Mathematical Biosciences and Engineering

MBE, 21 (2): 2991–3015.
DOI: 10.3934/mbe.2024133
Received: 10 October 2023
Revised: 03 December 2023
Accepted: 23 January 2024
Published: 30 January 2024

http://www.aimspress.com/journal/MBE

Research article

Lung adenocarcinoma identification based on hybrid feature selections and attentional convolutional neural networks

Kunpeng Li, Zepeng Wang, Yu Zhou and Sihai Li*

School of Information Engineering, Gansu University of Chinese Medicine, Lanzhou 730000, China

* **Correspondence:** Email: lisihai@gszy.edu.com.

Abstract: Lung adenocarcinoma, a chronic non-small cell lung cancer, needs to be detected early. Tumor gene expression data analysis is effective for early detection, yet its challenges lie in a small sample size, high dimensionality, and multi-noise characteristics. In this study, we propose a lung adenocarcinoma convolutional neural network (LATCNN), a deep learning model tailored for accurate lung adenocarcinoma prediction and identification of key genes. During the feature selection stage, we introduce a hybrid algorithm. Initially, the fast correlation-based filter (FCBF) algorithm swiftly filters out irrelevant features, followed by applying the k-means-synthetic minority over-sampling technique (k-means-SMOTE) method to address category imbalance. Subsequently, we enhance the particle swarm optimization (PSO) algorithm by incorporating fast-decay dynamic inertia weights and utilizing the classification and regression tree (CART) as the fitness function for the second stage of feature selection, aiming to further eliminate redundant features. In the classifier construction stage, we present an attention convolutional neural network (atCNN) that incorporates an attention mechanism. This improved model conducts feature selection post lung adenocarcinoma gene expression data analysis for classification and prediction. The results show that LATCNN effectively reduces the feature dimensions and accurately identifies 12 key genes with accuracy, recall, F1 score, and MCC of 99.70%, 99.33%, 99.98%, and 98.67%, respectively. These performance metrics surpass those of other comparative models, highlighting the significance of this research for advancing lung adenocarcinoma treatment.

Keywords: lung adenocarcinoma; tumor gene expression; feature selection; k-means-SMOTE; LATCNN; attention mechanism

1. Introduction

Global cancer statistics for 2020 projected approximately 19.29 million new cases and 9.96 million deaths [1]. Among cancers, lung cancer claimed the highest mortality rate worldwide, contributing to 18% of total cancer-related deaths. Over recent decades, lung adenocarcinoma, a prevalent form of non-small cell lung cancer, has exhibited an upward trajectory in both incidence and mortality across many nations [2]. Studies have shown that lung adenocarcinoma detected and diagnosed at an early stage can be treated by resection, and patients generally have a survival rate of about 90% within 5 years or even long-term survival [3]. However, sometimes hematogenous metastasis of lung adenocarcinoma occurs at an early stage, which requires combined chemotherapy and targeted drug therapy, and the survival period is shortened [4]. Therefore, early prevention of lung adenocarcinoma and the development of screening tools for this disease are crucial.

Clinical medicine and biomedical research have made tremendous progress with the development of gene chips and high-throughput sequencing technologies [5]. These technologies have revolutionized our understanding of the genome, proteome, and transcriptome, providing unprecedented opportunities for disease diagnosis, prevention, and treatment, and playing a key role in cancer research [6]. Advances in tumor genomics have helped scientists understand mutations in the genes that drive cancer, leading to the development of targeted therapies for specific mutations [7]. Common screening methods for lung adenocarcinoma include chest X-rays, blood biomarker tests, bronchoscopy, computed tomography (CT) scans, and genetic analysis. X-rays and blood biomarker tests are less effective in early screening, while bronchoscopy requires consideration of the patient's physical condition and is inconvenient to perform. On the other hand, CT scans and genetic analysis are widely employed in early detection, and with technological advances, they are also beginning to be integrated with artificial intelligence. Chen et al. developed a self-distillation training multitasking dense attention network based on CT scans [8]. They also proposed a deep learning-based protocol for CT image analysis that accurately identifies lung adenocarcinoma categories and high-risk tumor regions [9,10]. In a separate study, Gao [11] et al. utilized the particle swarm algorithm and the artificial bee colony algorithm to enhance support vector machine (SVM), achieving a prediction accuracy of 79.49% on lung adenocarcinoma gene expression data.

However, machine learning-based approaches to analyzing gene expression data for tumor prediction and key gene screening encounter several challenges. First, the categories of gene expression data typically display extreme imbalance, resulting in poor prediction performance for smaller categories, as classifiers tend to be more biased toward the dominant categories during training. Second, gene expression data often exhibit a large number of feature dimensions, which can easily lead to a dimensionality catastrophe, complicating the training and analysis of models, and increasing the difficulty of screening for key genes.

Many feature selection algorithms and classification prediction models have been applied to the study of gene expression profiling data [12]. Xie [13] et al. introduced a normalized mutual information method for screening key genes in unbalanced gene data, aiming to address the issue of traditional mutual information favoring multi-valued features. They approximated the gene subset size to 20–50 across multiple tumor gene datasets. Ye [14] et al. proposed a hybrid feature selection algorithm that combines FCBF and support vector machine recursive feature elimination (SVM-RFE). Through experiments on five public datasets, they reduced the gene subset size to approximately 10 and validated it using a k-nearest neighbor classifier, achieving an accuracy ranging from 83.87% to 100%. Ludwig [15] et al. suggested a fuzzy decision tree algorithm for analyzing gene expression in colon cancer data classification, achieving an accuracy of 80.28%. Zeebaree [16] et al. employed a simple

CNN model to classify 10 cancer datasets for prediction and compared it with SVM and random forest. The results were excellent on most datasets, with an average classification accuracy of 94.74%. Nguyen [17] et al. proposed an improved hierarchical analysis method and incorporated the probabilistic neural network into it, achieving an accuracy of 88.89% in colon cancer data diagnosis. Xiao [18] et al. conducted experiments on lung adenocarcinoma, gastric adenocarcinoma, and breast cancer by integrating multiple machine learning models using deep learning. They achieved a significant improvement in accuracy compared to a single machine learning classifier, reaching 99.20%, 98.78%, and 98.41%, respectively.

In summary, the primary challenges in the analysis and research of tumor gene data include data category imbalance and the difficulty of gene screening arising from high feature dimensions. In addition, due to the specificity of genetic data, there may be complex dependencies between certain genes, which should be considered during model construction. To address these challenges, we propose the LATCNN model, which is based on a hybrid feature selection approach and an attentional convolutional neural network, for analyzing gene expression data to predict lung adenocarcinoma. The initial step involves utilizing the FCBF algorithm for rapid filtering in feature selection to eliminate a substantial number of redundant genes. Subsequently, the k-means-SMOTE [19] method is employed to address the impact of data category imbalance. Following this, the dynamic inertia weight particle swarm optimization algorithm is applied to further refine the features, aiming to filter out key genes. Finally, atCNN is constructed for the prediction of lung adenocarcinoma. Accuracy, recall, F1 score, and MCC were used to measure the model performance. Extensive experiments demonstrate that LATCNN exhibits exceptional performance. The main highlights are as follows:

1) We propose a hybrid feature selection algorithm that combines FCBF and an improved PSO algorithm.

2) We propose a new adaptive inertia weight PSO (AWPSO) algorithm that decays rapidly to dynamically adjust the inertia weights based on the number of iterations.

3) We use the k-means-SMOTE method to deal with the data category imbalance problem.

4) We propose that attentional convolutional neural networks can better capture the dependencies between features through the attention mechanism.

2. Materials and methods

2.1. Architecture and workflow of the LATCNN model

In this paper, we propose the LATCNN model for predicting lung adenocarcinoma and screening its key genes based on gene expression data. LATCNN consists of three main components: Data preprocessing, feature selection, and the construction of a classification model. To obtain gene expression data for lung adenocarcinoma, we performed data correction and variance analysis on gene microarray data. The FCBF algorithm was then applied to rapidly filter out irrelevant feature genes. Subsequently, the AWPSO algorithm containing dynamic inertia weights was utilized to further approximate the feature subset, reduce the data size and identify key genes. Additionally, the k-means-SMOTE method was employed to address the issue of data category imbalance. The preprocessed feature data were fed into the constructed atCNN for the prediction of lung adenocarcinoma. Finally, a series of experiments were conducted to assess the model's performance in both key gene screening and lung adenocarcinoma prediction. Figure 1 illustrates the overall structure and flow of LATCNN.

