Artificial Intelligence Evolution
http://ojs.wiserpub.com/index.php/AIE/

UNIVERSAL WISER PUBLISHER

Research Article

Stroke Risk Prediction Using Artificial Intelligence Techniques Through Electronic Health Records

Song Jiang[1*], Yuan Gu[2], Ela Kumar[3]

[1]Department of Biochemistry, Huzhou Institute of Biological Products Co. Ltd., China
[2]Department of Statistics, The George Washington University, USA
[3]Department of Computer Science and Engineering, Koneru Lakshmaiah Education Foundation Deemed to be University, Vaddeswaram, India
E-mail: songjiang@hzbio.net

Received: 27 March 2023; **Revised:** 22 May 2023; **Accepted:** 24 May 2023

Abstract: Nowadays, Electronic Health Records (EHR) include critical information in the text format. In order to make medical decisions more efficient, the text should be processed and code deliberated. In this report, we applied Artificial Intelligence (AI) techniques to improve stroke risk prediction based on the EHR text. The system based on Natural Language Processing (NLP) generates structured text from EHR, followed by applying Machine Learning (ML) techniques to classify the text as a "good" or "bad" indicator, which is used for prediction. The ML models here we used include logistic regression and Support Vector Machine (SVM). Our results show that both models can classify the text precisely and make predictions accurately.

***Keywords*:** stroke risk, NLP, logistic regression, SVM

1. Introduction

EHR is a digital medical record that is used to store, manage, transmit, and reproduce with electronic devices (computers, health cards, etc.) to replace handwritten paper medical records [1-3]. It contains all the information of the paper medical records [2-4]. As we know, data types in EHR are very mixed and confused. It not only has well-structured data, but also many fully personalized data, such as doctors' own notes, ambulance records, authorization details, medication steps, etc., which are full of personal subjectivity [5-7]. These can be regarded as often as encountered in the field of NLP. To extract information from these data, the simplest and fastest method is to do it manually if the data is not large. However, when faced with massive PB-level data, the manual method becomes time-consuming, labor-intensive, error-prone and cannot be traced [8-10]. Therefore, it is natural to develop NLP to process automatically [11, 12].

NLP belongs to the sub-field of artificial intelligence. Its core purpose is to enable computers to understand and generate human natural language [13, 14]. The tasks mainly include information extraction, machine translation, sentiment analysis, abstract extraction, etc. The technologies of NLP include name recognition, semantic ambiguity elimination, reference resolution, speech tagging, structural analysis, etc. [15, 16]. The medical history, diagnosis, treatment methods, drugs, and other terms contained in a large number of medical texts provide the possibility for the

DOI: https://doi.org/10.37256/aie.4120232744

application of NLP. Using NLP to mine the hidden meaning is significant to the development of medicine [17].

To further extract the information from the text in EHR, ML models are applied, including logistic regression, SVM, random forest, decision tree, k-means, etc. [18-22]. Due to the limited pages, here, we only consider logistic regression and SVM as our top priorities. Logistic regression is a generalized linear regression analysis model, which is used in data mining, disease diagnosis, economic prediction, and other fields [23, 24]. The SVM is to find the best separation hyperplane in the feature space so that the positive and negative sample intervals can be maximized. SVM is a supervised learning algorithm for solving binary and multiclass classification problems [25, 26]. At the end of the report, we compared the logistic regression and SVM performance.

2. Methodology

The dataset used in this study is from Medical Information Mart for Intensive Care (MIMIC-III), which is a large, single-center database comprising information relating to patients admitted to critical care units at a large tertiary care hospital. The primary approach of our system, as seen in Figure 1, is to take EHR as input to preprocess, followed by NLP processing and analysis before applying classification algorithms (logistic regression and SVM).

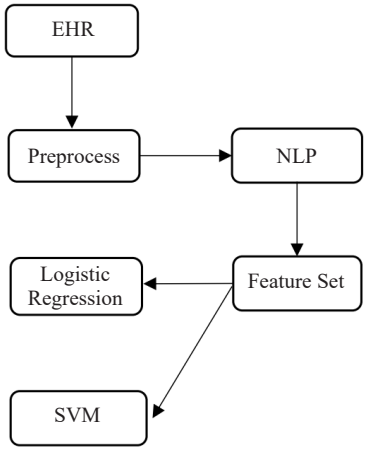

Figure 1. Analysis flowchart

In each algorithm, the data is split into train data (for model selection) and test data (for model validation), followed by NLP processing, such as stemming, computing the occurrence of words, etc. After that, the lasso feature selection is performed by applying a linear support vector classifier. Finally, each algorithm is implemented, optimized, and compared.

