Improving the generalization of unsupervised feature learning by using data from different sources on gene expression data for cancer diagnosis

Zhen Liu[ab], Ruoyu Wang[c], Wenbin Zhang[d]

[1a]School of Information Science and Technology, Guangdong University of Foreign Studies, Guangzhou, 510006, China

[b]School of Medical Information Engineering, Guangdong Pharmaceutical University, Guangzhou, 510006, China

[c]Information and Network Engineering Research Center, South China University of Technology, Guangzhou 510041, China

[d]University of Maryland, Baltimore County, MD 21250 USA

Abstract:

Machine learning techniques have been utilized on gene expression profiling for cancer diagnosis. However, the gene expression data suffers from the curse of high dimensionality. Different kinds of feature reduction methods have been proposed to decrease the features of specific cancer diagnosis. However, as the difficult of obtaining the samples of a particular tumor, the lack of training samples may lead to overfitting problem. In addition, the feature reduction model on a specific tumor may lead to the problem that the model is not scalable and can not be generalized to new cancer types. To handle these problems, this paper proposes an unsupervised feature learning method to reduce the data dimensionality of gene expression data. This method amplifies the training samples of feature learning by utilizing the unlabeled samples from different sources. Two heuristic rules are devised to check if the unlabeled samples could be used for amplifying the training set. The amplified training set is used to train the feature learning model based on Sparse Autoencoder. Since the method leverages the knowledge among the expression data from different sources, it improves the generalization of unsupervised feature learning and further boosts the cancer diagnosis performance. A series of experiments are carried out on the gene expression datasets from TCGA and other sources. Experimental results prove that our method improves the generalization of cancer diagnosis when unlabeled data is used for latent feature learning.

Keywords: unsupervised feature learning, generalization, gene expression data, cancer diagnosis, unlabeled data, sparse autoencoder

1 Introduction

Gene expression profiling has emerged as an effective technique for cancer diagnosis[1][2]. A variety of machine learning techniques[3][4] have been used on gene expression data, as its efficiency used in the field of bioinformatics[5][6]. Specifically, some of them [7][8][9] have been applied to build the cancer diagnosis models. However, one of the major drawbacks of gene expression data is the curse of dimensionality. There are usually more than ten thousand of features, however only hundreds of samples for a given tumor. For example, in TCGA datasets, most of the tumor types only have several hundred samples and some even have less than one hundred samples while there are 60,483 features for each sample [10]. This problem hinders the usefulness of information in datasets and leads to computational instability[11]. Another major drawback of gene expression data is the small sample size (usually less than 1000 for a given tumor). The trained model is in the risk of overfitting.

To handle the curse of dimensionality, a variety of feature selection and feature extraction methods [12] have been proposed to reduce the data dimensionality of gene expression data. For example, the filter feature selection includes Relief [13], Correlation-based feature selection[14] and mRMR[15]; the wrapper feature selection includes Genetic Algorithm-Support Vector Machine [16] and leave-one-out calculation sequential [17]. The

[1*] Corresponding Author: rywang@scut.edu.cn(Ruoyu Wang)

hybrid feature selection methods include Intelligent Dynamic Genetic Algorithm[18], Mutual Information Maximization and the Adaptive Genetic Algorithm [19]. The feature extraction methods include PCA based method [20] and Autoencoder[21]. Most of existing works are in the supervised way and they usually perform on the samples from a specific tumor. This leads to the problems with those methods that they are mostly not scalable and can not be generalized to new cancer types [22]. The effective information of tissue samples from other cancers are not used when building the feature learning models. As a result, the learned model can not be used for other cancers, for example, the breast cancer detection is learned, being restricted to only data from breast cancer and normal tissue when building the classifier.

To handle the small sample size problem, multi-task learning methods have been used to amplify the training samples. Recently, Liao et al. [23] proposed a multi-task deep convolutional neural network. The samples from multiple sources are used to train a multi-task model. It enhances the classification performance by leveraging the knowledge through the shared units in the hidden layers. However, the samples from new platforms can not be used to update the trained model. In addition, their method is supervised, as a result it can not utilize the unlabeled data from different sources.

The unlabeled samples that are used for enhancing the training set may be from different cases. (1) Case 1: The training data (S_1) and unlabeled data (S_2) are from the same platform but different tumors; (2) Case 2: The S_1 and S_2 are from different platforms but the same tumor; (3) Case 3: The S_1 and S_2 are from different platforms and different tumors.

Most works mainly consider the first case. It is more difficult to utilize the samples from the other two cases. In this paper, to handle the curse of high dimensionality and small sample size, we propose an unsupervised feature learning method. The experiments are carried out from the above three cases. The objects of our method include: (1) the unlabeled data from different sources (different tumors or platforms) could be used for feature learning; (2) the unsupervised feature learning can learn the latent representations in the data from different sources. The main contributions of this paper are as below:

(1) To handle the two problems of small sample size and high dimensionality, a new unsupervised feature learning method named SAEET(Sparse AutoEncoder based on Enhancing the Training set) is proposed. In which, two heuristic rules are devised to select the unlabeled samples that could be used for enhancing the training set. It mainly contains two stages. In the first stage, the unlabeled samples from other sources are checked and selected according to our heuristic rules, and then the selected samples are combined with the training set. In the second stage, the sparse autoencoder is used to learn the latent features on the enhanced training set. Our method can leverage the knowledge of the unlabeled data from different sources (even from different platforms).

(2) This paper carries out multiple experiments from above three cases by downloading the benchmark datasets from public databases. Our method is compared with recently proposed methods. The results show that our method is able to improve the performance of SAE through leveraging the knowledge of unlabeled data. Our experiments are carried out from more cases than existing papers.

This paper is an expanded version of our conference paper [24]. In contrast to our conference paper, this paper further analyzes the data characteristics of gene expression data, in order to illustrate the similarity between the gene expression data from different sources and the potential of utilizing the tumor data from different sources. In addition, we take more experiments in this paper to evaluate the performance of our proposed method and to discuss the parameters of our proposed method.

The rest of this paper is organized as follows. Section 2 overviews the related works of the feature selection and the feature extraction on gene expression data. Section 3 analyzes the gene expression data from different sources or different tumors and presents our unsupervised feature learning method. Section 4 introduces the experimental datasets and evaluation

metrics. Section 5 evaluates the performance of our proposed method by taking experiments into multiple cases. Section 6 concludes this paper.

2 Related Work

2.1 Feature selection methods on gene expression data

The feature extraction and feature selection approaches have been used for decreasing the feature dimensionality[10] [11][12][25]. Most of works focus on the feature selection methods in the field of gene expression data research. The feature selection methods aim at selecting a subset from the full feature set by removing the redundant and irrelevant features. The feature selection methods used on gene expression data could be summarized into filter feature selection, wrapper feature selection and hybrid feature selection[26]. Potharaju et al. [27] proposed a distributed feature selection strategy for microarray gene expression data. This method combines SU (Symmetric Uncertainty) and CFS (Correlation-based Feature Selection) to rank the features according the SU and CFS scores. The features are further clustered according to the rank index in the feature set. Then, MLP (Multi-Layer Perceptron) is trained on each cluster, and based on the highest accuracy and minimum RMS error rate nominate the dominant cluster. Wahid et al. [28] proposed a feature selection method exploiting correlation based overlapping analysis of expression data across classes. It selects features with low overlapping and high correlation to classes. Uzma et al. [29] presented an unsupervised two-stage feature selection technique. The first stage aggregates three filter methods, namely principal component analysis, correlation, and spectral-based feature selection techniques. Next, the genetic algorithm is used, which evaluates the chromosome utilizing the autoencoder-based clustering. Manosij et al. [37] proposed a method that combines the ReliefF, chi-square, and symmetrical uncertainty methods. It firstly obtains the union and intersection of the three filter feature selection methods. Next, the genetic algorithm (GA) is used on the union and intersection feature subset to get the fine-tuned results.

2.2 Feature extraction methods on gene expression data

The feature extraction methods reduce the dimensionality by building new features from the combinations (linear or nonlinear) of original features. The most well-known dimensionality reduction algorithm is principal component analysis (PCA) [20]. PCA and its variations have been applied as a way of reducing the dimensionality of the data in cancer microarray data[20] [30]. Recently, deep learning methods are used in the field of feature selection for gene expression data. Fakoor et al. [22] firstly utilized PCA to obtain an initial low dimension feature set. The obtained feature subset is combined with a subset of features randomly selected from the origin feature set. Then, the data with this feature set is used as the input of autoencoder to learn a low dimensional feature set. Huynh et al. [31] proposed a method named DCNN-SVM(deep convolutional neural networks-support vector machine). This method utilizes the DCNN to automatically extract features from microarray gene expression data, and then learns a non-linear SVM on gene expression data. Danaee et al. [32] applied stacked denoising autoencoder to deeply extract functional features from high dimension gene expression profiles. Using deep learning methods for feature extraction, the small sample sets (i.e. a small number of training examples) increase the risk of overfitting[22].

Fakoor et al. [22] utilized the spare autoencoder method to learn a concise feature subset from unlabeled data. In contrast to the previous methods where the data has to be strictly from the cancer type to be detected in order to provide the appropriate label for supervised learning, the unlabeled data can here be obtained by combing data from different tumor cells provided that they are generated using the same microarray platform (i.e. given that they contain the same gene expression information). However, the data from different platforms can not be utilized. In addition, there is no method to assess which kind of unlabeled data could be used to enhance the training data. In this paper, we aim at applying the unlabeled data to enhance the performance of unsupervised feature learning. The learned model could be used for reducing the data dimensionality from different sources. As the unlabeled data from other

tumors are used, the model trained on the data of one tumor could be used for detecting other kinds of tumors.

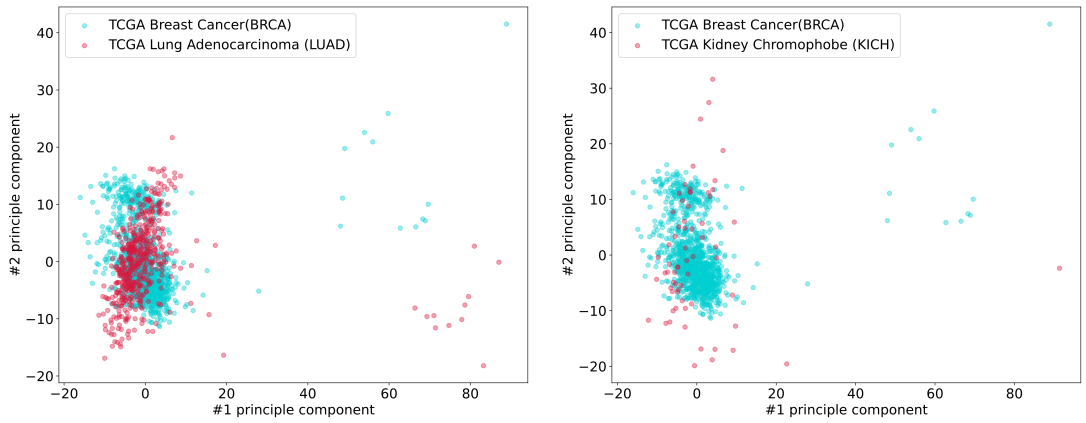
3. Methodology

3.1 Data Analysis

In this section, the gene expression datasets are visualized by the first two principal components obtained by PCA. Before performing PCA on these datasets, Min-Max normalization method is used to scale the feature values in the range of [0,1]. The three cases about the relation between the data from different sources illustrated in Section 1 will be analyzed as below.

