Research Article

Estimation of Chronic Illness Severity Based on Machine Learning Methods

Yue Chang¹ and Xudong Chen²

¹Department of Electrical and Computer Engineering, NYU Tandon School of Engineering, New York University, New York, USA

²State Key Laboratory of Turbulence and Complex Systems and Department of Mechanics and Engineering Science, College of Engineering, Peking University, Beijing, China

Correspondence should be addressed to Xudong Chen; cx_d_pku@163.com

Received 24 June 2021; Revised 24 August 2021; Accepted 27 August 2021; Published 15 September 2021

Academic Editor: Danfeng Hong

Copyright © 2021 Yue Chang and Xudong Chen. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chronic diseases are diseases that last one year or more and require a continuous medical care and monitoring. Based on this point, a dataset from an APP called Flaredown helps patients of chronic disease improve their symptoms and conditions. In this study, an illness severity-level model was proposed to give the patient an alert to his or her health condition into three different levels according to their severity. Personal information, treatment conditions, and dietary conditions were analyzed by a statistical measure, TD-IDF. Seven different machine learning models were used and compared to generate the illness severity-level model. The results revealed that the XGBoost model with a F_1 score of 0.85 and LightGBM model with a F_1 score of 0.84 have the best performance. We also applied feature selection and parameter tuning for these two models to attain better performance, and the final best F_1 scores achieved by the XGBoost model and LightGBM model were both 0.85. Sensitivity analysis has shown that the treatment feature and symptom feature have important effects on the classification of the illness severity-level. Based on this, a fusion model was designed to study the data and the final accuracy of the fusion model was 93.3%. Thus, this study provides an effective illness severity-level model for a reference and guidance for the management of high-risk groups of chronic diseases. Patients may use this illness severity-level model to self-monitor their illness conditions and take proactive steps to avoid deterioration of their illness and take further medical care.

1. Introduction

Chronic diseases are often referred to as chronic noncommunicable diseases, and they are a group of diseases with long-lasting or persistent effects and complex etiology. The leading chronic diseases such as heart disease, cancer, chronic obstructive pulmonary disease (COPD), stroke, and diabetes [1] are the most important public health problems in the world, and they bring great health and economic burdens both to developed and developing countries [2, 3]. Centers for Disease Control and Prevention (CDC) reported that 75% of the health care in the US is related to chronic conditions and, every year, 70% deaths are from chronic disease [4].

Chronic disease management is vital to patients who are struggling with chronic diseases. Not only do patients need to get regular disease assessments and scientific treatment guidelines but also need to be active self-managers to improve care for their disease [5]. Studies have shown that self-management plays a critical role in the primary care of chronic illness [6] and is considered an inevitable part of treatment [7]. With the popularization of mobile devices, APPs which can help patients record health information and provide health strategies are constantly emerging and they become effective tools for patients to take self-managements. Meanwhile, the data collected by these APPs will provide more information to healthcare professionals and give patients more accurate assessments of their own health conditions. Although

at this stage, we still need to rely on doctors to diagnose the conditions more accurately, the combination of big data from mobile APPs and artificial intelligence provides us with a very broad application prospect.

Artificial intelligence (AI) has a good application in healthcare improvements and disease detections. Jiang et al. provided an overview of the current status of artificial intelligence applications in healthcare and performed a research of three major categories of AI applications in stroke [8]. Davenport and Kalakota gave us an overall analysis of the potential for artificial intelligence in healthcare. The article elaborated different types of AI applications in healthcare and described the trends and challenges of future AI in healthcare [9]. A study of data mining applications and techniques in healthcare has been proposed by Koh and Tan, and it has shown that data mining techniques are very useful in healthcare [10]. Wiens and Shenoy analyzed how machine learning can affect healthcare epidemiology (HE) and provide some considerations for healthcare epidemiologists [11]. Also, due to the robust development of the state-of-the-art learning algorithms, the convolutional neural network (CNN) algorithm has brought the intelligent healthcare to a new era. Rehman et al. studied the applications of the latest CNN models in the healthcare industry and provided effective solutions for these applications to solve the conventional challenges of the CNN model [12]. At the same time, deep neural networks have been widely used in image classification in recent years [13] [14] and image classification can provide valuable information in medical diagnosis [15]. Hong et al. developed a novel version of graph convolutional networks (GCNs) called minibatch GCNs which provided a robust mechanism to train large-scale graph convolutional networks. The new strategy combined the applications of CNNs and GCNs and provided more distinguishable features for image classification [16]. Hong et al. also provided a new spectral mixture model called the augmented linear mixing model (ALMM) which takes the spectral variability issue of the hyperspectral image classification into consideration. Compared with the methods that ignored the spectral variability, the proposed model provided a more accurate estimation [17].

