Article

Early Risk Prediction of Diabetes Based on GA-Stacking

Yaqi Tan ¹, He Chen ¹, Jianjun Zhang ², Ruichun Tang ¹ and Peishun Liu ^{1,*}

¹ College of Information Science and Technology, Ocean University of China, Qingdao 266100, China; tanyaqi@stu.ouc.edu.cn (Y.T.); chenhe@stu.ouc.edu.cn (H.C.); tangruichun@ouc.edu.cn (R.T.)

² Qingdao Center for Disease Control and Prevention, Qingdao 266033, China; 13589346072@163.com

* Correspondence: liups@ouc.edu.cn

Abstract: Early risk prediction of diabetes could help doctors and patients to pay attention to the disease and intervene as soon as possible, which can effectively reduce the risk of complications. In this paper, a GA-stacking ensemble learning model is proposed to improve the accuracy of diabetes risk prediction. Firstly, genetic algorithms (GA) based on Decision Tree (DT) is used to select individuals with high adaptability, that is, a subset of attributes suitable for diabetes risk prediction. Secondly, the optimized convolutional neural network (CNN) and support vector machine (SVM) are used as the primary learners of stacking to learn attribute subsets, respectively. Then, the output of CNN and SVM is used as the input of the mate learner, the fully connected layer, for classification. Qingdao desensitization physical examination data from 1 January 2017 to 31 December 2019 is used, which includes body temperature, BMI, waist circumference, and other indicators that may be related to early diabetes. We compared the performance of GA-stacking with K-nearest neighbor (KNN), SVM, logistic regression (LR), Naive Bayes (NB), and CNN before and after adding GA through the average prediction time, accuracy, precision, sensitivity, specificity, and F1-score. Results show that prediction efficiency can be improved by adding GA. GA-stacking has higher prediction accuracy. Moreover, the strong generalization ability and high prediction efficiency of GA-stacking have also been verified on the early-stage diabetes risk prediction dataset published by UCI.

Citation: Tan, Y.; Chen, H.; Zhang, J.; Tang, R.; Liu, P. Early Risk Prediction of Diabetes Based on GA-Stacking. *Appl. Sci.* **2022**, *12*, 632. <https://doi.org/10.3390/app12020632>

Academic Editors: Giancarlo Mauri and Alexander N. Pisarchik

Received: 24 September 2021

Accepted: 5 January 2022

Published: 10 January 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: diabetes; machine learning; stacking; CNN; SVM; GA

1. Introduction

Diabetes is one of the most common chronic diseases, which may cause various complications such as cardiovascular disease and kidney disease. It not only reduces the patient's quality of life, but it also places a heavy burden on the patient [1].

Currently, researchers are trying to create a variety of systems and tools that can assist doctors in predicting diabetes. Machine learning algorithms are used in the research of diabetes risk prediction, which can predict and classify diabetes by analyzing early symptoms of diabetes. Kumari et al. [2] used the SVM as the classifier and Radial basis function (RBF) as the kernel function. The classification accuracy of SVM is increased to 78.0% in the Pima Indian diabetes dataset, which verifies the effectiveness of the support vector machine model for the diagnosis of diabetes. The system proposed by Islam et al. in [3] inputs user symptoms into Naive Bayes (NB), decision trees (DT), logistic regression (LR), and random forest (RF) for diabetes risk prediction. In the early-stage diabetes risk prediction dataset published by UCI, the accuracy of the random forest classifier is 97.40%, which is the highest accuracy. Alpan et al. [4] used WEKA Tool to conduct a study comparing diabetes data mining classification techniques. In the UCI dataset, seven algorithms, including Bayesian Network, Naive Bayes, decision tree (J48), random tree, random forest, KNN, and SVM, have been compared experimentally. Among them, the accuracy of the KNN algorithm is the highest after using the 10-fold cross-validation technique to split the training dataset and the test dataset, reaching 98.07%. With the advancement of research, the methods used for diabetes risk prediction have gradually

shifted from traditional machine learning algorithms to deep learning algorithms such as neural network (NN), CNN, and long short-term memory (LSTM) with higher accuracy. Chaves Luís and Marques Gonçalo [5] applied NN to diabetes risk prediction on early-stage diabetes risk prediction dataset. Its AUC and accuracy rates are 98.3% and 98.1%, respectively, which are better than machine learning algorithms such as NB, KNN, SVM, and RF. They proved that NN can be used for diabetes prediction. Rahman et al. [6] used the Conv-LSTM-based model for diabetes risk prediction for the first time. They compared Conv-LSTM, T-LSTM, CNN, and CNN-LSTM using the Pima Indian Diabetes dataset. Among them, the Conv-LSTM was superior to the other three models in terms of classification accuracy, which showed that the attribute extraction ability of CNN can improve the accuracy of the classification model.