3.1.1 Data analysis in Case 1

In Case 1, S_1 and S_2 are from the same platform but different tumors. The first two principal components of each dataset are shown in Fig. 1.



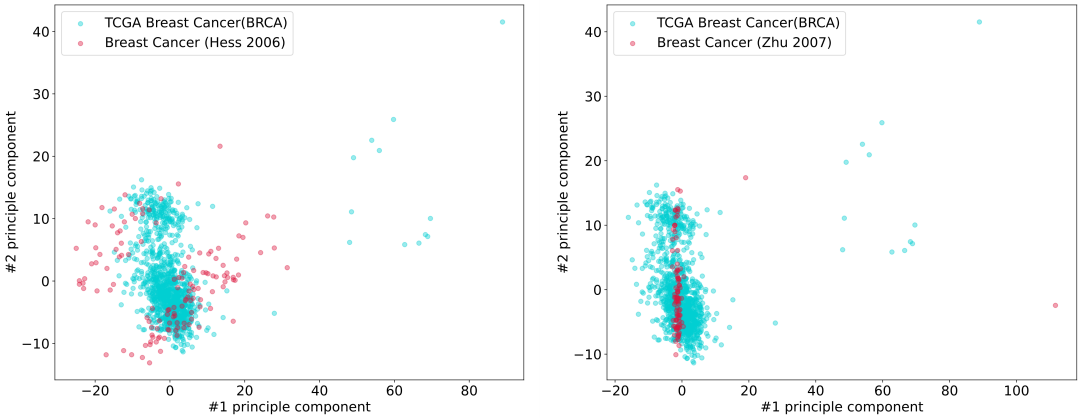
(a) BRCA vs. LUAD (b)BRCA vs. KICH

Fig.1 The first two principal components of the data in Case 1

In Fig. 1(a), S_1 denotes the BRCA data from TCGA and S_2 denotes the LUAD data from TCGA. In Fig. 1(b), S_1 denotes the BRCA data from TCGA and S_2 denotes the KICH data from TCGA. Fig. 1(a) shows that most of data points are overlapping but the data distribution is not similar between BRCA and LUAD. Fig. 1(b) shows that less samples are in overlapping area than those samples in Fig. 1(a).

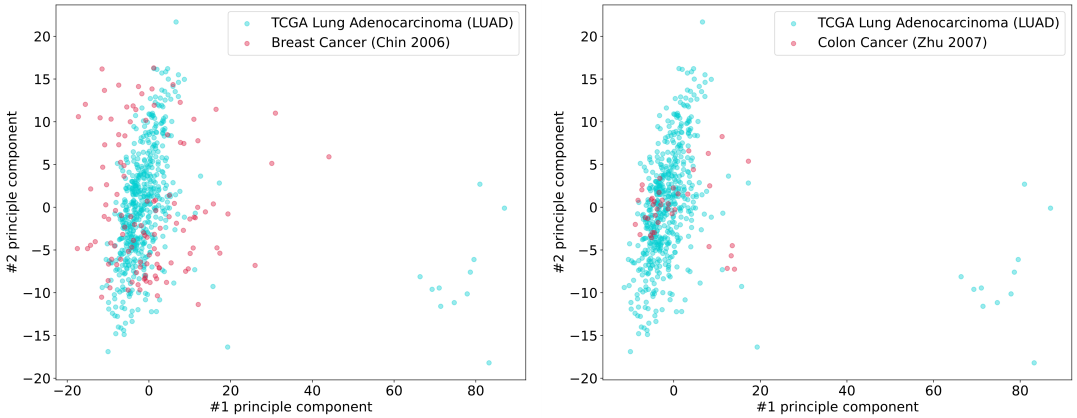
3.1.2 Data analysis in Case 2

In Case 2, S_1 and S_2 are from the same cancer but different platforms. The first two principal components of each dataset are shown in Fig. 2. In Fig. 2(a), S_1 denotes the Breast cancer data from TCGA, and S_2 denotes the Breast Cancer from the paper [33]. In Fig. 2(b), S_1 denotes the Breast cancer data from TCGA, and S_2 denotes the Breast Cancer from the paper[35]. Both of the two datasets are from Breast cancer but collected by different platforms. Fig. 2 shows that some data points are in the same feature area. They have some overlapping areas. It seems that the Breast Cancer (Zhu 2007) data is more similar to TCGA Breast Cancer than Breast Cancer(Hess 2006).



(a) Breast cancer (TCGA) vs. (Hess 2006) (b) Breast cancer (TCGA) vs. (Zhu 2007)

Fig. 2 The first two principal components of the data in Case 2



(a) LUAD vs Breast Cancer (b) LUAD vs Colon Cancer

Fig. 3 The first two principal components of the data in Case 3

3.1.3 Data analysis in Case 3

The visualization results of the datasets from different tumors and different platforms are shown in Fig. 3. In Fig. 3(a), S_1 is the dataset from Lung cancer of TCGA and S_2 is the dataset of Breast Cancer (Chin 2006) [34]. In Fig. 3(b), S_1 is the dataset from Lung cancer of TCGA and S_2 is the dataset of Colon Cancer (Zhu 2007). The two datasets in Figs. 3(a) and (b) are from different platforms and different tumors. Fig. 3 shows that less datapoints are in the overlapping region when compared with the datasets in Case 1 and Case 2.

Above figures visualize the data points distribution in the first two principal components space. These figures show that some samples from other sources (data from different tumors or different platforms) are close to the training data (i.e., the S_1 shown in the above three figures). This inspired us to utilize these samples to improve the performance of feature learning. Our method is introduced in the following section.

3.2 SAEET method

3.2.1 SAEET flowchart

As the high dimensionality of gene expression data, the feature learning to reduce the data dimension is a pre-processing for building a cancer classification model. A variety of feature learning algorithms have been proposed on gene expression data. However, it is lack of research on the feature learning for cross-cancer classification (the generalization of the feature learning model to new cancer types).

According to the analysis results in the above section, it is possible to use the data from other sources to improve the feature leaning performance. As labeled data are hard to obtain, we aim at utilizing the unlabeled data to improve the feature learning performance. The unsupervised feature learning algorithm Sparse Autoencoder (SAE) is used int this paper. In this section, we briefly introduce our proposed method. It is named as SAE based on Enhancing the Training set (SAEET). It aims at improving the generalization of SAE by utilizing the unlabeled samples from other sources. The flowchart of our method is shown as Fig. 4.

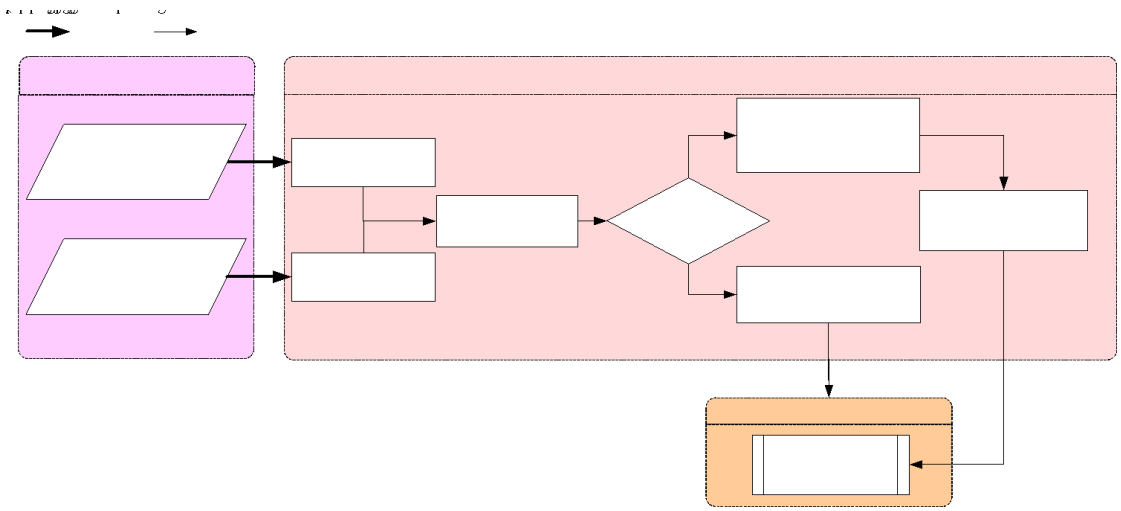


Fig. 4 The flowchart of SAEET method

As shown in Fig. 4, the input of SAEET is the training data (denoted by S_1) and unlabeled data (denoted by S_2). The output of SAEET is the feature learning model that is used to learn low dimensional features for cancer classification. Firstly, on the S_1 and S_2 data, the Min-Max normalization is performed. And then, we check if the samples in S_2 could be used for enhancing S_1. This is because some unlabeled data may be not good enough to enhance the performance of the unsupervised feature learning model. In this paper, we propose two heuristic rules to select the potential samples from S_2. If S_2 satisfy our heuristic rules, the selected samples from S_2 are combined with S_1, and then the combined sample set is used as the new training set of SAE algorithm. Otherwise, only the S_1 is used to training the SAE model.

We could finally obtain the SAE model by our method. The input of the SAE model is the data with full feature set and the output of SAE model is the data with reduce feature size. The unlabeled data is used for enhancing the training data of SAE model by using SAEET. It aims at improving the generalization performance of SAE model.

3.2.2 SAE (Sparse Autoencoder)

An auto-encoder is an unsupervised back-propagation neural networks algorithm for feature extraction [36]. It aims at minimizing the reconstruction error between input and output. It is a symmetrical structure of artificial neural networks as shown in Fig. 5.

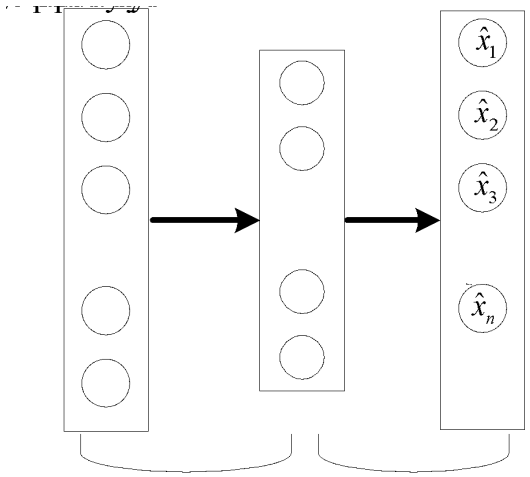


Fig. 5 The Auto-encoder diagram

A single auto-encoder consists of two stages: encoder and decoder. An encoder aims to map an input vector x into a hidden representation h through an encoding function:

$$h = f(x) = \gamma_1(W_1 x + b_1) \tag{1}$$

Where γ_1 is a non-linear activation function. The hidden representation h is then reconstructed using decoding function in order to generate the output vector y. The decoding function γ_2 is also a non-linear mapping function shown as:

$$y = g(h) = \gamma_2(W_2 h + b_2) \tag{2}$$

W_1 and W_2 represent the weight matrices of the hidden layer and output layer respectively. The b_1 and b_2 are the bias vectors of the hidden layer and output layer, respectively.