Chronic diseases are usually diagnosed and predicted by machine learning methods or deep learning methods. Kim and Tagkopoulos analyzed the advantages and limitations of these artificial intelligence methods and applied them in rheumatology [18]. In order to predict the hospital readmissions of patients who have diabetes, Bojja et al. evaluated several machine learning methods to predict the readmission possibilities [19]. Machine learning techniques in a three-staged study were applied by Lin and Hu to provide an accurate classification model for three types of immune disease. This study filled the gap of lacking analysis of autoimmune diseases and infectious diseases [20]. A comprehensive review of feature selection methods and classification systems for chronic disease prediction was presented by Jain and Singh [21]. The support vector machine (SVM) is usually a good estimator in classification of chronic disease. A new novel GPU cluster-based MapReduce system (gcMR) for big-health-data processing was created by Li et al. The

gcMR system had better performance compared to previous multi-GPU cluster-based MapReduce [22]. Three rheumatoid arthritis disease subtypes were identified from gene expression data from synovial histologic features via using the SVM classification system by Orange et al. [23]. Another great method to study the mechanism of chronic disease is the decision tree. In order to seek a more robust and accurate method to diagnose heart disease, Shouman et al. tested different combinations of decision tree types, desecration methods, and voting techniques [24]. Habibi et al. developed a new prediction model for early diagnosis of type 2 diabetes mellitus (T2DM). This new model combined decision tree techniques with the J48 algorithm and used a real dataset of 22398 records and the final precision achieved by this model was 71.7% [25]. Artificial neural networks (ANNs) are also effective ways to predict chronic diseases. A case study of hypertension was taken by Wang et al., and they integrated a binary logistic regression model with ANNs to predict hypertension [26]. Chen et al. proposed a disease risk prediction model based on CNN, which overcomes the difficulty of incomplete data. The prediction accuracy of their model is about 95% with a fast convergence speed [27]. Cascio et al. applied the CNN method to classify indirect immunofluorescence (IIF) images. The performance of the system has shown an accuracy of 96% compared to other representative works [28]. Besides traditional models, new methods were also promoted for the classification of chronic diseases. Soni et al. designed a smart and effective heart disease prediction system using the weighted associate classifier (WAC). The maximum accuracy of this approach was 82% with a support value of 25% and confidence of 80% [29]. In order to predict diabetes, a new approach which applied a modified J48 classifier to improve the accuracy of the data mining procedure was presented by Kaur and Chhabra and experimental results have shown that an accuracy of 99.87% was achieved by this classifier [30]. In the deep learning area, Srinivasu et al. proposed an effective model which can help doctors to diagnose skin disease efficiently. The system combined neural networks with MobileNet V2 and Long Short Term Memory (LSTM) with an accuracy rate of 85%, far exceeding other state-of-the-art deep models of deep learning neural networks [31].

All the above previous analyses have shown that classification models have great performance on disease classification and prediction.

This article uses machine learning methods to estimate the severity of chronic diseases based on personal information, treatment conditions, and dietary conditions which are recorded in the Flaredown app. The purpose of this article is twofold. On the one hand, the mechanism and characteristics of chronic disease severity were analyzed. On the other hand, we provided a model with relatively few features to help patients estimate their chronic disease severity and be aware of changes in the severity so that they can be alerted to get in-time medical treatment. The first section of this article is background introduction of chronic disease and related work of machine learning methods. Section 2 introduces the study of data, including the source of the data, the cleaning and filtering of the data, the selection of

features, and the determination of the objective function. Section 3 mainly presents machine learning models used in this paper. The conclusion analysis is shown in Section 4, which mainly analyzes the parameter settings of the above models and compares the prediction results of these models through several indicators. Section 5 focuses on the pros and cons of these models and partly puts forward the idea of multimodel fusion. The last part of this article is a summary of the full text and proposes further improvements.

2. Data Research

Many chronic diseases are caused by some risk behaviors such as tobacco use, poor nutrition, and excessive alcohol use. People who have these behaviors are at high risk of chronic disease and typical symptoms or signs may occur in their daily life. Table 1 shows the symptoms and warning signs of four chronic diseases. So, it is important to track the symptoms of patients with chronic disease to prevent their symptoms from getting worse. Flaredown is an app which helps patients with chronic disease to track their symptom severity, treatments, and other potential environment factors every day, and a dataset derived from this app is used in our research. In this app, patients may use a 0 to 4 scale to describe the severity of their symptoms and these self-explanatory rating scores were used as the trackable values to generate the final illness severity-level model.

After data deduplication and cleaning up the null values, the size of the available dataset after recombination is 13699. Table 2 lists all the features provided in the dataset, and these features were used as input to get an estimate of the patient's illness severity level. Among these features, age and weather are numerical features while sex, condition, symptom, treatment, tag, food, and check-in date are nominal features. As the nominal information should be transformed into numbers via text vectorization to make the machine learning algorithms understand them, TF-IDF (term frequency-inverse document frequency) is used as our text vectorizer. TF-IDF is very helpful for scoring words in machine learning algorithms, and the scores can be used to know the relevance of the words to our topic.

TF-IDF is a statistical measure that evaluates how relevant a word is to a document in a collection of documents. For a term t in document d , the TD-IDF score $W_{t,d}$ is

$$W_{t,d} = tf_{t,d} \times \log \left(\frac{N}{df_t} \right), \quad (1)$$

where $tf_{t,d}$ is the number of occurrences of t in d , df_t is the number of documents containing t , and N represents the total number of documents.

For a patient, the formula for his illness severity level which represented by y is

$$y = \sum_{i=1}^n (\text{trackableValue}_i + 1). \quad (2)$$

If a patient has n symptoms and the trackable_value for each symptom is trackableValue_i , then, we can divide the illness severity level into three levels, and the higher the level, the more severe the patient's symptoms are

$$\text{Level} = \begin{cases} 1, & y < 16, \\ 2, & 16 \leq y < 30, \\ 3, & y \geq 30. \end{cases} \quad (3)$$

Figure 1 describes the data statistics of different features like age, sex, weather information, and trackable value. The graph of trackable value presented the distribution of the illness severity level which is the output y in our research which is divided into three levels.