Generally, the performance of one classification model is limited. Besides, there are some problems of it such as weak generalization ability and poor fault tolerance, which makes the effect of disease prediction and diagnosis not ideal. Therefore, ensemble learning strategies have emerged [7]. Ensemble learning built a model based on the idea of integrating weak classifiers into strong classifiers. David H. Wolpert [8] proposed the stacking ensemble strategy that can minimize the error rate of one or more generalizations. Compared with the boosting method proposed by Schapire et al. and the bagging method proposed by Leo Breiman, the original training data set is used by stacking to train the primary learner. Then, the output of the primary learner is processed by the meta learner, which improved the ability of one model [9]. Stacking is now used in multiple fields. Ail et al. [10] used stacking ensemble strategy for breast cancer-related amino acid sequence prediction. NB, KNN, SVM, and RF are used as primary learners of the proposed model. Genetic programming (GP) is used as the meta learner. Compared to basic machine learning algorithms and other traditional ensemble strategies, the model proposed by Ail had more steady performance.

The key to using machine learning algorithms to assist diabetes risk prediction is to extract valid information from the attributes set. However, the large dimension of the attributes set has brought a huge impact on training [11,12]. Scholars began to try different feature selection methods. Ismail et al. [13] used three diabetes datasets to evaluate the prediction ability of the model combined with nine feature selection algorithms and 35 machine learning algorithms. They found that performing feature selection operations on the dataset before the classification model can reduce the execution time while avoiding data overfitting. Feature selection methods are mainly divided into filtering method, packaging method, and embedding method. GA based on the input wrapping method can successfully extract attribute subsets and improve the accuracy of prediction. Cerrada et al. [14] used GA to reduce the attribute set. GA was combined with the random forest to predict multiple types of fault diagnosis for spur gears. When the original condition attribute set is reduced by 34%, 97% classification accuracy can still be obtained. In the field of diabetes risk prediction, Li X et al. [15] combined GA with K-means clustering algorithm for feature selection and then used KNN for classification. Compared with previous related studies, the model performed better.

As an important means of early screening of risk factors of diabetes and other chronic diseases, physical examination data provide the possibility for machine learning to assist doctors in early diabetes risk prediction [16]. However, due to the large amount of attribute sets of physical examination data, the performance of the classification model is limited. Moreover, the models used for diabetes risk prediction generally have weak generalization ability and are only suitable for a single dataset. Therefore, GA-stacking ensemble learning model is proposed in this paper. Firstly, GA based on DT is used to extract attribute subsets of preprocessed data. Secondly, the training dataset is subjected to five-fold cross-validation, where the four-fold data are used to train CNN and SVM as the primary learners of stacking, and the remaining one-fold data are classified by the primary learner as the training dataset of the fully connected layer to avoid the risk of over-fitting. The main highlights of this article are as follows:

GA-stacking ensemble learning model is proposed in this paper. According to stacking, the proposed model is combined with the attribute extraction ability of CNN and the advantages of SVM to deal with binary classification problems. In addition, through the neurons in a fully connected layer, the classification error rate of the primary learner is decreased.

After preprocessing the physical examination dataset and early-stage diabetes risk prediction dataset, we implemented KNN, SVM, LR, NB, CNN, and XBNNet proposed in [17]. The proposed GA-stacking not only outperforms the prediction performance and efficiency of the basic machine learning algorithm and the model implemented in the latest studies, but it also has stronger generalization capabilities.

The rest of the paper is structured as follows. Materials and methods used are illustrated in Section 2. GA-stacking is proposed and presented in Section 3. Experiments and results analysis are presented in Section 4. The last section summarizes the main work and future research directions of this paper.

2. Materials and Methods

2.1. Stacking

Stacking is an ensemble strategy proposed by David H. Wolpert [8]. It can minimize the error rate of one or more generalizations. It integrates primary learners by adding meta learners.

The detailed steps of stacking are shown in the Figure 1. Firstly, the dataset is divided into training dataset and testing dataset. Secondly, the training dataset is divided into K equal parts, taking five parts as an example. The four parts of the dataset are used to train the primary learner, and the remaining part of the dataset predicted and classified by the primary learner is used as the training dataset of the meta learner. Once stacking is trained, it can be used for diabetes risk prediction. Given the testing dataset, the information extracted by the primary learner five times are averaged as the input of the meta learner for prediction. For the problems of limited predictive ability and weak generalization ability of one classification model for diabetes risk prediction, stacking could integrate the advantages of multiple learners to make the model suitable for multiple datasets.

Figure 1. Detailed steps for stacking integrated five-fold cross-validation.

2.2. CNN

CNN is a feedforward neural network with convolution calculation, which is one of the representative algorithms of deep learning [15]. CNN is a multi-layer neural network. Its core parts are the convolutional layer and pooling layer, which can effectively learn potential information from a large number of samples [18]. For prediction work with large attribute sets such as diabetes risk prediction, as one of the primary learners of stacking, CNN can give full play to the feature extraction advantages of itself and make up for the shortcomings of other primary learners that miss important information.

