BIOS 823 Final Project: Colorectal Cancer Histology Images Classification

Tengjie (TJ) Tang[1]

[1]Duke University Department of Statistical Science
Durham, North Carolina 27705 USA
tengjie.tang@duke.edu

Abstract

In this project, I implemented and compared the support vector machine, random forest, and three convolutional neural network architectures to classify textures in colorectal cancer histology. The results demonstrate that convolutional neural networks outperform the other two methods with higher classification accuracy for the image classification task.

Introduction

Image classification is important for biomedical science studies. Medical imaging, including X-ray imaging, CT (Computed Tomography) Scanner, PET (Positron Emission Tomography), and MRI (Magnetic Resonance Imaging), is widely used in medical science in the modern world to aid internal body structures assessment and disease diagnosis and treatment (Bajaj, Gupta, and Hasija 2018; Zhang and Sejdić 2019; Tchapga et al. 2021; Hussain et al. 2022). Precise medical imaging classification can help medical people to differentiate between normal and abnormal individuals and provide better treatments (Kang et al. 2021).

In the project, I explored several supervised learning methods, including two machine learning methods, support vector machine and random forest, and three deep learning convolution neural network architectures, and compared their performance on a biomedical image classification task.

Data set

The data set for this project is `colorectal_histology` (Kather et al. 2016) data set from the `TensorFlow` (Abadi et al. 2015) library. The data set has 5,000 histological images of human colorectal cancer. Each image is 150 x 150 x 3 RGB of one of eight tissue categories. The eight classes are tumor, stroma, complex, lympho, debris, mucosa, adipose, and empty. Figure 1 shows examples of images for each class with corresponding labels and class memberships.

Images are scaled before applying the proposed methods so that pixels range between 0 and 1. Scaling can also help the learning process be much faster.

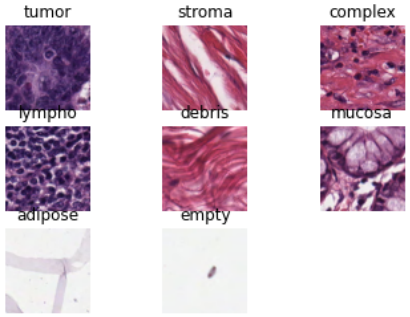

Figure 1: Sample images and corresponding labels in the data set

Methods

Support Vector Machine

A support vector machine (SVM) (Boser, Guyon, and Vapnik 1992) is a supervised machine learning method for classification and regression analysis that maximizes the margin between the training data points and the hyperplane. The hyperplane is $p-1$ dimensional decision boundary for the linear SVM, where p is the dimension of the feature space.

The SVM is a linear classifier in its base form, but it can also incorporate kernels to allow non-linear decision boundaries. The radial basis function (RBF) kernel is a commonly used SVM kernel. However, the SVM is defined only for binary classifications. Therefore, we need to consider multiclass SVM as an extension for our data analysis. In `scikit-learn` (Pedregosa et al. 2011), the multiclass SVM is handled according to a one-versus-one approach. Specifically, $K(K-1)/2$ pairwise binary classifiers are fit, where K is the total number of classes, and the class for a given data point (image) is the class that wins the most pairwise comparisons.

The SVM is appealing because it is a robust classifier and allows non-linear kernel and regularization. It has also shown good performance on image classification tasks (Miranda, Aryuni, and Irwansyah 2016).

Algorithm	Hyperparameters	Range	Best	Description
RF	max_depth	2; 5; 10; 100	100	Maximum depth of the tree
RF	n_estimators	100; 1000	1000	Number of trees
SVM	C	0.1; 1; 10	10	Regularization parameter
SVM	kernel	linear; rbf	rbf	Kernel type

Table 1: Hyperparameters tuning for random forest and support vector machine and hyperparameters for the final models

Random Forest

Random forests (RF) (Breiman 2001) is another popular machine learning method, which can also be used for classification and regression analysis. RF is an ensemble learning method that builds upon decision trees. And a decision tree is a tree-based machine learning approach built by the Classification and Regression Trees (CART) algorithm (Leo et al. 1984). CART selects the predictors for splitting and thresholds with the objective of minimizing misclassification errors at each split. CART is a greedy algorithm and gives the tree classifier high variance. RF controls the variance by building multiple trees on bootstrapped sub-datasets and averaging them. It also selects only a subset of predictors as the splitting candidates at each split. When making predictions, RF passes the new data point (image) to all trees, and the class that wins the most votes would be the class membership for that image.

