CLASSIFYING DRY SCLEROPHYLL FOREST: SUPPORT VECTOR MACHINE & DIFFERENT NEURAL NETWORK METHODS

Weihong Wang

Research School of Computer Science, Australian National University,
Canberra Australia

u6920831@anu.edu.au

Abstract. Detailed maps derived from geographical data are becoming increasingly desirable for use in forest management, in the previous work, the researchers proposed that the classification results of certain targets can be used as additional information for mapping by statistical analysis of satellite data, and some progresses have been achieved. This paper uses several popular machine learning and neural network models to extend the previous work. And these methods have reached a very high standard when performing some specific classification tasks.

Keywords: GIS · Machine Learning· Support Vector Machine· Deep Neural Network· Convolutional Neural Network· Recurrent Neural Network

1 Introduction

Collecting map information in a large area of land is very expensive, and the target area will change in a short period of time. Therefore, researchers rarely use this method to draw maps. Therefore, our goal is to analyze and classify the map in the GIS information, and distinguish whether the terrain belongs to the dry sclerophyll forest from the remote sensing image.

We will use the numerical characteristics of the samples in the dataset to process this information effectively, and then use different methods to build models and execute

classification tasks, compare different model methods from the results of the classification tasks to obtain the best performance. And analyze the reasons of why it achieves the best from the method principle and model structure.
From the subsequent results, the machine learning and neural network methods we used in this paper have achieved good performance (above 90% accuracy) on this data set, which is enough to reflect that we can use satellite data information sampling method more in the real-world process to carry out similar classification tasks, so as to achieve more accurate results while saving resources.

2 Dataset Preparation

2.1 Dataset Introduction

The geographical data which is from an area in the Nullica State Forest on the south coast of New South Wales. The area, approximately 20 by 10 km, is broken up into a grid of 179831 pixels, 30 by 30 m in size. The data has been collected from satellite imagery, soil maps and aerial photographs. From the aerial photographs it is possible to derive a terrain model and from this derive a number of terrain features. In this case each pixel has a value for altitude, aspect, slope, geology, topographic position, rainfall, temperature, and Landsat TM bands 1 to 7. For the purpose of training 190 detailed sample plots have been surveyed. This data gives us classifications for 190 of the pixels in the field area. For use in testing, we have 70 pixels that have been surveyed in less detail, for which we know the classes [1].

The constructor of the data set mainly hopes to use this batch of data to classify dry sclerophyll forest areas and non-dry sclerophyll forest areas. Therefore, of the 190 samples, 99 samples are classified as dry hard-leaf forest areas.

2.2 Data Preprocessing

The data needs to be preprocessed to facilitate the standardized input of each method. First, the data is removed from the header, serial number and other useless parts, so as to extract useful input data, and then convert the tags in the data set to One-Hot encoding to facilitate deep neural network training and optimization. Since there is only this batch of data, the training set, validation set and test set of the following methods can only use this batch of data.

3 Different Classification Methods and Results

Considering that this type of problem is a typical data classification problem, you can use Support Vector Machine (SVM) in machine learning methods. Deep Neural

Network (DNN) s, and convolutional neural networks (CNN), Recurrent Neural Network (RNN) to identify and classify data in deep learning method

3.1 Support Vector Machine

Principle and Architecture. Support vector machines are based on the principle of structural risk minimization and show many advantages in solving small sample, nonlinear and high-dimensional feature space pattern recognition problems [2]. The theoretical basis is to assume that there is a non-linear mapping $\emptyset: R^n \rightarrow H$.The non-linear mapping can output the data samples in the low-dimensional original input to a specific higher-dimensional space *H*. Then, according to the nature of the optimal classification surface algorithm, even if only the inner product between the vectors is performed, the required optimal classification surface can be found in the specific higher-dimensional space *H*, and there is no separate variable $\emptyset(x_i)$ appear. Therefore, if we can find an inner product operation function *K* such that:

$$K(x_i, x_j) = (\emptyset(x_i)^T \emptyset(x_i)) \quad . \tag{1}$$

Then in the high-dimensional space *H*, only the inner product operation of the vector is needed. This inner product operation function *K* is called the kernel function of SVM [3]. Typical kernel functions include linear kernel function, polynomial kernel function, radial basis kernel function and sigmoid kernel function.

