The effectiveness of the distinctiveness-based pruning and the generic algorithm-based feature selection on the performance of forward neural networks in sparse image recognition task

Tianqi Wang[1]

[1] Research
School of Computer Science, Australian National University,
U7149572@anu.edu.au

Abstract. Photographs of historical figures contain precious human memories, but for some reasons, it is difficult to identify the people on the photographs, which causes a lot of trouble for such as historical examinations. Therefore, identifing the people on these historical photos is a very meaningful work. In recent years, deep neural network structures such as CNN, for example, have very good performance on image recognition tasks. However, these deep neural networks usually require a large number of training samples to prevent the occurrence of overfitting. However, photos of historical figures are difficult to obtain (small number of samples), so deep networks such as CNNs are difficult to obtain good results on such small sample image recognition tasks. In this paper, we propose that a two-layer forward neural network is used for recognition. Since the number of samples in the dataset used in this paper is too small, but the number of features is relatively large, overfitting and model overparameterization are highly likely to occur. Therefore, to prevent the above problems, this paper uses a genetic algorithm-based feature selection method to reduce the feature dimensionality while maintaining the model performance. In addition, this paper uses a distinctiveness-based neuron pruning technique to effectively reduce the number of model parameters while ensuring model performance. Finally, this paper achieves the best average test accuracy of 79.17% and 81.67% on the FFMs dataset and Distances datasets, respectively.

Keywords: Neural Networks, Distinctiveness-based Pruning, Generic Algorithm-based Feature Selection

1 Introduction

The images of people in historical photographs contain a lot of important information, but many of them are difficult to identify due to war or natural disasters, etc. At present, the identification of people in historical photos is mainly achieved through manual identification by professionals, however, this essentially relies on the human eye for identification, and despite the use of professional and standard analysis means, there is still the possibility of incorrect identification. In addition, the manual identification method requires a large amount of human resources, while the time cost is also high.

Hence, it is quite intuitive that artificial intelligence may help a lot in this work. However, due to the sparse and old samples of these photos, it is difficult to implement them using cutting-edge image recognition techniques (e.g., networks such as CNN), which require a large number of samples to train the detection models, but this is not realistic for sparse historical images. Moreover, the resolution of most historical photos is not ideal and varies greatly from one photo to another due to the limitations of the shooting technology at that time, which makes it very difficult to extract accurate and uniform feature representations for multiple historical photos [1].

1.1 Data Set

The Facial Feature data set [1] is a sparse data set and it contains a total of 12 sets of photographs of historical figures. Each set consists of three photographs (as shown in Figure.1), with the identities of the people in the first two photographs being the same, and the third photograph being different from the identities of the people in the first two (as a comparison).

Each row of the dataset is represented by a feature for each two images with a label at the end (0 means that the person in these two photos has a different identity, and 1 means vice versa). Thus, the 12 sets of photos constitute a total of 36 rows.

Set #	Image A (Identity matches Image B)	Image B (Identity matches Image A)		Image C (Identity not a match for Images A and B)
2				

Figure. 1 Set No.2 in all 12 image sets [1].

Facial Feature Markers (FFMs). As Figure.2 shows, the faces on each image are marked with 14 markers in the form of coordinates (x,y). Therefore, the feature dimension of FFMs for is 56 (14×2×2). These FFMs can effectively describe the position relationship of bones as well as soft tissues of a human face, however, the representation with FFMs has the following problems: 1. Different shooting angles and head poses will lead to serious changes in the coordinates of FFMs, 2. Due to the inconsistent scale and resolution of different images, simply normalization during training will lead to the lack of relative position relationships [1].

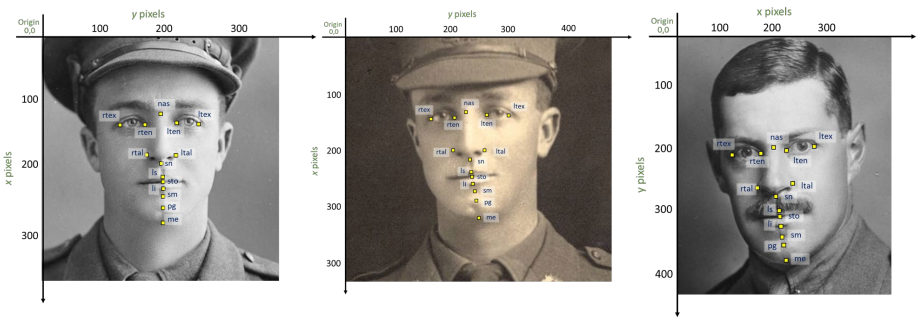

Figure. 2 FFMs on each image. Each marker is represented by x and y coordinates [1].

