Proceedings of the 3rd International Conference on Signal Processing and Machine Learning
DOI: 10.54254/2755-2721/4/20230352

Machine learning and emotion recognition in video clips

Yichen Chang

School of Artificial Intelligence and Advanced Computing, Xi'an Jiaotong-Liverpool University, Suzhou, Jiangsu, China, 215123

yichen.chang19@student.xjtlu.edu.cn

Abstract. The development and application of machine learning has been a hot topic in recent years, and enabling machines to read people's emotions is of great importance in many aspects of social development, such as healthcare, products development and human-computer interaction. This study recognizes seven human emotions by building a model of a neural network and training dataset. Using a large-scaled video dataset, Dynamic Facial Expression in-the-Wild (DFEW), trains the ResNet50 model and outputs the precisions. There are three possible training results, if the accuracies are all above 80%, then the neural network is well-performed; if partial classification accuracies is above 80% while others are above 50%, the neural network is partial well-performed; if the accuracies are all under 50%, the neural network is poor-performed.

Keywords: Machine Learning, Emotion Recognition, Residual Neural Network.

1. Introduction

Emotion recognition plays an important role in human-computer interaction, and people's emotions show how they feel when they do daily work and communication [1]. How to identify emotion has been researched from many aspects, such as pictures, speech and text. However, there are certain limitations in these emotion recognitions, people may dissemble their true feelings when detecting emotion from speech and text. For image recognition, an expression sometimes can have many meanings [2], and sometimes an image cannot completely represent the emotional changes. For example, the image of crying may be the expression of sadness, and at the same time, it may be the expression of tears of joy. Therefore, it is of great significance to analyze people's emotional changes in a period of time for emotion recognition.

This study identifies people's emotions by analyzing videos. There are a number of both image and video emotion datasets, but most of them are small size. Thus, this study chooses Dynamic Facial Expression in-the-Wild (DFEW) which contains 16372 challenging movie video clips [3]. These video clips are segmented into seven labels, happy, sad, neutral, angry, surprisde, disgusted and fearful. Convolutional neural network (CNN) is widely used for classification [4]. By using CNN to classify the large scaled dataset, the model is hard to be overfitted and the efficiency will not be low [4].

The model is to use ResNet (a kind of CNN), to classify the main character in each video clip, and then trains the model with 5-fold cross-validation to increase the accuracy of the model. Previous studies of such large-scale datasets have low accuracies; thus, this study chooses ResNet to reduce the error rate.

DOI: 10.54254/2755-2721/4/20230352

The research on emotion recognition does great help to medical treatment, product development and human-computer interactors [2]. In medical care, doctors can better understand the patient's state and feelings during the treatment by using the machine's emotion recognition, especially when the patients have an emotional expression disorder. Developers can have a more accurate understanding of user needs and satisfaction when developing products [2]. Also, it can make human-computer interaction more natural by recognizing human emotions.

2. Methodology

2.1. DFEW dataset

There are 16372 movie videos in this dataset. Each video shows the character's facial expressions and corresponding dialogues, and most of the movie clips are within 2-8 seconds in length, with a few exceeding 10 seconds. Using this dataset, not only can facial expression recognition be performed, but the corresponding dialogues can also be used to assist expression recognition by determining voice intonation. The dataset also provides each video clip's expression distribution in seven dimensions and emoticon annotation for a single marker of seven classic discrete emotions [3]. The expression distribution and the annotation help with the model building part and simplify the training work.

2.2. Data preprocessing

2.2.1. OpenCV. According to Jiang et al. [3], the data processing will use OpenCV. First, they extract frames from each video clip and use the face++ API to detect the positioning of faces, eliminating frames where the main character's front face is not detected and frames where the useful frame rate is below 50%. Thus, 362 clips are eliminated from training. Due to the pose, lighting, and position of the person, it is very difficult to accurately locate the face, so the face affine transformation will be used to locate the face in each frame to improve the recognition rate [5].

Affine Transformation:

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} R_0 & R_1 & T_X \\ R_2 & R_3 & T_Y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

where, (T_X, T_Y) is the translation of x-axis and y-axis; R_i reflects changes in image rotation, scaling, etc; (x, y) is the original coordinates; (x', y') is the coordinates after affine transformation. Finally, using time interpolation method to align each movie clip's length to 16 frames.

2.2.2. ResNet50. As for the dataset has provided the pre-processed frames, we use these frames directly for the training model. ResNet is a kind of deeper convolutional neural network. It is difficult to train deeper neural network, but for ResNet, He et al. [6] used a residual framework to simplify the training of neural network. They also proved that residual neural network can get a high accuracy as the number of layers increasing.