Table 1. Confusion matrix

		Predicted Class	
		Positive	Negative
Actual Class	Positive	True Positive (TP)	False Negative (FN)
	Negative	False Positive (FP)	True Negative (TN)

For logistic regression and SVM classification, the outcomes are summarized in Table 1. Equations 1 and 2 present how these values are obtained based on the predicted class types. Precision (equation 1) and recall (equation 2) are used to evaluate the classification performance.

$$precision = TP/(TP + FP) \tag{1}$$

$$recall = TP/(TP + FN) \tag{2}$$

2.1 *NLP*

Before applying algorithms, the medication abbreviations are replaced with full terms. Because the data are texts, they have to be transformed before applying them to a model. To do that, the text notes are first processed with word2vec and then transformed with term frequency-inverse document frequency with tfidf vectorizer.

2.2 *Classification*

After considering different classification algorithms, logistic regression and SVM were chosen. Logistic regression is preferred due to its explicit rule-based output that can be easily evaluated for content validity, whereas SVM is known to perform well in text classification tasks [27]. SVMs are also known to be robust to over-fitting [28] and they are working fast. It creates hyperplanes to separate data into classes, while logistic regression is based on statistical approaches. Both models are great and commonly used tools for classification problems.

3. Experiments and results

3.1 *Data collection*

The whole MIMIC-III database includes 26 tables. Here, the analysis focused on three tables-the NOTEEVENTS (including nursing and physician notes, ECG reports, radiology reports, and discharge summaries), DIAGNOSES_ICD (Hospital assigned diagnoses, coded using the International Statistical Classification of Diseases and Related Health Problems (ICD) system) and D_DIAGNOSES_ICD (Dictionary of International Statistical Classification of Diseases and Related Health Problems (ICD-9) codes relating to diagnoses) tables. The charge summaries of the NOTEEVENTS data are randomly sampled for this analysis.

3.2 *Preprocessing*

Prior to analysis and exploration, the data are cleaned including removing null values, removing incorrect reports, and cleaning non-characteristic letters, and replacing medical abbreviations. Then, the medical records in the NOTEEVENTS table are associated with the ICD9 codes in the DIAGNOSIS_ICD table and with the diagnosis name in the D_DIAGNOSIS_ICD table. Next, the records are labeled as stroke and non-stroke based on the ICD9 code, and the patient's past medical, social, and history information is retrieved. The top 10 diagnoses in the whole dataset are plotted and shown. The 1,904 records for stroke patients and 2,000 randomly sampled records for non-stroke patients are combined for cosine similarity and component analyses.

3.3 *NLP feature selection*

In this section, prediction models including Logistic Regression and Support Vector Machine Classifiers are implemented to predict the diagnosis, and the parameters for each model are optimized to obtain good prediction performance. Specifically, the following steps are followed: 1) split the data into train data (for model selection) and test data (for model validation), 2) use natural language processing (apply stemming to text, compute occurrence of words or term frequency-inverse document frequency to transform text notes) [29], 3) perform lasso feature selection using linear support vector classifier, 4) implement and optimize two popular text classifier: the Logistic Regression and Support

Vector Machine classifiers, 5) compare the performance of the optimized classifiers that are trained with the occurrence of words or term frequency-inverse document frequency.