Distances. The definition of distance is the Euclidean distance between every two markers. For 14 markers on each image, there are 91 distances in total. Therefore, the feature dimension of Distances is 182 (91×2).

1.2 Works Overview

For sparse samples, complex network structures seem difficult to apply, so can simple neural networks be used to solve it? Caldwell et al. tries to focus on the features, by deeply mining the facial features of people in images, combined with the analysis of the skeletal and soft tissue structures of faces, to find some features that can be well expressed on sparse samples, and it shows a relatively good identification accuracy (75%) can be obtained on a small number of test samples using, for example, FFMs and Distances [1].

In this paper, the same FFMs and Distances features are also used to train on a 2-layer forward neural network. However, the main problem in this task is that the dimensionality of these features is too large (there may be a large number of redundant features) and the relative dataset sample is limited (only 36 samples). Therefore, it is likely to cause the occurrence of overfitting and model overparameterization. In this paper, we try to filter the redundant features by a genetic algorithm-based feature selection method and prune the neurons using a distinction-based pruning technique [2] to reduce the model complexity and improve the generalization ability of the model while ensuring the model performance.

Ultimately, this paper achieved 81.67% test accuracy under the FFMs dataset and 79.17% test accuracy under the Distances dataset. This is close to the results of Caldwell (75%). It indicates that this paper maintains a relatively good model performance while reducing the model complexity.

2 Related Works

2.1 Distinctiveness-Based Pruning

Appropriate Model Complexity is Critical. The number of neurons is crucial to the performance of an Artificial Neural Network (ANN), but it is often difficult to directly determine the ideal number of neurons in a network when initializing it. The number of neurons in the input and output layers is determined by the dimensionality of the input data and the specific class of tasks to be solved, but the exact number of neurons in the hidden layer is not known. Intuitively, the higher the feature dimensionality, the more intermediate layers are needed to extract information from the features, the larger the data size, the more complex the network structure.

However, the complexity of the network is not absolutely proportional to the performance of the network. On the contrary, an overly complex network structure leads to an excessive number of model parameters, which tends to reduce the error back propagation speed; in addition, when the task itself is relatively simple, an overly complex network structure tends to lead to overfitting problems, which affects the performance of the model.

The problem to be solved in this paper is a simple binary classification problem, and the data samples are very sparse, on the other hand, the dimensionality of a single sample is very high (Distances feature dimension reaches 182). Training the model on such a small number of samples can easily lead to the occurrence of overfitting, so it is important to choose the appropriate model complexity.

Distinctiveness. Distinctiveness-based pruning provides a way to control the complexity of the model. As Gedeon et al. state [2], the distinctiveness of hidden units is determined from the unit output activation vector over the pattern presentation set, that is, for each hidden unit we construct a vector of the same dimensionality as the number of patterns in the training set, each component of the vector corresponding to the output activation of the unit (as shown in the figure. 3).

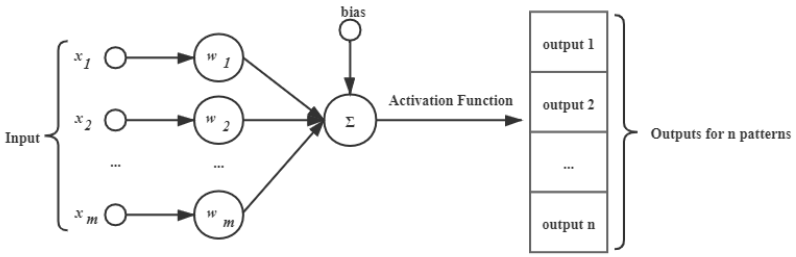

Figure. 3 On each hidden layer neuron, the output for n patterns constitutes a vector. This vector represents the functionality of the hidden unit in (input) pattern space.

Therefore, we can prune no functionality or same functionality neurons by such vectors of each neuron. Formally, we can represent such activation vectors of neuron i and neuron j as a_i and a_j, then we represent the angle between them as $\theta(a_i, a_j)$. If this vector is all zeros, it means that this neuron has basically no function, then we can remove it and if $\theta(a_i, a_j)$ is greater than certain degree $\theta_{\max}$, it means that the two neurons are functionally complementary, we can remove both, if $\theta(a_i, a_j)$ is less than certain degree $\theta_{\min}$, it indicates that the two neurons are functionally similar, we can add the weight of one of them to the other, then remove that one.