ResNet has a 7×7 convolutional kernel with 2 strides connected to a 3×3 max pool with 2 strides. The input (3, 224, 224) (channel, height, width) going through the convolutional layer and the MaxPooling layer will obtain a (64, 56, 56) output. Then, four series of residual structures are connected. The following table shows the structure of ResNet50.

DOI: 10.54254/2755-2721/4/20230352

Table 1. Structure of ResNet50.

layer name	output size	50-layer
conv1	112 x 112	7 x 7, 64, stride 2
		3 x 3 max pool, stride 2
conv2_x	56 x 56	$\begin{bmatrix} 1 \times 1, 64 \\ 3 \times 3, 64 \\ 1 \times 1, 256 \end{bmatrix} \times 3$
conv3_x	28 x 28	$\begin{bmatrix} 1 \times 1, 128 \\ 3 \times 3, 128 \\ 1 \times 1, 512 \end{bmatrix} \times 4$
conv4_x	14 x 14	$\begin{bmatrix} 1 \times 1, 256 \\ 3 \times 3, 256 \\ 1 \times 1, 1024 \end{bmatrix} \times 6$
conv5_x	7 x 7	$\begin{bmatrix} 1 \times 1, 512 \\ 3 \times 3, 512 \\ 1 \times 1, 2048 \end{bmatrix} \times 3$
	1 x 1	average pool, 1000-d fc, softmax
FLOPs		3.8×10^9

The first series (conv2_x) of residual structures contains three residual layers. Each residual layer has three convolutional kernels. (64, 56, 56) is the input for the first residual layer; after these three convolution kernels, the length and width of the feature matrix shrinks and the depth increases; the output of the first series is (256, 56, 56). Then the output repeats that for another two times. Then it goes into the second series, which contains four residual layers; then, goes into the third series with six residual layers and the fourth series with three residual layers. Finally, the output goes to the average pool and the fully-connected layer; then, use the softmax function for classification [6]. The softmax function is to map the output of multiple neurons to the interval of (0,1), so as to carry out multi-classification. If there is an array V, the i^{th} element in this array's softmax is $\text{softmax}(i) = \frac{e^i}{\sum_{j=1}^{k} e^j}$, where k represents the output of the neural network, j represents the j^{th} output.

In these four series, down-sampling begins to work in the first layer. However, the sub-branch in the first layer of the first series only aligns with the depth of the feature matrix. Hence, here we take the second series as an example. In picture 1, there are three layers in the main branch and another one in the sub-branch that is connected by the dotted line. That means the second series receive the (256, 56, 56) output from the first series. In the main branch, the first layer decreases the depth from 256 to 128 with 1 stride; the second layer decreases the height and width from 56 to 28 with a 3 x 3 convolutional layer and 2 strides; while the last layer increases the depth from 128 to 512 with 1 stride. The sub-branch only contains one layer which is used for down-sampling. In order to make the same output shapes of the main branch and sub-branch, the layer in the sub-branch decreases the height and width from 56 to 28 by 2 strides and increases the depth with 512 channels. The final output is the output of the main branch plus the output of the sub-branch. However, in the next layers, the sub-branch does not have any layers, and the final output is the output of the main branch plus the input. Besides, there is a ReLu function in each layer except the last convolutional layer, and after adding up two matrixes. The same operation is also present in the third (conv4_x) and fourth (conv5_x) series.

DOI: 10.54254/2755-2721/4/20230352

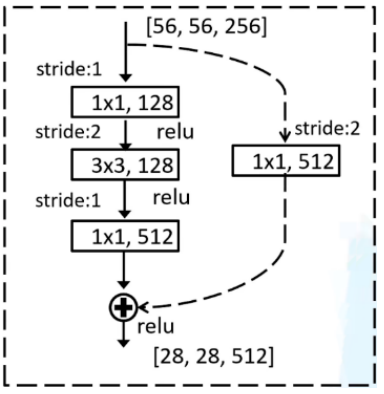

Figure 1. Residual structure of the second series' first layer.

2.2.3. Training with 5-fold cross-validation. In the training part, the frames from the pre-processing process are divided equally into five equal parts (fd1, fd2, fd3, fd4, fd5). For each training, one part is selected as the test set and the other four as the training set. Totally we have five training sections with different 5 test sets. Thus, the model was trained by all data, and the model is hardly suffering from overfitting.

2.3. The specific steps of the experiment

As the dataset has preprocessed the data and equally divided them into five folders, we just read the CSV files and import labels of these movie clips. Then we build the ResNet50 model with it conv_1, batch normalization, ReLu function, MaxPooling, the four series of residual structures, the average pooling layer, and the fully-connected layer.

