The Analysis of Cancer Domain and Mnist Data Prediction based on CNN Network

Shuchen Ye

Content

Learning and understanding of tumor purity estimation

Tumor purity refers to the proportion of tumor cells in tumor tissue. (LI, XU, & ZHA, 2019)Tumor content is one of the key factors affecting the quality of genomic analysis, and tumor content is also quantified as tumor purity. Studies have shown that tumor purity is significantly correlated with tumor patients' clinical characteristics, genomic expression, and biological characteristics. The samples need to have sufficient tumor content to detect genetic variation in tumor samples by next-generation sequencing. Therefore, accurate tumor purity estimation has important clinical implications.

From the paper (Oner, Chen, James, Revkov, & Egor, 2021), I learned that tumor purity is estimated by two main methods: tumor cell nuclei percentage estimation and genomic tumor purity inference. Tumor nuclei percentage estimation has broad applicability and cellular-level resolution. However, counting tumor cell nuclei is tedious and time-consuming. More importantly, there was interobserver variability between the pathologists' estimates. If tumor purity is inferred from different types of genomic data, we can investigate prior associations between tumor purity and clinical variables, but it also has major drawbacks, such as the inability of this method to work with low tumor content samples and the lack of cancer Spatial information of cell location.

The paper design a novel MIL model to predict tumor purity, which learns discriminative features of cancer versus normal histology from sample-level genomic tumor purity signatures without requiring exhaustive annotation by pathologists. Upon training, the MIL model was able to successfully predict tumor purity in histopathological sections in different TCGA cohorts.

Data processing

According to the subject requirements, in the Mnist data, we only need the handwritten number 0 and the handwritten number 7 as our analysis and training objects. Therefore, we need to extract the specified data from the total database before training the neural network.

The second command box in the Q1.ipynb file is the code for extracting data. We extract the required images and merge them into a folder named 'train_data' and 'test_data'. In addition, we also renamed the data required by this document and named its label to the file name, which is convenient for our subsequent neural network to read.

The construction of the neural network

The learning algorithm of the MIL model, according to the multi-instance learning overview, there are mainly 5 algorithms, Diverse Density, Citation-kNN, ID3-MI, RIPPER-MI, BP-MIP, which essentially correspond to Bayesian classifiers, respectively. KNN, Decision Trees, Rule Induction Algorithms and Neural Networks

Like the MIL model we mentioned earlier, we infer that such a neural network should be a CNN, and try to use a CNN neural network to train on the data.

The third instruction box in Q1.ipynb is the establishment of the model. Dropout() performs L2 regularization, where the learner uses Adam(), and the data processing process also uses MaxPooling2D().

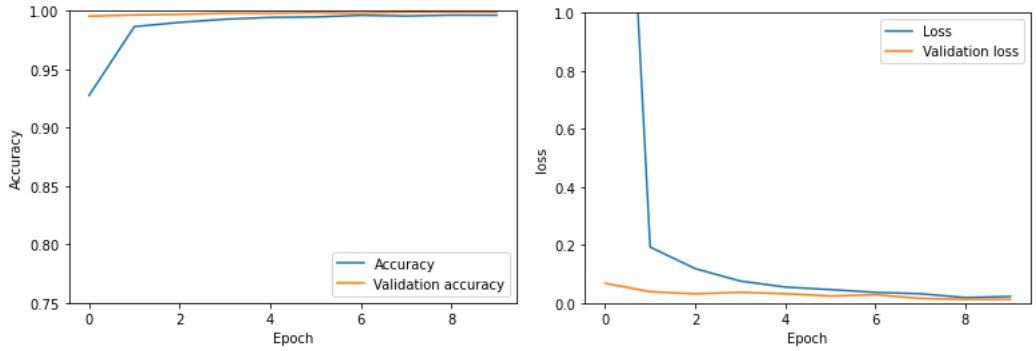
After the model is established, we need to train the model, where batch_size is set to 100 and epochs is set to 10. Then we evaluate the model.

```
Test Socre: 0.012550437822937965
Test Accuracy: 0.9990040063858032
```

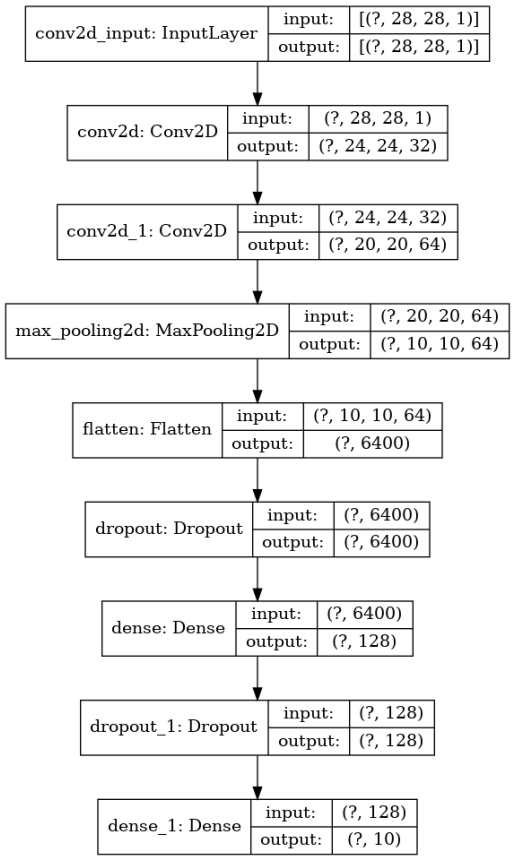
We can see that our test has an accuracy of 0.999, which is a high number. Since the learning rate and other parameters of the model are set according to the specification of the paper, the parameters are no longer adjusted here.

Results and Model Visualization

The visualization of the test results is as follows:



The visualization of our neural network is shown below:



Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 24, 24, 32)	832
conv2d_1 (Conv2D)	(None, 20, 20, 64)	51264
max_pooling2d (MaxPooling2D)	(None, 10, 10, 64)	0
flatten (Flatten)	(None, 6400)	0
dropout (Dropout)	(None, 6400)	0
dense (Dense)	(None, 128)	819328
dropout_1 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 10)	1290

Total params: 872,714
Trainable params: 872,714
Non-trainable params: 0

For the specific code, please see Q1.ipynb.

Reference

LILan, XUBin, & ZHALi. (2019). Tumor Purity: A Potential Ignored Confounder in Tumor Research[J]. Cancer Research on Prevention and Treatment, Page 46(6): 570-574.

OnerUmitMustafa, ChenJianbin, JamesAnne, RevkovEgor, & EgorNeslihanArife. (2021). Obtaining Spatially Resolved Tumor Purity Maps Using Deep Multiple Instance Learning In A Pan-cancer Study. bioRxiv.