CNN consists of an input layer, convolutional layer, pooling layer, fully connected layer, and output layer [19]. The input layer is used to input the dataset that needs to be trained. CNN supports multiple types of datasets, such as image data, audio data, etc. The

convolutional layer is the core part of CNN [20]. The convolution kernel regularly slides through the input matrix, which can extract physical examination potential information. The work of the convolution kernel is to multiply and sum the corresponding elements of the matrix in the receptive field and accumulate the offset:

$$ConvLay_j^l = \sum_{i \in M_j} X_i^{l-1} \times K_{i,j}^l + b_j^l \quad (1)$$

where l indicates the current network layer number, X_i^{l-1} represents the input of neurons at layer $l - 1$, $K_{i,j}^l$ is the convolution kernel function of the current layer, and b_j^l is the offset parameter.

Another core part of CNN is the pooling layer. It reduces the dimensionality of the information extracted by the convolutional layer, which improves the robustness of the information. Commonly used pooling methods are maximum pooling and average pooling. The output of the pooling layer is calculated by the Equation (2).

$$PoolLay_j^l = \beta_j^l Pooling(X_i^{l-1}) + b_j^l \quad (2)$$

where $Pooling (*)$ represents the pooling function. The maximum pooling function can extract the maximum value of the specified size area in the input matrix, and the output of the average pooling function is the average value of the specified size area in the input matrix. After training, the output of each pooling layer will correspond to an optimal multiplicative bias β and additive bias b [21].

The fully connected layer is located after several convolutional layers and pooling layers. Its function is to perform advanced reasoning on CNN. Neurons in this layer are fully connected to all activated neurons in the upper layer and transmit signals to other fully connected layers. The last layer is the output layer, which usually completes different tasks according to the purpose of the research. SoftMax function is generally used for classification.

2.3. SVM

SVM is a generalized linear classifier proposed by Vapnik in 1963. It optimizes model performance by minimizing the average classification error during training. SVM has a strong ability of binary classification [22]. As one of the primary learners of stacking for diabetes risk prediction, it can provide more accurate information for the mate learners.

For the training dataset x_i and the class label $y_i \in \{1 \text{ or } 0\}$, SVM constructs the optimal hyperplane decision function with the largest margin as shown in Equation (3) to achieve the maximum classification of x_i .

$$f(x) = \sum_{i=1}^n \alpha_i y_i K(x, x_i) - b \quad (3)$$

where $\alpha_i \geq 0, i = 1, 2, \dots, n$ is the Lagrangian multiplier, b is the offset, and K is the kernel function.

3. GA-Stacking

3.1. GA for the Feature Selection in Classifiers Based on DT

The feature dimension of physical examination data is relatively large, which is not suitable for direct use in diabetes prediction. Extracting representative attribute subsets from it can reduce not only the training time of model but also the probability of overfitting. There are three main methods of feature selection, namely the packing method, embedding method, and filtering method. GA based on the input packing method combined with DT is used in this paper. Its pseudocode is shown in Algorithm 1. The basic operation process of GA for feature selection is as follows:

1. Initialization: We set the number of evolutions to 100, the crossover probability to 0.6, the mutate probability to 0.01, and the number of chromosomes to 32, that is, the number of attributes in the physical examination dataset. We initialize the evolution counter $t = 0$. For the training data x_i of the physical examination dataset, whether to be selected or not is encoded in binary. When $x_i = 1$, it indicates that this attribute is selected, and $x_i = 0$ indicates that it is not selected. Each generation produces a population $P(t)$ composed of 20 individuals, and the first generation is the initial population $P(0)$.
2. Calculate individual fitness: Calculate the fitness of each individual in the population $P(t)$ of the current evolutionary generation. The attributes coded as 1 in the individual are classified by DT, and its F1-Score is used as the fitness of this individual.
3. Select, crossover, and mutate: The roulette wheel selection is applied to the population based on fitness. The higher the fitness, the higher the probability of being selected. The selected individual will be the parent of $P(t + 1)$. Firstly, according to the crossover probability $P_c = 0.6$, select two consecutive individuals in the population, and then exchange their gene selected by a random function to form a new individual. Secondly, according to the mutate probability $P_m = 0.01$, select some individuals in the population and select the position of the mutated gene through a random function. If the original gene is 0, it will be mutated to 1, and vice versa. After the above operations, $P(t + 1)$ is generated.

Repeat steps 2 and 3 until the number of evolutions is 100, then stop the loop. The individual with the greatest fitness in $P(100)$ is the output. The attribute coded as 1 in this output is the most representative attribute subset we need.

Algorithm 1 presents the pseudocode of the genetic algorithm for the feature selection in classifiers based on decision tree.

Algorithm 1 Genetic Algorithm for the feature selection in classifiers based on Decision Tree

Input: Attributes: $X = \{x_i\}_{i=1}^m$, Class: $Y \in \{0, 1\}$

Output: Binary coded attributes which are selected

1: Initial the number of evolutions: $T = 100$, the number of attributes: $n = 32$, Crossover probability: $P_c = 0.6$, Mutate probability: $P_m = 0.01$, Evolution counter: $t = 0$, Initial population with 20 randomly generated individuals: $P(0)$.