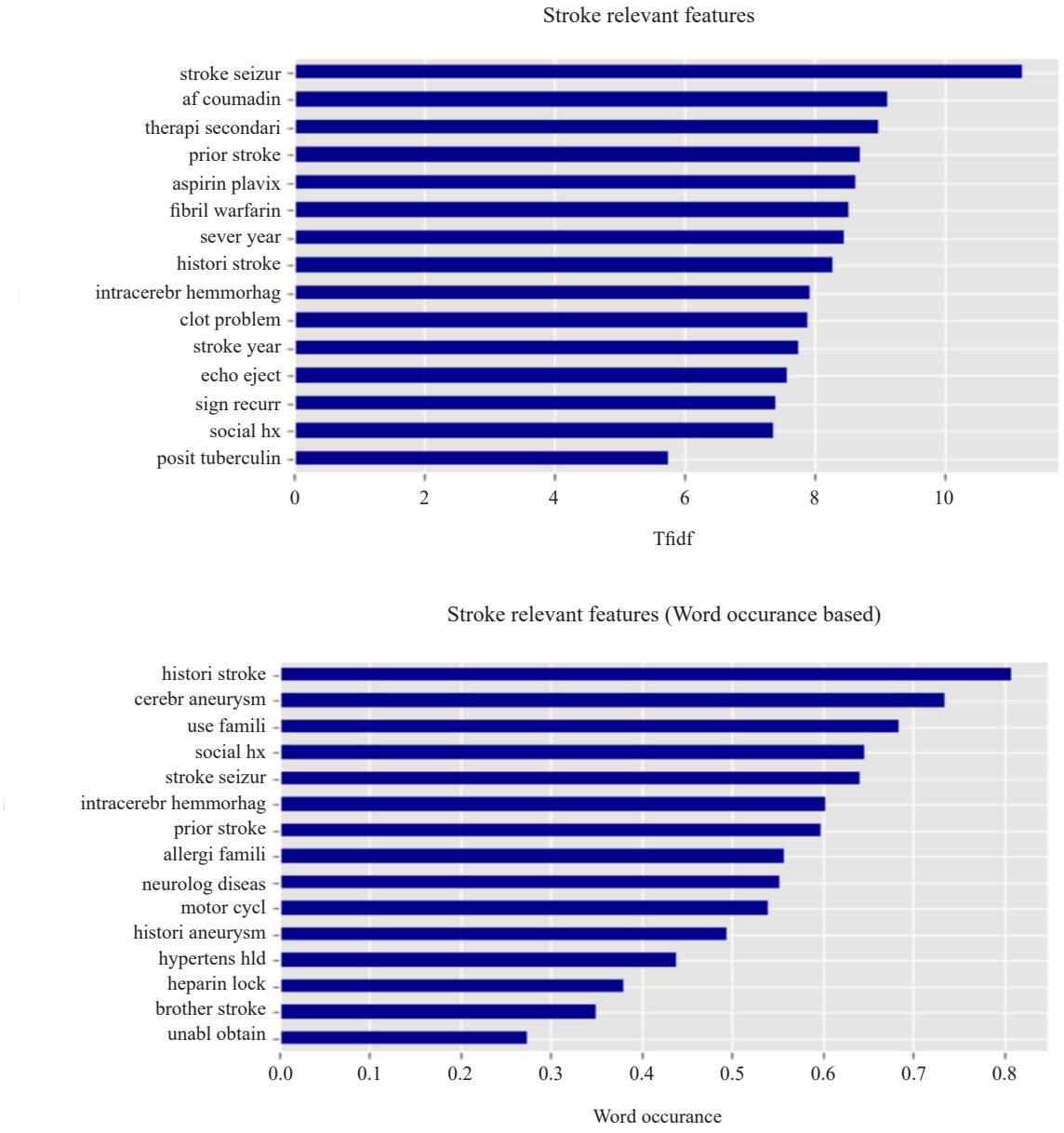

Figure 2. Top 10 topics based on tfidf vectorizer and word count occurrence vectorizer

To test a model, the data are first separated into train (80%) and test (20%) data [27]. The training data are used for training and cross-validation for each model. The test data are used to evaluate the final performance of each model. Because the data are texts, they have to be transformed before applying them to a model. To do that, the history notes are tokenized with stemming and converted into the occurrence of words with sklearn count vectorizer or term frequency- inverse document frequency with tfidf vectorizer [28].

Parameter C for the linear Support Vector Machine was searched among [0.1, 0.2, 0.25, 0.5, 10] using GridSearchCV, and 5-fold cross-validation was performed to determine the best parameter. We obtained C = 0.1 and

C = 10 as the best parameters for the count vectorizer and tfidf vectorizer, respectively. The top 10 topics for count vectorizer and tfidf vectorizer are presented respectably (Figure 2).