Figure 2 describes the relevance of the numerical features like age, sex, and weather information to the value of the illness severity level. It can be seen in the figure that the correlation between these numerical features and the illness severity level is not great. It means that we should use both the numerical features and nominal features in our study.

3. Model

3.1. Logistic Regression. Logistic regression is often used to deal with regression problems where the dependent variable is a categorical variable. Commonly, the regression problems are binomial or binomial distribution problems but it can also handle multiclassification problems [32]. Logistic regression is simple to implement and is widely used in data mining, automatic disease diagnosis, economic prediction, and other fields. At the same time, it is not expensive to calculate and easy to understand and implement due to few parameters of the model. Besides, it is convenient to observe the sample probability through logistic regression which predicts a probability between 0 and 1.

The implementation of logistic regression has three steps. The first step is to choose the logistic function. The logistic function is an important representative of the Sigmoid function (a function of shape S), and it is selected as the activation function in logistic regression. The second step is to obtain the prediction function of logistic regression. The prediction function is shown as $h_\theta(x)$ where θ is the parameter vector and $\theta^T x$ represents the decision boundary.

$$h_\theta(x) = g(\theta^T x) = \frac{1}{1 + e^{-\theta^T x}}, \quad (4)$$

$$\theta^T x = \sum_{i=1}^n \theta_i x_i = \theta_0 + \theta_1 x_1 + \dots + \theta_n x_n.$$

The third step is to find the optimal solution for the classification task. For an input value x , the probability for classification can be calculated where the value of y could be 0 or 1. The loss will be calculated under each condition. For m -independent distributed training samples which is represented by x , the likelihood function is $L(\theta)$ and the log-likelihood function is $l(\theta)$. The maximum

TABLE 1: Symptoms and warning signs of four chronic diseases.

Chronic disease	Symptoms and warning signs
Heart disease	(i) Chest pain or discomfort
	(ii) Upper back or neck pain
	(iii) Indigestion
	(iv) Heartburn
	(v) Nausea or vomiting
	(vi) Upper body discomfort, dizziness
	(vii) Fluttering feelings in the chest
	(viii) Shortness of breath
	(ix) Fatigue
	(x) Swelling of the feet, ankles, legs, abdomen, or neck veins
Cancer	(i) Breast changes—lump in your breast
	(ii) Coughing, chest pain, and breathlessness
	(iii) Changes in bowel habits
	(iv) Unexplained bleeding
	(v) Moles
	(vi) Unexplained weight loss
COPD	(i) An ongoing cough or a cough that produces a lot of mucus (often called smoker's cough)
	(ii) Shortness of breath, especially with physical activity
	(iii) Wheezing or a whistling or squeaky sound when you breathe
	(iv) Chest tightness
Diabetes	(i) Urinating a lot, often at night, and are very thirsty
	(ii) Losing weight without trying and are very hungry
	(iii) Having blurry vision
	(iv) Having numb or tingling hands or feet
	(v) Feeling very tired and having very dry skin
	(vi) Having sores that heal slowly
	(vii) Having more infections than usual

TABLE 2: Features of chronic illness.

No	Factor	Information factor	Description
1	Age	Numerical	Age of a user
2	Sex	Numerical	Sex of a user
3	Condition	Nominal	An illness or diagnosis which is rated on a scale of 0 to 4
4	Symptom	Nominal	Self-explanatory, also rated on a 0–4 scale
5	Treatment	Nominal	Anything a patient uses to improve their symptoms
6	Tag	Nominal	A string representing an environmental factor
7	Food	Nominal	Food seeded from the USDA database or manually added
8	Weather	Numerical	Weather information based on user's postal code
9	Check-in date	Nominal	The check-in date of the user

likelihood estimation shows that the optimal solution will be obtained and the value of θ maximizes $l(\theta)$.

$$l(\theta) = \log L(\theta) = \sum_{i=1}^m y_i \log h_{\theta}(x_i) + (1 - y_i) \log (1 - h_{\theta}(x_i)). \quad (5)$$

3.2. Random Forest

3.2.1. Decision Tree. A decision tree is a tree-like decision structure and Figure 3 illustrates the basic structure of the

design tree. Each internal node of the decision tree represents a test of a feature, each branch represents the test result of the feature, and each leaf node represents a classification category [33, 34]. The goal of the decision tree is to classify the dataset according to the corresponding class labels, and the quality of feature selection determines the prediction accuracy of the decision tree. Appropriate feature selection can make the samples contained in the branch nodes of the decision tree belonging to the same class as much as possible. There are three criteria that can be used to judge the purity of a node: the information gain, the information gain rate, and the Gini index. This article uses the Gini

FIGURE 1: Description of numerical features.

FIGURE 2: Correlation of numerical features and frequency histogram.

index as the criterion for determining the purity of a node, that is, the smaller the Gini index, the smaller the uncertainty of the sample.

Decision trees have many advantages. They are easy to understand and can be visually analyzed. They can process both nominal and numerical data and are very suitable for samples with missing attributes. They also can handle irrelevant features and make feasible and effective decision results for large data sources in a short period of time [35].