2: **function** FITNESS ($P(t)$):

3: **for** $i = 0; i < \text{sizeof}(P(t)); i++$ **do**

4: // Determine the selected attributes according to $P(t)$.

5: **for** $j = 0; j < \text{sizeof}(P(t)[i]); j++$ **do**

6: **if** $P(t)[i][j] == 1$ **then**

7: Attributes_{position}.push(j);

8: **end if**

9: **end for**

10: $X = \{x_i\}$ Attributes_{position};

11: Score($p(t)$).push(F1score(X, Y));

12: **end for**

13: **return** Score($p(t)$);

14: **end function**

15: **while** $t < T + 1$ **do**

16: // Individuals with higher fitness are more likely to be selected.

17: Parent1=Select($P(t)$, FITNESS($P(t)$)), Parent2=Select($P(t)$, FITNESS($P(t)$));

18: // Randomly select the position of the gene value and exchange the selected gene value of Parent1 and Parent2.

19: Child=Crossover(Parent1, Parent2, P_c);

20: // Randomly select the position of the mutated gene value. If the original gene value is 0, it will be mutated to 1, and vice versa.

21: Child=Mutate(Child, P_m);

22: $P(t+1)$.push(Child);

23: **end while**

24: **return** Max(FITNESS($P(T)$))

3.2. Stacking Based on CNN and SVM

CNN rely on the convolutional layer and the pooling layer to extract potential information in the input data. The fully connected layer is used to integrate the information, and the output layer is used to classify. For the physical examination dataset, CNN with a small number of layers can not only take advantage of its fewer parameters, no gradient explosion, and gradient disappearance, etc., but it can also efficiently and accurately explore the potential relationships between attributes [23]. In order to make the CNN more suitable for diabetes risk prediction on physical examination dataset, stacking based on CNN and SVM is proposed. The fully connected layer is used as the mate learner to process the output results of CNN and SVM. The model network structure is shown in Figure 2.

Figure 2. Network structure of GA-stacking. (a) shows the structure of GA-Stacking. (b) shows the structure of simplified CNN. (c) shows the structure of SVM.

The amount of physical examination datasets is relatively small, so we simplified the deep CNN structure, as shown in the Figure 2b. After the input attributes are expanded by the fully connected layer, they are processed twice by the convolutional layer, the pooling layer, and the activation layer. Then, the fully connected layer is connected for information integration. Finally, the SoftMax layer is used to obtain the classification result. The following is a detailed introduction to the used parts:

After the original 32-dimensional physical examination data has undergone feature selection, a 20-dimensional attribute subset is obtained. To avoid the convolutional layer and the pooling layer from ignoring the necessary information, a fully connected layer is added after the input layer to expand the 20-dimensional to 36-dimensional:

$$X'_{36 \times 1} = W_{36 \times 20}^T X_{20 \times 1} + B_{36 \times 1} \quad (4)$$

where X is the input matrix, X' is the matrix after expanding, and W and B are the parameter matrices of weight and bias.

Convolutional layer: Since the attribute information of the matrix is relatively scattered, padding is used to add 0 to the matrix, in order to prevent the information of the matrix edge from being lost during the convolution.

Pooling layer: For the physical examination dataset, maximum pooling is used to extract the maximum value of the local area, which not only retains the information that has the greatest impact on diabetes classification, but also effectively avoids information loss during dimensionality reduction.

Activation layer: The activation layer is used after convolutional and pooling layers, which can capture the non-linear factors of the physical examination data and extract effective information. We use *ReLU*, which converges fastest, as the activation function:

$$f(x) = \max(0, x) \quad (5)$$

Fully connected layer: After potential information extraction, the fully connected layer converts the learned information of the physical examination attributes matrix into feature vectors for advanced reasoning. The neurons in this layer are fully connected to the neurons in the activation layer and transmit signals to the output layer or the second part of the convolutional layer.

For another primary learner SVM, our main purpose is to construct an optimal decision function. It divides the 20-dimensional vector physical examination data into two categories. For the attribute x_i and the class label $y_i \in \{1 \text{ or } 0\}$, different kernel functions are used to construct the optimal hyperplane decision function with the largest margin. In this study, we compared the linear kernel, polynomial kernel, and radial basis function. Among them, the radial basis function, as shown in the Equation (6), has the best classification effect.

$$K(x, x_i) = \exp\left(-\frac{\|x - x_i\|^2}{2\sigma^2}\right) \quad (6)$$

where, σ is the width parameter of the function, which controls the radial range of the function.

Stacking improves the classification ability and generalization ability of the model by combining multiple learners. Therefore, based on the idea of it, the fully connected layer is used to learn the output of CNN and SVM. It adjusts the dimensionality to serve as the input to the SoftMax layer for classification. In the output layer, Equation (7) is used to calculate the probability in the fully connected neurons.