A logistic regression model with default parameters was also utilized to predict stroke. The metric we used is recall. Recall is the ratio tp/(tp + fn) where tp is the number of true positives and fn the number of false negatives. The recall is intuitively the ability of the classifier to find all the positive samples.

The learning curve of both the count vectorizer and tfidf vectorizer for default logistic regression are displayed (Figure 3). We can see that the performance of Logistic Regression trained with tfidf vectorizer transformed data is highly biased and Logistic Regression trained with count vectorizer transformed data is overfitted with the default parameters.

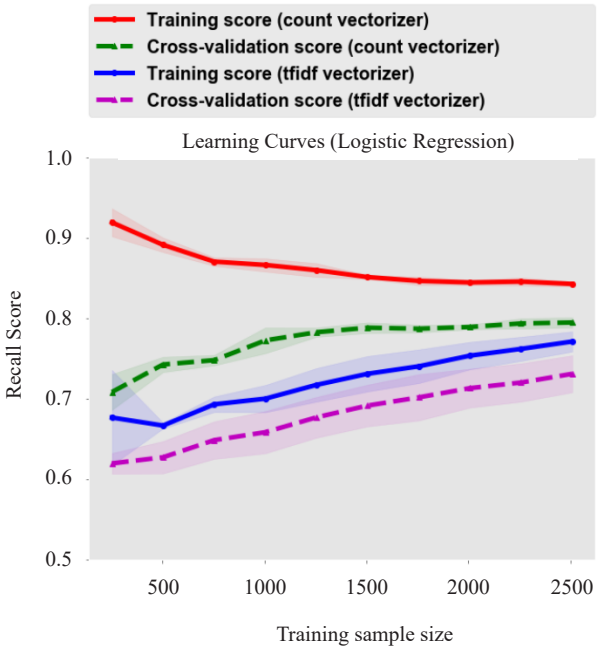

Figure 3. Learning curves of logistic regression trained on count vectorizer

Then we optimized parameters for logistic regression with parameters including regulator C, tolerance, and penalty. And we performed 5-fold split cross-validation to determine the best parameter. The best parameters for count vectorizer (Logistic Regression) are: {'C': 3.981071705534969, 'penalty': 'l2', 'tol': 0.001} and the best parameters for tfidf vectorizer (Logistic Regression) is: {'C': 1000.0, 'penalty': 'l2', 'tol': 1e-08}. Using recall as a metric again, we get a learning curve (Figure 4). We can see that the performance of Logistic Regression trained with tfidf vectorizer transformed data has been improved and Logistic Regression trained with count vectorizer transformed data are still overfitted with the optimized parameters.

Compare the learning curve of tfidf vectorizer for logistic regression with default parameters and logistic regression with optimized parameters (Figure 5). With the optimized parameters, there are improvements in the performance of prediction for the Logistic Regression classifier trained with tfidf vectorizer transformed data according to the cross-validation curves (Figure 5).

Next, we applied Linear Support Vector Classification with default parameters on personal stroke history information. We plotted the learning curve for both count vectorizer and tfidf transformed data (Figure 6). We can see that the performance of Linear Support Vector Classification trained with tfidf vectorizer transformed data is fine and Linear Support Vector Classification trained with count vectorizer transformed data are overfitted with the default parameters.