3.2.2. Random Forest. Random forest is an algorithm that integrates multiple trees through the idea of ensemble learning. Its basic unit is a decision tree, but there is no correlation between each decision tree. The final classification

category of the random forest is determined by the mode of the classes of output by all decision trees in the forest, that is, when a new sample is an input, each decision tree in the forest will be classified, and then, the most selected category will be used to label this sample [36, 37].

Random forests are easy to implement and fast to train, making it easy to make parallel methods. At the same time, it can process data with very many features, without dimensionality reduction and feature selection. It has a strong ability to process datasets as it can handle both discrete and continuous data. The introduction of two randomness makes the random forest hard to overfit and has good antinoise ability. In addition, it can judge the importance of features and the interaction between different features.

FIGURE 3: Structure of the decision tree.

FIGURE 4: Process of constructing a random forest.

Even if a large part of the features were lost, the random forest can still maintain accuracy.

The process of constructing a random forest is shown in Figure 4. First of all, randomly select N samples from a dataset and use these selected samples to train a decision tree. Each of these samples has M features. m features are randomly selected from the M features, and the condition $m \ll M$ is satisfied. Then, use a certain strategy such as information gain to select one feature from these m features as the splitting feature for the node. During the formation of the decision tree, each node must be split by the above steps until it cannot be split again, that is, if the next selected feature of the node is the feature used by its parent node when splitting, then, the node has reached the leaf node and no longer needs to be split. Each decision tree grows to the greatest extent, and no pruning is performed during the entire decision tree formation process. Finally, a large num-

ber of decision trees are established in the above manner to form a random forest [38].

3.3. Support Vector Machine. The support vector machine is a class of generalized linear classifiers that perform binary classification of data by supervised learning. Its decision boundary is the maximum-margin hyperplane solved for the learning samples [39–41]. SVM can solve high-dimensional problems as well as machine learning problems based on small dataset, and a nonlinear SVM model with a kernel function is shown in Figure 5. Compared with algorithms such as neural networks, it has no local minimum problems. It also can handle interactions of nonlinear features. In addition, it does not need to rely on the entire dataset and its generalization ability is relatively strong [42, 43].

The purpose of SVM is to find a classification hyperplane so that the classification interval between the samples

FIGURE 5: SVM model.

of two categories is the largest. A set of coefficient vectors $\vec{w} = (w_1, w_2, \dots, w_p)$ should be found to maximize b in the following formula and satisfy the constraints for each sample (x_i, y_i) . The ε_i in the constraints is called a relaxation variable, and the sum of all relaxation variables should be less than the penalty coefficient C .

$$\begin{aligned} y_i(w_1 x_{i1} + w_2 x_{i2} + \dots + w_p x_{ip}) &\geq b(1 - \varepsilon_i), \\ \sum_{i=1}^n \varepsilon_i &\leq C, \quad \varepsilon_i \geq 0, \\ \sum_{j=1}^p w_j^2 &= 1. \end{aligned} \quad (6)$$

3.4. XGBoost. XGBoost (Extreme Gradient Boosting) combines hundreds and thousands of tree models into one highly accurate model, and this model generates new trees through continuous iteration. Each tree only contains a few branches and is not very powerful. Each individual tree will try to correct for the mistakes of all the trees before it and makes the model better via using gradient boosting. A set of decision trees will be produced after many iterations. The final prediction will be the sum of the output from each tree [44]. Figure 6 presents the constructing process of XGBoost.

The basic idea of XGBoost is to continuously add new trees during the implementation and continuously perform feature splitting to grow a tree. Each time that a tree is added, a new function is learned to fit the last predicted pseudoresidual. When k trees are obtained after training, the score of a sample can be predicted, that is, one leaf node will be selected to represent a sample feature in each tree and the leaf node will get a corresponding score, and then, the scores of these leaf nodes will be added up as the predicted score of the sample.

The objective function of XGBoost consists of training loss and regularization. In practical training process, XGBoost uses the greedy method to split the nodes. When the depth of the tree is 0, each leaf node in the tree is attempted to split. After each split, the original leaf node is continuously split into two nodes—a left child node and a right child node and the samples of the original leaf node will be dispersed into these two child nodes. We need to calculate the gain of each split to find the best split point and control the growth of the tree [45].

3.5. LightGBM. LightGBM is a fast, distributed, high-performance gradient boosting framework and can be used for sorting, classification, regression, and many other machine learning tasks. LightGBM has a lot of advantages. It has lower memory usage and better accuracy, and it supports parallel learning and has the capability of handling large-scaling data. Besides, it supports categorical feature directly.

The implementation details of LightGBM are listed in the following aspects. LightGBM uses a leaf-wise leaf growth strategy with depth limitation to train each decision tree and split the data. In order to find the best split for each leaf node within a reasonable time cost, LightGBM applied a histogram-based algorithm to reduce the number of splits. Besides that, LightGBM mainly uses a gradient-based one-side sampling (GOSS) technique to train data as this method can effectively reduce the number of samples, that is, excluding most small gradient samples. LightGBM also applies an exclusive feature bundling method to effectively reduce the number of features [46, 47]. The leaf-wise tree growth strategy of LightGBM is displayed in Figure 7.

4. Results

80% of the dataset were used as the training set, and 20% of the dataset were used as the test set. The dataset is trained and tested with seven different models, and the performance of each model is measured by accuracy, precision, recall, and F_1 score.

Accuracy is the ratio of correctly predicted observations to the total observations and described as follows:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}}, \quad (7)$$

where TP represents the correctly predicted positive values, TN represents the correctly predicted negative values, FP represents actual negative values but predicted as positive values, and FN represents actual positive values but predicted as negative values.