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{c=1}^2 e^{z_c}} \quad (7)$$

where z_i represents the i -th input signal received from the fully connected layer, and the denominator represents the exponential sum of all input signals in the two output neurons. We judge whether the patient will suffer from diabetes or not by the probability value of the two output neurons. If the probability value is more than 0.5, the patient is judged to have higher risk of diabetes, and if the value is less than or equal to 0.5, the patient is considered to have lower risk of diabetes.

4. Results

4.1. Data Set and Data Preprocessing

The dataset used in this paper is 8787 desensitization physical examination data from Qingdao CDC. Each piece of data in the dataset includes the desensitization information of the user, such as sex, date of birth, date of physical examination, body temperature, breathing rate, pulse rate, and other individual examination items, as show in Table A1. At the same time, in order to verify the generalization ability of GA-stacking, the early-stage diabetes risk prediction dataset published by UCI [3] is used, as show in Table A2. It has 520 instances and 16 attributes related to diabetes, such as age, sex, polyuria, polydipsia, etc.

Before classification, we performed data preprocessing, including data cleaning, coding, discretization, normalization, and dataset division.

1. Data cleaning: Before data modeling, data cleaning can make the model more effectively extract the actual group characteristics. For the physical examination dataset, mode imputation was used to fill the samples with a few missing attribute columns. Samples with more than 10 attribute columns missing were deleted.
2. Data encoding: One-hot encoding was performed on the sex attribute to make the calculation of the loss function more reasonable and improve the accuracy of the model.
3. Data discretization: In the physical examination data, some samples were at the same age. After discretization, the model will be more stable. Moreover, the risk of fitting will be reduced.
4. Data normalization: After the dataset was normalized using the Equation (8), the convergence speed and accuracy of the model could be effectively improved.

$$x' = \frac{x - \mu}{\sigma} \quad (8)$$

where, μ is the mean value of the age, and σ is the standard deviation.

Dataset division: If the output data of training the primary learner is used to directly train the mate learner of stacking, it will cause the risk of overfitting. Therefore, we used a five-fold cross-validation method to process the two datasets. Firstly, datasets are divided into training dataset and testing dataset in 7:3. Then, the training dataset is divided into five equal parts, of which four-fold data are used to train CNN and SVM, and the remaining one-fold data are predicted and classified by the primary learner as the training dataset of the meta learner. For the testing dataset, the information extracted by the primary learner five times are averaged as the input of the mate learner for classification output.

4.2. Evaluation Criteria

To evaluate the classification effect of the GA-stacking model, accuracy, precision, sensitivity, and specificity defined by the confusion matrix parameters and F1-measure are used as the Performance metrics.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (9)$$

$$Precision = \frac{TP}{TP + FP} \quad (10)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (11)$$

$$Specificity = \frac{TN}{TN + FP} \quad (12)$$

$$F1 = \frac{2 \times Precision \times Sensitivity}{Precision + Sensitivity} \quad (13)$$

Accuracy represents the proportion of the correct samples predicted by the model. Precision is the proportion of the samples diagnosed as diabetes by the model that actually have diabetes. Sensitivity in this article refers to the probability that the diagnosis is correct in patients with diabetes. Specificity in this article refers to the probability that the diagnosis is correct in people without diabetes. The F1-measure is the harmonic mean value between the precision in Equation (10) and the sensitivity of Equation (11).

4.3. Experiments and Results

The software environment of the experiment is Windows 10, which is equipped with Intel Core (TM) i5-8400 CPU@2.8GHz, 8G RAM, NVIDIA 1050ti graphics card. The deep learning algorithms used are Pytorch 1.4.0.

4.3.1. Evaluation between Different Models on Qingdao Physical Examination Dataset

In order to have a clear understanding of the correlation of the attributes in the Qingdao physical examination dataset, we have drawn a feature correlation heatmap, as shown in Figure 3. The bluer the color in the picture, the smaller the correlation, and the more purple the color, the higher the correlation. It can be seen from Figure 3 that the redundancy between most of the attributes is small, but the redundancy between a small part of the attributes is relatively large. So, it is necessary to extract a representative attribute subset.

Figure 3. Feature correlation heatmap.

Table 1 shows the performance comparison of KNN, SVM, LR, NB, CNN, and stacking using GA or not. It can be seen from the table that adding GA before data input to the model can reduce the prediction time, while the accuracy, precision, sensitivity, specificity, and F1-score increase by more than 0.1%. It also shows that on the redundant dataset, using GA not only effectively improves the prediction efficiency, but is also helpful to improve the performance of the model.

Comparing with the performance of KNN, SVM, LR, NB, and CNN, GA-CNN showed the best performance. The average accuracy is 85.08%, and the average precision, sensitivity, specificity, and F1-score are 100%, 30.10%, 100%, and 46.27%, respectively. The performance of GA-SVM is second only to GA-CNN, with an accuracy of 84.48%, and its precision, sensitivity, specificity, and F1-score are better than other machine learning

algorithms. Finally, SVM is selected as another primary learner of stacking, so that the proposed algorithm could extract potential information while being suitable for binary classification problems.

Table 1. Performance of different modules on the Qingdao physical examination data set.