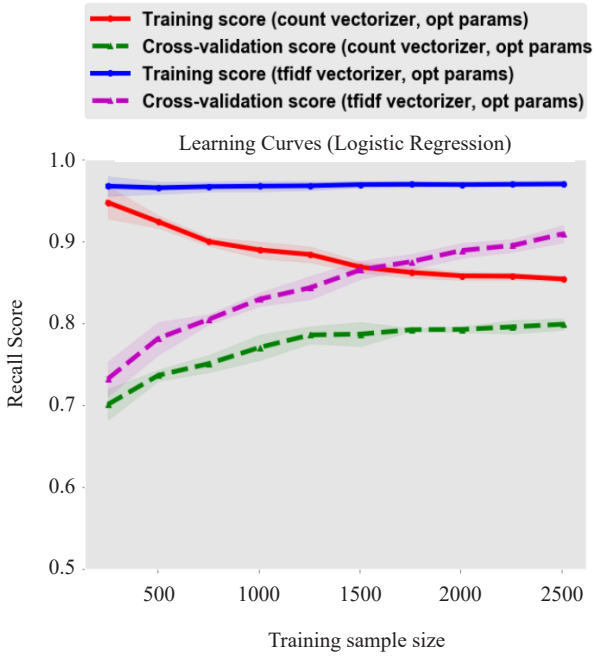

Figure 4. Learning curves of logistic regression trained on tfidf vectorizer

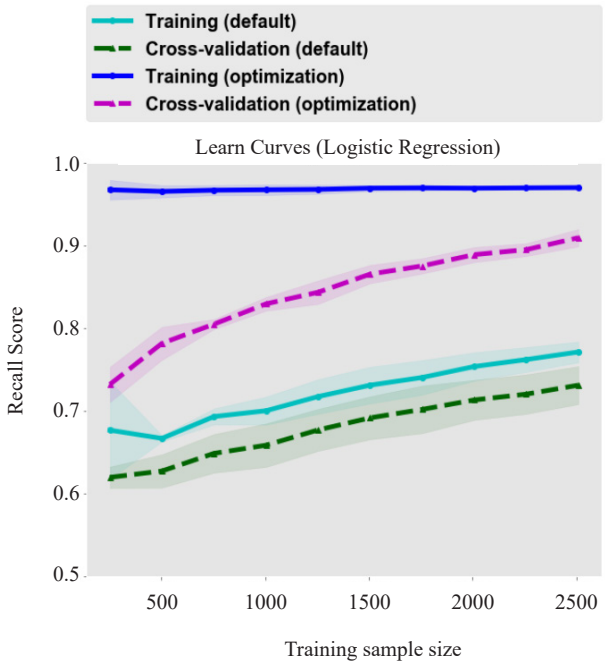

Figure 5. Learning curves of logistic regression trained on tfidf vectorizer with default and optimized parameters

Then we optimized parameters tolerance, regulator C, max iteration, and loss for SVM. And we performed 5-fold split cross-validation to decide the best parameter. The best parameters for the count vectorizer (SVM) are: {'C': 3.981071705534969, 'loss': 'squared_hinge', 'max_iter': 200, 'tol': 1e-08}. The best parameters for tfidf vectorizer (SVM): {'C': 63.09573444801933, 'loss': 'hinge', 'max_iter': 200, 'tol': 1e-08}. The learning curve is as Figure 7.

We can see that the performance of Linear Support Vector Classification trained with tfidf vectorizer transformed data is better than that with a default parameter, and Linear Support Vector Classification trained with count vectorizer transformed data is still overfitted with optimized parameters.