Convolutional Neural Network

A convolutional neural network (CNN) is a deep learning method designed to extract relevant features directly from pixel images and is shown to have promising performance for image pattern recognition over other methods (Lecun et al. 1998; Valueva et al. 2020).

CNN architectures consist of the image input, convolutional (Conv) layers, max-pooling layers, fully connected layers (dense layers), and the softmax layer. Here, I propose three CNN architectures with different numbers of Conv layers and dense layers:

- 1 Conv layer + 1 dense layer
- 3 Conv layers + 2 dense layers
- 5 Conv layers + 3 dense layers.

I trained the model in 500 epochs for all three CNN architectures, and early stopping was used. For this classification task, I used ReLU as the activation function, the sparse categorical crossentropy as the loss function, and the Adam algorithm as the optimizer with a learning rate of 0.0001.

Results

Machine learning and deep learning methods can suffer from overfitting issues. In particular, a model can perform exceptionally well on a training set but poorly on external data sets. I performed hyperparameter tuning for RF and SVM to mitigate model overfitting. Specifically, I shuffled the whole data set and split it into training (60%), validation (20%), and test (20%) sets and then considered grid search to find the optimal combinations of hyperparameters that produced

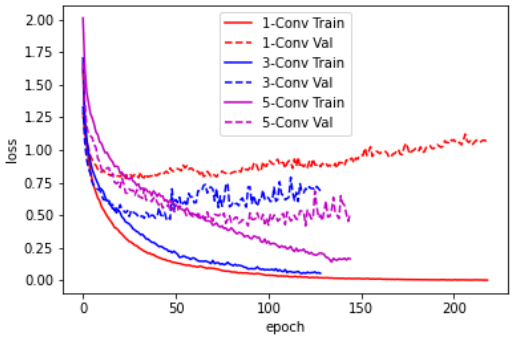

Figure 2: Loss curves for the three convolutional neural network architectures

Model	Test Accuracy
1-Conv CNN	0.789
3-Conv CNN	0.866
5-Conv CNN	0.88
RF	0.822
SVM	0.803

Table 2: Test accuracy for all the models

the best validation accuracy for RF and SVM. The validation set was used during the CNN training process to monitor the model's sparse categorical crossentropy validation loss at each epoch. Additionally, I added "dropout" to each CNN, a regularization technique that randomly ignores each hidden unit with a probability of 0.5, to avoid overfitting (Hinton et al. 2012).

Table 1 shows the hyperparameter tuning and hyperparameters for the final model. For RF, the number of trees in the forest is 1,000, and the maximum depth of the tree is 100. The RBF kernel is used for the SVM, and the regularization parameter is 10. Figure 2 shows the training and validation loss curves for all three CNN architectures. We can see that the 5 Conv layers CNN has the lowest validation loss at the end. Even though there are still overfitting issues, it is acceptable to use the 5 Conv layers CNN architecture for classifications.

The evaluation is performed on the test set. Table 2 reports the classification accuracy across all five proposed methods. From this, we can conclude that the 3 Conv layers and 5 Conv layers CNN has the best accuracy on the test set, and the 5 Conv layers CNN is slightly much better. The 1 Conv layer CNN has the worst performance and is worse than RF the SVM.

Figure 3 is the confusion matrix, which shows the classification performance for the best 5 Conv layers CNN model. It illustrates that the classification is very accurate across most of the classes. We can find that the model may misclassify stroma, debris, and complex tissue types. From Figure 1, one of the reasons is probably because these tissues have similar colors and patterns.