Model	Time	Acc (%)	Pre (%)	Sen (%)	Spe (%)	F1-Score (%)
KNN	0.637	79.15	50.74	27.41	92.91	35.60
GA-KNN	0.614	81.07	58.22	35.21	93.27	43.89
SVM	1.671	84.36	100	30.37	100	46.59
GA-SVM	1.572	84.48	100	30.91	100	47.22
LR	0.085	84.07	95.76	29.57	99.64	45.18
GA-LR	0.067	84.10	98.21	30.37	99.85	46.39
NB	0.175	84.25	89.92	32.53	98.99	48.92
GA-NB	0.137	84.45	97.58	33.60	99.78	48.92
CNN	24.360	84.97	100	29.30	100	45.32
GA-CNN	21.857	85.08	100	30.10	100	46.27
Stacking	52.500	85.14	94.30	36.29	98.74	52.41
GA-stacking	48.652	85.88	96.12	39.24	99.92	55.73

Table 1 shows the performance of the proposed GA-stacking on the Qingdao physical examination dataset. The average accuracy is 85.88%, and the average precision, sensitivity, specificity, and F1-score are 96.12%, 39.24%, 99.92%, and 55.73%, respectively. Comparing the performance of machine learning algorithms in the table, the accuracy has increased by more than 1%, and the F1-score has increased by more than 7%. The above results indicate that GA-stacking is superior to other machine learning methods in performance metrics such as accuracy, precision, recall, and F1-score. So, GA-stacking is more suitable for the early prediction of diabetes.

4.3.2. Evaluation between Different Models on the Early-Stage Diabetes Risk Prediction Dataset Published by UCI

In addition, we also verified the generalization ability of GA-stacking on the early-stage diabetes risk prediction dataset published by UCI. To obtain higher-accuracy results, data preprocessing is performed, which includes coding, discretization, normalization, and dataset division. Table 2 lists the performance comparisons of KNN, SVM, LR, NB, CNN, and GA-stacking. The model we proposed is more effective in predicting diabetes risk. In terms of accuracy, compared with traditional machine learning algorithms, it has been improved. It is also 9.61%, 5.17%, and 1.92% higher than NB, LR, and KNN, respectively. In addition, comparing the accuracy of the two primary learners of CNN and SVM, the accuracy has been increased by 1.92% and 1.91%. In terms of precision, sensitivity, and FI score, GA-stacking reached 100%, 96.77%, and 98.36%. Except for the slightly lower sensitivity, all other performance metrics achieved the best results.

Table 2. Performance of different modules on the UCI data set.

Model	Acc (%)	Pre (%)	Sen (%)	Spe (%)	F1-Score (%)
KNN	96.79	93.84	98.38	95.74	96.06
SVM	96.79	96.72	95.16	97.87	95.93
LR	93.54	93.33	90.32	95.74	91.80
NB	89.10	90.90	80.64	94.68	85.47
CNN	96.79	98.86	93.30	98.93	95.86
GA-stacking	98.71	100	96.77	100	98.36

Several similar works are available in the literature. Table 3 presents the comparative results between studies. The system architecture proposed by M. M. Faniqul Islam et al. used NB, J48, LR, and RF for risk prediction. Among them, RF had the highest accuracy

rate of 97.40% [3]. Alpan et al. conducted experimental comparisons on seven algorithms including BN, NB, J48, RT, RF, KNN, and SVM. Among them, the accuracy of the KNN after using the 10-fold cross-validation technique to split the training data set and the test data set was the highest, reaching 98.07% [4]. Marques Gonçalo applied neural network to diabetes risk prediction on the early-stage diabetes risk prediction dataset published by UCI. The AUC and accuracy were 98.3% and 98.1%, respectively, which were better than machine learning algorithms such as NB, KNN, SVM, and RF [5]. Tushar Sarkar et al. proposed XBNet, using gradient boosted tree updates the weights of each layer in the neural network, which increases the model's interpretability and performance. We conducted experiments on the UCI dataset according to the code provided by the author, and the final accuracy was 80.00% [17]. Comparing [3–5], we can find that the accuracy of NB, SVM, KNN, etc., with different data preprocessing methods is lower than GA-stacking. Moreover, in [17], when XBNet is used on different datasets, the accuracy fluctuates greatly, and the generalization ability is weak.

Table 3. Studies results comparison.

Study	NB	BN	J48	RF	RT	KNN	SVM	LR	AdaBoost	XBnet	GA-stacking
[3]	87.4	-	95.6	97.4	-	-	-	92.4	-	-	-
[4]	87.11	86.92	95.96	97.50	96.15	98.07	92.11	-	-	-	-
[5]	86.92	98.08	97.31	96.92	-	97.31	97.1	-	-	-	-
[17]	-	-	-	-	-	-	-	-	-	80.00	-
This study	89.10	-	-	-	96.79	-	96.79	93.54	-	-	98.71

In summary, comparing the algorithms implemented in [3–5,17] as well as KNN, SVM, LR, NB, and CNN, GA-stacking has achieved the highest accuracy rate. It is a good predictor of diabetes risk and is suitable for early diabetes risk prediction of patients.