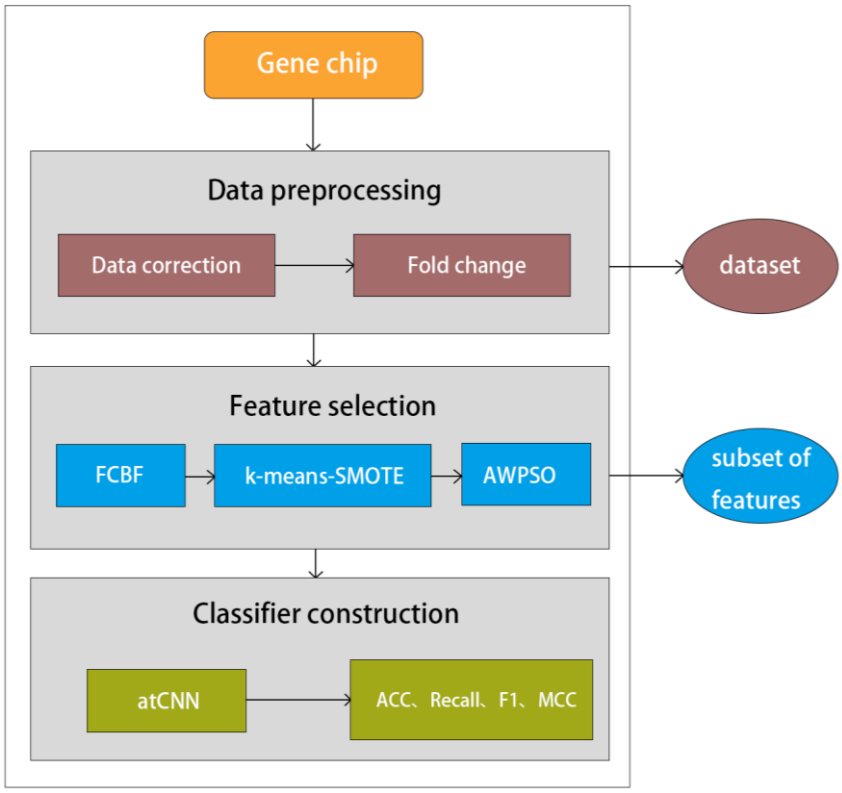

Figure 1. Architecture of the LATCNN model.

2.2. Data collection and processing

The original lung adenocarcinoma data utilized in this study were obtained from the GEO database, with the gene chip number GSE31210 [20,21]. Gene chips are composed of thousands of biomolecular probes, and each probe point corresponds to a gene. However, there was a situation where a gene was represented by more than one probe, so the downloaded cancer tissues and paracancerous tissues were subjected to probe transformation. During this process, gene probe IDs are replaced by gene names, the data with the same gene names were merged by averaging, and the data of the genes that had no differentiation were deleted. The number of samples with lung adenocarcinoma in the integrated dataset was 226, and the number of normal samples was 20, and the number of genes was 20,248. A column of labels was added to the dataset to indicate whether the disease was present or not, with 1 label for lung adenocarcinoma samples and 0 for normal samples.

Gene expression data from gene chips or high-throughput sequencing typically exhibit right-skewed or left-skewed distributions, i.e., non-normal distributions, due to obeying different distributions. Logarithmic processing of the data can transform the data into a form that is closer to a normal distribution, which helps to satisfy the assumption of normality in statistical analysis and thus apply a wider range of statistical methods. Due to the variation introduced by experimental conditions or microarray differences, the abundance of gene expression is much different, and for the convenience of subsequent modeling, gene expression data need to be normalized. The specific steps of normalization are: First, calculate the quantile of each gene in each sample, and then adjust the data of

each sample to a common distribution of quantiles (usually the median), which helps to make the distribution of data consistent among different samples. The specific formula is given below:

$$x' = (x / q) * Q \tag{1}$$

where x' is the normalized gene data, x is the original expression data, q is the sample quartile, and Q is the global quartile, and a comparison of the box-and-line plots before and after data correction is shown in Figures 2 and 3.

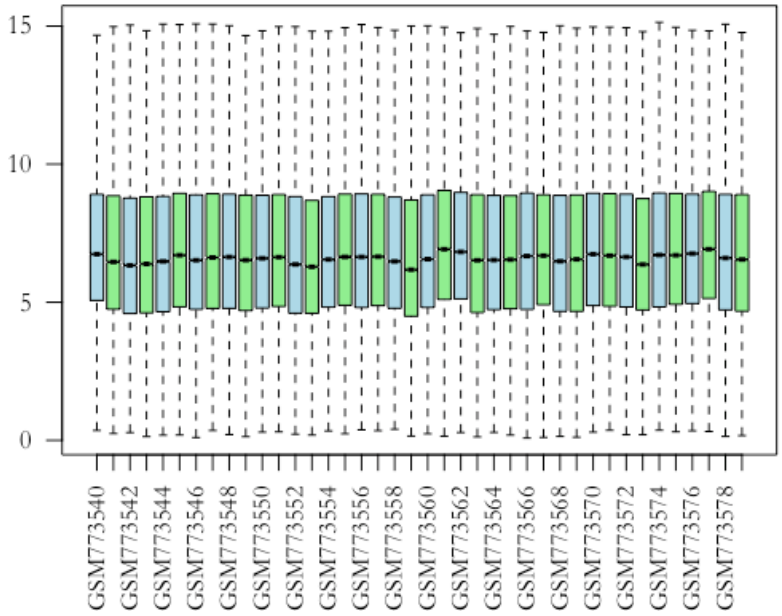

Figure 2. Data before correction.

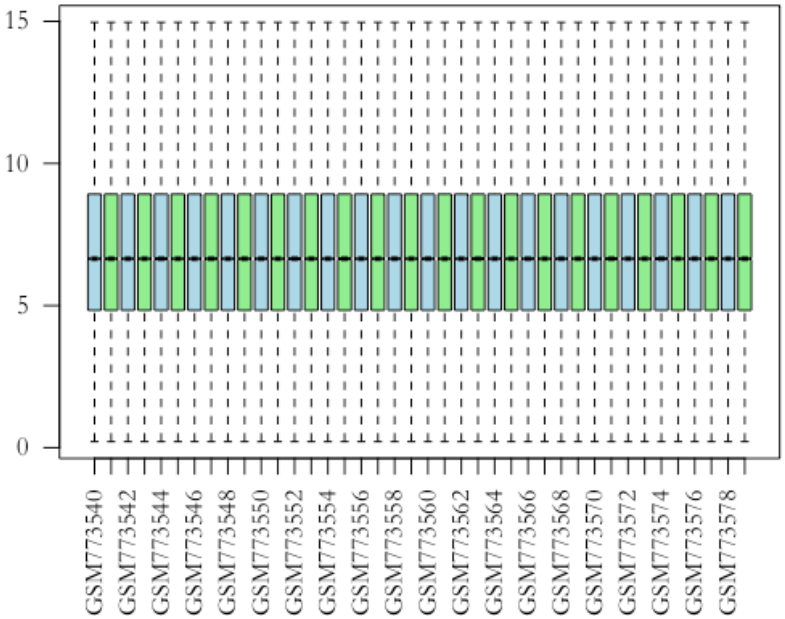

Figure 3. Data after correction.

The fold change method is the most basic method for gene expression analysis, which calculates

the fold difference between the diseased and normal gene data, and if the fold difference is greater than a specified threshold, the gene is identified as differentially expressed. If the multiplicity difference is greater than a specified threshold, the gene is identified as differentially expressed, and the expression formula is as follows:

$$\text{FC(i)}=\log2(\frac{\overline{x}_1(i)}{\overline{x}_2(i)}) \tag{2}$$

where $\overline{x}_1(i)$ and $\overline{x}_2(i)$ denote the mean value of gene i in the two sets of data, respectively, and whether it is a differential gene is judged by comparing $FC(i)$ with the specified threshold. The multiplicative change method can effectively reduce the data dimension, but it is not precise enough, and the limitation of this method is that the threshold value is not well determined, which is likely to result in a high proportion of false-positive results for genes. There are many characterized genes after screening, and other characterization screening methods should be used subsequently to further narrow down the screening scope.

The fold change method specifies a fold threshold of 1 and a threshold of 0.05 for the parameter p. The genes were selected as up-regulated genes if the expression difference was greater than 1-fold, and down-regulated genes if the expression difference was less than 0.5-fold. A total of 3099 differential genes were screened by the fold change method of differential analysis.

Comparison of gene expression data before and after all pretreatment steps is shown in Tables 1 and 2.

Table 1. Original data set.

Sample	DDR1	RFC2	…	TMEM231	LOC100505915	Label
GSE773540	2955.73	316.62	…	409.69	113.80	1
GSE773541	2278.31	496.10	…	453.71	138.83	1
GSE773542	2789.68	460.95	…	392.21	55.69	1
…	…	…	…	…	…	…
GSE773783	900.72	178.59	…	812.82	256.50	0
GSE773784	2301.62	327.82	…	5591.64	382.99	0
GSE773785	1476.74	277.21	…	2831.72	296.60	0

Table 2. Preprocessed data set.

Sample	DDR1	RFC2	…	TMEM231	LOC100505915	Label
GSE773540	11.56	8.30	⋯	8.69	6.76	1
GSE773541	11.12	9.02	⋯	8.90	7.29	1
GSE773542	11.39	8.99	⋯	8.78	6.11	1
⋯	⋯	⋯	⋯	⋯	⋯	⋯
GSE773783	9.87	7.47	⋯	9.72	8.03	0
GSE773784	11.21	8.16	⋯	12.61	8.40	0
GSE773785	10.51	7.90	⋯	11.56	8.01	0

In addition, in order to evaluate the performance of our proposed LATCNN model more comprehensively, we also chose three public tumor gene expression datasets for our experiments, namely, breast cancer, lung cancer, and DLBCL, which were obtained from the Kent Ridge Bio-

medical dataset repository as mentioned in the literature [22]. It is worth noting that these datasets cover different types of cancers and are not limited to lung adenocarcinoma. By examining data from different cancer types, we aim to validate the generality and applicability of the LATCNN model, as well as its performance in handling diverse cancer expression data. Such a design allows us to gain a more comprehensive understanding of the potential broad application value of our proposed approach. Details of all datasets are shown in Table 3.