Precision is shown as follows:

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}. \quad (8)$$

Recall is the ratio of correctly predicted positive observations to the sum of actual positive observations and actual negative observations.

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}. \quad (9)$$

The F_1 score is calculated as

$$\begin{aligned} F_1 \text{ score} &= \frac{1}{n} \left(\sum f1_k \right)^2, \\ f1_k &= \frac{2 \times \text{precision}_k \times \text{recall}_k}{\text{precision}_k + \text{recall}_k}. \end{aligned} \quad (10)$$

For logistic regression, we applied both L1 and L2

FIGURE 6: XGBoost constructing process.

FIGURE 7: Leaf-wise tree growth in LightGBM.

regularization and adjust the inverse of regularization strength to train the model. We also used the Newton method and stochastic average gradient descent as our solvers to find optimize parameters for the cost function. The final precision of L1 regularization is 0.79, and the F_1 score is 0.79; the final precision of L2 regularization is 0.79, and the F_1 score is 0.66. We trained the model with two types of kernels for the SVM-SVM with linear kernel and SVM with radial basis function (RBF) kernel. The final precision of linear SVM is 0.80, and the F_1 score is 0.79; the final precision of SVM RBF is 0.48, and the F_1 score is 0.31. The performance of SVM RBF is not very satisfactory as the RBF kernel is too complex to our dataset. We tested different numbers of decision trees in the forest to train the random forest model. The maximum depth of the tree was set to four different values: 10, 50, 100, and 200. Besides that, the minimum number of samples required to split an internal node is tested with three values: 2, 5, and 10. The final precision of random forest is 0.82, and the F_1 score is 0.82. For XGBoost, we set the number of decision trees as 1000 and the maximum depth of the tree as 6. To prevent overfitting, the value of the subsample is set to 0.8 and the step size shrinkage used in the update is set to 0.1. The final precision of XGBoost is 0.85 and the F_1 score is 0.85. For LightGBM, the boosting type of the LightGBM classifier is the gradient boost decision tree (GBDT). The maximum tree depth for base learners and the boosting learning rate were set to multiple values to train the model. The L1 regularization was set to 0.6 and L2 regularization was set to 0 for the model. The final precision of LightGBM is 0.84 and the F_1 score is 0.84.

Among the above models, the performance of the SVM with the linear model is similar to the L1 logistic regression

TABLE 3: Classifier performance comparison.

	Precision	Recall	F_1 score
LR with L1 regularization	0.79	0.79	0.79
LR with L2 regularization	0.79	0.72	0.66
SVM linear	0.80	0.79	0.79
SVM RBF	0.48	0.48	0.31
Random Forest	0.82	0.82	0.82
XGBoost	0.85	0.85	0.85
LightGBM	0.85	0.84	0.84

model and the performance of the L2 logistic regression is slightly inferior to the former two and the random forest model is slightly better than the SVM with linear model and L1 logistic regression models. Of the four models, the SVM with RBF kernel has the largest result deviation, while the XGBoost model and the LightGBM model have the best results, as shown in Table 3.

Table 4 listed the running time, time complexity, and space complexity of these models. Combining the impact of the computer hardware on the running time, the logistic regression model has the shortest time, and compared with the random forest model, the running time of XGBoost and LightGBM is significantly reduced.

5. Discussion

5.1. Model Comparison. As can be seen in Table 3, the XGBoost model and LightGBM model have the best performance. LR with L1 regularization, SVM with linear kernel,

TABLE 4: Time complexity and space complexity comparison.

	Running time	Time complexity	Space complexity	Parameters
Logistic regression	23.3 s	$O(nd)$	$O(d)$	n = samples d = dimension of features
SVM	42.7 s	$O(n^2)$	$O(k)$	n = samples, k = support vectors
Random Forest	45.3 s	$O(mkn \log n)$	$O(n + dm)$	n = samples, d = number of features, m = randomized trees, k = variables drawn at each node n = samples
XGBoost	33.4 s	$O(Kd\ x\ _0 \log n)$	$O(Kq)$	K = number of trees, d = max <i>depth</i> of the tree, $\ x\ _0$ = non-missing entries, q = candidates in the dataset
LightGBM	41.8 s	$O(0.5dm)$	$O(m)$	d = dimension of features, m = number of bins

and random forest also have higher precision, recall, and F_1 score. But the F_1 score of the LR with L2 regularization model is not very high as our dataset has many features, and the LR with L2 regularization model itself is relatively simple, so it is not the most suitable model for our dataset. Besides that, the F_1 score of the SVM with RBF kernel model is relatively low, which shows that the kernel function of RBF for classification of our dataset is too complicated and is not suitable for our dataset.

For XGBoost and LightGBM, their own advantages make them have better performance than other models. For the XGBoost model, as it uses column sampling and adds a regular term to the objective function, XGBoost has high accuracy. XGBoost also uses the sparse perception algorithm to process missing values, which makes XGBoost have a strong advantage in the processing of missing values. Finally, XGBoost supports parallelism in feature granularity, so it has a high calculation rate. At the same time, it uses leaf-wise splitting methods to generate more complex trees than level-wise splitting methods, so LightGBM also has a perfect performance. As the XGBoost and LightGBM models have the best performance among these models, we have a further analysis of the feature selection and parameter tuning for these two models.