Eq. (1) corresponds to the compression function, from which an encoded input representation is obtained. Eq. (2) corresponds to the decoder part, where the AE reconstructs the input from the information contained in the hidden layer.

The object of AEs is to minimize the reconstruction error, which measures the differences between origin input and the consequent reconstruction. As suggested in [22], the sparse autoencoder is used on gene expression data. The overall cost function is defined as

$$J_{sparse}(W,b) = J(W,b) + \frac{\lambda}{2}\sum_{l=1}^{n_l-1}\sum_{i=1}^{s_l}\sum_{j=1}^{s_{l+1}}(W_{ji})^2 + \beta\sum_{j=1}^{s_l} KL(\rho \,\|\, \hat{\rho}_j) \tag{3}$$

$$J(W,b) = \left[\frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{2}\| h_{W,b}(x^{(i)}) - x^{(i)} \|^2\right)\right] \tag{4}$$

$$KL(\rho \,\|\, \hat{\rho}_j) = \rho \log\frac{\rho}{\hat{\rho}_j} + (1-\rho)\log\frac{1-\rho}{1-\rho_j} \tag{5}$$

Where n_l is the number of layers, s_l is the number of neurons in the layer l, β controls the weight of sparsity penalty term. The first term of Eq.(3) $J(W,b)$ is an average sum-of-squares error term. The second term of Eq.(3) is a regularization term that tends to decrease the magnitude of the weights, and helps prevent overfitting. The third term of Eq.(3) is the

Kullback-Leibler(KL) divergence between a Bernoulli random variable with mean ρ and a Bernoulli random variable with mean $\hat{\rho}_j$. The $\hat{\rho}_j$ is the average activation of hidden unit j (averaged over the training set). It is defined as

$$\hat{\rho}_j = \frac{1}{m}\sum_{i=1}^{m}[a_j^{(2)}(x^{(i)})] \tag{6}$$

Where $a_j^{(2)}(x^{(i)})$ denotes the activation of hidden unit (the second layer) j in the autoencoder when the network is given a specific input $x^{(i)}$. The ρ is a sparsity parameter, typically a small value close to zero.

3.2.3 Heuristic rules

In SAEET, we devise two heuristic rules to check if the unlabeled data potentially enhance the generalization of SAE. The heuristic rules are based on the similarity evaluation between the labeled data and unlabeled data. The data similarity are evaluated from two aspects in this paper. First one is from the data probability distribution. Second one is from the aspect of data points position. The first one mainly reflects the data distribution. The second one reflects the data positions in the feature space. The detail of heuristic rules and evaluation metrics are illustrated as below.

Heuristic rule #1 data probability distribution similarity: Given two datasets S_1 and S_2, if the maximum p value of the K-S test among the p values obtained by each dimension of S_1 and S_2 is higher than 0.05, their data distributions are not significantly different.

To evaluate the data probability distribution similarity between the two datasets, K-S(Kolmogrov-Smirnov) test is used. K-S test aims at evaluating the agreement between two cumulative distributions. Assume S_1 is represented by the ith feature (S_1={ $x_{1i}, x_{2i},\ldots,x_{mi}$ }) and S_2 represented by the ith feature (S_2={$x'_{1i}, x'_{2i},\ldots,x'_{ni}$}). Suppose that one sample set S_1 of size m has distribution with c.d.f $F(x)$ and another sample set S_2 of size n has distribution with c.d.f $G(x)$, we want to test

$$H_0: F=G \quad \text{vs.} \quad H_1: F\neq G \tag{7}$$

If $F_m(x)$ and $G_n(x)$ are corresponding empirical c.d.f then the statistic is

$$D_{mn} = \left(\frac{mn}{m+n}\right)^{1/2} \sup_x |F_m(x) - G_n(x)| \tag{8}$$

If the $D_{mn}<D(a/2, n)$(i.e., the p value of K-S test is higher than 0.05), we accept the assumption. This denotes that the data distributions of F and G are not significantly different when the data is represented by the ith feature.

Another heuristic rule is the data overlapping evaluation metric. It is defined as below.

Heuristic rule #2 data overlapping: Given two datasets S_1 and S_2, if the overlapping ratio calculated by Eq. (9) between S_1 and S_2 is higher than a threshold ε, they are overlapping.

The data samples may be clustered into different areas of the feature space. To assess the overlapping ratio in different clusters, a density-based clustering algorithm (DBSCAN) is firstly performed on the union set of S_1 and S_2. Among all clusters obtained by DBSCAN, the overlapping ratio is assessed by Eq.(9).

$$\textit{Overlapping ratio} = N_s/N_{total} \tag{9}$$

Where N_s is the number of samples from S_2 that are in the same cluster with the samples from S_1. N_{total} is the sample size of S_2. For example, in Fig. 6, assuming that the stars are from S_2 and the dots are from S_1. The overlapping ratio equals to the number of star samples in the

big oval divided by the total number of star samples. In Rule #2, the threshold ε is set as 0.6. This means that more than 60% of samples from S_2 are in the overlapping area.

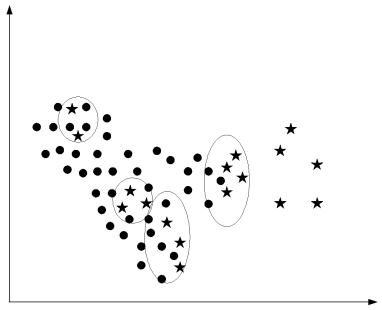


Fig. 6 The overlapping ratio calculation example

Because some samples may be far from the samples of S_1 (not in the same cluster with the samples of S_1) even though the overlapping ratio is higher than 0.6. Therefore, not all samples from S_2 are combined in S_1. The unlabeled samples that used for enhancing the training set are selected from the clusters that with the samples from S_1. The samples in the same clusters are more similar to S_1.

The K-S test is calculated between two sequences. Each time, the K-S test is performed on the samples represented by only one feature dimension. It is difficult to find the unlabeled samples that could be significantly similarity on all feature dimensions. Therefore, the rule #1 that used to check if the data could be used for enhancing the training set is that the p value of K-S test on at least one dimension is higher than 0.05.

Base on the above two heuristic rules, the pseudocode of SAEET is shown in Algorithm 1. The MINMAX is the function to do the min-max normalization on S_1 and S_2. The **getRuleMetrics** is the function to calculate the p value of K-S test on NS_1 and NS_2, and the overlapping ratio between NS_1 and NS_2. The *udata* returned by **getRuleMetrics** is the dataset that in the same cluster with the samples of NS_1. The *udata* will be combined with the training set when $p >= 0.05$, $r >=0.6$.

On the aspect of combing the samples from S_1 and the samples from S_2. If the they are from the same platform (their feature sizes are the same), they could be directly combined. The feature size means the number of features on a dataset. But if they are not the same, extra processing is required to make the features in the same size. In this paper, the feature size of the data from TCGA is 60483 that are more than the feature size of other datasets. Inspired from the zero-padding used in the image recognition, it is utilized in our algorithm as shown in the line 4 of Algorithm 1. The N_f is the feature size of S_1, i.e., the feature size of S_2 after zero-padding.

Algorithm1 SAEET algorithm

Input: labeled data S_1, unlabeled data S_2

Output: feature learning model **F**

1 NS_1= **MINMAX**(S_1)

2 NS_2=**MINMAX** (S_2)

3 if NS_1 and NS_2 are in Case 2 or Case 3:

4 NS_2=zeroPadding(NS_2,N_f)

5 *p, r, udata* = **getRuleMetrics**(NS_1,NS_2)

6 if $p >= 0.05$, $r >=0.6$:

7 NS=**Union**(NS_1, *udata*)

8 **F**=SAE(*NS*)

4. Experiment datasets and performance evaluation metrics

4.1 Experimental datasets

To evaluate the performance of cancer detection of our method , the publicly shared datasets from different sources are collected and used in the following experiments.

(1)TCGA datasets

In TCGA datasets[2], there are 11,093 human samples for mRNA gene expression quantification, which were collected from 26 different human anatomical organ sites and covering 33 different Cancer tumor types. Each individual human sample represents the whole transcriptome and includes a total of 60,483 genes annotated against a reference genome. In this paper, mRNA-seq data from the TCGA public dataset were downloaded from UCSC Xena. The data is in the Illumina HiSeq 2000 log2(fpkm+1) scaled. Due to the lack of samples and the imbalance between normal and tumor samples in most cancer types, we considered the 12 cancer types (15 datasets) as shown in Table 1. The features (i.e., genes) in these cancer data are the same and the number of features is 60483. The number of cancer and normal samples in each dataset is shown in Table 1. Table 1 illustrates that some tumor only owns less than 100 samples (such as KICH etc.), but the number of features is high. The feature reduction is necessary before training a cancer classification model.

Table 1 The detail of the TCGA datasets

Datasets	#Normal	#Tumor	Datasets	#Normal	#Tumor
BRCA	113	1104	ESCA	11	162
LUAD	59	526	HNSC	24	65
LUSC	49	501	LIHC	50	374
KICH	24	65	PRAD	52	499
KIRC	72	535	STAD	32	375
KIRP	32	289	THCA	58	510
BLCA	19	411	UCEC	35	548
COAD	41	471			

(2) Other datasets

On UCSC Xena, we also downloaded the Breast Cancer on UCSC Public Hub from different literatures (shown as the first five datasets in Table 2). The processed detail can be found on the UCSC Xena website. The last dataset (Breast Cancer(Zhu 2007)) in Table 2 is downloaded from http://csse.szu.edu.cn/staff/zhuzx/Datasets.html.

Table 2 The detail of other datasets

Datasets	No. samples	No. genes
Breast Cancer (Caldas 2007)	135	15340
Breast Cancer (Chin 2006)	118	21815
Breast Cancer (Hess 2006)	133	21815
Breast Cancer (vantVeer 2002)	117	11885
Breast Cancer (Vijver 2002)	295	19484
Breast Cancer(Zhu 2007)	97	24481

4.2 Performance evaluation metrics

The classification performance evaluation metrics are also calculated to evaluate the performance of feature reduction algorithm. The *Acc* and *Fscore* of cancer data are used as evaluation metrics. The *Acc* is defined as

[2] The Cancer Genome Atlas (TCGA) Research Network. Available: https://www.cancer.gov/ tcga.

$$Acc = \frac{TP + TN}{n} \tag{10}$$

The *Fscore* is the composite evaluation of *recall* (*R*) and *precision* (*P*). If *recall* is improved but *precision* drops significantly, and the *Fscore* could not be improved.

$$Fscore = \frac{2RP}{R + P} \tag{11}$$

where $R = \frac{TP}{TP + FN}$ $P = \frac{TP}{TP + FP}$

In the above equations, *TP* denotes the True Positives, that is the number of correctly identified cancer samples; *TN* denotes the True Negatives, that is the number of correctly identified normal samples, *FP* denotes the False Positives, that is the wrongly identified normal samples; and *FN* denotes the False Negatives, that is the number of wrongly identified cancer samples. In the following experiments, the *Fscore* of tumor class and that of normal class will be given and analyzed.

