SDAE-based feature selection method for biological Omics data

Zhangqi Jiang*
1The School of Mathematical Sciences, Shanxi University, Taiyuan, Shanxi, China
*Corresponding author’s e-mail: 201900903121@email.sxu.edu.cn, sxdxjzq@gmail.com

Abstract. The advancement of Omics technology has led to a surge in molecular and cell profiling data for mechanism study. Large amount of data and complex data structure pose a great challenge to data analysis. Modern machine learning methods such as deep learning are expected to take advantage such big data for accurate disease prediction or other related tasks. However, large feature number may bring large amount redundant information and adversely affect the accuracy of a classifier. To this end, feature selection methods can remove redundant information and help the model achieve higher accuracy by selecting informative features. In this paper, we propose a two-step deep learning-based method combining stacked denoising autoencoders (SDAE) with SVM-RFE to accomplish the task of feature selection. We compared our method with other related methods and the results showed that our approach achieved a better performance than other methods when using the TCGA datasets.

1. Introduction

Omics technology provides a new perspective on studying the function of molecules constituting a cell, a basic building unit for living organisms [1]. It has a broad range of applications in biology, which primarily aims at a comprehensive detection of genes (genomics), mRNAs (transcriptomics), proteins (proteomics) and metabolites (metabolomics) in a specific biological sample.

It is worthwhile mentioning that a large number of features can be obtained from an Omics experiment, which enables a systemic way of learning the function of a biological organism. Cancer is a systematic disease therefore Omics is a promising technique in the area of cancer research or clinical practice, such as mechanism study, diagnosis, treatment decision making [2]. Machine learning (ML) methods have been widely explored and applied in the task of cancer classification and biomarker discovery [3]. However, due to a large feature number and relatively small sample size, using ML to analyze data has become problematic and challenging. For example, overfitting issue is always a major problem for ML model training where the excessive features make the model memorize training data rather than learning to generalize from a trend. To this end, various feature selection methods are proposed to solve this problem.

Liang et al. developed a Support Vector Machine-Recursive Feature Elimination (SVM-RFE) feature selection method to compute the ranking of the features based on their importance to accuracy by training SVM, and then recursively removed the feature with the lowest ranking [4]. In another study [5], Zhang et al. present a two-stage selection method by combing ReliefF and minimal-redundancy-maximal-relevance (mRMR). In the first stage, ReliefF is used to find a candidate feature set as the algorithm is able to effectively provide quality estimates for attributes. And then, mRMR would be applied to select the features that have the highest relevance with the target class and are also maximally...
dissimilar to each other from the candidate set. Nowadays, deep learning (DL) has been drawing increasing attention and become a mainstream force in ML due to their excellent performance in various tasks [6]. In several previous feature selection pipelines using DL, a specific neural network architecture is used to obtain a high-level representation of the original input data and then the transformed input features are further selected using traditional feature selection methods. Li et al. developed a deep feature selection (DFS) model to select input features for multiclass data where a deep neural network is adopted [7]. They used elastic-net to add a sparse one-to-one linear layer between the input layer and the first hidden layer of an MLP, and select the important features based on the weights of the input layer after training. Chen et al. proposed a flexible neural tree (FNT) model, based on the pre-defined instruction or operator sets, for breast cancer classification [8]. The FNT structure adopts genetic programming and the parameters are optimized by the memetic algorithm. However, there are still many unexplored deep learning structures that can be tried for feature selection.

In this paper, we propose a new deep learning-based feature pipeline. In our method, we apply stacked denoising autoencoders (SDAE) [10] as a deep architecture for feature abstraction at a higher level and classical SVM-RFE to finalize the process of feature selection. We have conducted a comprehensive evaluation for our proposed method by comparing with other setups and our results show that our method achieves best classification accuracy and lower error rate.

The remainder of this paper is organized as follows. Section II describes the proposed method. Section III shows results of our method and compares them to results achieved using other methods. Finally, Section IV concludes the paper.

2. Methods

2.1. Gene Expression Data

In our experiment, we analyzed miRNA-seq expression data from The Cancer Genome Atlas (TCGA) database which molecularly characterized over 20,000 primary cancer and matched normal samples spanning 33 cancer types [11]. We downloaded three datasets of three different cancer types, namely Breast invasive carcinoma (BRCA), Lung adenocarcinoma (LUAD), Stomach adenocarcinoma (STAD). Table 1 shows the size of the datasets. Furthermore, we deleted all genes whose expression was zero in all samples, and we kept the same number of features for all three datasets in order to have a fair comparison.

| Dataset Name | Number of Genes | Tumor Samples | Healthy Samples |
|--------------|-----------------|---------------|-----------------|
| BRCA         | 762             | 87            |
| LUAD         | 656             | 452           | 46              |
| STAD         | 389             | 41            |

2.2. Feature Learning Using Stacked Denoising Autoencoders

The method proposed in this study is a two-step feature selection method with a SDAE to obtain a high-level feature representation for original features and SVM-RFE to further select informative features and reduce feature number. The workflow of our approach is shown in Figure 1.
Figure 1. The Workflow of the Proposed Method

In this pipeline, firstly SDAE is applied to learn the high-level representation of the original data. And then, we use the SVM-RFE algorithm to further screen out the most representative features.

