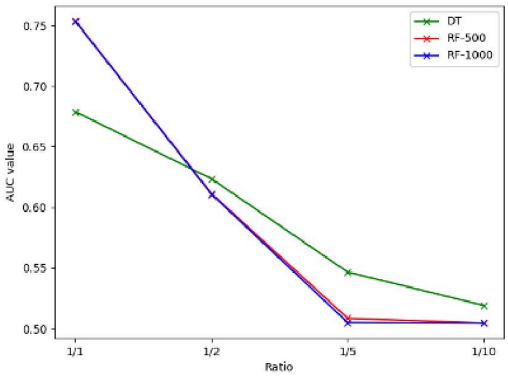
In this paper, we proposed to use imbalanced learning for 30-day hospital readmission prediction. The main goal is to predict, at the time of a hospital discharge, whether



(a) Classification Accuracy



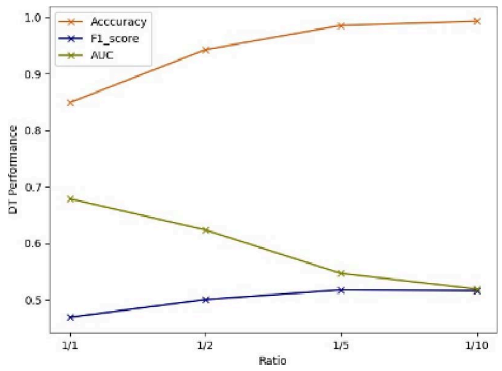
(b) F1_score



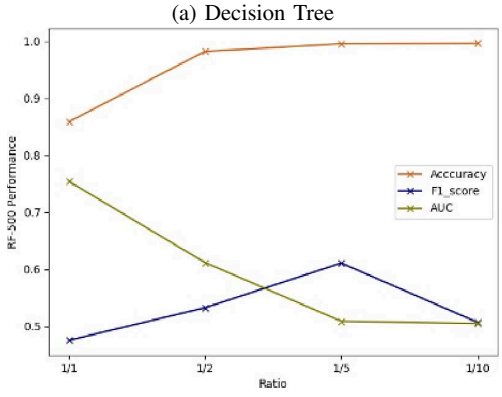
(c) AUC values

Fig. 7: Performance comparisons using different class sampling ratios 1:1, 1:2, 1:5, 1:10

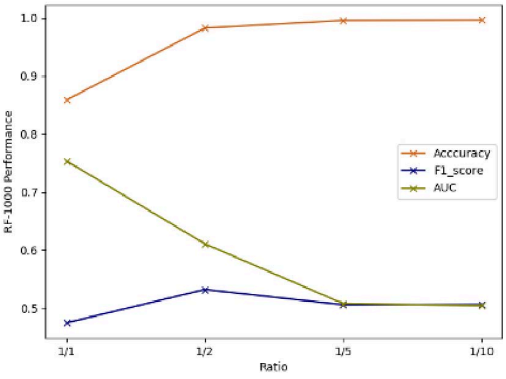
the patient may return in 30 days or not in the future. To build a machine learning task, we used National Readmission Databases (NRD) to extract features from patient visits. We created a set of features, using simple patient demographics, ICD-10 clinical modification (CM), and Clinical Classification Software Refined (CCSR) conversion, to represent each hospital visit. Because patient readmission is only a small portion of all patient visits, the machine learning task is severely challenged by the imbalanced class distributions. To solve the challenge, we used random under sampling (RUS) to create different copies of balanced sample sets. Ensemble classifiers



(a) Decision Tree



(b) Random Forest (number of trees = 500)



(c) Random Forest (number of trees = 1,000)

Fig. 8: Performance comparisons between decision trees (a), and random forest with 500 trees (b), and 1,000 trees (c)

were trained from balanced sample sets to build classifiers for readmission prediction. Experiments on the NRD databases confirm that Random Forests, with 1,000 trees, deliver the best AUC scores for 30-day hospital readmission prediction.

ACKNOWLEDGEMENT

This research was sponsored by the U.S. National Science Foundation (NSF) through Grants IIS-1763452 & CNS-1828181.

REFERENCES

[1] H.M. Krumholz, A.R. Merrill, E.M. Schone, et al., "Patterns of hospital performance in acute myocardial infarction and heart failure 30-day mortality and readmission," Circ Cardiovasc Qual Outcomes, vol. 2, pp.407-413, 1955.