2.2 Generic Algorithm-Based Feature Selection

High feature dimensionality may lead to curse of dimensionality problem, therefore, performing feature selection can alleviate this problem to some extent, Genetic algorithms (GA) can be useful for feature selection [3]. GA is based on Darwin's theory of evolution. It is a slow gradual process that works by making changes to the making slight and slow changes. The flow of GA is shown by the following figure.

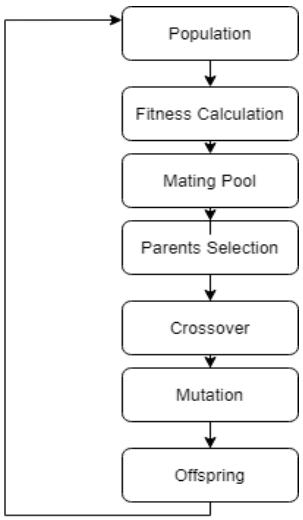

Figure. 4 Generic Algorithm Steps

GA works on a population consisting of some solutions. Each solution is called individual. It represents a chromosome, and a chromosome has k genes. In our task case, each chromosome is a k-dimensional vector consisting of 0, 1 (k is the dimensionality of the feature). Each gene above it is represented by a 1 or a 0, indicating that the corresponding feature was selected or not selected, respectively.

After that, we calculate its fitness for each solution. Fitness expresses the quality of this solution, the higher the fitness of an individual, the greater the chance that it will be selected for mating in the "mating pool". Those selected individuals are called parents.

These parents are then combined in pairs by crossover to produce children. We hope to eliminate the bad genes by combining two individuals with good genes.

However, combining the genes of the parents by crossover alone will result in the same bad part of the genes being inherited from the parents. Therefore, we hope to solve it by mutation.

By replacing their parents in the original population with the children and repeating the above process several times, we will obtain the optimal or acceptable solution. Each repetition, called a generation.

3 Experimental Design

The main problem in our task is that overfitting and model overparameterization due to the case of too large feature dimension and too few samples can degrade the performance of the model. Therefore, this paper addresses the above problems from two aspects. First, reduce the model complicity by decreasing the number of neurons in the model as much as possible by pruning technique without significantly degrading the model performance. Second, improving the model performance through feature selection.

Therefore, the experiments in this paper will be divided into two stages, 1. Model Selection: testing different pruning levels with the test accuracy as metric. And the number of neurons in hidden layer corresponding to the pruning degree with the highest test accuracy is used as the baseline for subsequent experiment. 2. Feature Selection: testing on the feature selection method with different parameters, using the test accuracy as metric to filter out a large number of redundant features. Then improving the model performance by identifying the most valuable features.

3.1 Model Selection Stage

we examine nine different pruning levels using a 2-layer fully connected neural network. The 9 different pruning degrees are shown specifically by the table below.

Table 1. Pruning Degrees. Note that, the first pruning degree represent the baseline (not applying pruning).

Degree No.	1	2	3	4	5	6	7	8	9
Complementary	180	175	170	165	160	155	150	145	140
Similar	0	5	10	15	20	25	30	35	40

Data pre-processing. Since there are scale differences in the data (different resolutions of the images), we first normalize the FFMs dataset and the Distances dataset to between 0, 1. This will favor the convergence of the model.

Network Architecture. In this paper, we use a 2-layer fully connected neural network, with the hidden layer followed by a sigmoid activation, and the output layer is also followed by a sigmoid activation due to the binary classification problem. Due to sigmoid activation, the output of the neuron will be limited between (0, 1). This leads to the pruning where the resulting activation vector clip will be restricted between $(0, \pi / 2)$. To ensure that the vector pinch angle is between $(0, \pi)$, we need to normalize the activation vector to between (-0.5, 0.5) by subtracting 0.5 when pruning.

Optimization. Since it is a binary classification problem, we choose cross-entropy loss. In addition, we use the Adam as optimizer.

Hyperparameters. The training epoch is 500 and the learning rate is 0.01. Intuitively, we set the number of hidden layer neurons to a larger value (14 for the FFMs dataset and 50 for the Distances dataset) at the beginning.

Evaluation Metric. Due to the small sample size, we evaluated by 10-fold cross-validation. By average recognition accuracy as metric.

3.2 Feature Selection Stage

In the feature selection phase, we conduct experiments on the FFMs and Distances datasets, respectively. The average test accuracy of all features will be utilized as a baseline. After that, the features selected by different population and crossover rates are tested. The feature combination that gives the final testing accuracy will be used as the optimal result and compared with the baseline. The population is set to these 5 values of [20, 40, 60, 80, 100] and the crossover rate is set to [0.1,0.2, ... ,0.9] these 9 values, and we set hyperparameter number of generations to be 50 and mutation rate to be 0.002. Therefore, we need to perform 10-fold cross-validation for each of the 45 different cases. The time cost of this process is large.