5. Discussion

In this work, we proposed a GA-stacking ensemble learning model for diabetes risk prediction. Experiments have shown that the average precision, sensitivity, specificity, and F1-score of the proposed model are better than other machine learning models such as KNN, SVM, LR, NB, CNN, XBNet, etc. The average accuracy is 85.88%. So, GA-stacking has obvious advantages for diabetes risk prediction.

1. Using GA based on DT for feature selection effectively improves the speed of prediction and accuracy of the model.
2. Based on stacking, using a fully connected layer combined with two primary learners, CNN and SVM, can process the input more accurately and make the model have great generalization capabilities.

Although GA-stacking effectively improves the accuracy and speed of diabetes risk prediction and has a good generalization ability, it still has some limitations. For example, the model faces challenges in datasets with small attribute sets and unbalanced data. In the future, we will conduct research on how to scientifically and effectively expand the feature, solve the problem of class imbalance, and improve the accuracy and generalization ability of the model.

Author Contributions: Conceptualization, Y.T.; methodology, Y.T. and H.C.; software, H.C.; resources, J.Z.; data curation, J.Z.; writing—original draft preparation, Y.T.; writing—review and editing, Y.T., and R.T.; supervision, R.T. and P.L.; funding acquisition, J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This work is supported by Qingdao Science and Technology Development Plan (21-1-4-rkjjk-14-nsh).

Acknowledgments: This work is supported by Qingdao Science and Technology Development Plan (21-1-4-rkjjk-14-nsh). We are grateful to the anonymous reviewers for comments on the original manuscript.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A

Table A1. Attributes and description of desensitization physical examination data from Qingdao CDC.

Attributes	Attributes Description
Sex	1. Male 2. Female
Date of birth	1916/1–1969/1
Date of physical examination	2016–2019
Body temperature	35.8–37.3
Breathing rate	
Pulse rate	
Right side high blood pressure	
Right side low blood pressure	
Left side high blood pressure	
Left side blood pressure	
Height	
Waistline	
Weight	
BMI	Body Mass Index
WBC	White Blood Cell Count
HGB	Hemoglobin
PLT	Platelet count
PRO	Proteinuria < 0.1g/24 h: –; 0.1 g/24 h < Proteinuria < 0.2 g/24 h: ±; 0.2 g/24 h < Proteinuria < 1.0 g/24 h: +; 1.0 g/24 h < Proteinuria < 2.0 g/24 h: ++; 2.0 g/24 h < Proteinuria < 4.0 g/24 h: +++; Proteinuria > 4.0 g/24 h: ++++.
GLU	Glucose in urine < 2.8 mmol/L: –; Glucose in urine < 5.5 mmol/L: ±; Glucose in urine < 27.8 mmol/L: +; 27.8mmol/L < Glucose in urine < 55 mmol/L: ++; 55mmol/L < Glucose in urine < 111.1 mmol/L: +++; Glucose in urine > 111.1 mmol/L: ++++.
KET	Ketone < 0.5 mmol/L: –; 0.5 mmol/L < Ketone < 1.5 mmol/L: +; 1.5 mmol/L < Ketone < 3.9 mmol/L: ++; 3.9 mmol/L < Ketone < 7.8 mmol/L: +++; 7.8 mmol/L < Ketone < 15.6 mmol/L: ++++;
ERY	At high magnification, less than 10 RBC cells were found.: –; At high magnification, 10 erythrocytes were found.: +; At high magnification, 20 erythrocytes were found.: ++; At high magnification, 30 erythrocytes were found.: +++;
Uric acid	
SCR	Serum creatinine
BUN	
ALT	
AST	
ALB	
TBIL	
SCR	
BUN	
CHO	
TG	
LDLC	
HDLC	
Class	Fasting plasma glucose (FPG) < 7.5: 0; Fasting plasma glucose (FPG) ≥ 7.5: 1;

Table A2. Attributes and value of early-stage diabetes risk prediction dataset published by UCI.

Attributes	Values
Age	16–90
Sex	1. Male 2. Female
Polyuria	1. Yes 2. No
Polydipsia	1. Yes 2. No
Sudden weight loss	1. Yes 2. No
Weakness	1. Yes 2. No
Polyphagia	1. Yes 2. No
Genital thrush	1. Yes 2. No
Visual blurring	1. Yes 2. No
Itching	1. Yes 2. No
Irritability	1. Yes 2. No
Delayed healing	1. Yes 2. No
Partial paresis	1. Yes 2. No
Muscle stiffness	1. Yes 2. No
Alopecia	1. Yes 2. No
Obesity	1. Yes 2. No
Class	1. Positive 2. Negative