Table 3. Details of datasets.

Datasets	Samples	Attributes	Classes
Breast cancer	97	24481	2
Lung cancer	39	2880	2
DLBCL	58	7129	2
GSE31210	246	3099	2

2.3. Feature selection methods

2.3.1. FCBF

FCBF [23] is a fast-filtering feature selection algorithm with symmetric uncertainty (SU) instead of information gain (IG) as a relevance measure.

The information entropy [24], is a measure of uncertainty of a random variable in information theory. Assuming that there is a random variable $X = \{x_1, x_2, ..., x_n\}$, with probability $p(x)$, the information entropy $H(X)$ of X is expressed as:

$$\mathrm{H}(X) = -\sum_{x \in X} p(x) \log(x) \tag{3}$$

The greater the information entropy, the greater the amount of information it contains. Assuming that there is a random variable Y, the information entropy of X, i.e., the conditional entropy $H(X \mid Y)$, after observing the entropy value of Y, is expressed as:

$$H(X \mid Y) = -\sum_{y \in Y} p(x) \sum_{x \in X} p(x \mid y) \log(x \mid y) \tag{4}$$

where $p(y)$ is the probability of y, and $p(x \mid y)$ denotes the probability of event x occurring under the given condition y.

The information gain $IG(X \mid Y)$, on the other hand, measures the extent to which the information entropy of the random variable X is reduced, conditional on the information entropy of the random variable Y being computed. In short, the information gain quantifies the amount of information provided by Y for the prediction of X. The larger its value, the higher the correlation between X and Y. The expression is:

$$IG(X \mid Y) = H(X) - H(\mathrm{X} \mid Y) \tag{5}$$

The SU value is a measure that describes the degree of correlation between two nonlinearly correlated variables, which is calculated using information entropy, a normalized form of mutual information, capable of solving the problem of the tendency of mutual information to refer to more than one characteristic, and a normalized form of information gain, with symmetric uncertainty

$SU(X,Y)$ expressed as:

$$SU(X,Y) = 2\frac{IG(X \mid Y)}{H(X) + H(Y)} \tag{6}$$

From the above equation, when the value of $SU(X,Y)$ is 0, it indicates that there is no correlation between variables X and Y. When the value of $SU(X,Y)$ is 1, it indicates that variables X and Y are perfectly correlated.

2.3.2. ReliefF algorithm

Feature weight is a metric for evaluating the importance of features in the ReliefF algorithm and is used to update the weights of features in each iteration [25]. Feature weight $w[A]$ is an important metric used to measure the relationship between the distance value between the original feature set and feature A. The expression for $w[A]$ is:

$$w[A] = w[A] - \frac{\sum_{j=1}^{k} diff(A,R,H_j)}{mk} + D(C) \tag{7}$$

$$D(C) = \sum_{C \neq class(R)} \left[\frac{P(C)}{1 - P(class(R))} \sum_{j=1}^{k} diff(A,R,M_j(C))\right] / (mk) \tag{8}$$

R represents a random sample drawn from the initial sample set. $class(R)$ represents the sample class containing sample R. $P(C)$ then denotes the ratio of the number of selected sample features to the total number of sample features among all sample classes. H is the k-nearest neighbor sample among samples of the same class as R, and M is the k-nearest neighbor sample among samples that are not of the same class as R. $diff(A,R_1,R_2)$ denotes the difference between samples R_1 and R_2 on feature A, and $M_j(C)$ denotes the jth k-nearest neighbor sample of class C. The expression for $diff(A,R_1,R_2)$ is given by:

$$diff(A,R_1,R_2) = \begin{cases} \dfrac{|R_1[A] - R_2[A]|}{\max(A) - \min(A)} & continuous \\ 0 & discrete, R_1[A] \neq R_2[A] \\ 1 & discrete, R_1[A] = R_2[A] \end{cases} \tag{9}$$

The features are ranked according to the feature weights w and the top-ranked features are selected as the final feature subset.

2.3.3. SVM-RFE

The objective of SVM is to construct an optimal hyperplane to maximize the classification interval to achieve accurate classification of samples. When the samples are linearly differentiable and satisfy the constraints, the objective function of SVM can be expressed as:

$$\min(\frac{1}{2}\| w\|^2 + c\sum_{i=1}^{n}\xi_i) \ \ s.t.\ y_i(w^T x_i + b) \geq 1-\xi_i, \xi_i \geq 0, i = 1,2,...,n \tag{10}$$

where w represents the weight vector of the hyperplane, b represents the classification threshold, and y represents the category labels.

In solving the nonlinear high-dimensional pattern recognition problem, SVM uses the kernel function $K(x_i \cdot x_j)$ to map the data into a high-dimensional space to find the optimal classification hyperplane to achieve linear differentiability. Ultimately, SVM computes the discriminant function by applying quadratic programming and Lagrange's dyadic theorem:

$$f(x) = \text{sgn}\{w \cdot x + b\} = \text{sgn}\{\sum_{i=1}^{n}\alpha_i y_i K(x_i, x) + b\} \tag{11}$$

SVM-RFE uses a linear kernel function and the ordering coefficients can be defined as:

$$Rank(i) = w_i^2, w_i = \sum_{i=1}^{n}\alpha_i y_i x_i^2 \tag{12}$$

According to the above equation, the weight w represents a linear combination of non-zero α support vectors, and an increase in weight indicates that the feature contains more categorical information.

2.3.4. PSO

The PSO algorithm is a heuristic global optimization method that achieves an approximate optimal solution by simulating the social behavior of a flock of birds in a multidimensional space [26]. Each object in the PSO algorithm is referred to as a particle, and each particle has the attributes of position and velocity. Assuming that the algorithm runs in n-dimensional space, the pth iteration position and velocity of particle i can be denoted as $x_i^p = [x_{i1}^p, x_{i2}^p, \cdots, x_{in}^p]$, $v_i^p = [v_{i1}^p, v_{i2}^p, \cdots, v_{in}^p]$, respectively. In the iterative process, each particle judges the superiority of the current position according to the calculated value of the fitness function, and denotes the value of the best position of particle i in the previous p iterations in the dth dimension by $pbest_{id}^p$, and denotes the value of the best position of all particles in the previous p iterations in the dth dimension by $gbest_{id}^p$. The particle decides the next position and velocity by the current position, velocity and the above two best positions, and the update formula for particle i is as follows.

$$v_{id}^{p+1} = \omega v_{id}^p + c_1 r_1 (pbest_{id}^p - x_{id}^p) + c_2 r_2 (gbest_{id}^p - x_{id}^p) \tag{13}$$

$$x_{id}^{p+1} = \begin{cases} 1 & sigmoid(v_{id}^{p+1}) = \dfrac{1}{1+e^{-v_{id}^{p+1}}} > U(0,1) \\ 0 & otherwise \end{cases} \tag{14}$$

Here, c_1, c_2 are called learning coefficients, r_1 and r_2 take the random numbers from 0 to 1, ω are the inertia weights of the particles during the iteration [27], which are generally taken as constants, $U(0,1)$ denotes the random numbers from 0 to 1 obeying a uniform distribution. In addition, the velocity of each particle during iteration must be limited to $[v_{\min}, v_{\max}]$.

2.3.5. Fast-decay adaptive inertia weights PSO algorithm (AWPSO)

Although the PSO algorithm possesses strong search ability, it often misses the optimal solution by falling into the local optimum when facing multi-polar problems. In order to further enhance the overall search ability of the particles, this paper introduces adaptive inertia weights to update the particle velocity:

$$\omega_{iter} = \omega_{\min} + \frac{\omega_{\max} - \omega_{\min}}{1 + e^{-\beta(iter-\alpha)}}\left(1 - \frac{iter}{\max iter}\right) \tag{15}$$

Here, ω_{max} and ω_{min} are the maximum and minimum inertia weights, respectively; α and β are the empirical value of 0.01 and 50; $iter$ is the current iteration number; $\max iter$ is the preset maximum iteration number. From Eq (15), the value of $iter$ eventually decreases from max to min with iterations. inertia weights have the function of balancing the global and local search of the particle swarm algorithm. When $iter$ is larger, it can improve the global search ability of PSO; when it is smaller, PSO possesses stronger local search ability.

The fitness function is an index for evaluating the quality of individual particles, and the selection of fitness function has a great influence on the convergence speed of PSO algorithm and whether it can find the optimal solution. The fitness function selected in this paper is adapted from the literature [28], and the specific formula is as follows:

$$f(x) = \alpha(1-P) + (1-\alpha)\left(1 - \frac{N_f}{N_t}\right) \tag{16}$$

where α is a hyperparameter that decides the tradeoff between the classifier performance P, and the size of the feature subset N_f concerning the total number of features N_t. The classifier performance can be the accuracy, F-score, precision, and so on. In this paper, the accuracy of CART and the current number of features are used as the fitness function, because it has the ability of feature selection, which will lead to better verification of the feature subset and better evaluation of the quality of individual particles.