The Matthews correlation coefficient(MCC) is used in this paper to measure the quality of binary classification. It is generally regarded as a balanced measure. The MCC is in essence a correlation coefficient value between -1 and +1. A coefficient of +1 represents a perfect prediction, 0 an average random prediction and -1 an inverse prediction. It is defined as

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{12}$$

As the label information is not used in the feature learning process, the data could also be used for unsupervised learning. Therefore, the unsupervised learning method (such as K-means) is also performed on the data with reduced dimensionality. The normalized mutual information is used as the clustering metric in this paper. It is defined as

$$NMI(Y, C) = \frac{2 \times I(Y; C)}{H(Y) + H(C)} \tag{13}$$

Y is the class label set, and C the cluster label set. H denotes the entropy, I(Y;C) the mutual information between Y and C. NMI is a normalization of the Mutual Information (MI) score to scale the results between 0 and 1. The value of NMI is closer to 1, the performance of clustering is better.

5 Experimental results

In this section, we carry out a series of experiments to evaluate the performance of unsupervised feature learning method in the three cases illustrated in Section 1. The experiments are mainly performed on the TCGA datasets. The other datasets are used as the unlabeled datasets used in Cases 2 and 3. The Random Forest is used as the basic classification algorithm. The number of trees in Random Forest is set as 100. In the following experiments, Relu is used as the activation function as it is a popularly used activation function; the AdaGrad is used to optimize the weights and biases in SAE as it can set different learning rate for different parameters. The number of learned features by SAE is set as 100.

5.1 Heuristic rule metrics

This section mainly illustrates the metrics used in our heuristic rules. The evaluation results on the different real datasets are shown in Table 3. The column of "*P* value of K-S test" denotes the maximum of *P* value of K-S test among all dimensions of S_1 and S_2. In Case 1, all datasets are from TCGA datasets. In Case 2, S_1 is TCGA BRCA, S_2 includes the breast cancer

data as shown in Table 2. In Case 3, S_1 is TCGA UCUE, S_2 includes the breast cancer data as shown in Table 2. Among the three cases, the results show that more datasets in the second case satisfy our heuristic rules. According to above two rules, in the following experiments, the samples in TCGA UCUE could be used to enhance the training data with TCGA Breast cancer for training the SAE model in the Case 1. The samples in Breast Cancer (Caldas 2007) and Breast Cancer (Zhu 2007) datasets could be used for enhancing the training data with TCGA Breast cancer in the Case 2. The samples in Breast Cancer (Zhu 2007) dataset could be used to enhance the TCGA UCUE in Case 3.

Table 3 The results of the K-S test and the overlapping ratio metrics in the three cases

S_1	S_2	*P* value of K-S test	Overlapping ratio
Case 1: Same platform different cancers			
TCGA BRCA	KIRC	0.000	0.443
	LIHC	0.000	0.276
	THCA	0.000	0.283
	LUAD	0.000	0.602
	HNSC	0.000	0.408
	STAD	0.000	0.312
	BLCA	0.373	0.463
	KIRP	0.000	0.215
	PRAD	0.003	0.387
	UCEC	**0.094**	**0.662**
	COAD	0.000	0.660
	ESCA	0.000	0.116
	KICH	0.005	0.022
	LUSC	0.005	0.376
Case 2: Same cancer but different platforms			
TCGA BRCA	Breast Cancer (Chin 2006)	0.008	0.203
	Breast Cancer (VantVeer 2002)	0.026	0.667
	Breast Cancer (Vijver 2002)	0.026	0.487
	Breast Cancer (Hess 2006)	0.160	0.135
	Breast Cancer (Caldas 2007)	**0.713**	**0.578**
	Breast Cancer (Zhu 2007)	**0.013**	**0.814**
	Case 3: different cancer different platform		
TCGA UCUE	Breast Cancer (Chin 2006)	0.357	0.113
	Breast Cancer (VantVeer 2002)	0.161	0.171
	Breast Cancer (Vijver 2002)	0.014	0.488
	Breast Cancer (Hess 2006)	0.043	0.144
	Breast Cancer (Caldas 2007)	0.100	0.289
	Breast Cancer (Zhu 2007)	**0.578**	**0.67**

5.2 Performance evaluation on cross cancer classification

To evaluate the generalization of our feature learning method, the feature learning model is trained on the dataset with one cancer and tested on the datasets with different cancers. As illustrated in section 1, three cases will be analyzed in this section. Each experiment is repeated 10 times, and the average over the ten runs are shown in the flowing figures.

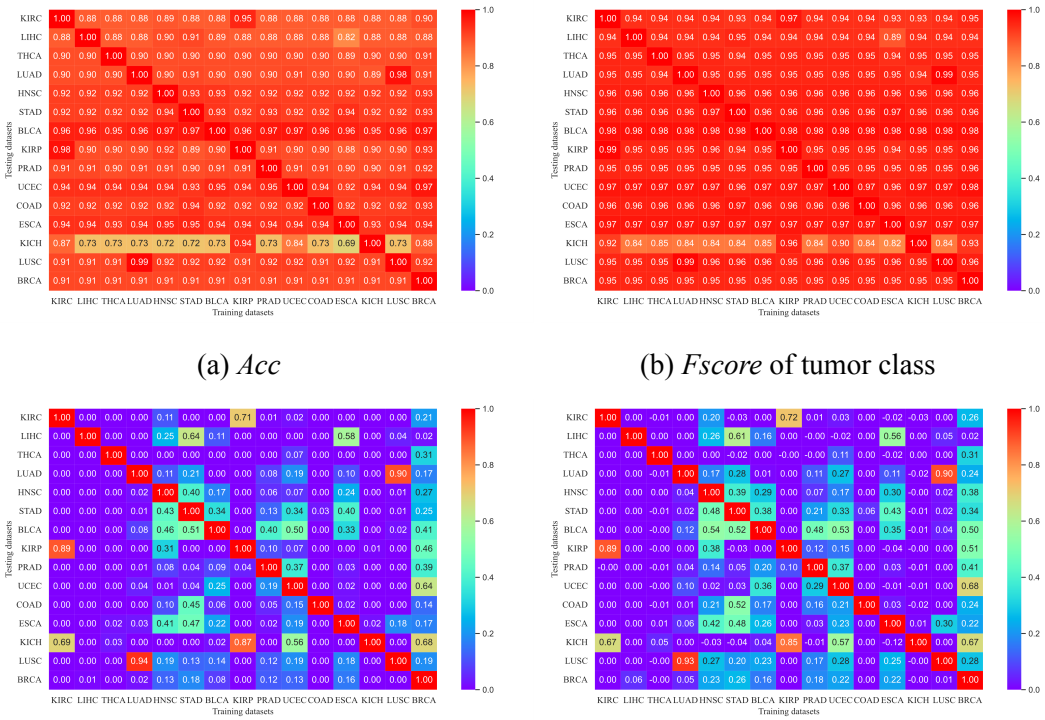
5.2.1 Results without training enhancement

The results without training set enhancement are shown in Fig. 7. The datasets in the x-axis denotes the training sets and those in the y-axis denotes the testing sets. For example, when KIRC is the training set, the model is tested on KIRC, LIHC, THCA ,…, and BRCA. Most of *Accs* and *Fscores* of tumor class are higher than 90%. However, the *MCC* and the *Fscore* of normal class are lower than 50% on most of testing datasets. This indicates that the model

mis-classify the normal class into the tumor class. The model biases towards the tumor class. This is resulted by the class imbalance problem.

The results also suggest that the best testing result on a dataset is obtained by the model trained on the data that is correlation to the testing data. For example, the best *MCC* on KIRC (0.72) and the best *MCC* on KICH(0.85) are all obtained by the model trained on the KIRP. The best *MCC* on LUAD(0.9) is obtained by the model trained on the LUSC dataset. the best *MCC* on LUSC(0.93) is obtained by the model trained on the LUAD dataset. This could be explained by the domain knowledge. These datasets contain the cancer samples from the same organ. For example, the KIRC, KICH and KIRP are the cancer on the Kidney. We also found another interesting result is that the best *MCC* on UCEC(0.67) is obtained by the model trained on the BRCA. The two cancers happen on women in high probability.

This suggests that it is possible to use the correlation dataset to improve the generalization performance of the feature learning model.



(a) *Acc* (b) *Fscore* of tumor class

(c) *Fscore* of normal class (d) *MCC*

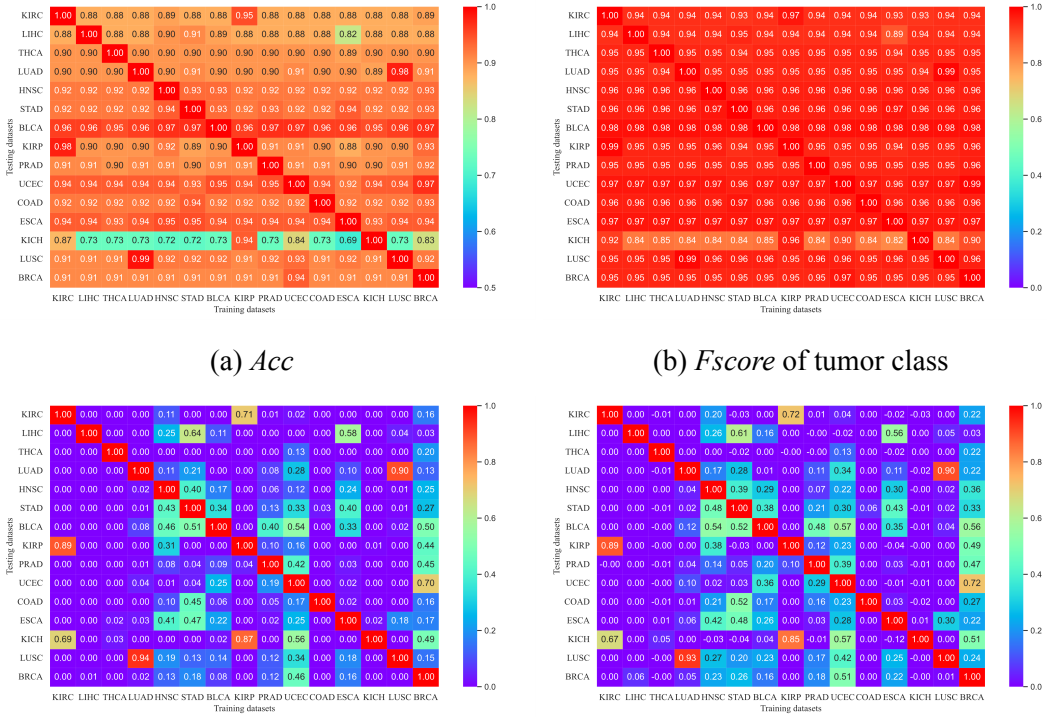
Fig. 7 The cross cancer results without enhancing the training set of feature learning

5.2.2 Results in Case1

This section mainly analyzes the performance in Case 1. That is, on TCGA dataset as shown in Table 1, one dataset is used as training set, other datasets are used as training set. When one dataset used as training set, all rest datasets are used as the unlabeled datasets and evaluated by the overlapping metric and K-S test. If the dataset satisfies the heuristic rules, the selected samples could be used to enhance the training set for building the SAE model as shown in SAEET algorithm. The unlabeled samples only participate in training the SAE model for reducing the data dimension, and not used for training the cancer classifier.