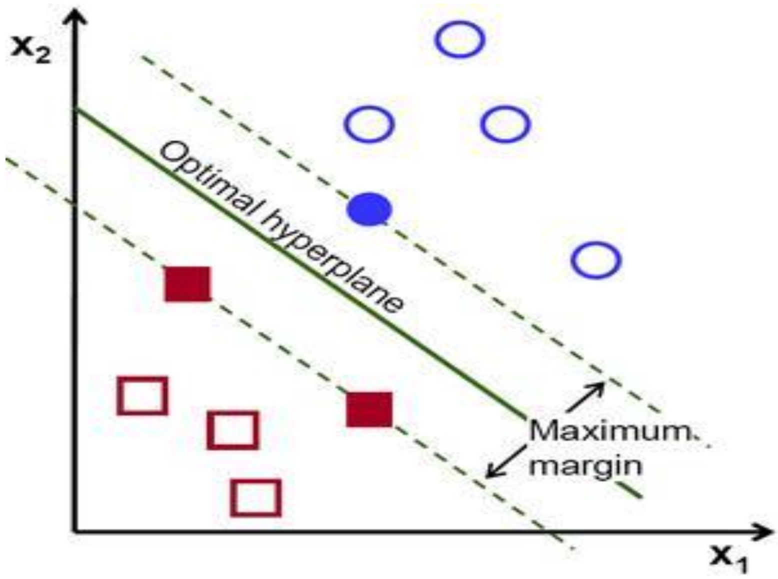

Fig.1 The SVM algorithm finds the optimal segmentation

Implementation. This paper calls the "SVC" function in the "SVM" package in the "sklearn" library to implement the support vector machine for classification tasks. The same batch of data sets are used to train and test the support vector machine, and the influence of different kernel functions on the classification results is also investigated. The correct number and correct rate of classification are shown in the following table:

Table.1 Classification Results with different Kernel Function SVM

Kernel Function	Correct No.	Accuracy
Sigmoid	96	50.53%
Linear	137	72.11%
Polynomial	190	100%
Radial Bias	190	100%

Performance. It can be seen from the above table that the classification performance of the polynomial kernel function and the radial basis kernel function is the best, followed by the linear kernel function, and the classification performance of the Sigmoid kernel function is the worst. Due to the small amount of sample data, the support vector machine method is particularly good at solving classification problems under small samples, nonlinear and high-dimensional features, and the same batch of data sets used for training and testing, so based on polynomial and radial basis kernel function classification The correct rate can reach 100%

3.2 Deep Neural Network

Principle and Architecture. A deep neural network refers to a feedforward neural network (also called a multi-layer perceptron) that contains multiple hidden layers, and the nodes of two adjacent layers are fully connected. Generally, an unsupervised pre-training method is used to initialize the network weights, and a classifier (such as logistic regression, support vector machine or SoftMax layer, etc.) is built between the last hidden layer and the output layer, and finally the supervised training method is used to the weight of the entire network is adjusted. Figure.2 shows a deep neural network with 3 hidden layers.

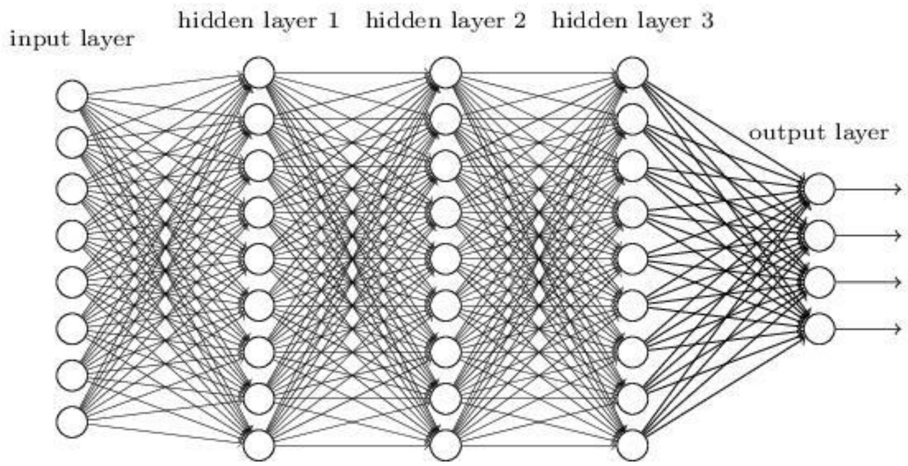

Fig.2 A deep neural network with 3 hidden layers

Implementation. For a DNN, the main parameters that determine the network structure include the number of hidden layers, the number of neurons in the input layer, the hidden layer and the output layer, and the activation function of each neuron. The network structure of reference [1], the DNN used in this article contains 3 hidden layers, the number of neurons are 16, 14, and 2, respectively, and the last hidden layer uses the "SoftMax" function for data classification.