2.4. K-means-SMOTE methods

In gene expression profiling data, category imbalance is an important issue. This means that the sample instances are unevenly distributed across different classes and hence the results of unbalanced data classification are biased towards the majority class. SMOTE is an improved algorithm for the random oversampling method, as the random oversampling method is a direct re-adoption of a few classes, which will result in a lot of duplicate samples in the training set, and is prone to cause overfitting problems in the resulting model [29]. The basic idea of the SMOTE algorithm is that for each minority class sample x_i, randomly select a sample $\hat{x}_i$ from its nearest neighbors ($\hat{x}_i$ is a sample in the minority class and defaults to 1), and then randomly select a point on the line between x_i and $\hat{x}_i$ as the newly synthesized minority class sample. The distance formula is:

$$dist(X, Y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \tag{17}$$

For each randomly selected nearest neighbor b, respectively, a new sample is constructed according to the following formula:

$$x_{new} = x_i + rand(0,1) \times (\hat{x}_i - x_i) \tag{18}$$

The process of the SMOTE algorithm to synthesize few class samples is shown in Figure 4.

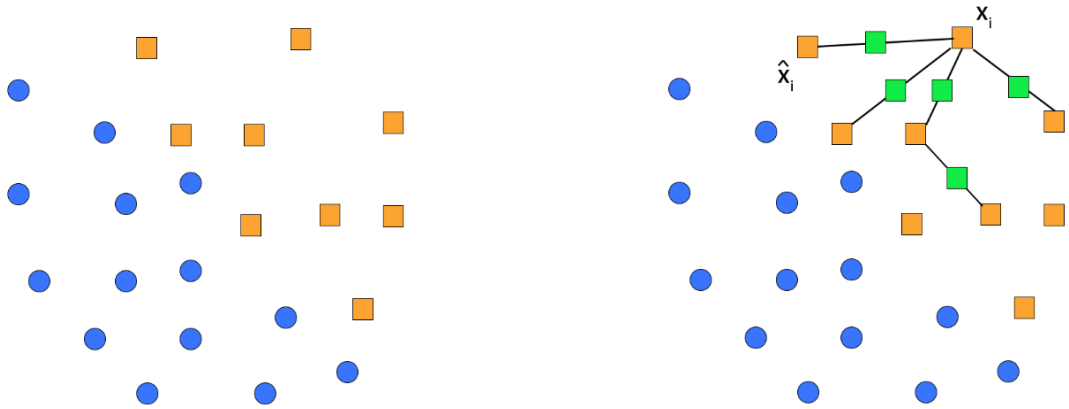

Figure 4. SMOTE algorithm process.

K-means-SMOTE is an enhanced version of the SMOTE algorithm proposed by Douzas et al. in 2018 [19], introducing a clustering operation on the basis of the SMOTE algorithm. Initially, the spatial domain is partitioned into k clusters using the K-means clustering method. Subsequently, based on the imbalance ratio, clusters requiring oversampling are selected, and additional samples are allocated to the sparser minority class clusters. For each selected cluster, the SMOTE algorithm is then applied to perform oversampling. The final output consists of a new dataset containing both the original samples and the synthesized samples.

2.5. Improved convolutional neural networks based on attention mechanism (atCNN)

To predict lung adenocarcinoma more accurately, we propose a neural network classification model that becomes atCNN, which consists of an input layer, two convolutional layers, two pooling layers, two dropout layers, an attention module layer, two dense fully connected layers, a flat layer, and an output layer. AtCNN's network structure and the overall structure of the LATCNN model are shown in Figure 5.

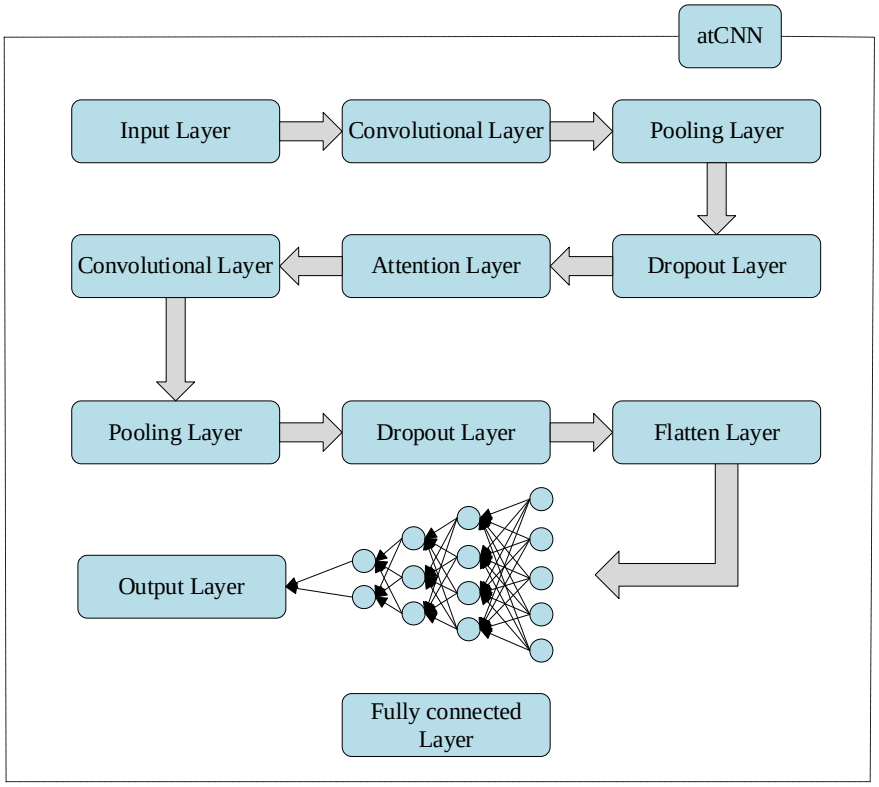

Figure 5. Network structure of atCNN.

The feature genes extracted by different feature selection algorithms are considered input features. In the convolutional layer, the convolutional kernel width is defined as 3. The convolutional layer's activation function employs the LeakyRelu function, a derivative of the Relu function that solves the Relu dead zone problem by introducing a small non-zero slope. When presented with an input sample, the convolutional layer functions in the subsequent manner:

$$convolution(x)_{ik} = \mathrm{Re}\,lu(\sum_{p=0}^{P-1}\sum_{q=0}^{Q-1} w_{\mathrm{pq}}^{\mathrm{k}} x_{i+p,q}) \tag{19}$$

$$\mathrm{Re}\,lu(x) = \max(0, x) \tag{20}$$

$$Leaky\,\mathrm{Re}\,lu(x) = \begin{cases} x & if\ x \geq 0 \\ ax & if\ x < 0 \end{cases} \tag{21}$$

Where w denotes the convolution kernel size and $w_{pq}^{k} x_{i+p,q}$ denotes the weight matrix of the kth convolution kernel with matrix size $P \cdot Q$.

To achieve enhanced precision in results, the utilization of pooling layers is deemed essential for the reduction of output dimensionality derived from the convolutional layer. Typically, the pooling layer can be categorized into two types: average pooling and maximum pooling. In the context of this research, the choice has been made to employ the maximum pooling layer for processing. The maximum pooling operation involves the selection of the highest value within the filtering range. Two distinct pooling window sizes, namely 2 and 3, are applied respectively. The process of channeling the convolution-processed data into the maximum pooling layer is detailed as follows:

$$pooling(X)_{ik} = \max(X_{iM,k}, X_{iM+1,k}, X_{iM+2,k}, ..., X_{iM+M-1,k}) \tag{22}$$

To prevent overfitting, a stochastic deactivation layer is attached after the pooling layer with the parameter set to 0.5.

An attention module is added before the second convolutional layer to introduce a self-attention mechanism to improve the performance of the model. The self-attention mechanism serves to introduce a kind of context-dependent weighted pooling in the intermediate layer of the model in order to capture long-distance dependencies in the input data. The relationships between different features can be complex, and the self-attention mechanism can help the model better understand these relationships. The core formula for the attention mechanism is:

$$Attention(Q_i, K_i, V_i) = SoftMax(\frac{Q_i K_i^T}{\sqrt{d_k}})V_i \tag{23}$$

where,

$$\begin{aligned} Q_i &= W_Q \bullet X_i \\ K_i &= W_K \bullet X_i \\ V_i &= W_V \bullet X_i \end{aligned} \tag{24}$$

Here, X_i is the ith element of the input sequence, W_Q, W_K, W_V are the corresponding weight matrices, and d_k is the dimension of the attention header. Q (Query) is a query vector that is used to query all other Key vectors to determine the weight given to a given Key in the attention mechanism, K (Key) is a key vector used to compare the query vectors, which denotes the importance of each element in the input sequence, and the V (Value) vectors are the values associated with each Key, which are used to generate the final output. In the self-attention mechanism, for each query vector, a weight is computed based on the Key vectors corresponding to it, and then the Value vectors associated with these Keys are weighted and summed using these weights. The final output is this weighted sum.