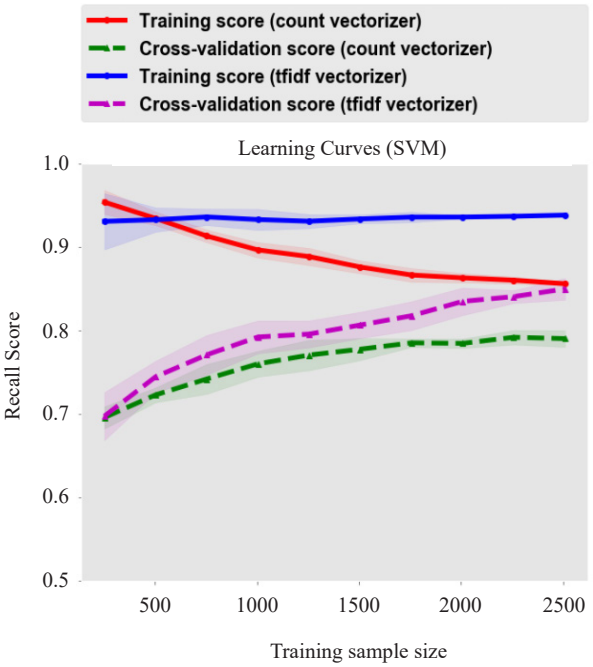

Figure 6. Learning curves of SVM trained on count and tfidf vectorizer

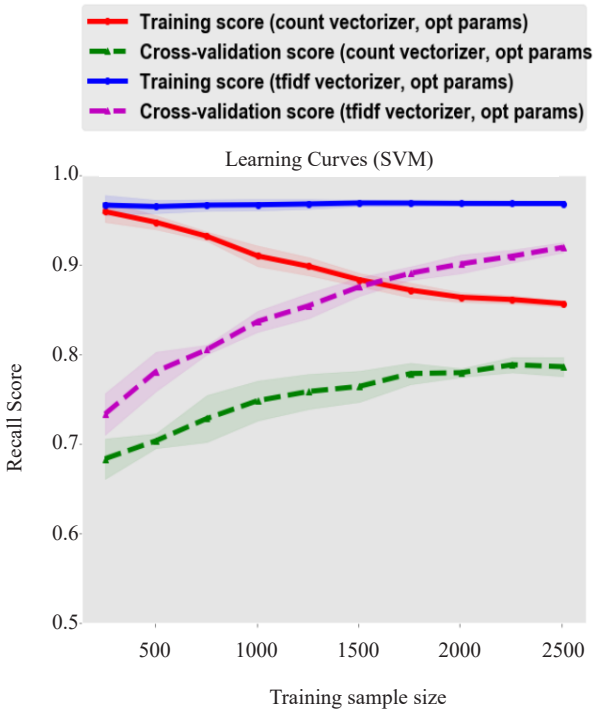

Figure 7. Learning curves of SVM trained on count and tfidf vectorizer with optimized parameters

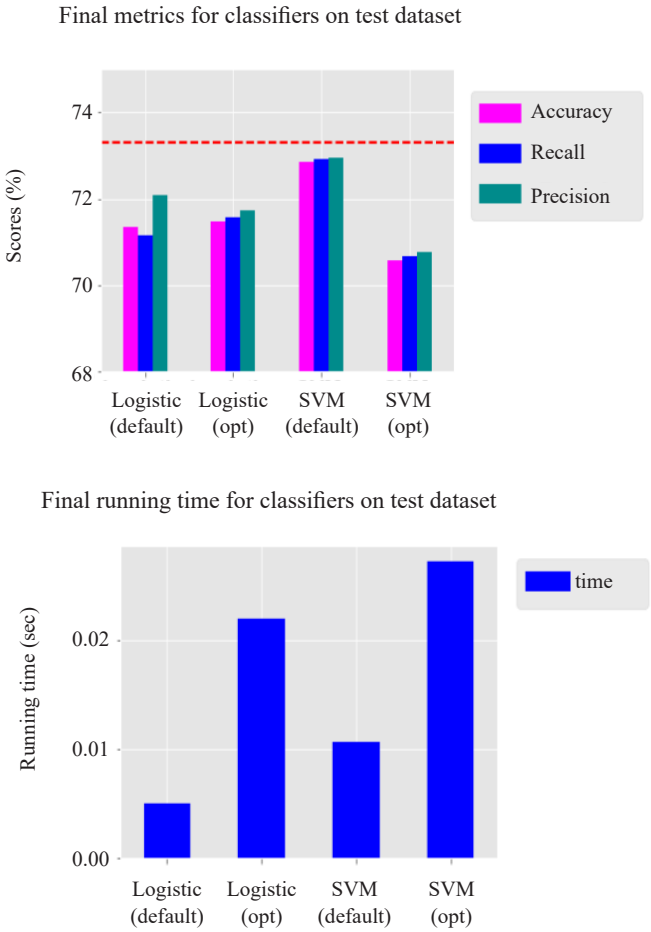

Figure 8. Model results and running time comparison across 4 models: Logistic with default parameter, Logistic with optimized parameter, SVM with default parameter and SVM with optimized parameter

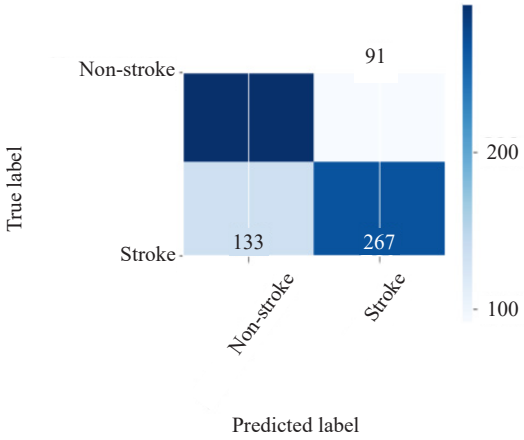

Figure 9. Non-normalized confusion matrix for logistic regression trained on tfidf vectorizer with optimized parameters