2.2.1. Stacked Denoising Autoencoders

In our experiment, SDAE [10] is applied to learn the feature expressions of the original data set. In order to better understand SDAE, we should first understand how denoising autoencoders (DAE) works [9]. Different from sparse autoencoder or undercomplete autoencoder that constrain the hidden layer to have fewer neurons than the input layer, denoising autoencoder tries to achieve a good representation for features by changing the reconstruction criterion [12]. DAE takes a partially corrupted input and is trained to recover the original undistorted input. The objective of DAE is that of cleaning the corrupted input, or denoising. In other words, denoising is advocated as a training criterion for learning to extract useful features that will constitute better higher-level representations of the input [10].

The structure of a DAE can be divided into two parts, encoding and decoding, as shown in Figure 2. The training process of a DAE works as follows:

**Encoding Process:**
- The original input $x$ is corrupted into $\tilde{x}$ through stochastic mapping $\tilde{x} \sim q_\theta(\tilde{x}|x)$.
- The corrupted input $\tilde{x}$ is then mapped to a hidden representation $h = f_\theta(\tilde{x}) = \sigma(W\tilde{x} + b)$.

**Decoding Process:**
- From the hidden representation the model reconstructs $z = g_{\theta'}(h) = \sigma'(W'h + b')$.

Here, $\sigma$ and $\sigma'$ are element-wise activation function such as the sigmoid function or the rectified linear unit. $W$ ($W'$) is a weight matrix and $b$ ($b'$) is a bias vector. We can define the weight matrix $W$ and bias $b$ as $\theta = \{W, b\}$ and similarly $\theta' = \{W', b'\}$. The model’s parameters $\theta$ and $\theta'$ are trained to minimize the reconstruction error which often refers to the squared error:

$$\mathcal{L}(x, z) = \|x - z\|^2 = \|x - \sigma'(W\sigma(W\tilde{x} + b) + b')\|^2$$

SDAE is constructed as a series of DAE mappings with parameters $\theta_1, \theta_2, ..., \theta_n$. By stacking, the neural network can learn more robust features which contain more information about the original input data. The training algorithm for obtaining parameters of SDAE is based on greedy layer-wise strategy [13].
In our proposed method, we adopt the deep architecture of stacked denoising autoencoders as shown in Figure 3. SDAE can be divided into two parts: encoding and decoding. In the encoding part, a dropout layer is added before each hidden layer to erase some features to increase noise. In this way, the SDAE algorithm can be forced to learn more robust features which are less affected by noise. At the same time, gradually reduce the number of neurons in hidden layers to achieve the effect of dimensionality reduction. After training, the high-level representations are given from middle layer which formed a candidate feature set.

2.2.2. SVM-RFE

In the second step, SVM-RFE method is used to further select the most important features in the new dataset. In the first round of algorithm training, all the features from the candidate set are used to train SVM. As the training is completed, the trained SVM will score the importance of each feature through calculating the sorting coefficient, and then remove the feature with the lowest score from the current set to form a new set [14]. The algorithm repeats the above process until the required number of features is finally reached.
Table 2. Evaluating feature selection methods using SVM and RFs models

| Dataset | DL Methods | SVM Accuracy | SVM F1 | SVM MSE | RFs Accuracy | RFs F1 | RFs MSE |
|---------|------------|--------------|--------|--------|--------------|--------|--------|
| BRCA    | CNN        | 96.58%       | 98.09% | 0.0341 | 96.81%       | 98.43% | 0.0365 |
|         | SAE        | 97.06%       | 98.34% | 0.0293 | 96.70%       | 98.95% | 0.0270 |
|         | SDAE       | 97.92%       | 98.84% | 0.0208 | 97.91%       | 98.91% | 0.0202 |
|         |            | **98.50%**   | **99.16%** | **0.0150** | **98.73%** | **99.29%** | **0.0114** |
| LUAD    | CNN        | 98.99%       | 99.44% | 0.0100 | 97.39%       | 99.45% | 0.0160 |
|         | SAE        | 99.37%       | 99.53% | 0.0262 | 98.37%       | 99.55% | 0.0128 |
|         | SDAE       | 99.25%       | 99.59% | 0.0075 | 98.79%       | 99.33% | 0.0133 |
|         |            | **99.05%**   | **99.48%** | **0.0095** | **98.61%** | **99.29%** | **0.0117** |
| STAD    | CNN        | 96.28%       | 97.93% | 0.0371 | 96.02%       | 97.32% | 0.0489 |
|         | SAE        | 97.90%       | 98.84% | 0.0209 | 96.98%       | 97.94% | 0.0348 |
|         | SDAE       | 98.33%       | 99.07% | 0.0167 | 96.03%       | 97.75% | 0.0372 |
|         |            | **98.75%**   | **99.39%** | **0.0109** | **97.51%** | **98.27%** | **0.0288** |

2.3. Evaluation

In order to evaluate the proposed feature selection pipeline, SVM and Random Forest (RF) were used as classifiers trained with selected features by different feature selection methods. And then the 10-fold cross validation were used for reporting final model performance, which can derive a more accurate estimate of classifier prediction accuracy by averaging measures of fitness in prediction [15]. In our evaluation, MSE, F1 and Accuracy were used to compare our proposed method with other related setups.