After hyperparameter adjustment in the network training, the following optimized hyperparameters are obtained: the learning rate is set to 0.003, the optimization algorithm uses the adaptive momentum estimation method [4] (Adaptive Moment Estimation, Adam), the number of iterations is 1000 epochs, and the loss function is the cross-entropy loss function. At the same time, this paper compares the classification results and network parameters when using different activation functions in the hidden layer, as shown in Table.2, which includes the classical Sigmoid, Tanh, and ReLU , as well as parametric rectified linear unit (PReLu) [5]. In the code implementation of various neural networks, this paper calls the necessary modules and functions in libraries such as "Tensorflow", "Keras", and "Numpy"

Table.2 Different activation function methods and results

ActivationFunction	Correct No.	Accuracy	Parameter No.
Sigmoid	160	84.21%	540
Tanh	156	82.11%	540
ReLU	173	91.05%	540
PReLU	174	91.58%	570

Performance. From the table we find that the classification accuracy using the Tanh activation function is the lowest, the Sigmoid function is slightly better, and the performance of the ReLU and PReLU functions is the best and there is little difference between the two. However, from the perspective of the amount of parameters, the Sigmoid, Tanh, and ReLU functions did not increase the amount of network parameters, while the PReLU function added some network parameters in order to obtain better performance, although increasing is slightly.

3.3 Conventional Neural Network

Principle and Architecture. Convolutional neural network is a special neural network structure. The nodes of two adjacent layers are not fully connected. They are generally composed of alternate cascades of convolutional layers and pooling layers. The convolutional layer generally includes multiple convolution kernels, and multiple feature sub-maps are obtained by convolution operation with the input of the previous layer to achieve feature extraction of data. The pooling layer implements feature dimensionality reduction processing by down-sampling the input data of the convolutional layer. Because the convolutional neural network uses perceptual field, weight sharing and pooling technology, so that the extracted features have the advantages of invariance, etc., it is widely used in computer vision and image recognition fields [6]. Figure.3 shows the classic LeNet-5 network structure [7]

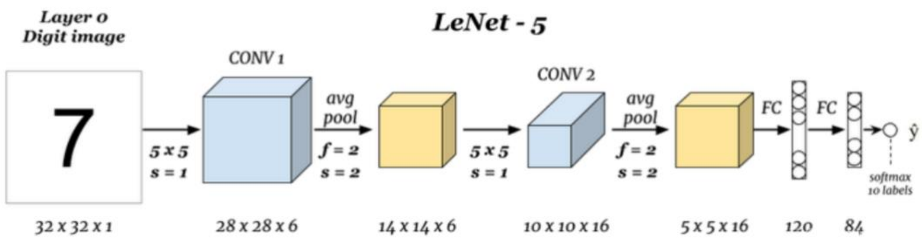

Fig.3 CNN LeNet-5 Network Structure

Implementation. With reference to the classical CNN structure and considering that the dataset is small, this paper uses a shallow CNN with only 2 convolutional layers and 1 fully connected layer. The first convolution layer uses 128 1×3 convolution kernels, the second convolution layer uses 32 1×3 convolution kernels, the fully connected layer contains 2 neurons, and the "SoftMax" function is also used in data classification. The learning rate is set to 0.002, the optimization algorithm still uses Adam, the number of iterations is 500 epoch, and the loss function is the cross-entropy loss function.

Table.3 CNN Classification Results of different Activation Functions

ActivationFunction	Correct No.	Accuracy	Parameter No.
Sigmoid	177	93.16%	14370

Tanh	174	91.58%	14370
ReLU	175	92.11%	14370
PReLU	173	91.05%	16546

Performance. From the table we can find the accuracy of classification using different activation functions is actually not much different. In the above CNN framework, for the dataset in given paper, the activation function has little influence. Theoretically, it is more appropriate to use the Sigmoid function to achieve binary classifications, so the classification accuracy rate is also higher in experiment result.

3.4 Recurrent Neural Network

Principle and Architecture. Different from feedforward network models such as DNN and CNN, recurrent neural networks generally have a loop network structure and short-term memory capabilities. Each network node in RNN can accept input information from itself and other nodes. The output information. The training and optimization of RNN can also be achieved by back propagation algorithm [8] through time series, but if the input timing information is very long, RNN training is still prone to encounter gradient explosion and gradient disappearance problems [9]. In response to these problems, studies have found that the introduction of a gating mechanism is the most effective improvement method, and the classic Long Short-Term Memory (LSTM) network [10] has been created from this. Figure.4 shows the structure diagram of an LSTM unit.