The input matrix transforms an output vector through the utilization of a flattening layer after the application of data filtering. Within this model, two dense fully connected layers with 128 and 64 neurons, respectively, are employed. The processing of these dense layers is segregated into two phases: Forward propagation and backpropagation. The former is employed for the computation of the desired output results, while the latter is utilized to iteratively optimize the parameters of the forward propagation process. The parameters, which are updated post-backpropagation, are subsequently reused for the computation of the output values. An activation function known as Relu is applied to each output. The mathematical expression representing the dense fully connected layer is presented below:

$$a_i = \sum_{j=1}^{n} w_{ij} x_j + b_i \tag{25}$$

$$loss = a_i^* - a_i \tag{26}$$

$$w_{ij}' = w_{ij} - \frac{n}{m} \sum \frac{\partial loss}{\partial w_{ij}} \tag{27}$$

$$b_i' = b_i - \frac{n}{m}\sum \frac{\partial loss}{\partial b_i} \tag{28}$$

The output of the dense layer, denoted as a_i, is influenced by the weights, symbolized by w_{ij}. The input to the output of the pooling layer, which is referred to as x_j, contributes to this process. Additionally, a bias parameter, represented by b_i, is incorporated. The updated weights and bias parameter, denoted as w_{ij}' and b_i', respectively, play a crucial role in this context. It is imperative to highlight that the loss function quantifies the cumulative error.

In the classifier's ultimate output layer, the SoftMax activation function is incorporated, facilitating the assignment of a numerical range of 0 to 1 to each element within the vector. Furthermore, the cumulative sum of these elements invariably equals 1. This characteristic proves particularly advantageous in conveying the likelihood of the presence of lung adenocarcinoma. The mathematical expression for the SoftMax function is provided below:

$$SoftMax(y_i) = \frac{e^{y_i}}{\sum_{j=1}^{k} e^{y_k}} \tag{29}$$

2.6. Model evaluation indicators

The most effective approach for illustrating the model's robustness and utility is regarded as the utilization of k-fold cross-validation [30]. In this research, a cross-validation technique is employed for the evaluation of the method's predictive capability. Furthermore, several potent assessment metrics have been chosen to validate the model's feasibility. These chosen metrics encompass accuracy (ACC), sensitivity (Sn), specificity (Sp), precision (Pre), the F1 index (F1), and Matthew's correlation coefficient (MCC).

A confusion matrix usually evaluates the model's output for a classification query. Table 4 shows the confusion matrix.

Table 4. Confusion matrix.

	Predicted positive	Predicted negative
Actual positive	TP	FP
Actual negative	FN	TN

For the binary classification model, after the classifier classifies instances into positive and negative classes, there are four classification outcomes. True positive (TP): A positive class instance that is also classified as a positive class. True negative (TN): A negative class instance that is classified as a negative class. False positive (FP): An instance of a negative category, but is judged to be a positive example. False negative (FN): An instance of a positive category, but is judged to be a negative category. The relevant definitions and formulas for the six indicators are represented below:

Accuracy is the degree to which an instance is correctly described and is calculated as:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN} \tag{30}$$

Specificity is the ability to correctly identify negative class samples and is calculated as:

$$Sp = \frac{TN}{TN + FP} \tag{31}$$

Sensitivity is the proportion of correctly predicted positive class instances to the total number of actual positive class samples and is calculated as:

$$Sn = \frac{TP}{TP + FN} \tag{32}$$

Precision is the accuracy of predicting positively classified samples and is the proportion of samples predicted to be positively classified that are correctly predicted, calculated by the formula:

$$\Pr e = \frac{TP}{TP + FP} \tag{33}$$

Accuracy and sensitivity are conflicting indicators. In general, when accuracy is high, sensitivity tends to be low, while when accuracy is low, sensitivity tends to be high. The F1 index is a weighted average of accuracy and sensitivity, which enables the two indicators to be taken into account together, and is calculated using the following formula:

$$F1 = \frac{2TP}{2TP + FP + FN} \tag{34}$$

Matthews correlation coefficient is a comprehensive evaluation index that takes into account the sensitivity and specificity and takes the value between [0,1], the larger the value of MCC, the better the classification effect of the model, and the calculation formula is:

$$MCC = \frac{TP * TN - FP * FN}{\sqrt{(TP + FN)(TP + FP)(TN + FP)(TN + FN)}} \tag{35}$$

In addition, the model was evaluated using the AUC value of the area under the curve of the ROC curve, which describes the performance of the classification model as a function of threshold, with no point on the curve reflecting the perceptibility of the same signal stimulus. It is a curve plotted with the true positive rate as the vertical coordinate and the false positive rate as the horizontal coordinate. When comparing two or more models, it is possible to visualize the strengths and weaknesses of the models. The closer the curve is to the upper left corner, the better the model performance. The formula for calculating the true positive rate is the same as for the sensitivity, and the formula for calculating the false positive rate is:

$$\mathrm{FDR} = \frac{FP}{TN + FP} \tag{36}$$

The larger the area under the curve, the better the performance of the corresponding model, and the AUC value takes the range.

2.7. Algorithm description and pseudo-code

When using the hybrid feature selection algorithm proposed in this paper, the FCBF algorithm is first used to do the initial feature selection to quickly filter out a large number of irrelevant features

using symmetric uncertainty as a metric, next we propose a PSO optimization algorithm that dynamically adjusts the inertia weights to do the second stage of the feature selection using the performance of the CART and the number of features together to form the fitness function in order to further narrow down the the optimal gene subset. The following is the pseudo-code for this description.

Input: Tumor gene expression data (X, C), where $X = \{x_1, x_2, \ldots, x_m\}$, $C = \{c_1, c_2, \ldots, c_n\}^T$.

Output: The best subset of features *G*.

1) $G = \{\}$ # Initialize *G* as an empty set
2) for *i* in range(1, *n*+1):
3) Calculate C-correlation value $SU(c_i, d)$
4) $G = G + \{c_i\}$
5) end
6) Sort *G* in descending order based on *SU* values
7) *G* = Top 100 elements of *G*
8) For each particle *i*:
9) Calculate initial fitness value *fit[i]*
10) If *fit[i]* is better than individual best *fitP[i]*, update *fitP[i]* and individual best position *Xpb[i]*
11) If *fitP[i]* is better than global best *fitG*, update *fitG* and global best position *Xgb*
12) For each particle *i*:
13) Update particle velocity *V[i]*, including the update for adaptive inertia weight
14) Update particle position *X[i]*
15) Repeat steps 8–14 until reaching the maximum iteration count max_iter
16) Return the global best feature subset *Xgb*
17) *G* = *Xgb*

3. Result and discussion

In order to verify the effectiveness of LATCNN, the dataset made from gene chip GSE31210 and three publicly available datasets are chosen for experiments in this paper. The experiments in this paper were all conducted at Google Colab with an Intel Xeon Dual Core 2.2 GHz CPU and NVIDIA Tesla K80 GPU.

3.1. Comparison of feature selection methods

For the comparison of feature selection algorithms, we used SVM as a classifier and performed ten trials on the best subset of features screened by each dataset. Five-fold cross-validation was used for each experiment, and the average result of the ten trials was used as the final accuracy. In order to better evaluate the hybrid feature selection algorithm combining FCBF and AWPSO proposed in this paper, this paper also conducts comparative experiments with the following algorithms, including FCBF-PSO, ReliefF-AWPSO, ReliefF-SVMRFE, and FCBF-SVMRFE used in the literature [14]. Among them, the filtering algorithms FCBF and ReliefF's are both selected to rank the top 100 features. Table 5, Figures 6 and 7 show the experimental results for the four datasets under the five algorithms, where ats denotes the size of the subset of optimal genes and acc denotes the final accuracy.

Table 5. Feature selection results.

	ReliefF-AWPSO		ReliefF-SVMRFE		FCBF-PSO		FCBF-SVMRFE		FCBF-AWPSO	
	Atts	Acc(%)	Atts	Acc(%)	Atts	Acc(%)	Atts	Acc(%)	Atts	Acc(%)
GSE31210	21	97.01	27	94.18	19	98.7	22	97.01	12	100
Breast cancer	29	91.71	33	83.26	27	91.71	33	90.55	21	94.7
Lung cancer	13	81.22	17	79.4	13	81.67	13	80.32	6	89.03
DLBCL	25	87.67	27	84.51	24	86.4	25	88.56	19	91.07

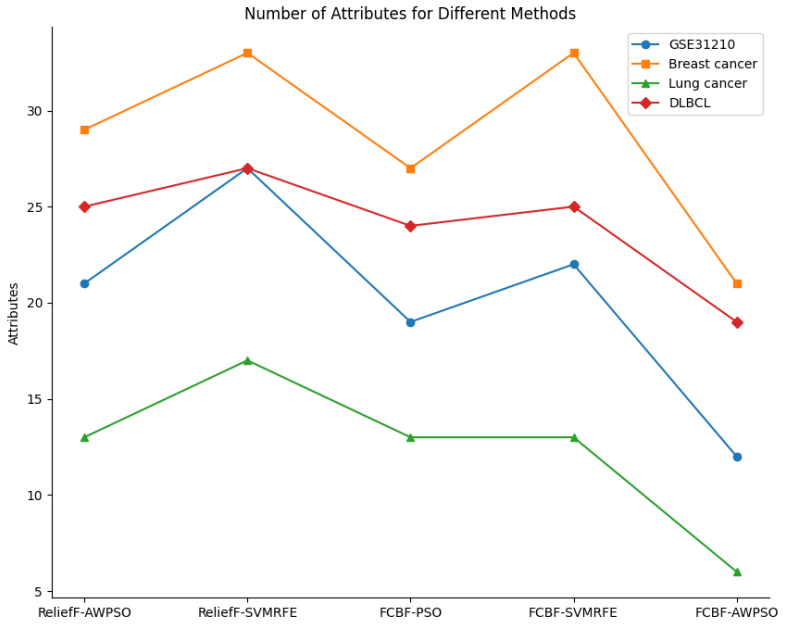

Figure 6. Number of attributes for different methods.