The *Accs*, *Fscores* and *MCCs* are shown in Fig. 8. When compared with results without enhancing the training set (results in Fig. 7), only on the UCUE and BRCA datasets, the two heuristic rules are satisfied. This means that only the training sets of the two datasets are enhanced. So, on the other datasets, the results are the same as those shown in Fig. 7. On average, the *Fscore* of normal class is improved about 6.2%, and *MCC* is improved about 5.2% when UCUE is the training set and other datasets are the testing sets. Especially when

tested on BRCA, the *MCC* is improved from 0.22 to 0.51. When BRCA is the training set, even though it reduces the *MCC* on some testing datasets, it improves the *MCC* from 0.68 to 0.72 on UCUE dataset.



(a) *Acc* (b) *Fscore* of tumor class

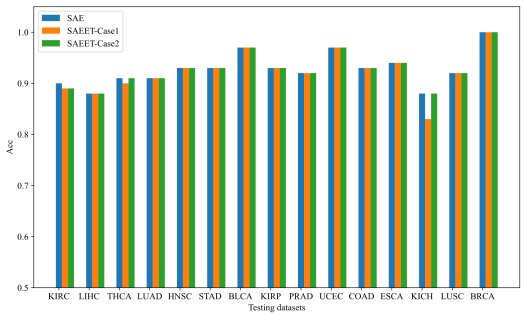
(c) *Fscore* of normal class (d) *MCC*

Fig. 8 The cross cancer results with enhancing the training sets of feature learning in Case 1

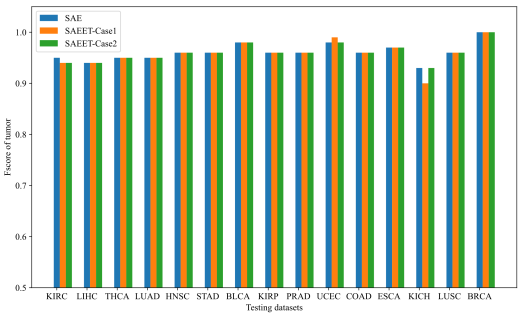
5.2.3 Results in Case 2

The above experimental results show that in the same platform, SAEET is able to improve the performance when the training set is enhanced by the unlabeled data. In this section, we further discuss the experiments in the Case 2. Taking the TCGA Breast data as the training set, the datasets from other platforms (i.e., the other datasets in Table 2) were used as the unlabeled data. The unlabeled samples that satisfies the heuristic rules could be added into the training data for learning the parameters of SAE model.

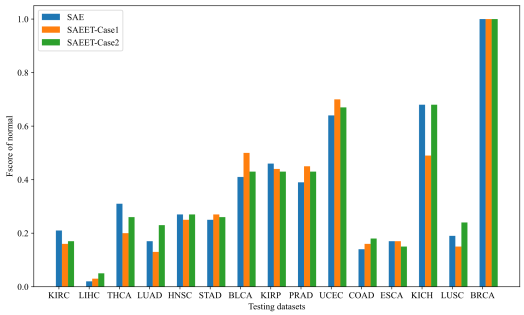
The model trained on BRCA dataset is tested on the datasets of TCGA. The *Acc*, *Fscore* and *MCCs* are shown in Fig. 9. The x-axis denotes the testing datasets and y-axis denotes the performance. The "SAE" denotes the results of SAE without extending training set, i.e. the 14th column in Fig. 7. The "SAEET-Case1" denotes the results when the same platform data is used to extend the training set, i.e., the 14th column in Fig. 8. The "SAEET-Case2" denotes the results when different platform data is used to extend the training set. Even though the *Acc* and *Fscore* on some datasets are decreased, the results on LIHC, LUAD, LUSC, UCUE are improved by using the same tumor data from the other platform to enhance the training set. On average, "SAEET-Case2" obtains 92.7% *Acc*, 96% *Fscore* of tumor clas, 36.3% *Fscore* of normal class and 41.6% *MCC*, which are higher than the "SAEET-Case1" (92.3% *Acc* and 95.9% *Fscore* of tumor class, 34% *Fscore* of normal class and 39.1% *MCC*). This demonstrates that the data from the same tumor but different platforms could be used for improving the performance of unsupervised feature learning.



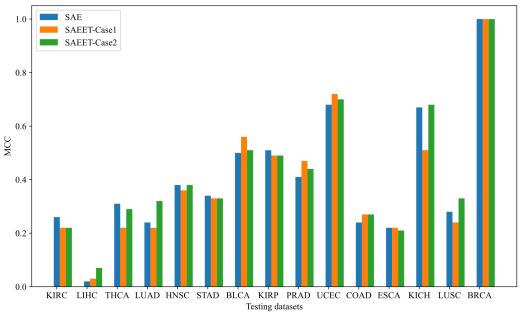
(a) *Acc*



(b) *Fscore* of tumor class

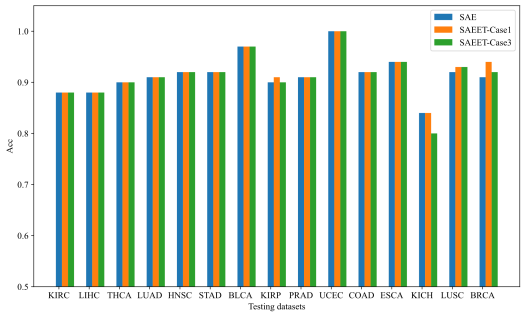


(c) *Fscore* of normal class

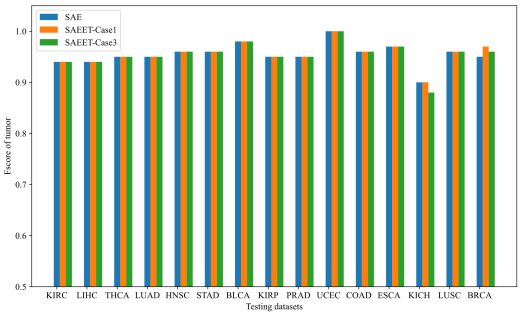


(d) *MCC*

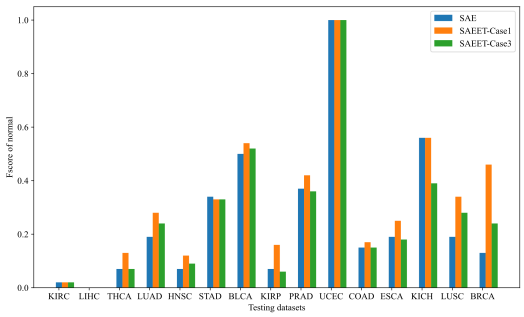
Fig. 9 The cross cancer results with enhancing the training sets of feature learning in Case 2



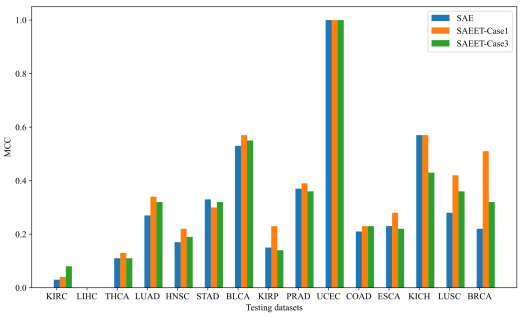
(a) *Acc*



(b) *Fscore* of tumor class



(c) *Fscore* of normal class



(d) *MCC*

Fig. 10 The cross cancer results with enhancing the training sets of feature learning in Case 3

5.2.4 Results in Case 3

In the Case 3, taking the UCUE dataset as the training set, the other datasets shown in Table 2 used for enhancing the training set. The *Accs, Fscores* and *MCCs* on the datasets of TCGA are shown in Fig. 10. The x-axis denotes the testing sets and y-axis denotes the performance results.

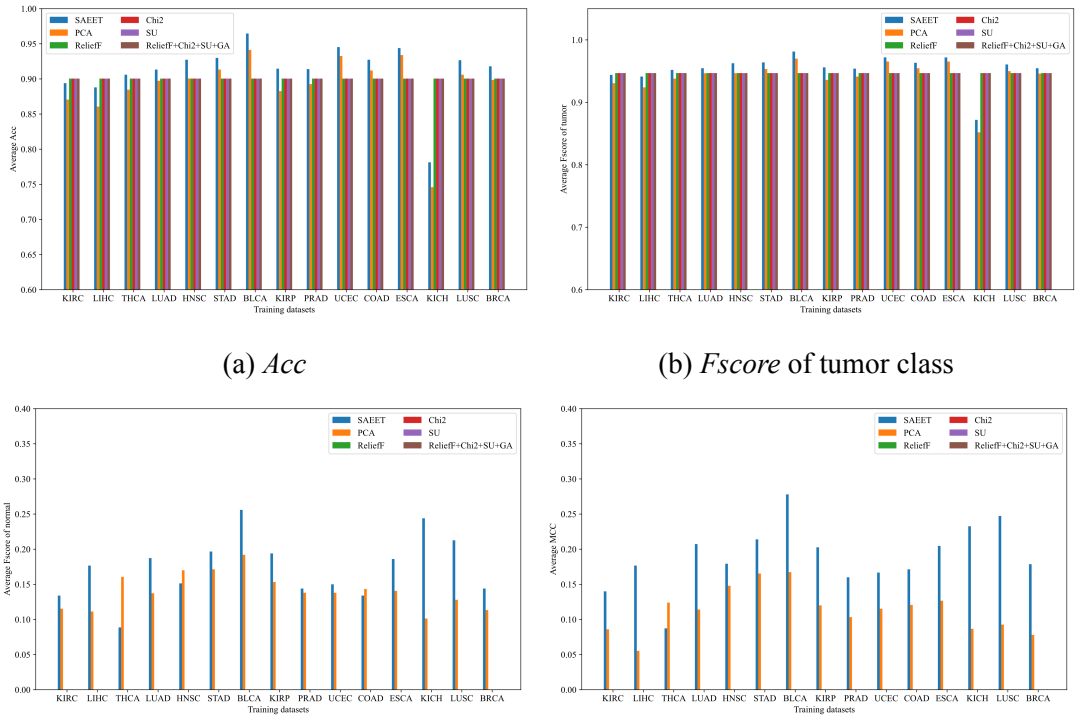
Similarly, "SAE" and "SAEET-Case1" respectively denote the results obtained by the model trained on UCUE without enhancing the training set and with enhancing the training set using the data from the same platform. The "SAEET-Case3" denotes the results obtained by the model trained on UCUE with enhancing the training set from different platforms and different tumor. The performance is worse than Case2. Nevertheless, the *MCC* on LUAD, LUSC and BRCA are significantly improved. This further demonstrate that the SAEET could improve the performance when the model is evaluated on some datasets. But it also suggests that it is difficult to improve the performance in Case3. Because the samples are from different platforms and different tumors, the informative samples are less than those samples in Cases 1 and 2.