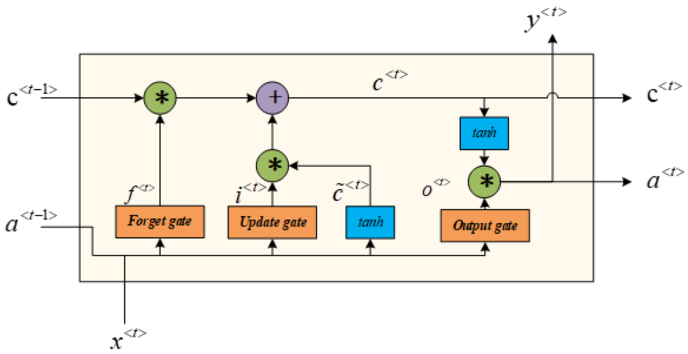

Fig.4 LSTM Structure

Implementation. In training codes, the learning rate is set to 0.001, the optimization algorithm still uses Adam, the number of iterations is 2000 epoch, and the loss function is the cross-entropy loss function. Considering that the activation functions in the LSTM network are all given, this paper focuses on the impact of using different unit numbers of the LSTM network on the classification results.

Table.4 Performance with Different Unit Numbers LSTM

Unit No.	Correct No,	Accuracy	Parameter No.
16	163	85.79%	2146
32	175	92.11%	6338
64	173	91.05%	20866

Performance. From the table above we can infer, when the network contains 16 LSTM units, the correct number of classifications is 163, and the correct rate is 85.79%; when the number of units increases to 32, the correct rate rises significantly, reaching 92.11%; and when it continues to increase to 64 LSTM units, The classification performance of the network has fallen into a bottleneck, which is not much different from the classification performance of 32 units. Considering the classification accuracy and the amount of parameters comprehensively, for the data set in this paper, using 32 LSTM units can get the best classification performance.

4 Limitation and Future Work

The function of deep neural network may be more effective and powerful in processing large-scale sample data ets. Therefore, if large-scale sample data sets are obtained, we can use aboce methods for comparison and analysis to summarize the different application scenarios of these methods. .

We used a small number of dataset samples, through several methods, judging from the performance of the results, we still achieved good results. In the future next step, we plan to use the pre-train model and Fine-Tune approach to adjust the network to obtain higher accuracy.

5 Discussion and Conclusion

After analyzing the parameters of each method, this paper compares the classification performance between different methods, as shown in Table.5, where the classification performance of each method is selected with the best parameters.

The comparison results show that for the given dataset, the SVM method has the best classification performance and can achieve 100% classification accuracy; the CNN method is second, which can achieve a classification accuracy of 93.16%; the classification accuracy of DNN and RNN is relatively high.

Considering that the number of samples in the dataset used in this paper is small, and the same dataset is used to train and test each method, theoretically, under the condition of a small number of samples, the performance of traditional machine learning methods can match or even surpass neural networks. The performance of the SVM method in this article has the best performance. Furthermore, the performance of the neural network largely depends on its network architecture. If the number of samples is large and the architecture is optimized, the performance of the neural network will be better.

Table.5 Different Methods with Best Parameters Performances

	Correct No.	Accuracy
SVM	190	100%
DNN	174	91.58%
CNN	177	93.16%
RNN	175	92.11%

In our real-world work, we can make more use of satellite data for similar classification tasks, so as to achieve the goal of saving costs while achieving higher accuracy.

6 Acknowledge

There is no extended dataset available for I used in the first attempt, so this paper focus more on the technic and approaches.

References

1. Milne L K, Gedeon T D, Skidmore A K. Classifying Dry Sclerophyll Forest from Augmented Satellite Data: Comparing Neural Network, Decision Tree & Maximum Likelihood[J]. Training, 1995.
2. Y. Liu, X. Yan, C. Zhang, W. Liu. An Ensemble Convolutional Neural Networks for Bearing Fault Diagnosis Using Multi-Sensor Data[J]. Sensors, 2019, 19(23): 5300
3.Zhou Zhihua. Machine learning [M]. Beijing: Tsinghua University Press, 2016, 181-185
4. K. D, Ba. J. Adam: A method for stochastic optimization[C]. Proceedings of International Conference on Learning Representations, San Diego, USA, 2015
5. K. He, X. Zhang, S. Ren, J. Sun. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[C]. 2015 IEEE International Conference on Computer Vision (ICCV), Santiago, USA, 2015, 1026-1034
6. Qiu Xipeng. Neural network and deep learning [M]. Beijing: Machinery Industry Press, 2020
7. Lecun Y, Bottou L. Gradient-based learning applied to document recognition[J]. Proceedings of the IEEE, 1998, 86(11): 2278-2324.
8. P. Werbos. Backpropagation through time: what it does and how to do it[J]. Proceedings of the IEEE, 1990, 78(10): 1550-1560
9. Y. Bengio, P. Simard, P. Frasconi. Learning long-term dependencies with gradient descent is difficult[J]. IEEE Transactions on Neural Networks, 1994, 5(2): 157-166
10. S. Hochreiter, J. Schmidhuber. Long short-term memory[J]. Neural computation, 1997, 9(8): 1735-1780