5.2. Analysis of Parameter Sensitivity

5.2.1. Feature Selection Analysis. Figure 8 is obtained by removing one feature each time from the feature set and calculating the F_1 score using all other features. From the graph in Figure 8, treatment feature and symptom feature have important effects on the classification of the illness severity level. Not using treatment feature leads to a 0.26 lesser F_1 score of the XGBoost model and a 0.25 F_1 score of the LGBM model than using all the features. For the symptom feature, not using it will lead to a 0.25 lesser F_1 score of the XGBoost model and a 0.24 F_1 score of the LGBM model. The drop in F_1 scores indicates that treatment and symptoms are very relevant features to the illness severity level. The condition feature also has a great impact on the performance. Not using the condition feature leads to a 0.15 lesser F_1 score of the XGBoost model and a 0.20 F_1 score of the LGBM model. The F_1 score is affected slightly by the

FIGURE 8: F_1 scores of XGBoost and LightGBM with different feature selection settings.

weather condition. Not using this feature leads to a 0.12 lesser F_1 score of the XGBoost model and a 0.05 F_1 score of the LGBM model. Removing the sex feature or age feature does not change the F_1 score of the LGBM model and only have a 0.01 lesser F_1 score of the XGBoost model. This also shows that these numerical features illustrated in Figure 2 are not very relevant to the illness severity level.

5.2.2. Analysis of Parameter Tuning. In Figure 9, the maximum number of iterations is set in a range from 25 to 45 and both XGBoost and LightGBM have the best F_1 score when the value is set to 35. The learning rate tested for these two gradient boosting models is set in a range between 0.05 and 0.2, and its default value is 0.1. For better accuracy, we usually use a small learning rate with a large number of iterations; however, this may lead to a slower classification process. Both XGBoost and LightGBM reach the best F_1 score when the learning rate is set to 0.15. The maximum depth which is used to limit the tree depth is set between 3 and 10. A proper value for the maximum depth should be

FIGURE 9: Score of parameter tuning for XGBoost and LightGBM.

selected because the unconstrained tree depth can induce overfitting. XGBoost gets the best F_1 score when the maximum depth is set to 7, and LightGBM gets the best F_1 score when the value is set to 9. The number of leaves which is used to control the complexity of the tree model is set in a range from 30 to 70. XGBoost has the best F_1 score when the number of leaves is set to 50, and LightGBM has the best F_1 score when the value is set to 60.

5.3. Model Fusion. Model fusion is an effective way to improve the effect of machine learning. It is a machine learning method that integrates one model by training multiple submodels according to a certain method. The basic theoretical assumption is that different submodels have different capabilities on different data. By optimizing and combining the parts that they are good at, they can obtain a preferred model that is “accurate” in all aspects. Model fusion can integrate multiple “weak” models to obtain a “strong” model, which improves the final prediction ability on limited data and achieves better prediction results.

Stacking is one of the powerful model fusion methods. This method usually considers learning them in parallel and using a model to fuse the prediction results of each single model to reduce the generalization error of the single model. In stacking, effective features can be extracted well through multiple learners in the first layer.

In this part, we borrowed the ideas of stacking and merged the three models of ridge logistic regression, Gauss-

ian kernel SVM method, and LightGBM. Among them, ridge regression can create simple and efficient powerful models in the presence of a large number of features. By using L2 regularization, the error between the predicted value and the true value is minimized. SVM is a supervised learning model that distinguishes the results of each classification by fitting a boundary function. Using the Gaussian kernel function, the boundary function can be divided nonlinearly. LightGBM is an efficient gradient promotion decision tree that supports efficient parallel training. It has faster training speed, lower memory consumption, better accuracy, and distributed support, which can quickly process massive data. The specific structure is illustrated in Figure 10.

The results show that the accuracy of classification was improved to 93.5% by the model fusion method which is better than all previous models.

The F_1 score of sensitivity analysis has been displayed in Table 5 by removing one critical feature each time. Compared with all other features, the symptom features have the largest drop in score, followed by the treatment feature and the condition feature, which shows that the fusion model also effectively reflects the important impacts of these three features on the illness severity level. The table also shows that the three numerical features of weather, sex, and age only have a slight effect on the F_1 score, indicating that they are not very relevant to the illness severity level of the fusion model.

FIGURE 10: Structure of model fusion.

TABLE 5: Sensitivity analysis of model fusion.

Excluded feature	F_1 score
No excluded	93.5%
Age	91.2%
Sex	92.0%
Weather	90.1%
Condition	88.3%
Treatment	86.9%
Symptom	84.1%

6. Conclusion

This paper presented an illness severity-level model to predict the severity of a patient's chronic autoimmune disease. Firstly, we selected ten relative features based on the initial dataset and compared seven different classification models. The comparison has shown that the XGBoost model and LightGBM model have the best performance. Then, feature selection was used to select the most relative features to analyze the feature effects on chronic diseases. Meanwhile, we performed parameter tuning for the XGBoost model and LightGBM model to achieve better a F_1 score. The final F_1 score of the XGBoost model is 0.85 as well as the LightGBM model. At last, we applied a model integration system for the illness severity-level model to improve the accuracy of our result and the final accuracy that we got was 93.3%. The results indicated that our illness severity-level model has great prediction on the illness severity and patients may use this model to remind themselves the changes in health.