5.3 Compared with existing methods

In this section, we further compare our method with existing feature selection algorithms including the supervised feature selection, unsupervised feature extraction and feature learning methods.

5.3.1 Compared with traditional feature selection/extraction methods

Taking the random forest as classification algorithm, the results compared with traditional feature selection/extraction methods are shown in Fig. 11. The x-axis represents the training sets and the y-axis represents the average *Accs*, *Fscores* and *MCCs* over all testing sets. On the TCGA datasets, one dataset is used as the training set and all rest of datasets are used as the testing sets.



(a) *Acc* (b) *Fscore* of tumor class

(c) *Fscore* of normal class (d) *MCC*

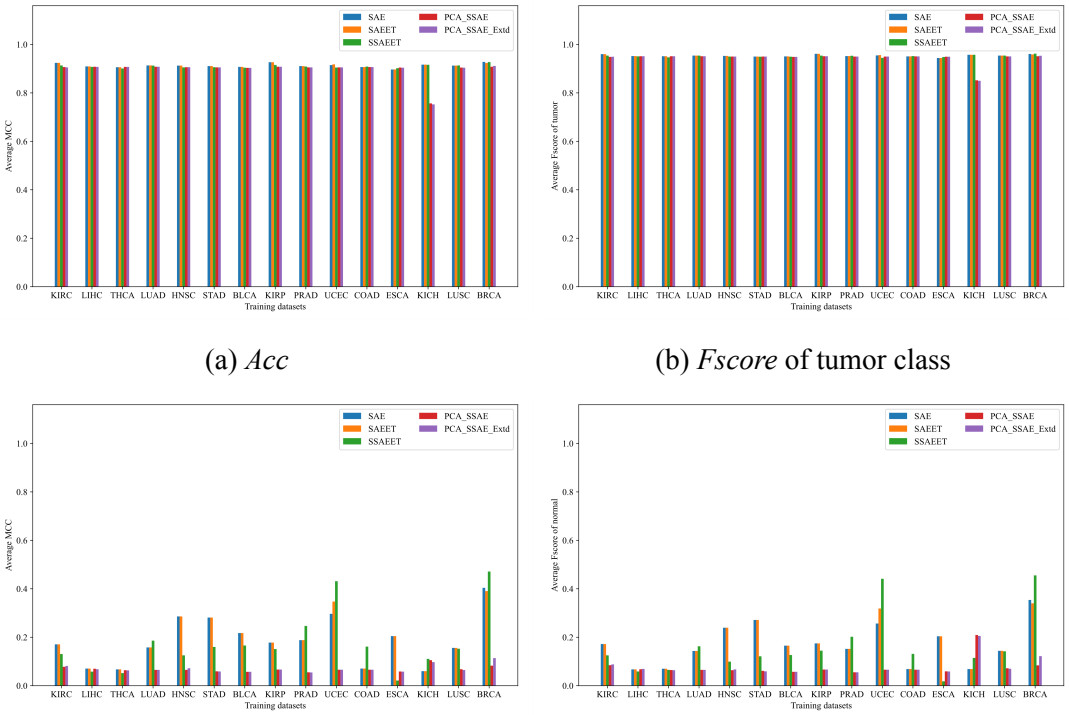
Fig. 11 The classification results when comparing our method with existing methods

In Fig. 11, "SAEET" denotes the results of the model obtained with SAEET with extending training sets using the unlabeled data in the same platform. Relief, Chi2 and SU are the supervised feature selection methods. The "ReliefF+Chi2+SU+GA" denotes the results of the

feature selection method proposed recently in [37]. The ReliefF, Chi2, SU and "ReliefF+Chi2+SU+GA" are supervised feature learning method. They perform feature selection algorithm on the labeled data. PCA is the unsupervised feature learning method. Fig. 11 shows that our method with extending the training set obtains the best *Acc*, *Fscore* and *MCC* on most datasets. The models with the supervised feature learning methods obtain 0% *Fscore* of normal class, as a result they obtain 0% *MCC* as shown in Figs11(c) and (d). This may be resulted by the class imbalance problem. The tumor class own more samples than the normal class. The selected feature subset biases towards the tumor class.

5.3.2 Compared with feature learning methods

In this section, our method is compared with the method proposed in [22] that utilizes the PCA to preprocess the origin feature set. The output of PCA and a subset of origin feature set is used as the input of SSAE (stacked SAE). It builds SSAE model to learn low dimensional features and utilizes Softmax as the classification algorithm. To fairly comparing, the stacked SAE is also used in the SAEET and the method is called SSAEET, and the Softmax is also used as the classification algorithm. The *Acc*, *Fscore* and *MCC* results are shown in Fig. 12. Similar to the results in Fig. 11, the x-axis is the training sets and the y-axis is the average *Accs*, *Fscores* and *MCCs* over all testing sets. The "SSAEET" denotes the results obtained by the SSAEET with extending the training set. The "PCA_SSAE" and "PCA_SSAE_Extd" respectively denote the results of the methods in [22] without and with extending the training data of SSAE. The extending method is the same as the SAEET, i.e., the two heuristic rules are used for selecting samples to extend the training set.



(a) *Acc* (b) *Fscore* of tumor class

(c) *Fscore* of normal class (d) *MCC*

Fig. 12 The comparison results with existing feature learning methods

When comparing the SSAEET with PCA_SSAE and PCA_SSAE_Extd, the results show that the SSAEET outperforms PCA_SSAE and PCA_SSAE_Extd. We further analyze the results and found that the features obtained by PCA resulted to the model obtaining 0% *Fscore of normal* on some testing datasets. Our method directly uses the origin feature set rather than the combination of PCA features and the subset of origin feature set. It avoids the shortcoming of PCA. When SSAEET compared with SAE and SAEET, it performs better on LUAD, PRAD, UCUE, COAD, and BRCA, but performs worse on KIRC, LIHC, HNSC,

STAD, BLCA, KIRP and ESCA. Softmax algorithm is used as the classification algorithm in SSAEET. Random Forest is still used in SAE and SAEET for classification. The results indicate that the Random forest used performs better than the Softmax when the number of training samples is small.

5.4 Discussions

5.4.1 Results evaluated by the clustering metrics

Since our method is a kind of unsupervised feature learning method, it does not utilize the labeled information. This method could be used in the case of cancer detection using clustering algorithms. The performance could be evaluated by the clustering evaluation metrics such as the *NMI* defined in section 4.2.

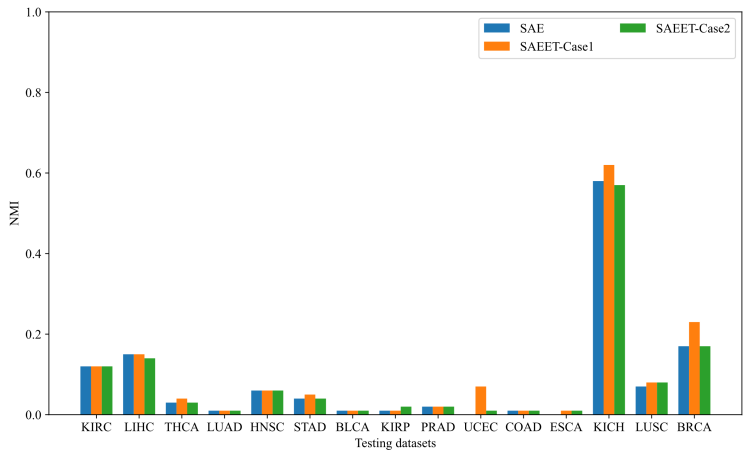


Fig. 13 The *NMI* results when the BRCA is the training set

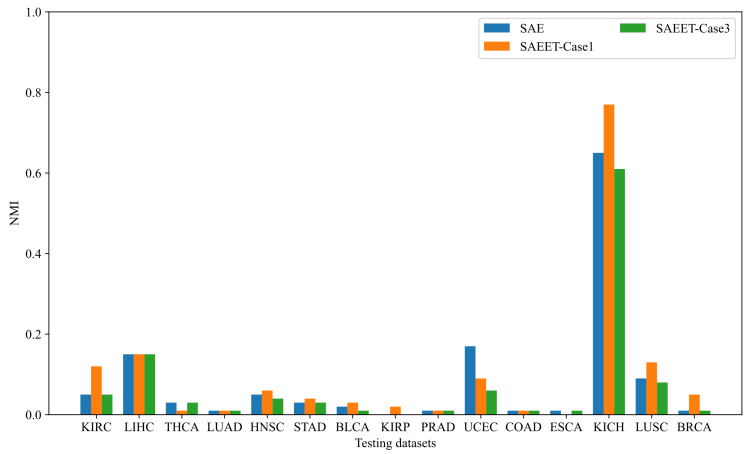


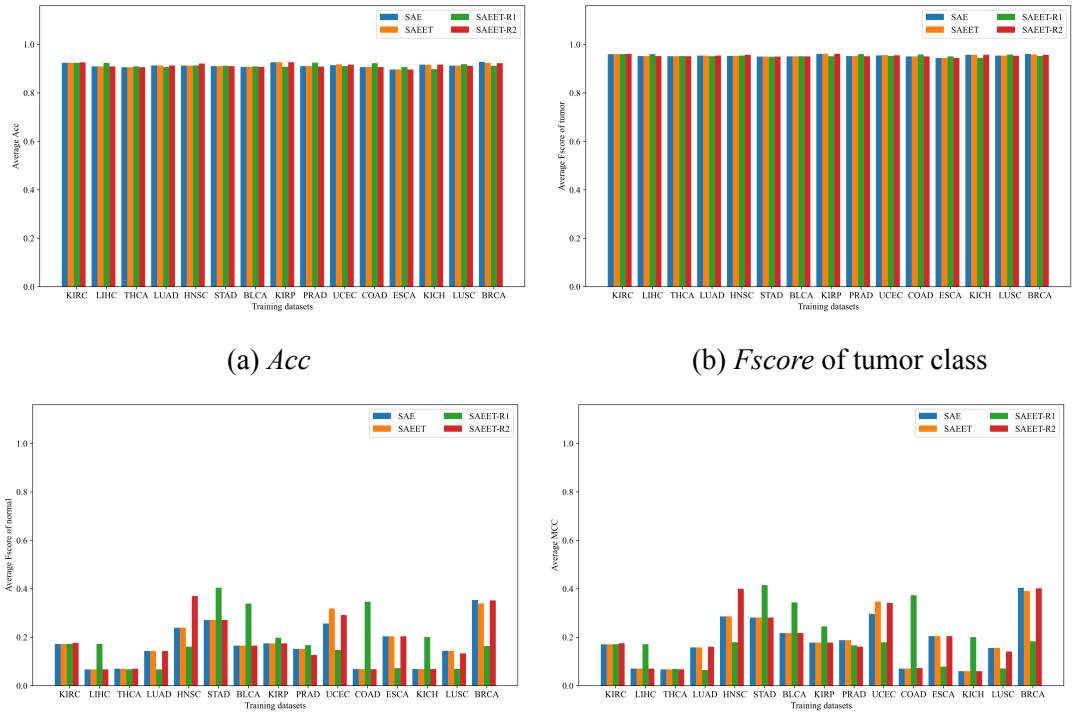
Fig. 14 The *NMI* results when the UCUE is the training set