We perform experiments using the neural network architecture selected from the model selection phase. The raw data needs to be preprocessed as before. Optimization method, hyperparameters setting and evaluation metric are as same as in model selection phase.

4 Result

4.1 Model Selection

We apply distinctiveness-based model pruning to the FFMs and Distances data, respectively. It is found that the performance of the models (test precision) generally decreases as the degree of pruning increases, which is in line with our expectation. However, we found on both the FFMs and Distances datasets that the model still has a performance close to that of the model without pruning when the pruning degree is (165, 15). And the number of post-pruning neurons corresponding to (165, 15) is around 5 for both. This indicates that by using a distinctiveness-based pruning technique, we can significantly reduce the complexity of the model and, at the same time, maintain the performance of the model. The detailed experimental results are shown in the figure.5 below. Therefore, we set the number of hidden layer neurons to 5 as a benchmark for subsequent experiments.

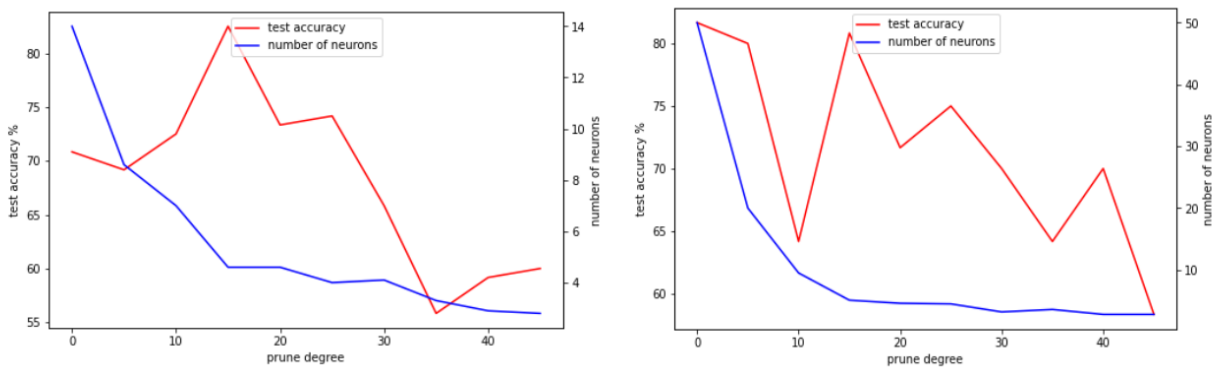

Figure. 5 Test accuracy (red) and number of hidden layer neurons (blue) for distinctiveness-based pruning under the FFMs and Distances datasets. On the left are the results under FFMs with a benchmark of 70.83% and 82.5% at a pruning degree of (165, 15). On the right is the result under Distances with 81.67% benchmark and 80.83% pruning at degree of (165, 15).

4.2 Feature Selection

We use the model obtained from the above experiments to perform feature selection. The experimental results on FFMs and Distances are shown in Figure. 6 and Figure. 7 respectively. Figure 6 shows that, at a population of 80 and a crossover rate of 0.7, a precision of 79.17% was achieved by GA-selected FFMs features, which is close to the benchmark (82.5%). Figure 7 shows that, at a population of 100 and a crossover rate of 0.7, a precision of 81.67% was achieved by GA-selected Distances features, which is close to the benchmark (80.83%).