Since our study only applied a simple model for the illness severity level which ranged from zero to three, further medical information should be used to generate a more scientific model. Furthermore, our research needs to be improved in the setting of the illness severity level, because the lack of a corresponding medical background makes evaluation of illness worthy of medical significance. In addition, we can try different multimodel fusion strategies to get better research results. Different kinds of model ensemble strategies could enhance the ability of estimating the severity of a chronic autoimmune disease.

Data Availability

All data are available on request.

Conflicts of Interest

This work has no conflicts of interest.

References

- [1] W. Raghupathi and V. Raghupathi, "An empirical study of chronic diseases in the United States: a visual analytics approach to public health," *International Journal of Environmental Research and Public Health*, vol. 15, no. 3, p. 431, 2018.
- [2] G. Anderson and J. Horvath, "The growing burden of chronic disease in America," *Public Health Reports*, vol. 119, no. 3, pp. 263–270, 2004.
- [3] R. Nugent, "Chronic diseases in developing countries: health and economic burdens," *Annals of the New York Academy of Sciences*, vol. 1136, no. 1, pp. 70–79, 2010.

- [4] H. Promotion, *The Power of Prevention; Chronic Disease . The Public Health Challenge of the 21st Century*, Health Promotion, 2009.
- [5] E. H. Wagner, "Chronic disease management: what will it take to improve care for chronic illness?," *Effective Clinical Practice*, vol. 1, no. 1, 1998.
- [6] T. Bodenheimer, K. Lorig, H. Holman, and K. Grumbach, "Patient self-management of chronic disease in primary care," *JAMA*, vol. 288, no. 19, pp. 2469–2475, 2002.
- [7] D. Fu, H. Fu, P. McGowan et al., "Implementation and quantitative evaluation of chronic disease self-management programme in Shanghai, China: randomized controlled trial," *Bulletin of the World Health Organization*, vol. 81, no. 3, pp. 174–182, 2003.
- [8] F. Jiang, Y. Jiang, H. Zhi et al., "Artificial intelligence in healthcare: past, present and future," *Stroke and Vascular Neurology*, vol. 2, no. 4, pp. 230–243, 2017.
- [9] T. Davenport and R. Kalakota, "The potential for artificial intelligence in healthcare," *Future Healthcare Journal*, vol. 6, no. 2, pp. 94–98, 2019.
- [10] H. C. Koh and G. Tan, "Data mining applications in healthcare," *Journal of Healthcare Information Management*, vol. 19, 2011.
- [11] J. Wiens and E. S. Shenoy, "Machine learning for healthcare: on the verge of a major shift in healthcare epidemiology," *Clinical Infectious Diseases*, vol. 66, no. 1, pp. 149–153, 2018.
- [12] S. u. Rehman, S. Tu, Z. Shah et al., "Deep learning models for intelligent healthcare: implementation and challenges," in *International Conference on Artificial Intelligence and Security*, pp. 214–225, Springer, 2021.
- [13] X. Cao, J. Yao, Z. Xu, and D. Meng, "Hyperspectral image classification with convolutional neural network and active learning," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 58, no. 7, pp. 4604–4616, 2020.
- [14] D. Hong, L. Gao, N. Yokoya et al., "More diverse means better: multimodal deep learning meets remote-sensing imagery classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 5, pp. 4340–4354, 2020.
- [15] M. A. Calin, S. V. Parasca, D. Savastru, and D. Manea, "Hyperspectral imaging in the medical field: present and future," *Applied Spectroscopy Reviews*, vol. 49, no. 6, pp. 435–447, 2014.
- [16] D. Hong, L. Gao, J. Yao, B. Zhang, A. Plaza, and J. Chanussot, "Graph convolutional networks for hyperspectral image classification," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 59, no. 7, pp. 5966–5978, 2021.
- [17] D. Hong, N. Yokoya, J. Chanussot, and X. X. Zhu, "An augmented linear mixing model to address spectral variability for hyperspectral unmixing," *IEEE Transactions on Image Processing*, vol. 28, no. 4, pp. 1923–1938, 2018.
- [18] K.-J. Kim and I. Tagkopoulos, "Application of machine learning in rheumatic disease research," *The Korean Journal of Internal Medicine*, vol. 34, no. 4, pp. 708–722, 2019.
- [19] G. R. Bojja and O. El-Gayar, "Predicting hospital readmissions of diabetic patients-a machine learning approach," *Annual Research Symposium*, vol. 24, 2019.
- [20] K.-H. Lin and Y.-J. Hu, "Application of machine learning to immune disease prediction," *International Journal of Engineering and Innovative Technology*, vol. 7, 2018.
- [21] D. Jain and V. Singh, "Feature selection and classification systems for chronic disease prediction: a review," *Egyptian Informatics Journal*, vol. 19, no. 3, pp. 179–189, 2018.
- [22] J. Li, Q. Chen, and B. Liu, "Classification and disease probability prediction via machine learning programming based on multi-GPU cluster MapReduce system," *The Journal of Supercomputing*, vol. 73, no. 5, pp. 1782–1809, 2017.
- [23] D. E. Orange, P. Agius, E. F. DiCarlo et al., "Identification of three rheumatoid arthritis disease subtypes by machine learning integration of synovial histologic features and RNA sequencing data," *Arthritis & Rheumatology*, vol. 70, no. 5, pp. 690–701, 2018.
- [24] M. Shouman, T. Turner, and R. Stocker, "Using decision tree for diagnosing heart disease patients," *Proceedings of the Ninth Australasian Data Mining Conference*, vol. 121, pp. 23–30, 2011.
- [25] S. Habibi, M. Ahmadi, and S. Alizadeh, "Type 2 diabetes mellitus screening and risk factors using decision tree: results of data mining," *Global Journal of Health Science*, vol. 7, no. 5, pp. 304–310, 2015.
- [26] A. Wang, N. An, Y. Xia, L. Li, and G. Chen, "A logistic regression and artificial neural network-based approach for chronic disease prediction: a case study of hypertension," in *2014 IEEE International Conference on Internet of Things (iThings), and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom)*, pp. 45–52, Taipei, Taiwan, September 2014.
- [27] M. Chen, Y. Hao, K. Hwang, L. Wang, and L. Wang, "Disease prediction by machine learning over big data from healthcare communities," *Ieee Access*, vol. 5, pp. 8869–8879, 2017.
- [28] D. Cascio, V. Taormina, and G. Raso, "Deep CNN for IIF images classification in autoimmune diagnostics," *Applied Sciences*, vol. 9, no. 8, 2019.
- [29] J. Soni, U. Ansari, D. Sharma, and S. Soni, "Intelligent and effective heart disease prediction system using weighted associative classifiers," *International Journal on Computer Science and Engineering*, vol. 3, no. 6, pp. 2385–2392, 2011.
- [30] G. Kaur and A. Chhabra, "Improved J48 classification algorithm for the prediction of diabetes," *International Journal of Computer Applications*, vol. 98, no. 22, pp. 13–17, 2014.
- [31] P. N. Srinivasu, J. G. SivaSai, M. F. Ijaz, A. K. Bhoi, W. Kim, and J. J. Kang, "Classification of skin disease using deep learning neural networks with MobileNet V2 and LSTM," *Sensors*, vol. 21, no. 8, 2021.
- [32] I. Ruczinski, C. Kooperberg, and M. LeBlanc, "Logic regression," *Journal of Computational and Graphical Statistics*, vol. 12, no. 3, pp. 475–511, 2003.
- [33] A. Ramezankhani, O. Pournik, J. Shahrabi, D. Khalili, F. Azizi, and F. Hadaegh, "Applying decision tree for identification of a low risk population for type 2 diabetes. Tehran Lipid and Glucose Study," *Diabetes Research and Clinical Practice*, vol. 105, no. 3, pp. 391–398, 2014.
- [34] S. R. Safavian and D. Landgrebe, "A survey of decision tree classifier methodology," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 21, no. 3, pp. 660–674, 1991.
- [35] Y. Kumar and G. Sahoo, "Prediction of different types of liver diseases using rule based classification model," *Technology and Health Care*, vol. 21, no. 5, pp. 417–432, 2013.
- [36] M. Khalilia, S. Chakraborty, and M. Popescu, "Predicting disease risks from highly imbalanced data using random forest," *BMC Medical Informatics and Decision Making*, vol. 11, no. 1, 2011.
- [37] V. Svetnik, A. Liaw, C. Tong, J. C. Culberson, R. P. Sheridan, and B. P. Feuston, "Random forest: a classification and