According to the above results, we found that only the UCUE and BRCA are enhanced when using our method. Therefore, only the results of the two datasets are discussed in this section. The results of the methods when BRCA is used as the training set are shown in Fig. 13. The SAEET_Case1 denotes the result of the model trained on the BRCA in Case 1. The SAEET_Case2 denotes the results of the model trained on the BRCA in Case2. Also, the TCGA UCUE dataset is used as the training set and the testing sets are shown in Fig. 14. The results show that the SAEET_Case2 could improve the *NMI* when compared with SAE on most of testing datasets. In Fig. 13, when testing on UCUE and BRCA dataset, the *NMI* is improved about 7% and 6% respectively. In Fig. 14, when testing on KIRC, KICH and BRCA, the NMI is improved about 7%,4% and 4% respectively. The results also show that the SAEET in Case3 performs worse than the SAE on average. This may be resulted by that

the samples are from different platforms and different tumors. The informative samples that could be used to enhance the training set are less than those samples in Cases 1 and 2.

5.4.2 Discussion on the heuristic rules

This section mainly discusses the heuristic rules used in our method. SAEET-R1 denotes the method that only using the heuristic Rule #1 (i.e., the data probability distribution rule) to select the unlabeled samples for training set enhancement. Similarly, SAEET-R2 denotes the method that only using the heuristic Rule #2(i.e., the overlapping ratio rule). The *Acc*, *Fscores*, and *MCCs* are shown in Fig. 15. The results show that SAEET-R1 performs better than SAEET on some datasets (such as LIHC, STAD, BLCA, COAD and KICH), but it also decreases the *Fscore* of normal class and *MCC* one some other datasets (such as LUAD, HNSC, UCUE, ESCA and BRCA). When using SAEET-R1, there are more unlabeled samples for enhancing the training set. But some unlabeled samples may be not good enough to enhance the performance, as a result, it reduces the performance. Similarly, the SAEET-R2 has the same problem. SAEET with the two heuristic rules performs more stable than SAEET-R1 and SAEET-R2.



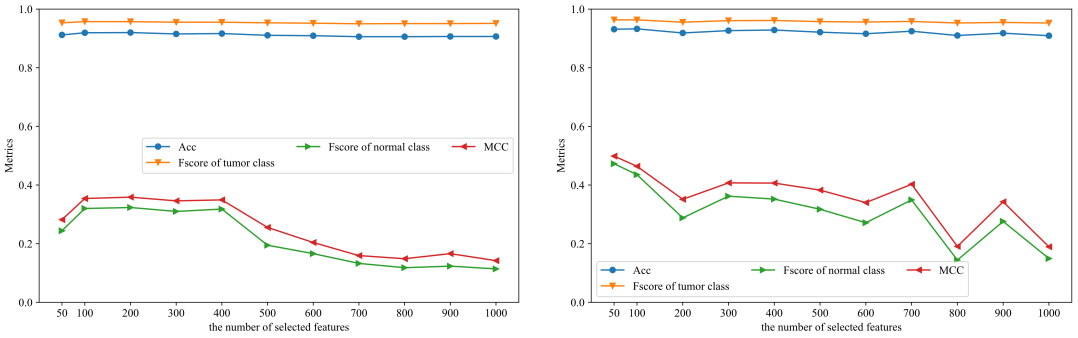
(a) *Acc* (b) *Fscore* of tumor class

(c) *Fscore* of normal class (d) *MCC*

Fig. 15 The results of SAEET with different heuristic rules

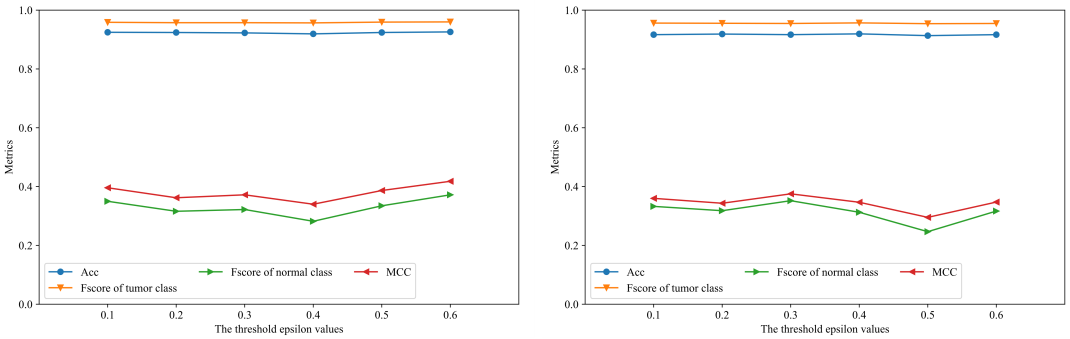
5.4.3 Discussion on the number of learned features

This section mainly discusses the influence of the number of learned features (i.e., the number of hidden neuros in SAE) in SAEET. The number of learned features is set in the range of [50, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000]. Similar to the heuristic rules' discussion experiments, only the UCUE and BRCA datasets are used as the training sets. The results show that there is no fluctuation when evaluated by the *Acc* and *Fscore* of tumor class. The *Fscore* of normal and *MCC* are decreased when learned more features. This may be resulted by the overfitting when more features are learned.



(a) Results on UCUE (b)Results on BRCA

Fig. 16 The results with different number of learned features in SAEET



(a) Results on UCUE (b)Results on BRCA

Fig. 17 The results with different thresholds of overlapping ratio in SAEET

5.4.4 discussion on the thresholds

This section mainly discusses the threshold ε in heuristic rule #2, which is set in the range of [0.1,0.2,0.3,0.4, 0.5,0.6]. Since there is no unlabeled samples satisfy our heuristic rules when ε>0.6, the values in the range from 0.1 to 0.6 are discussed in this section. The results with different values of ε are shown in Fig. 17. The results show that there is no fluctuation on *Acc* and *Fscore* of tumor class. According to the results of *Fscore* of normal class and *MCC*, 0.3 and 0.6 perform better than other values. To guarantee the quality of the selected samples, 0.6 is a conservation choice in our above experiments.

6. Conclusions and future work

Gene expression data have been used for cancer classification. However, it faces the high dimensionality and small sample size problems. To handle these problems, we propose an unsupervised feature learning method based on SAE, named SAEET. This method utilizes the unlabeled gene expression data from different sources to improve the performance of SAE. Before building the SAE model, the unlabeled data from different sources (different tumors or different platforms) could be combined with the labeled training data to build the SAE in the unsupervised way. So that more information is contained in the training data of SAE, and the generality of SAE model could be improved. TCGA and other Breast Cancer datasets are used in our experiments. A series of experiments are carried out to evaluate the performance of our method. The experimental results are summarized as below.

(1) The experimental results in the three cases demonstrate that SAEET could further improve the generalization performance of the SAE. Because SAEET utilizes the information of the unlabeled data, and more data information could be learned for training the SAE model.
(2) The experimental results of comparing SAEET with existing feature selection/extraction method show that SAEET obtains the best *Acc*, *Fscore* and *MCC* on most datasets.

(3) When SAEET is compared with PCA_ SSAE method [22] that is the one with the similar object with our method, the results show that SSAEET outperforms PCA_ SSAE method that applied PCA and SSAE as the feature learning algorithm.

In future, we will further research the feature alignment when the unlabeled data are from different platforms. The class imbalance problem leads to the classifier biases towards the tumor class. To handling the class imbalance problem and improving the *MCC* is also an interesting work in the future. In order to adapt to different domains, the transfer deep learning will be researched in the classification of cancers with gene expression data.

Acknowledgements

We thank the anonymous reviewers for their constructive comments. An earlier version of this paper [24] was presented at the International Conference on the 13th International Conference on Machine Learning and Computing. This work is supported by the Key research platforms and projects of colleges and universities in Guangdong Province [Grant No. 2020ZDZX3060], National Natural Science Foundation of China [Grant No. 61501128], financial support from China Scholarship Council, Natural Science Foundation of Guangdong Province [Grant Nos. 2017A030313345].

Compliance with ethical standards

Conflict of interest The authors declare that there is no conflict of interest regarding this work.

References

[1] Salem H, Attiya G, El-Fishawy N (2017) Classification of human cancer diseases by gene expression profiles, Applied Soft Computing 50: 124-134.
[2] Liu JX, Xu Y, Zheng C-H, H. Kong, Z.-H. Lai(2015) RPCA-based tumor classification using gene expression data. IEEE/ACM Transactions on Computational Biology & Bioinformatics 12 (4): 964-970.
[3] Mignone P, Pio G, Džeroski S, Ceci M(2020). Multi-task learning for the simultaneous reconstruction of the human and mouse gene regulatory networks. Scientific reports, 10: 22295.
[4] Erola P, Björkegren JLM, Michoel T(2020). Model-based clustering of multi-tissue gene expression data. Bioinformatics, 36(6):1807-1813.
[5] Bao W, Yuan C A, Zhang Y, Han K., Nandi A k, Honig B., Huang D(2017). Mutli-features prediction of protein translational modification sites. IEEE/ACM Transactions on Computational Biology & Bioinformatics, 15(5): 1453-1460.
[6] Bao W, Dong W, Chen Y(2017). Classification of protein structure classes on flexible neutral tree. IEEE/ACM transactions on computational biology and bioinformatics, 14(5): 1122-1133.
[7] Yuan F, Lu L, Zou Q (2020) Analysis of gene expression profiles of lung cancer subtypes with machine learning algorithms, BBA-Molecular Basis of Disease 866: 165822.
[8] Khorshed TA (2020) Deep learning for multi-tissue cancer classification of gene expressions (GeneXNet), IEEE Access 8: 90615-90629.
[9] Tirumala SS, Narayanan A (2016) Attribute selection and classification of prostate cancer gene expression data using artificial neural networks, Pacific-asia Conference on Knowledge Discovery & Data Mining, 2016, pp. 206-34.
[10] Khorshed T, Moustafa MN, Rafea A (2020) Multi-tissue cancer classification of gene expressions using deep learning, IEEE Sixth International Conference on Big Data Computing Service and Applications (BigDataService), 2020, pp. 128-135.
[11] Abdulla M, Khasawneh MT (2020) G-Forest: An ensemble method for cost-sensitive feature selection in gene expression microarrays, Artificial Intelligence in Medicine 108: 101941.
[12] Hira ZM, Gillies DF (2015) A review of feature selection and feature extraction methods applied on microarray data. Advances in Bioinformatics, 2015: 198363.