3. Results

In order to demonstrate the generality and applicability of the proposed method, we obtained 3 different datasets, BRCA, LUAD and STAD, from TCGA. We removed all features with 0 values from the original dataset and finally 656 features were kept in all three data sets. After feature representation learning using SDAE, the dimension of feature vectors is reduced to 100. In the next step, SVM-RFE, 20 features are output as final selected features. We used SVM and RF as the classifier and accuracy, F1 score and Mean Square Error (MSE) are used as metrics to evaluate our proposed method and compare with other related methods.

As shown in Table 2, we compared our proposed method with a series of other related setups. We also included a pipeline without deep learning method. The results show that the final classification performance of pipelines using deep learning methods are significantly better than the one without deep learning methods. For example, in the BRCA dataset, the final classification accuracy of pipelines using deep learning methods is about 98%, while the accuracy of pipelines without deep learning methods is only about 96.5%. In the comparison between deep learning methods, we found that the performance of the autoencoder as the deep architecture methods (SAE, SDAE) achieves better performance compared to the convolutional neural network (CNN) as the deep architecture method. In the three data sets used in the experiment, the final classification accuracy of the SAE or SDAE method is about 1% to 2% higher than that of the CNN method. In the autoencoder-based methods, the results show that SDAE performs slightly better than SAE. In the two datasets of BRCA and STAD, the SDAE method is even 0.5% higher than the SAE method on the basis of high accuracy.

4. Conclusion

In this paper, we proposed a method to improve cancer classification from miRNA-seq expression data using deep learning method and traditional feature selection method. The proposed approach uses SDAE
to address the high dimensionality of the initial feature space followed by SVM-RFE to further filter out the most important features for the final classification step. We applied our method in three different datasets from TCGA. The results not only prove that using deep learning methods will improve the final classification accuracy to a certain extent, but also show that our method is slightly better than the approaches that use CNN and SAE as the deep architecture.

References

[1] Karahalil, B. (2016). Overview of systems biology and omics technologies. Current medicinal chemistry, 23(37), 4221-4230.

[2] Chakraborty, S., Hosen, M., Ahmed, M., & Shekhar, H. U. (2018). Onco-multi-OMICS approach: a new frontier in cancer research. BioMed research international, 2018.

[3] Tan, A. C., & Gilbert, D. (2003). Ensemble machine learning on gene expression data for cancer classification.

[4] Liang, Y., Zhang, F., Wang, J., Joshi, T., Wang, Y., & Xu, D. (2011). Prediction of drought-resistant genes in Arabidopsis thaliana using SVM-RFE. PLoS one, 6(7), e21750.

[5] Zhang, Y., Ding, C., & Li, T. (2008). Gene selection algorithm by combining reliefF and mRMR. BMC genomics, 9(S2), S27.

[6] LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. nature, 521(7553), 436-444.

[7] Li, Y., Chen, C. Y., & Wasserman, W. W. (2016). Deep feature selection: theory and application to identify enhancers and promoters. Journal of Computational Biology, 23(5), 322-336.

[8] Chen, Y., Abraham, A., & Yang, B. (2006). Feature selection and classification using flexible neural tree. Neurocomputing, 70(1-3), 305-313.

[9] Vincent, P., Larochelle, H., Bengio, Y., & Manzagol, P. A. (2008, July). Extracting and composing robust features with denoising autoencoders. In Proceedings of the 25th international conference on Machine learning (pp. 1096-1103).

[10] Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., Manzagol, P. A., & Bottou, L. (2010). Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. Journal of machine learning research, 11(12).

[11] Wang, Z., Jensen, M. A., & Zenklusen, J. C. (2016). A practical guide to the cancer genome atlas (TCGA). In Statistical Genomics (pp. 111-141). Humana Press, New York, NY.

[12] Goodfellow, I., Bengio, Y., Courville, A., & Bengio, Y. (2016). Deep learning (Vol. 1, No. 2). Cambridge: MIT press.

[13] Bengio, Y., Lamblin, P., Popovici, D., & Larochelle, H. (2006). Greedy layer-wise training of deep networks. Advances in neural information processing systems, 19, 153-160.

[14] Weston, J., & Guyon, I. (2012). U.S. Patent No. 8,095,483. Washington, DC: U.S. Patent and Trademark Office.

[15] Rodriguez, J. D., Perez, A., & Lozano, J. A. (2009). Sensitivity analysis of k-fold cross validation in prediction error estimation. IEEE transactions on pattern analysis and machine intelligence, 32(3), 569-575.