[2] W. Zhang, S. Galloway, "Ten-year secular trends for congestive heart failure hospitalizations: an analysis of regional differences in the United States," Congest Heart Fail,14(5):266-271, 2008.

[3] H. Bueno, J.S. Ross, Y. Wang, et al, "Trends in length of stay and short-term outcomes among Medicare patients hospitalized for heart failure, 1993-2006," JAMA, 303(21):2141-7, 2010.

[4] Medicare Payment Advisory Commission, "Report to the Congress Medicare Payment Policy," 2019.

[5] E.W.Lee, "Selecting the Best Prediction Model for Readmission," Journal of Preventive Medicine and Public Health, 45,4(2012),259,2012.

[6] R. N. Axon, M.V.Williams, "Hospital readmission as an accountability measure," JAMA, 305(5):504-5, 2011.

[7] J. Harkey, R. Vraciu, "Quality of health care and financial performance: is there a link?," Health Care Manage Rev.17(4):55-63, 1992.

[8] S. F. Jencks, M. V. Williams, E. A. Coleman, "Rehospitalizations among Patients in the Medicare Fee-for-Service Program, ". The New England journal of medicine. 360. 1418-28, 2009.

[9] Jordan Rau, New Round of Medicare Readmission Penalties Hits 2,583 Hospitals, Kaiser Health News, October, 2019. https://khn.org/news/hospital-readmission-penalties-medicare-2583-hospitals/

[10] Hospital Readmission Reduction Program(HRRP), CMS.gov, Medicare, Value-Based Programs.

[11] Overview of HCUP, hcup-us.ahrq.gov.

[12] Yang, C., Delcher, C., Shenkman, E., & Ranka, S. (2016). "Predicting 30- day all-cause readmissions from hospital inpatient discharge data," 2016 IEEE 18th International Conference on e-Health Networking, Applications and Services (Healthcom), 1-6.

[13] D. He, S. C. Mathews, A. N. Kalloo, and S. Hutfless, "Mining high-dimensional administrative claims data to predict early hospital readmissions," Joournal of the American Medical Informatics Association, vol. 21, no. 2, pp.272-279,2014.

[14] A. G. Singal, R. S. Rahimi, C. Clark,et al, "An automated model using electronic medical record data identifies patients with cirrhosis at high risk for readmission," Clin Gastroenterol Hepatol, 11(10):1335-1341,2013.

[15] R. Caruana, Y. Lou, J. Gehrke et la, "Intelligible Models for HealthCare," the 21th ACM SIGKDD International Coference, 2015.

[16] S. Roy, A. Teredesai, K. Zolfaghar et la, "Dynamic hierarchical classification for patient risk-of readmission," the 21th ACM SIGKDD International Coference,2015.

[17] H. Wang, Z. Cui, Y. Chen, M. Avidan, A. Abdallah and A. Kronzer. "Cost-sensitive Deep Learning for Early Readmission Prediction at A Major Hospital." 16 th SIGKDD Workshop on Data Mining in Bioinformatics, 2017.

[18] X. Min, B. Yu, F. Wang, "Predictive Modeling of the Hospital Readmission Risk from Patients' Claims Data Using Machine Learning: A Case Study on COPD," Sci Rep, 20:9(1):2362,2019.

[19] Healthcare Cost and Utilization Project (HCUP) National Readmission Database Overview, https://www.hcup-us.ahrq.gov/db/nation/nrd/nrddbdocumentation.jsp.

[20] Clinical Classifications Software Refined (CCSR) for ICD-10-CM Diagnoses, https://www.hcup-us.ahrq.gov/toolssoftware/ccsr/ccs_refined.jsp#overview.

[21] Ankur Agarwal, Christopher Baechle, Ravi S. Behara, Xingquan Zhu, A Natural Language Processing Framework for Assessing Hospital Readmissions for Patients With COPD. IEEE J. Biomed. Health Informatics 22(2): 588-596, 2018.

[22] Xingquan Zhu, Jose Hurtado, Haicheng Tao, Localized sampling for hospital re-admission prediction with imbalanced sample distributions. Proc. of International Joint Conference on Neural Network (IJCNN). pp.4571-4578, 2017.