[13] Hall MA, Smith LA (1998) Practical feature subset selection for machine learning, 21st Australasian Computer Science Conference (ACSC '98), 1998, pp. 1-11.
[14] Hall MA (2000) Correlation-based feature selection for discrete and numeric class machine learning, 17th International Conference on Machine Learning (ICML'00), 2000, pp. 359-366.
[15] Peng H, Long F, Ding C (2005) Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy, IEEE Transactions on Pattern Analysis and Machine Intelligence 27 (8): 1226-1238.
[16] Perez M, Marwala T (2012) Microarray data feature selection using hybrid genetic algorithm simulated annealing, IEEE 27th Convention of Electrical and Electronics Engineers in Israel (IEEEI '12), 2012, pp. 1–5.
[17] Tang EK, Suganthan PN, Yao X (2006) Gene selection algorithms for microarray data based on least squares support vector machine, BMC Bioinformatics **7**(95): 1-16.
[18] Dashtban M, Balafar M (2017) Gene selection for microarray cancer classification using a new evolutionary method employing artificial intelligence concepts, Genomics 109 (2): 91-107.
[19] Lu H, Chen J, Yan K, Jin Q, Xue Y, Gao Z (2017) A hybrid feature selection algorithm for gene expression data classification, Neurocomputing 256: 56-62.
[20] Jonnalagadda S, Srinivasan R (2008) Principal components analysis based methodology to identify differentially expressed genes in time-course microarray data, BMC Bioinformatics 9: 267.
[21] Sevakula RK, Singh V, Verma NK, Kumar C, Cui Y (2019) Transfer learning for molecular cancer classification using deep neural networks, IEEE/ACM Transactions on Computational Biology and Bioinformatics 16 (6): 2089-2100.
[22] Fakoor R, Ladhak F, Nazi A, Huber M (2013) Using deep learning to enhance cancer diagnosis and classification, the 30th International Conference on Machine Learning (ICML 2013), 2013, pp. 1-8.
[23] Liao Q, Ding Y, Jiang ZL, Wang X, Zhang C, Zhang Q (2019) Multi-task deep convolutional neural network for cancer diagnosis. Neurocomputing 348: 66-73.
[24] Liu Z, Wang R, Zhang W, Tang D (2021) An unsupervised feature learning method for enhancing the generalization of cancer diagnosis. 13th International Conference on Machine Learning and Computing, 2021, pp. 252-257.
[25] Sun L, Zhang X, Qian Y, Xu J, Zhang S (2019) Feature selection using neighborhood entropy-based uncertainty measures for gene expression data classification, Information Sciences 502: 18-41.
[26] Almugren N, Alshamlan H (2019) A Survey on hybrid feature selection methods in microarray gene expression data for cancer classification, IEEE Access 7: 78533-78548.
[27] Potharaju SP, Sreedevi M (2019) Distributed feature selection (DFS) strategy for microarray gene expression data to improve the classification performance, Clinical Epidemiology & Global Health 7: 171-176.
[28] Wahid A, Khan DM, Iqbal N, Khan SA, Ali A, Khan M, Khan Z (2020) Feature selection and classification for gene expression data using novel correlation based overlapping score method via Chou's 5-Steps rule, Chemometrics and Intelligent Laboratory Systems 199: 103958.
[29] Uzma, Al-Obeidat F, Tubaishat A, Shah B, Halim Z (2020) Gene encoder: a feature selection technique through unsupervised deep learning-based clustering for large gene expression data, Neural Computing and Applications 2020: 1-23 (published online).
[30] Nikulin V, McLachlan GJ (2009) Penalized principal component analysis of microarray data, in Computational Intelligence Methods for Bioinformatics and Biostatistics 2009: 82-96.
[31] Huynh PH, Nguyen VH, Do TN (2018) A coupling support vector machines with the feature learning of deep convolutional neural networks for classifying microarray gene expression data. In book: Modern Approaches for Intelligent Information and Database Systems, 2018, pp. 233-243.

[32] Danaee P, Ghaeini R, Hendrix DA (2016) A deep learning approach for cancer detection and relevant gene identification, Pacific Symposium on Biocomputing Pacific Symposium on Biocomputing 22: 219-229.
[33] Zhu Z, Ong YS, Dash M (2007) Markov blanket-embedded genetic algorithm for gene selection, Pattern Recognition 49 (11): 3236-3248.
[34] Chin K, DeVries S, Fridlyand J, Spellman PT, Roydasgupta R, Kuo W, Lapuk A, Neve RM, Qian Z, Ryder T, et al. (2006) Genomic and transcriptional aberrations linked to breast cancer pathophysiologies, Cancer Cell 10 (6): 529-41.
[35] Hess KR (2006) Pharmacogenomic predictor of sensitivity to preoperative chemotherapy with paclitaxel and fluorouracil, doxorubicin, and cyclophosphamide in breast cancer. Journal of Clinical Oncology 24 (26): 4236-44.
[36] Telikani A, Gandomi AH (2009) Cost-sensitive stacked auto-encoders for intrusion detection in the Internet of Things, Internet of Things 14: 100122.
[37] Manosij G, Sukdev A, Kanti GK, Aritra S, Shemim B, Ram S (2019) Genetic algorithm based cancerous gene identification from microarray data using ensemble of filter methods, Medical & Biological Engineering & Computing 57: 159-176.
[38] Wenbin Zhang and Eirini Ntoutsi. Faht: an adaptive fairness-aware decision tree classifier. In International Joint Conference on Artificial Intelligence (IJCAI), pages 1480–1486, 2019.
[39] Wenbin Zhang, Xuejiao Tang, and Jianwu Wang. On fairness-aware learning for non-discriminative decision-making. In International Conference on Data Mining Workshops (ICDMW), pages 1072–1079, 2019.
[40] Wenbin Zhang and Albert Bifet. Feat: A fairness-enhancing and concept-adapting decision tree classifier. In International Conference on Discovery Science, pages 175–189. Springer, 2020.
[41] Wenbin Zhang et al. Flexible and adaptive fairness-aware learning in non-stationary data streams. In IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI), pages 399–406, 2020.
[42] Wenbin Zhang and Liang Zhao. Online decision trees with fairness. arXiv preprint arXiv:2010.08146, 2020.
[43] Wenbin Zhang. Learning fairness and graph deep generation in dynamic environments. 2020.
[44] Wenbin Zhang, Albert Bifet, Xiangliang Zhang, Jeremy C Weiss, and Wolfgang Nejdl. Farf: A fair and adaptive random forests classifier. In Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 245–256. Springer, 2021.
[45] Wenbin Zhang and Jeremy Weiss. Fair decision-making under uncertainty. In 2021 IEEE International Conference on Data Mining (ICDM). IEEE, 2021.
[46] Wenbin Zhang and Jeremy C Weiss. Longitudinal fairness with censorship. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, pages 12235–12243, 2022.
[47] Wenbin Zhang, Shimei Pan, Shuigeng Zhou, Toby Walsh, and Jeremy C Weiss. Fairness amidst non-iid graph data: Current achievements and future directions. arXiv preprint arXiv:2202.07170, 2022.
[48] Wenbin Zhang, Tina Hernandez-Boussard, and Jeremy C Weiss. Censored fairness through awareness. In Proceedings of the AAAI Conference on Artificial Intelligence, 2023.
[49] Wenbin Zhang and Jeremy Weiss. Fairness with censorship and group constraints. Knowledge and Information Systems, 2022.
[50] Wenbin Zhang and Jianwu Wang. A hybrid learning framework for imbalanced stream classification. In IEEE International Congress on Big Data (BigData Congress), pages 480–487, 2017.
[51] Wenbin Zhang, Jian Tang, and Nuo Wang. Using the machine learning approach to predict patient survival from high-dimensional survival data. In IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2016.

[52] Wenbin Zhang and Jianwu Wang. Content-bootstrapped collaborative filtering for medical article recommendations. In IEEE International Conference on Bioinformatics and Biomedicine (BIBM), 2018.
[53] Xuejiao Tang, Liuhua Zhang, et al. Using machine learning to automate mammogram images analysis. In IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pages 757–764, 2020.
[54] Liuhua Zhang et al. A comparison of different pattern recognition methods with entropy based feature reduction in early breast cancer classification. European Scientific Journal, 3:303–312, 2014.
[55] Mingli Zhang, Xin Zhao, et al. Deep discriminative learning for autism spectrum disorder classification. In International Conference on Database and Expert Systems Applications, pages 435–443. Springer,2020.
[56] Xuejian Wang, Wenbin Zhang, Aishwarya Jadhav, and Jeremy Weiss. Harmonic-mean cox models: A ruler for equal attention to risk. In Survival Prediction-Algorithms, Challenges and Applications, pages 171–183. PMLR, 2021.
[57] Wenbin Zhang, Jianwu Wang, Daeho Jin, Lazaros Oreopoulos, and Zhibo Zhang. A deterministic self-organizing map approach and its application on satellite data based cloud type classification. In IEEE International Conference on Big Data (Big Data), 2018.
[58] Xuejiao Tang, Xin Huang, et al. Cognitive visual commonsense reasoning using dynamic working memory. In International Conference on Big Data Analytics and Knowledge Discovery. Springer, 2021.
[59] Wenbin Zhang, Liming Zhang, Dieter Pfoser, and Liang Zhao. Disentangled dynamic graph deep generation. In Proceedings of the SIAM International Conference on Data Mining (SDM), pages 738–746, 2021.
[60] Zhen Liu, Ruoyu Wang, Nathalie Japkowicz, Deyu Tang, Wenbin Zhang, and Jie Zhao. Research on unsupervised feature learning for android malware detection based on restricted boltzmann machines. Future Generation Computer Systems, 120:91–108, 2021.

Zhen Liu received the Ph.D. degree from the School of Computer Science and Technology of South China University of Technology in 2013. She is now an associated professor in Guangdong University of Foreign Studies. She worked in Guangdong Pharmaceutical University from Sep. 2014 to July 2021. Her research interests include machine learning and bioinformatics.

Ruoyu Wang received the Ph.D. degree from the school of Computer Science and Engineering, South China University of Technology in 2015. He is now a senior engineer at the Information and Network Engineering and Research Center, South China University of Technology. His research interests include machine learning.

Wenbin Zhang received his Ph.D. at the University of Maryland, Baltimore County, USA. His research investigates the theoretical foundations of machine learning. Other interests include computer systems and deep generative models.