Code of ResNet50 model:

```
class ResNet(nn.Module):
    """
    __init__
        block: stacked vasic modules
        block_num: resnet50=[3,4,6,3]
        num_classes: classification feature dimension after full connection

    _make_layer
        block: stacked basic modules
        channel: 64, 128, 256, 512
        stride: conbolutional strides
    """
    def __init__(self, block, block_num, num_classes=1000):
        super(ResNet, self).__init__()
        self.in_channel = 64 # conv1's output channel

        self.conv1 = nn.Conv2d(in_channels = 3,
out_channels=self.in_channel, kernel_size=7, padding=3, bias=False) #H/2,
W/2, C:3->64
        self.bn1 = nn.BatchNorm2d(self.in_channel)
```

DOI: 10.54254/2755-2721/4/20230352

```
        self.relu = nn.ReLU(inplace=True)
        self.maxpool = nn.MaxPool2d(kernel_size=3, stride=2, padding=1)
#H/2, W/2, C remain
        self.layer1 = self._make_layer(block=block, channel=64,
block_num=block_num[0], stride=1)
        #H, W remain, downsample controls shortcut, out_channel=64*4=256
        self.layer2 = self._make_layer(block=block, channel=128,
block_num=block_num[1], stride=2)
        #H/2, W/2, downsample controls shortcut, out_channel=128*4=512
        self.layer3 = self._make_layer(block=block, channel=256,
block_num=block_num[2], stride=2)
        #H/2, W/2, downsample controls shortcut, out_channel=256*4=1024
        self.layer4 = self._make_layer(block=block, channel=512,
block_num=block_num[3], stride=2)
        #H/2, W/2, downsample controls shortcut, out_channel=512*4=2048

        self.avgpool = nn.AdaptiveAvgPool2d((1, 1))
        self.fc = nn.Linear(in_features=512*block.expansion,
out_features=num_classes)

        for m in self.modules(): # initialize weights
            if isinstance(m, nn.Conv2d):
                nn.init.kaiming_normal_(m.weight, mode='fan_out',
nonlinearity='relu')
```

Then we create _make_layer() to adjust the width, height and depth of each layers. Finally, the forward () contains the execution orders of the ResNet50 model.

Using numpy and StartifieldKFold from sklearn to use 5-fold cross-validation, in order to train five times, we divide the frames and corresponding labels into X_train and Y_train and X_test and Y_test, where the ratio of training section to test section is 4:1.

Code of KFold:

```
import numpy as np
from sklearn.model_selection import KFold
kfold = KFold(n_splits = 5, shuffle = True) # there are 5 parts of data
which means the kfold function will implement 5 times
```

After the training, well-performed weights will be obtained; thus, we use these weights to classify the whole dataset. Then, output the confusion matrix and the classification report to see the efficiency and accuracy of the model. Normally, when the accuracy of large-scaled dataset reaches 80%, the neural network has been well trained.

3. Result and discussion

There are three cases of experimental results.

3.1. Well performance

The accuracies for each category of emotions are 80% or more. That means the model was successfully trained, the weights are good and the residual network structure fits the dataset very well. If the desired

DOI: 10.54254/2755-2721/4/20230352

results are higher, people can use deeper residual neural networks, such as ResNet101 or ResNet152. As the network get deeper, the performance gets better because the residual neural network solved the problem of gradient disappearance and gradient explosion [7].

3.2. Partial well performance

The accuracies for two to three categories were less than 50%, and the rest was above 80%. There are two possible reasons for this result. One is the frames obtained during data pre-processing do not reflect the real emotions of the main character of the video. For example, the model only captures some frames without expressions or blurs the boundaries of neutral expressions. To solve this problem, the temporal bottleneck can be used for the preprocessing part to obtain more information in each video clip [8]. Another is that these emotions correspond to a smaller number of fragments than the others, so more segments containing these emotions can be added to the dataset.

3.3. Poor performance

The accuracies for each category of emotions were less than 50%. The reason may be that the model gets the wrong parameters in data processing, and no normalization is performed on the frames we get in the preprocessing part. The 0-single-labeled videos do not be moved from the training set and the test set. So first we need to remove the 0-single-labeled videos before preprocessing part, and after intercepting frames from the video, the frames should be normalized. Besides, the weights obtained through training may not fit this data set, so retrain for new weights.

When doing experiments, it is not enough to just analyze the accuracy rate. We need to further analyze the recall and precision of the model using the confusion matrix. The training time of this model is very long because the neural network has many layers. We can choose the AdaFrame proposed by that Wu et al. [9], which can increase the speed of video processing. It is a conditional computation framework to adapt the allocation of computing resources according to input video conditions. Besides, we can use the neural network Shen, Xu and Zhuang proposed [10] to enhance the edge features of faces with the combination of sobel operator and Resnet, so as to improve accuracy. Also, if the precisions are high, people can consider using a network with fewer layers, such as ResNet20 and ResNet34 to see the differences.