References

- Richard, H. The neglected epidemic of chronic disease. *Lancet* **2005**, *366*, 1514.
- Kumari, V.A.; Chitra, R. Classification of diabetes disease using support vector machine. *Int. J. Eng. Res. Appl.* **2013**, *3*, 1797–1801.
- Islam, M.M.F.; Ferdousi, R.; Rahman, S.; Bushra, H.Y. Likelihood Prediction of Diabetes at Early Stage Using Data Mining Techniques. In *Computer Vision and Machine Intelligence in Medical Image Analysis*; Springer: Singapore, 2020; pp. 113–125.
- Alpan, K.; Ilgi, G.S. Classification of Diabetes Dataset with Data Mining Techniques by Using WEKA Approach. In Proceedings of the 2020 4th International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), Istanbul, Turkey, 22–24 October 2020; pp. 1–7.
- Chaves, L.; Marques, G. Data Mining Techniques for Early Diagnosis of Diabetes: A Comparative Study. *Appl. Sci.* **2021**, *11*, 2218. [\[CrossRef\]](#)
- Rahman, M.; Islam, D.; Mukti, R.J. A deep learning approach based on convolutional LSTM for detecting diabetes. *Comput. Biol. Chem.* **2020**, *88*, 107329. [\[CrossRef\]](#) [\[PubMed\]](#)
- Kearns, M.; Valiant, L.G. *Learning Boolean Formulae or Finite Automata Is as Hard as Factoring*; Technical report TR-14-88; Harvard University Aiken Computation Lab: Cambridge, CM, USA, 1988.
- Wolpert, D.H. Stacked generalization. *Neural Netw.* **1992**, *5*, 241–259. [\[CrossRef\]](#)
- Schapire, R. The strength of weak learnability. *Mach. Learn.* **1990**, *5*, 197–227. [\[CrossRef\]](#)
- Ali, S.; Majid, A. Can-evo-ens: Classifier stacking based evolutionary ensemble system for prediction of human breast cancer using amino acid sequences. *J. Biomed. Inform.* **2015**, *54*, 256–269. [\[CrossRef\]](#) [\[PubMed\]](#)
- Cilia, N.D.; Stefano, C.D.; Fontanella, F. Variablelength representation for EC-based feature selection in high-dimensional data. In Proceedings of the International Conference on the Applications of Evolutionary Computation, Leipzig, Germany, 24–26 April 2019; pp. 325–340.
- Xue, B.; Zhang, M.; Browne, W.N.; Yao, X. A survey on evolutionary computation approaches to feature selection. *IEEE Trans. Evol. Comput.* **2015**, *20*, 606–626. [\[CrossRef\]](#)
- Ismail, L.; Materwala, H.; Tayefi, M. Type 2 Diabetes with Artificial Intelligence Machine Learning: Methods and Evaluation. *Arch. Comput. Methods Eng.* **2021**, *29*, 1–21. [\[CrossRef\]](#)
- Cerrada, M.; Zurita, G.; Cabrera, D.; Sánchez, R.V.; Artés, M.; Li, C. Fault diagnosis in spur gears based on genetic algorithm and random forest. *Mech. Syst. Signal Process.* **2016**, *70*, 87–103. [\[CrossRef\]](#)
- Li, X.; Zhang, J.; Safara, F. Improving the Accuracy of Diabetes Diagnosis Applications through a Hybrid Feature Selection Algorithm. *Neural Processing Lett.* **2021**, 1–17. Available online: <https://link.springer.com/article/10.1007/s11063-021-10491-0#citeas> (accessed on 1 December 2021). [\[CrossRef\]](#) [\[PubMed\]](#)
- Wei, J.; Shaofu, L. The Risk Prediction of Type 2 Diabetes based on XGBoost. In Proceedings of the 2019 2nd International Conference on Mechanical, Electronic and Engineering Technology (MEET 2019), Xi'an, China, 19–20 January 2019; pp. 156–161.
- Sarkar, T. XBNNet: An Extremely Boosted Neural Network. *arXiv* **2021**, arXiv:2106.05239. Available online: <https://arxiv.53yu.com/abs/2106.05239> (accessed on 1 December 2021).
- Gu, J.; Wang, Z.; Kuen, J. Recent Advances in Convolutional Neural Networks. *Comput. Sci.* **2015**, *77*, 354–377. [\[CrossRef\]](#)
- Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, CM, USA, 2016; pp. 326–366.
- Lecun, Y.; Kavukcuoglu, K.; Cle'ment, F. Convolutional networks and applications in vision. In Proceedings of the 2010 IEEE International Symposium on Circuits and Systems, Paris, France, 30 May–2 June 2010; pp. 253–256.

21. Waibel, A.; Hanazawa, T.; Hinton, G. Phoneme recognition using time-delay neural networks. *IEEE Trans. Acoust.* **2002**, *37*, 328–339. [[CrossRef](#)]
22. Mohandes, M.A.; Halawani, T.O.; Rehman, S.; Hussain, A.A. Support vector machines for wind speed prediction. *Renew Energy* **2004**, *29*, 939–947. [[CrossRef](#)]
23. Pedregosa, F.; Varoquaux, G.; Gramfort, A. Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* **2011**, *12*, 2825–2830.