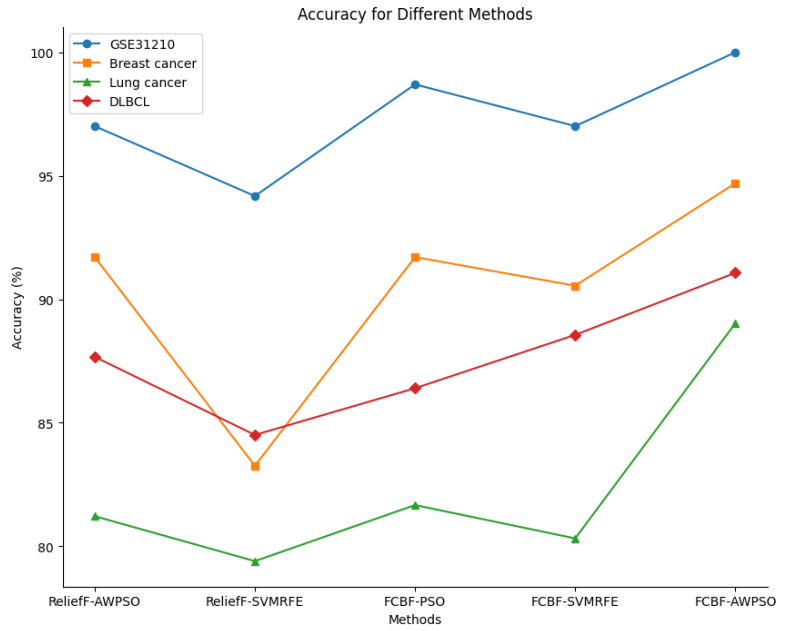

Figure 7. Accuracy for different methods.

In summary, it can be seen that the FCBF-AWPSO algorithm proposed in this paper has the smallest size of the best gene subset selected on each dataset, and none of the classification accuracies are lower than the other combinations. This indicates that the combination of FCBF and AWPSO is a better feature selection algorithm. It can also be seen that compared to ReliefF, FCBF is more suitable for gene priming in the first stage of the hybrid feature selection algorithm.

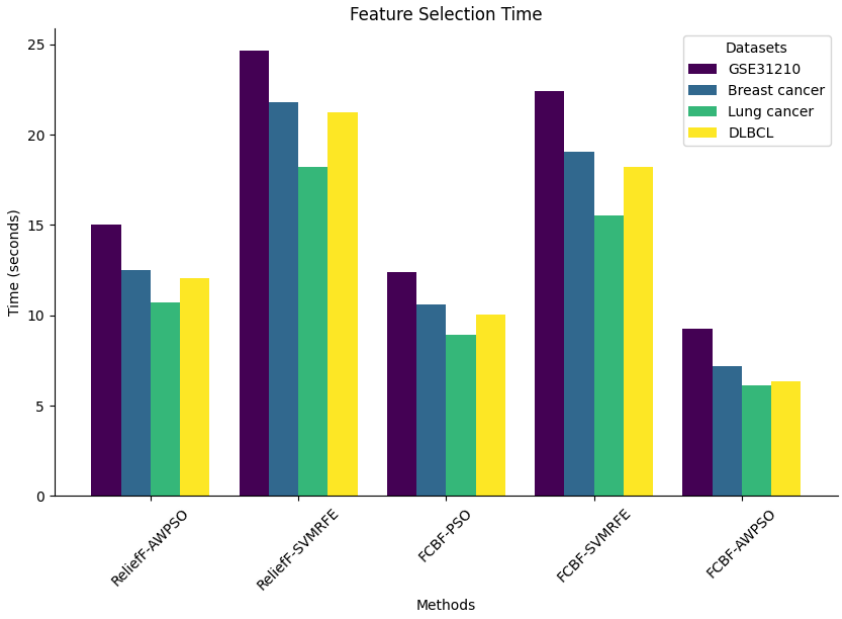

Figure 8. Feature selection time.

As shown in Figure 8, we also recorded the time taken to run these algorithms, and it can be seen that the algorithms in this paper took the least amount of time on each dataset.

Based on the above analysis, the hybrid feature selection algorithm using FCBF combined with AWPSO can quickly and accurately capture the feature information of tumor gene data. Therefore, we adopt this method to construct the LATCNN model.

3.1.1. Analysis of key genes

The 12 key genes of GSE31210 after screening were COL10A1, AFAP1-AS1, MNX1, ENTREP2, GALNT7, LGI3, RCC1, B3GNT3, ST6GALNAC3, KDELR3, STIL, and TACC1. Among these genes, COL10A1, AFAP1-AS1, MNX1, B3GNT3, and TACC1 have been documented to be associated with lung adenocarcinoma [31−35]. GO functional enrichment analysis of these genes was performed, and the results are shown in Figure 9, in which RCC1, STIL, and TACC1 belong to the same 3 pathways: GO:0007052, GO:1902850, GO:0007051. GALNT7, B3GNT3, and ST6GALNAC3 belong to the same 5 pathways: GO:0006486, GO:0043413, GO:0070085, GO:0009101, GO:0009100, and GO:0016757. The function of each gene is described in Table 6.

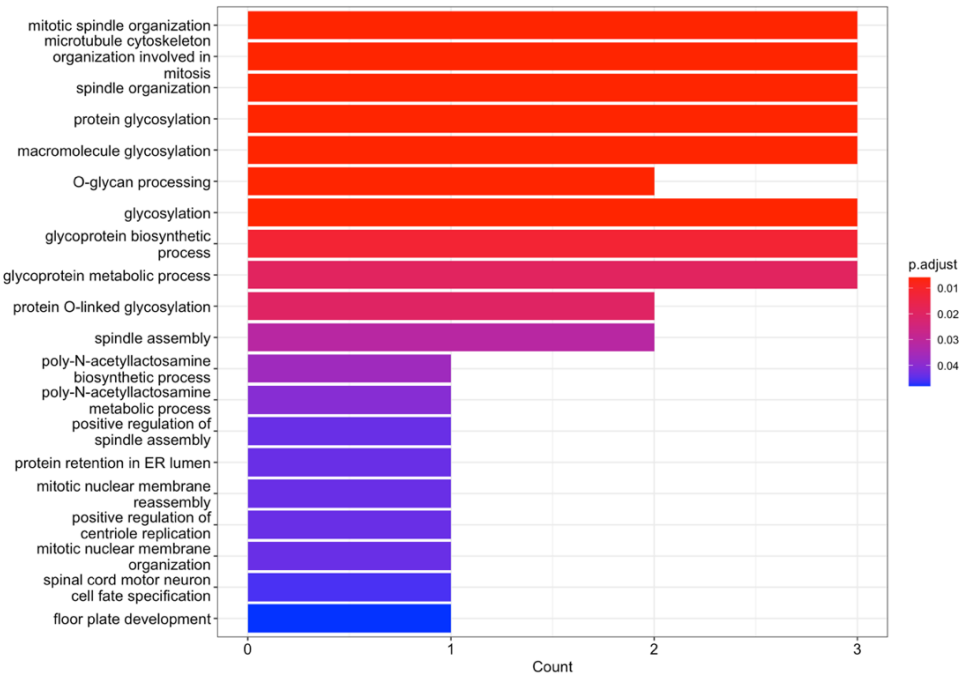

Figure 9. GO enrichment analysis histogram.

Table 6. Functional description of key genes.

Gene name	Description	Gene biotype
COL10A1	collagen type X alpha 1 chain	Protein coding
AFAP1-AS1	AFAP1 antisense RNA 1	lncRNA
MNX1	motor neuron and pancreas homeobox 1	Protein coding
ENTREP2	endosomal transmembrane epsin interactor 2	Protein coding
GALNT7	polypeptide N-acetylgalactosaminyltransferase 7	Protein coding
LGI3	leucine rich repeat LGI family member 3	Protein coding
RCC1	regulator of chromosome condensation 1	Protein coding
B3GNT3	UDP-GlcNAc:betaGal beta-1,3-N-acetylglucosaminyltransferase 3	Protein coding
ST6GALNAC3	ST6 N-acetylgalactosaminide alpha-2,6-sialyltransferase 3	Protein coding
KDELR3	KDEL endoplasmic reticulum protein retention receptor 3	Protein coding
STIL	STIL centriolar assembly protein	Protein coding
TACC1	transforming acidic coiled-coil containing protein 1	Protein coding

3.1.2. *K-means-SMOTE algorithm result*

The K-means-SMOTE algorithm is used after the initial screening of features for FCBF to balance the minority samples. Since the second stage of feature selection is influenced by the classifier, it is affected by the unbalanced data. The balancing of the dataset is achieved by adding new minority instances and making them equal to the majority instances. Table 7 compares the results before and

after using the SMOTE algorithm on the original dataset.

Table 7. Data balance before and after.

	Dataset (original)	Dataset (using SMOTE)
LUAD class instances	226	226
Normal class instances	20	226
Total no. of instances	246	452

3.2. Comparison of multiple methods

Table 8. Classifier Performance Comparison.