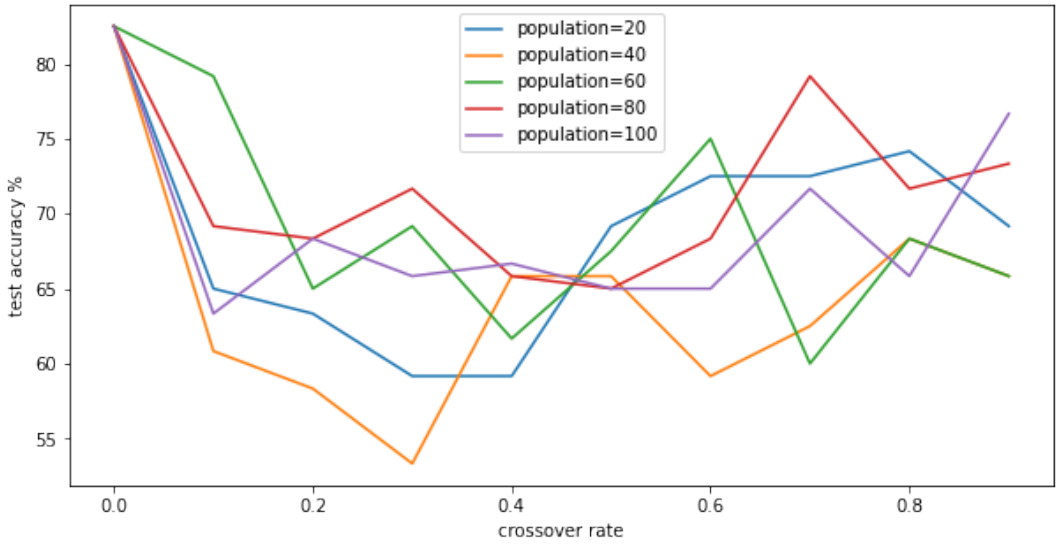

Figure. 6 Accuracy of testing FFMs features selected by GA under different parameter settings.

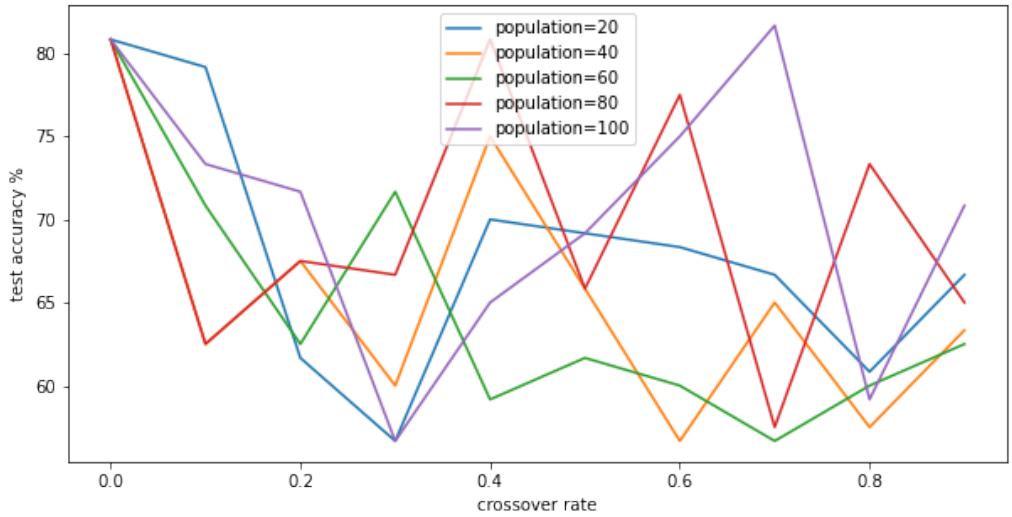

Figure. 7 Accuracy of testing Distances features selected by GA under different parameter settings.

The number of FFMs features selected by GA is only about 1/2 of the original features, while the number of Distances features selected by GA is 82, which is less than 1/2 of the original features This indicates that there exist redundant in FFMs and Distances datasets and by screening out a large number of redundant features, the model can achieve close to or even better performance than the benchmark.

5 Conclusion

For datasets with too few samples like historical images, the model complexity can be significantly reduced by distinctiveness-based pruning techniques while ensuring the model performance. there is a large amount of redundancy in FFMs and Distances features, and a large number of redundant features can be screened out by GA-based feature selection without degrading the model performance. However, the time cost of GA is too high.

6 Discussion and Future Works

We can try to introduce a regularization term in the loss function to further control the model complexity.

Too few samples lead to large fluctuations in test results, which poses a challenge for the determination of optimal parameters. Therefore, the existing data set needs to be augmented in the future.

The GA-based feature selection method is effective, but it also has the obvious problem that it is too costly in terms of time. Other feature selection methods can be tried like Principal component analysis.

FFMs features and Distances features proved to be heavily redundant and further attempts can be made to reduce the redundancy on the one hand. On the other hand, attempts to utilize other features, such as more complex quadrilateral relations or proportional relations features, can be considered.

References

1. Caldwell, S. (2021) "Human interpretability of AI-mediated comparisons of sparse natural person photos," CSTR-2021-1, School of Computing Technical Report, Australian National University.
2. Gedeon, T. D., & Harris, D. (1992, June). Progressive image compression. In Neural Networks, 1992. IJCNN., International Joint Conference on (Vol. 4, pp. 403-407). IEEE.
3. Chandrashekar, G., & Sahin, F. (2014). A survey on feature selection methods. Computers & Electrical Engineering, 40(1), 16-28.