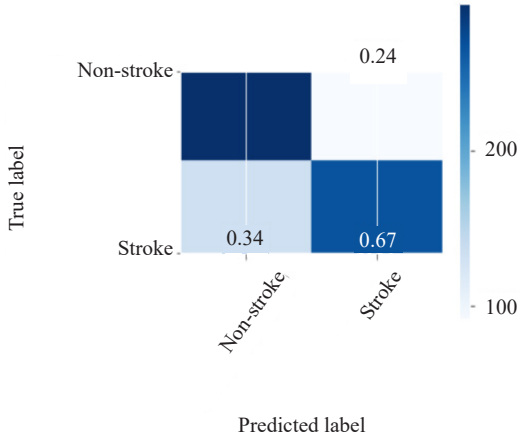

Figure 10. Normalized confusion matrix for logistic regression trained on tfidf vectorizer with optimized parameters

Four models (logistic regression with default parameters, logistic regression with optimized parameters, Linear Support Vector Classification with default parameters, and Linear Support Vector Classification with optimized parameters) running times were compared (Figure 8). Three metrics on four models were compared. The logistic regression classifier trained with tfidf vectorizer transformed data yields the best performance of the prediction based on the metrics of recall. We also examined the confusion matrix for Logistic Regression trained with tfidf features with optimized parameters, non-normalized (Figure 9), and normalized (Figure 10).

4. Conclusion

In this analysis, we analyzed the real de-identified medical records and implemented predictive models to predict the risk of stroke. We implemented two predictive models, a Logistic Regression classifier and a Linear support vector machine classifier, and compared their performances for prediction. The model process is as follows: 1) split the data into train data (for model selection) and test data (for model validation), 2) use natural language processing (apply stemming to text, compute occurrence of words, or term frequency-inverse document frequency to transform text notes, 3) perform lasso feature selection using linear support vector classifier, 4) implement and optimize two popular text classifier: the Logistic Regression and Support Vector Machine classifiers, 5) compare the performance of the optimized classifiers that are trained with the occurrence of words or term frequency-inverse document frequency. And their final performances are measured and compared with the test dataset. Based on the final performance (Confusion matrix in Figure 10) with the test dataset, the optimized Logistic Regression trained with tfidf transformed data showed the best performance for prediction.

Conflict of interest

The authors declare no competing financial interest.

References

[1] Jha AK, DesRoches CM, Campbell EG, Donelan K, Rao SR, Ferris TG, et al. Use of electronic health records in US hospitals. *New England Journal of Medicine*. 2009; 360(16): 1628-1638.

[2] Luo Y, Thompson WK, Herr TM, Zeng Z, Berendsen MA, Jonnalagadda SR, et al. Natural language processing for EHR-based pharmacovigilance: a structured review. *Drug Safety*. 2017; 40: 1075-1089.