Data	Classifier	Acc(%)	Recall(%)	F1(%)	MCC(%)
GSE31210	SVM	81.01	78.39	71.71	72.76
	RF	87.65	92.80	75.43	77.22
	XGBOOST	86.29	70.48	76.80	79.93
	CNN	96.37	96.66	96.90	95.17
	DNN	97.57	99.02	98.84	97.13
	LSTM	93.08	92.20	89.33	88.66
	atCNN	**99.70**	**99.33**	**99.98**	**98.67**
Breast cancer	SVM	87.63	84.39	78.92	77.40
	RF	94.75	92.37	94.06	92.10
	XGBOOST	94.26	95.33	91.17	92.05
	CNN	93.92	96.15	91.83	91.22
	DNN	95.60	**100**	94.31	94.72
	LSTM	93.08	93.10	92.37	90.14
	atCNN	**96.99**	**100**	**97.47**	**97.50**
Lung cancer	SVM	79.49	82.30	74.06	74.66
	RF	88.18	89.35	84.62	83.51
	XGBOOST	87.22	91.73	84.67	82.15
	CNN	92.78	93.35	90.41	91.69
	DNN	93.54	**98.06**	91.32	90.75
	LSTM	89.25	89.44	86.93	86.20
	atCNN	**95.73**	96.67	**94.56**	**94.99**
DLBCL	SVM	81.03	82.57	81.26	81.86
	RF	89.43	87.54	83.20	82.33
	XGBOOST	89.03	86.44	87.62	85.48
	CNN	93.57	94.80	91.55	91.32
	DNN	93.02	**95.28**	90.71	91.60
	LSTM	91.46	79.92	90.29	88.74
	atCNN	**96.00**	95.10	**94.42**	**94.84**

In order to objectively evaluate LATCNN, we launched a series of experiments on the independent dataset GSE31210 and three public datasets. Each method was subjected to 5 experiments on each dataset, and 10-fold cross-validation was used each time, with the most average value as the final result. For classifier comparison, we chose three machine learning and three deep learning

baseline methods, namely SVM, Random Forest, XGBOOST, CNN, DNN, and LSTM, and accuracy, recall, F1 score, and MCC are used as evaluation metrics, respectively. Table 8 shows the results of these methods on each dataset.

As can be seen from Table 8, atCNN performs better on all datasets compared to other classifiers. On the lung adenocarcinoma independent dataset GSE31210, the accuracy, recall, F1 score, and MCC reached 99.70%, 99.33%, 99.98%, and 98.67%, respectively.

We also compare LATCNN with some of the existing models in the literature, including the proposed FCBF-PA-SVM in literature [11], the FCBF-SVMRFE feature selection using FCBF-SVMRFE features and prediction using four machine learning classifiers as mentioned in literature [14], and a CNN model as mentioned in literature [16].

Table 9. Comparison with existing models.

	Literature [11]	Literature [14]	Literature [16]	LATCNN
GSE31210	81.50	94.29	95.33	**99.70**
Breast cancer	88.49	85.71	92.10	**97.85**
Lung cancer	78.06	90.26	94.03	**99.71**
DLBCL	87.15	93.06	93.89	**98.53**

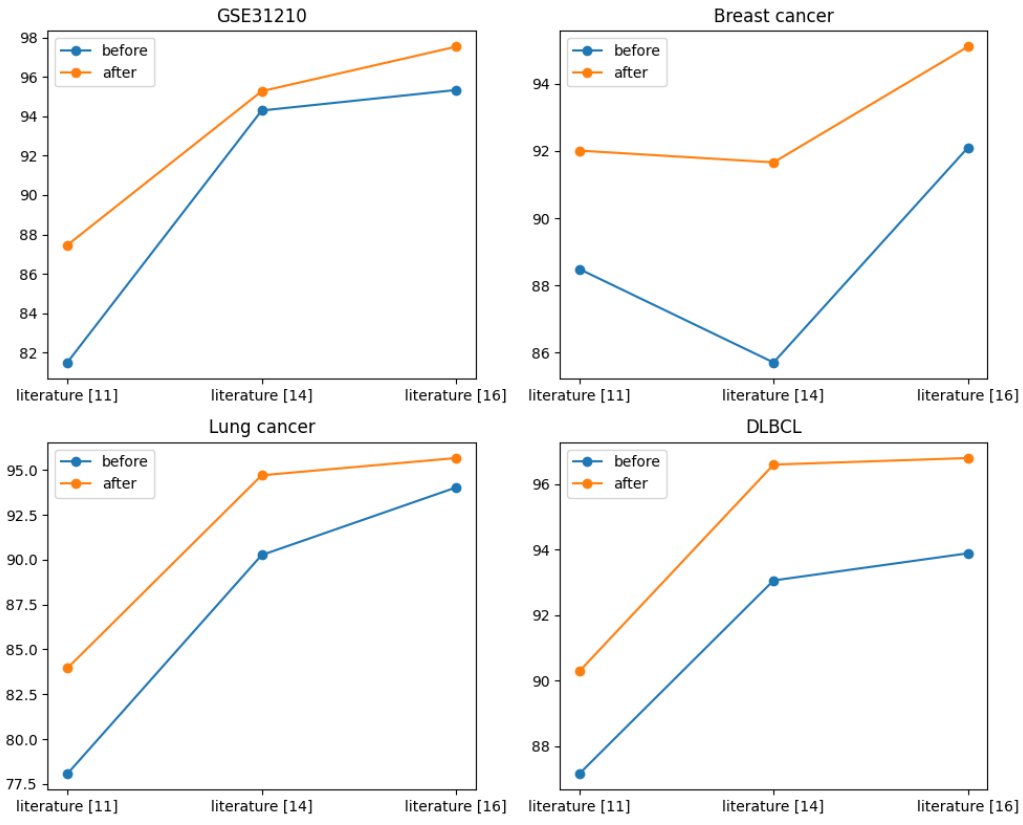

Figure 10. Accuracy before and after model modification.

Table 9 shows the accuracy comparison between LATCNN and the above three models on each dataset, and the model in this paper achieves the best performance on all four datasets, reaching 99.64, 97.85, 99.71, and 98.53, respectively. We also modified the models in the above three literatures by replacing their original methods with the feature selection method proposed in this paper, and Figure

10 shows the comparison between before and after replacement. It can be seen that after the replacement, the accuracy of all these models is improved, proving that the feature selection method proposed in this paper can indeed improve the model performance.

Compared with other models, LATCNN has two advantages: First, the hybrid feature selection algorithm can remove irrelevant and redundant features and accurately select the optimal subset of features, which on the one hand facilitates the subsequent classification prediction of the model, and on the other hand, it also filters out the key genes at the same time. Second, the convolutional neural network with the introduction of the attention mechanism performs better than traditional machine learning classifiers and also performs better than other deep learning models. However, our model has some shortcomings, compared with machine learning, deep learning models take longer to train and are more susceptible to hardware factors. Thus, finding better hyperparameters will be the key to optimize the model in the future. In summary, LATCNN can be applied to tumor genetic data for prediction and key gene screening.

4. Conclusions

Tumors are extremely harmful to humans, and early identification and screening of key genes are particularly critical. We take lung adenocarcinoma as an example and propose the LATCNN model, which aims to accurately predict lung adenocarcinoma and screen key genes. We face the problem of how to better downsize the gene data to narrow the scope and more accurately make predictions.

Initial steps include processing the gene chip data to prepare a usable tumor gene expression dataset. The next step is feature extraction using a hybrid feature selection algorithm, which includes fast filtering using FCBF, unbalanced categories using K-means-SMOTE, and further feature reduction using AWPSO to achieve optimal feature extraction. Finally, atCNN was constructed for prediction, achieving an accuracy of 99.70 and the best results on the other three public datasets. Comparison with existing methods shows that LATCNN performs better in lung adenocarcinoma prediction and key gene screening, and has the potential to analyze other oncogene data.

Our aim of this study is to explore better methods for oncogene screening as well as prediction. In the future, better integration with clinical data is suggested to further improve prediction accuracy. In addition, for gene chips prepared in different ways, a suitable batch correction method is searched for to obtain a better dataset through joint analysis of multi-chip data. By solving these problems, it will eventually help our research on tumors.

Use of AI tools declaration

The authors declare that they have not used Artificial Intelligence (AI) tools in the creation of this article.

Acknowledgments

This study is financially supported by the Gansu Provincial Science and Technology Program Project of China (grant no. 21JR1RA272) and the Gansu Provincial Department of Education: University Teachers Innovation Fund Project (grant no. 2023B-105).

Conflict of interest

The authors declare that there are no conflicts of interest.