- regression tool for compound classification and QSAR modeling,” *Journal of Chemical Information and Computer Sciences*, vol. 43, no. 6, pp. 1947–1958, 2003.
- [38] C. Strobl, A.-L. Boulesteix, T. Kneib, T. Augustin, and A. Zeileis, “Conditional variable importance for random forests,” *BMC Bioinformatics*, vol. 9, no. 1, 2008.
- [39] T. S. Furey, N. Cristianini, N. Duffy, D. W. Bednarski, M. Schummer, and D. Haussler, “Support vector machine classification and validation of cancer tissue samples using microarray expression data,” *Bioinformatics*, vol. 16, no. 10, pp. 906–914, 2000.
- [40] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, “Gene selection for cancer classification using support vector machines,” *Machine Learning*, vol. 46, no. 1/3, pp. 389–422, 2002.
- [41] D. A. Omondigbe, S. Veeramani, and A. S. Sidhu, “Machine learning classification techniques for breast cancer diagnosis,” *IOP Conference Series: Materials Science and Engineering*, vol. 495, no. 1, 2019.
- [42] S. Tong and D. Koller, “Support vector machine active learning with applications to text classification,” *Journal of Machine Learning Research*, vol. 2, pp. 45–66, 2001.
- [43] W. S. Noble, “What is a support vector machine?,” *Nature Biotechnology*, vol. 24, no. 12, pp. 1565–1567, 2006.
- [44] T. Chen and C. Guestrin, “Xgboost: a scalable tree boosting system,” in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, San Francisco, CA, USA, August 2016.
- [45] B. Pan, “Application of XGBoost algorithm in hourly PM_{2.5} concentration prediction,” *IOP Conference Series: Earth and Environmental Science*, vol. 113, no. 1, 2018.
- [46] G. Ke, Q. Meng, T. Finley et al., “Lightgbm: a highly efficient gradient boosting decision tree,” *Advances in Neural Information Processing Systems*, vol. 30, pp. 3146–3154, 2017.
- [47] D. Wang, Y. Zhang, and Y. Zhao, “LightGBM: an effective miRNA classification method in breast cancer patients,” in *Proceedings of the 2017 International Conference on Computational Biology and Bioinformatics*, pp. 7–11, Newark, NJ, USA, October 2017.