[3] Aguirre RR, Suarez O, Fuentes M, Sanchez-Gonzalez MA. Electronic health record implementation: a review of resources and tools. *Cureus*. 2019; 11(9): e5649.
[4] Kalra D. Electronic health record standards. *Yearbook of Medical Informatics*. 2006; 15(01): 136-144.
[5] Cohen GR, Friedman CP, Ryan AM, Richardson CR, Adler-Milstein J. Variation in physicians' electronic health record documentation and potential patient harm from that variation. *Journal of General Internal Medicine*. 2019; 34: 2355-2367.
[6] Hausvik GI, Thapa D, Munkvold BE. Information quality life cycle in secondary use of EHR data. *International Journal of Information Management*. 2021; 56: 102227.
[7] Ohno-Machado L. Realizing the full potential of electronic health records: the role of natural language processing. *Journal of the American Medical Informatics Association*. 2011; 18(5): 539.
[8] Wang X, Hripcsak G, Markatou M, Friedman C. Active computerized pharmacovigilance using natural language processing, statistics, and electronic health records: a feasibility study. *Journal of the American Medical Informatics Association*. 2009; 16(3): 328-337.
[9] Chopard D, Treder MS, Corcoran P, Ahmed N, Johnson C, Busse M, et al. Text mining of adverse events in clinical trials: Deep learning approach. *JMIR Medical Informatics*. 2021; 9(12): e28632.
[10] Agrawal R, Prabakaran S. Big data in digital healthcare: lessons learnt and recommendations for general practice. *Heredity*. 2020; 124(4): 525-534.
[11] Juhn Y, Liu H. Artificial intelligence approaches using natural language processing to advance EHR-based clinical research. *Journal of Allergy and Clinical Immunology*. 2020; 145(2): 463-469.
[12] Li Y, Salmasian H, Harpaz R, Chase H, Friedman C. Determining the reasons for medication prescriptions in the EHR using knowledge and natural language processing. In *AMIA Annual Symposium Proceedings*. American Medical Informatics Association. 2011. p. 768.
[13] Oussalah M. AI explainability. A bridge between machine vision and natural language processing. In *Pattern Recognition. ICPR International Workshops and Challenges*. Springer International Publishing; 2021. p. 257-273.
[14] Nadkarni PM, Ohno-Machado L, Chapman WW. Natural language processing: an introduction. *Journal of the American Medical Informatics Association*. 2011; 18(5): 544-551.
[15] Balahur A, Turchi M. Comparative experiments using supervised learning and machine translation for multilingual sentiment analysis. *Computer Speech & Language*. 2014; 28(1): 56-75.
[16] Jiang J, Zhai C. Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*. Stroudsburg, PA, United States: Association for Computational Linguistics. 2007. p. 264-271.
[17] Liao KP, Cai T, Savova GK, Murphy SN, Karlson EW, Ananthakrishnan AN, et al. Development of phenotype algorithms using electronic medical records and incorporating natural language processing. *BMJ*. 2015; 350: h1885.
[18] Milosevic N, Dehghantanha A, Choo KK. Machine learning aided Android malware classification. *Computers & Electrical Engineering*. 2017; 61: 266-274.
[19] Moosmann F, Triggs B, Jurie F. Fast discriminative visual codebooks using randomized clustering forests. In *Advances in Neural Information Processing Systems*. 2006; 19.
[20] Jiang S, Gu Y, Kumar E. Magnetic Resonance Imaging (MRI) brain tumor image classification based on five machine learning algorithms. *Cloud Computing and Data Science*. 2023; 4(2): 122-133.
[21] Sachdev V, Tian X, Gu Y, Nichols J, Sidenko S, Li W, et al. A phenotypic risk score for predicting mortality in sickle cell disease. *British Journal of Haematology*. 2021; 192(5): 932-941.
[22] Cure-Cure CA, Cure P, Gu Y, Tian X, Patel T, Wu CO, et al. Predictors of all cause mortality and their gender differences in a hispanic population from barranquilla-colombia using machine learning with random survival forests. *Circulation*. 2018; 138(Suppl_1): A16252.
[23] LaValley MP. Logistic regression. *Circulation*. 2008; 117(18): 2395-2399.
[24] Steinwart I, Christmann A. Support vector machines. *Springer Science & Business Media*. 2008.
[25] Guo Y, Zhang Z, Tang F. Feature selection with kernelized multi-class support vector machine. *Pattern Recognition*. 2021; 117: 107988.
[26] Shah K, Patel H, Sanghvi D, Shah M. A comparative analysis of logistic regression, random forest and KNN models for the text classification. *Augmented Human Research*. 2020; 5: 1-6.
[27] Bentley P, Ganesalingam J, Jones AL, Mahady K, Epton S, Rinne P, et al. Prediction of stroke thrombolysis outcome using CT brain machine learning. *NeuroImage: Clinical*. 2014; 4: 635-640.
[28] Alexopoulos E, Dounias GD, Vemmos K. Medical diagnosis of stroke using inductive machine learning. *Machine*

Learning and Applications: Machine Learning in Medical Applications. 1999.
[29] Asadi H, Dowling R, Yan B, Mitchell P. Machine learning for outcome prediction of acute ischemic stroke post intra-arterial therapy. *PloS one*. 2014; 9(2): e88225.