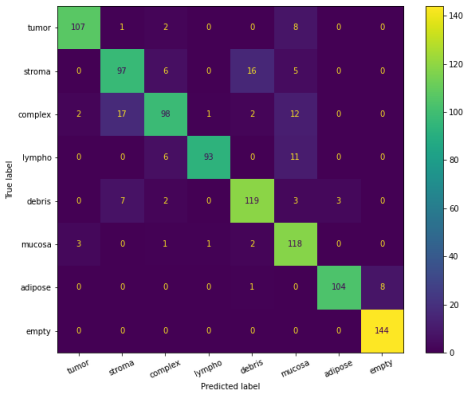

Figure 3: Confusion matrix for the best five convolutional layers convolutional neural network

Discussion and Conclusion

In conclusion, among all CNN architectures, the 3 Conv layers and 5 Conv layers model has the best classification accuracy on the test sets. These two deep learning CNN architectures perform better than the two machine learning methods (SVM and RF) in this image classification task. The 1 Conv layer CNN has the worst performance, which is because the model is too simple to learn the data set well. Due to the limited computing power, I did not perform an exhaustive grid search and cross-validation parameters tuning for SVM and RF, and I did not perform fine-tuning for the three CNN architectures. Future studies could explore much deeper and different CNN architectures. We may also consider other useful techniques, including data augmentation and autoencoder.

References

Abadi, M.; Agarwal, A.; Barham, P.; Brevdo, E.; Chen, Z.; Citro, C.; Corrado, G. S.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Goodfellow, I.; Harp, A.; Irving, G.; Isard, M.; Jia, Y.; Jozefowicz, R.; Kaiser, L.; Kudlur, M.; Levenberg, J.; Mané, D.; Monga, R.; Moore, S.; Murray, D.; Olah, C.; Schuster, M.; Shlens, J.; Steiner, B.; Sutskever, I.; Talwar, K.; Tucker, P.; Vanhoucke, V.; Vasudevan, V.; Viégas, F.; Vinyals, O.; Warden, P.; Wattenberg, M.; Wicke, M.; Yu, Y.; and Zheng, X. 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

Bajaj, L.; Gupta, K.; and Hasija, Y. 2018. *Image Processing in Biomedical Science*, 185–211. Cham: Springer International Publishing. ISBN 978-3-319-63754-9.

Boser, B. E.; Guyon, I. M.; and Vapnik, V. N. 1992. A training algorithm for optimal margin classifiers. In *Proceedings of the fifth annual workshop on Computational learning theory*, 144–152.

Breiman, L. 2001. Random Forests. *Machine Learning*, 45(1): 5–32.

Hinton, G. E.; Srivastava, N.; Krizhevsky, A.; Sutskever, I.; and Salakhutdinov, R. R. 2012. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*.

Hussain, S.; Mubeen, I.; Ullah, N.; Shah, S. S. U. D.; Khan, B. A.; Zahoor, M.; Ullah, R.; Khan, F. A.; and Sultan, M. A. 2022. Modern Diagnostic Imaging Technique Applications and Risk Factors in the Medical Field: A Review. *BioMed Research International*, 2022.

Kang, D.; Kim, S.; Jung, Y.; and Ryoo, H. S. 2021. Generating Interpretable Patterns for Biomedical Image Classification. In *2021 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, 1658–1660.

Kather, J. N.; Weis, C.-A.; Bianconi, F.; Melchers, S. M.; Schad, L. R.; Gaiser, T.; Marx, A.; and Z"ollner, F. G. 2016. Multi-class texture analysis in colorectal cancer histology. *Scientific reports*, 6: 27988.

Lecun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Leo, B.; Friedman, J. H.; Olshen, R. A.; and Stone, C. J. 1984. Classification and regression trees. *Wadsworth International Group*, 8: 452–456.

Miranda, E.; Aryuni, M.; and Irwansyah, E. 2016. A survey of medical image classification techniques. In *2016 International Conference on Information Management and Technology (ICIMTech)*, 56–61.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12: 2825–2830.

Tchapga, C. T.; Mih, T. A.; Kouanou, A. T.; Fonzin, T. F.; Fogang, P. K.; Mezatio, B. A.; and Tchiotsop, D. 2021. Biomedical Image Classification in a Big Data Architecture Using Machine Learning Algorithms. *Journal of Healthcare Engineering*, 2021.

Valueva, M.; Nagornov, N.; Lyakhov, P.; Valuev, G.; and Chervyakov, N. 2020. Application of the residue number system to reduce hardware costs of the convolutional neural network implementation. *Mathematics and Computers in Simulation*, 177: 232–243.

Zhang, Z.; and Sejdić, E. 2019. Radiological images and machine learning: Trends, perspectives, and prospects. *Computers in Biology and Medicine*, 108: 354–370.