References

1. World Health Organization, *Global health estimates 2020: Deaths by cause, age, sex, by country and by region*, 2000–2019, Switzerland, (2020).
2. V. Gedvilaitė, E. Danila, S. Cicėnas, G. Smailytė, Lung cancer survival in Lithuania: changes by histology, age, and sex from 2003–2007 to 2008–2012, *Cancer Control*, **26** (2019). https://doi.org/10.1177/1073274819836085
3. K. Chansky, F. C. Detterbeck, A. G. Nicholson, V. W. Rusch, E. Vallières, P. Groome, et al., The IASLC lung cancer staging project: External validation of the revision of the TNM stage groupings in the eighth edition of the TNM classification of lung cancer, *J. Thorac. Oncol.*, **12** (2017), 1109–1121. https://doi.org/10.1016/j.jtho.2017.04.011
4. T. Tamura, K. Kurishima, K. Nakazawa, K. Kagohashi, H. Ishikawa, H. Satoh, et al., Specific organ metastases and survival in metastatic non-small-cell lung cancer, *Mol. Clin. Oncol.*, **3** (2014), 217–221. https://doi.org/10.3892/mco.2014.410
5. G. Lightbody, V. Haberland, F. Browne, L. Taggart, H. Zheng, E. Parkes, et al., Review of applications of high-throughput sequencing in personalized medicine: Barriers and facilitators of future progress in research and clinical application, *Brief. Bioinf.*, **20** (2019), 1795–1811. https://doi.org/10.1093/bib/bby051
6. F. S. Collins, H. Varmus, A new initiative on precision medicine, *N. Engl. J. Med.*, **372** (2015), 793–795. https://doi.org/10.1056/NEJMp1500523
7. B. Vogelstein, N. Papadopoulos, V. E. Velculescu, S. B. Zhou, L. A. Diaz, K. W. Kinzler, Cancer genome landscapes, *Science*, **339** (2013), 1546–1558. https://doi.org/10.1126/science.1235122
8. L. Y. Chen, Z. J. Zhang, The self-distillation trained multitask dense-attention network for diagnosing lung cancers based on CT scans, *Med. Phys.*, (2023). https://doi.org/10.1002/mp.16736
9. L. Y. Chen, H. Y. Qi, D. Lu, J. X. Zhai, K. K. Cai, L. Wang, et al., A deep learning based CT image analytics protocol to identify lung adenocarcinoma category and high-risk tumor area, *STAR Protoc.*, **3** (2022), 101485. https://doi.org/10.1016/j.xpro.2022.101485
10. L. Y. Chen, H. Y. Qi, D. Lu, J. X. Zhai, K. K. Cai, L. Wang, et al., Machine vision-assisted identification of the lung adenocarcinoma category and high-risk tumor area based on CT images, *Patterns*, **3** (2022), 100464. https://doi.org/10.1016/j.patter.2022.100464
11. L. Y. Gao, M. Q. Ye, C. R. Wu, Cancer classification based on support vector machine optimized by particle swarm optimization and artificial bee colony, *Molecules*, **22** (2017), 2086. https://doi.org/10.3390/molecules22122086
12. M. Yousef, A. Kumar, B. Bakir-Gungor, Application of biological domain knowledge based feature selection on gene expression data, *Entropy*, **23** (2020), 2. https://doi.org/10.3390/e23010002
13. J. Y. Xie, M. Z. Wang, Y. Zhou, H. C. Gao, S. Q. Xu, Differential expressed gene selection algorithms for unbalanced gene datasets, *J. Comput.*, **42** (2019), 1232–1251. https://doi.org/10.11897/SP.J.1016.2019.01232

14. M. Q. Ye, L. Y. Gao, C. R. Wu, C. Y Wan, Informative gene selection method based on symmetric uncertainty and SVM recursive feature elimination, *Patt. Recog. Artif. Intell.*, **30** (2017), 429–438. https://doi.org/10.16451/j.cnki.issn1003-6059.201705005
15. S. A. Ludwig, S. Picek, D. Jakobovic, Classification of cancer data: Analyzing gene expression data using a fuzzy decision tree algorithm, *Oper. Res. Appl. Health Care Manage.*, **262** (2018), 327–347. https://doi.org/10.1007/978-3-319-65455-3_13
16. D. Q. Zeebaree, H. Haron, A. M. Abdulazeez. Gene selection and classification of microarray data using convolutional neural network, in *2018 International Conference on Advanced Science and Engineering (ICOASE)*, (2018), 145–150. https://doi.org/10.1109/ICOASE.2018.8548836
17. T. Nguyen, A. Khosravi, D. Creighton, S. Nahavandi, A novel aggregate gene selection method for microarray data classification, *Pat. Recog. Lett.*, **60** (2015), 16–23. https://doi.org/10.1016/j.patrec.2015.03.018
18. Y. W. Xiao, J. Wu, Z. L. Li, X. D. Zhao, A deep learning-based multi-model ensemble method for cancer prediction. *Comput. Methods Programs Biomed.*, **153** (2018), 1–9. https://doi.org/10.1016/j.cmpb.2017.09.005
19. G. Douzas, F. Bacao, F. Last, Improving imbalanced learning through a heuristic oversampling method based on k-means and SMOTE, *Inf. Sci.*, **465** (2018), 1–20. https://doi.org/10.1016/j.ins.2018.06.056
20. H. Okayama, T. Kohno, Y. Ishii, Y. Shimada, K. Shiraishi, R. Iwakawa, et al., Identification of genes upregulated in ALK-positive and EGFR/KRAS/ALK-negative lung adenocarcinomas, *Cancer Res.*, **72** (2012), 100–111. https://doi.org/10.1158/0008-5472.CAN-11-1403
21. M. Yamauchi, R. Yamaguchi, A. Nakata, T. Kohno, M. Nagasaki, T. Shimamura, et al., Epidermal growth factor receptor tyrosine kinase defines critical prognostic genes of stage I lung adenocarcinoma, *PLoS One*, **7** (2012), e43923. https://doi.org/10.1371/journal.pone.0043923
22. X. H. Cao, I. Stojkovic, Z. Obradovic, A robust data scaling algorithm to improve classification accuracies in biomedical data, *BMC Bioinf.*, **17** (2016). https://doi.org/10.1186/s12859-016-1236-x
23. L. Yu, H. Liu, Efficient feature selection via analysis of relevance and redundancy, *J. Mach. Learn. Res.*, **5** (2004), 1205–1224.
24. J. Liang, Z. Shi, D. Li, M. J. Wierman, Information entropy, rough entropy and knowledge granulation in incomplete information systems, *Int. J. Gen. Syst.*, **35** (2006), 641–654. https://doi.org/10.1080/03081070600687668
25. L. M. Pan, M. H. Zhang, P. Ju, H. He, M. Ishii, Vertical co-current two-phase flow regime identification using fuzzy C-means clustering algorithm and ReliefF attribute weighting technique, *Int. J. Heat Mass Transfer*, **95** (2016), 393–404. https://doi.org/10.1016/j.ijheatmasstransfer.2015.11.081
26. R. Sheikhpour, M. A. Sarram, R. Sheikhpour, Particle swarm optimization for bandwidth determination and feature selection of kernel density estimation based classifiers in diagnosis of breast cancer, *Appl. Soft Comput.*, **40** (2016), 113–131. https://doi.org/10.1016/j.asoc.2015.10.005
27. M. Taherkhani, R. Safabakhsh, A novel stability-based adaptive inertia weight for particle swarm optimization, *Appl. Soft Comput.*, **38** (2016), 281–295. https://doi.org/10.1016/j.asoc.2015.10.004
28. S. M. Vieira, L. F. Mendonca, G. J. Farinha, J. M. Sousa, Modified binary PSO for feature selection using SVM applied to mortality prediction of septic patients, *Appl. Soft Comput.*, **13** (2013), 3494–3504. https://doi.org/10.1016/j.asoc.2013.03.021
29. D. Ramyachitra, P. Manikandan, Imbalanced dataset classification and solutions: A review, *Int. J. Comput. Bus. Res.*, **5** (2014).

30. J. Wieczorek, C. Guerin, T. McMahon, K-fold cross-validation for complex sample surveys, *Stat*, **11** (2022), e454. https://doi.org/10.1002/sta4.454
31. T. T. Li, H. P. Huang, G. Y. Shi, L. Y. Zhao, T. J. Li, Z. Zhang, et al., TGF-β1-SOX9 axis-inducible COL10A1 promotes invasion and metastasis in gastric cancer via epithelial-to-mesenchymal transition, *Cell Death Dis.*, **9** (2018), 849. https://doi.org/10.1038/s41419-018-0877-2
32. Y. Zhong, L. T. Yang, F. Xiong, Y. He, Y. Y. Tang, L. Shi, et al., Long non-coding RNA AFAP1-AS1 accelerates lung cancer cells migration and invasion by interacting with SNIP1 to upregulate c-Myc, *Signal Transduction Targeted Ther.*, **6** (2021), 240. https://doi.org/10.1038/s41392-021-00562-y
33. Q. Q. Zhu, C. G. Zhang, T. Y. Qu, X. Y. Lu, X. Z. He, W. Li, et al., MNX1-AS1 promotes phase separation of IGF2BP1 to drive c-Myc–mediated cell-cycle progression and proliferation in lung cancer, *Cancer Res.*, **82** (2022), 4340–4358. https://doi.org/10.1158/0008-5472.CAN-22-1289
34. Y. Z. Wu, J. M. Luo, H. Li, Y. Huang, Y. R. Zhu, Q. Q. Chen, B3GNT3 as a prognostic biomarker and correlation with immune cell infiltration in lung adenocarcinoma, *Ann. Transl. Med.*, **10** (2022), 295. https://doi.org/10.21037/atm-22-493
35. Y. Y. Wang, M. Li, L. Zhang, Y. T. Chen, M. W. Ha, LINC01140 inhibits nonsmall cell lung cancer progression and cisplatin resistance through the miR-4742-5p/TACC1 axis, *J. Biochem. Mol. Toxicol.*, **36** (2022), e23048. https://doi.org/10.1002/jbt.23048

AIMS Press