4. Conclusion

This thesis uses residual neural networks for video emotion recognition. In general, the residual neural network has a simple structure, high performance, and no gradient explosion as the neural network level deepens. However, the training speed is low, especially when the dataset is large-scaled. Thus, in future research, people should work on that to increase the speed of training. Moreover, when characters have multiple feelings, the model cannot classify the results. Subsequent research should be devoted to solving this problem.

Emotion recognition has significant implications for the advancement of machine learning. Hence, there should be more research on emotion recognition, not just focus on text, images and videos. Since facial expressions and voice intonation can be disguised, machine recognition cannot detect the real emotions of people. Therefore, people can try to perform micro-expression recognition and muscle signal recognition, and at the same time, one can build on this to perform physiological signal recognition, such as heart rate, blood oxygen, eye movement, etc. in future research.

Acknowledgment

First of all, I would like to express my deep respect and gratitude to my teachers, Ike and professors, Dr. M.J. and Dr. Su. Their lectures and guidance have equipped me to complete this thesis independently, while they have provided guidance and criticism in various aspects while writing this thesis. In addition, I would like to thank my friends and family. My friends helped me to make inferences about the cause of the errors when the code was reported; meanwhile, my family gave me professional advice on psychology and great encouragement and support. Their help and support enabled me to finish this thesis.

DOI: 10.54254/2755-2721/4/20230352

References

[1] Deng, L., Wang, X., Jiang, F., & Doss, R. (2021). EEG-based emotion recognition via capsule network with channel-wise attention and LSTM models. CCF Transactions on Pervasive Computing and Interaction, 3(4), 425–435. https://doi.org/10.1007/s42486-021-00078-y

[2] Chen, J., Ro, T., & Zhu, Z. (2022). Emotion Recognition With Audio, Video, EEG, and EMG: A Dataset and Baseline Approaches. IEEE Access, Access, IEEE, 10, 13229–13242. https://doi.org/10.1109/ACCESS.2022.3146729

[3] Jiang, X., Zong, Y., Zheng, W., Tang, C., Xia, W., Lu, C. & Liu, J. (2020). DFEW: A Large-Scale Database for Recognizing Dynamic Facial Expressions in-the-Wild. ACM Multimedia.

[4] Zelenina, L., Khaimina, L., Khaimin, E., Khripunov, & Zashikhina, I. (2022). Convolutional Neural Networks in the Task of Image Classification. Mathematics & Informatics, 24(1), 19–29. https://doi.org/10.53656/math2022-1-2-con

[5] Myers, A. J., & Megherbi, D. B. (2014). An efficient computational intelligence technique for affine-transformation-invariant image face detection, tracking, and recognition in a video stream. 2014 IEEE International Conference on Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), Computational Intelligence and Virtual Environments for Measurement Systems and Applications (CIVEMSA), 2014 IEEE International Conference On, 88–93. https://doi-org.ez.xjtlu.edu.cn/10.1109/CIVEMSA.2014.6841444

[6] He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference On, 770–778. https://doi.org/10.1109/CVPR.2016.90

[7] Gupta, A., Arunachalam, S., & Balakrishnan, R. (2020). Deep self-attention network for facial emotion recognition. Procedia Computer Science, 171, 1527–1534. https://doi-org.ez.xjtlu.edu.cn/10.1016/j.procs.2020.04.163

[8] Carvalho, S. R., Bertagnolli, N. M., & Folkman, T. (2021). Temporal Bottleneck Attention for Video Recognition. 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), Machine Learning and Applications (ICMLA), 2021 20th IEEE International Conference on, ICMLA, 1400–1406. https://doi-org.ez.xjtlu.edu.cn/10.1109/ICMLA52953.2021.00226

[9] Wu, Z., Li, H., Xiong, C., Jiang, Y., & Davis, L. S. (2022). A Dynamic Frame Selection Framework for Fast Video Recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence, Pattern Analysis and Machine Intelligence, IEEE Transactions on, IEEE Trans. Pattern Anal. Mach. Intell, 44(4), 1699–1711. https://doi.org/10.1109/TPAMI.2020.3029425

[10] Shen, X., Xu, X., & Zhuang, Y. (2021). Facial Emotion Recognition Based On Sobel-Resnet. 2021 IEEE 5th Information Technology,Networking,Electronic and Automation Control Conference (ITNEC), Information Technology,Networking,Electronic and Automation Control Conference (ITNEC), 2021 IEEE 5th, 5, 484–488. https://doi-org.ez.xjtlu.edu.cn/10.1109/ITNEC52019.2021.9587235