Classifying Television Commercials by Convolutional Recurrent Neural Networks

Takahiro Murakami s1240099 Supervised by Prof. Kazuyoshi Mori

Abstract

In this paper, we describe the development of a system to classify television commercial categories. In modern times, people watch televisions on a daily basis. Investigating the television commercials can be useful for social analysis because the television commercials have influences on people and the social culture. I decided to develop a Convolutional Recurrent Convolutional Neural Network (CRNN). CRNN is a combination of Recurrent Neural Network (RNN) and Convolutional Neural Network (CNN). In order to evaluate the system with recurrent, we decided to compare it with CNN under various conditions. In this study, poor results were obtained, so additional experiments were performed to infer the cause.

1 Introduction

In modern times, many people have television and often watch television commercials. Television commercial provides information on various products and services to us. It affects buying intention of viewers. We consider that investigating television commercials is useful for social analysis because television commercials reflect the social situation and trends at the time. In order to investigate television commercials, it is important to classify television commercials according to their categories. We decided to develop a system that automatically classifies television commercials using convolution neural network (CNN) because it is inefficient to classify television commercials manually. In addition, we have developed a convolutional recurrent neural network (CRNN). This is because we wanted to make it more efficient to classify commercials by adding not only image processing evaluations but also time evaluations. In this study, by increasing or decreasing the convolutional and pooling layers, we found out which was better compared to CNN. We will also experiment on how many CRNN convolutional layers and pooling layers will be most effective for CRNN.

2 Neural Network

Neural network [1] is a computing system modeled human on the brain and nervous system, which is often used in the field of pattern recognition such as character recognition and speech recognition. The neural network consists of an input layer, hidden layers, and an output layer. Machine learning by a neural network with multiple hidden layers is called deep learning. An example of the neural network is shown in Fig. 1. The circle in Fig. 1 is called a neuron. It is calculated with a weight and a bias and passes the value to the next neuron. Adjusting this parameter makes it possible to develop a neural network that eventually output the expected result.

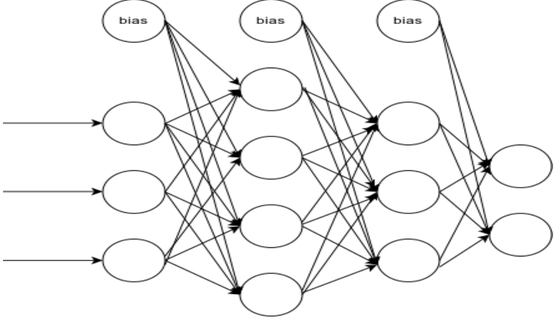

Fig. 1 Neural Network

2.1 Convolutional Neural Network

Convolutional Neural Network is one of neural networks with deep learning. It is a neural network with convolution layers and pooling layers. Neural network is often used in the field of image recognition. In the convolution layer, CNN calculates the input data with convolution and obtains local feature of the data. The data calculated in this way is called a feature map [2]. The pooling layer is usually applied behind the convolution layer and compresses the information to transform the input data into a more manageable form. The pooling layer can be several types such as average pooling [2] and max pooling [2]. In this research, max pooling, which outputs maximum value of input, is employed.

2.2 Recurrent Neural Network

Recurrent Neural Network (RNN) is also one of neural networks with deep leaning. The hidden layers of RNN is different from those of other neural networks in the sense of its structure. The hidden layer of the RNN can transmit the output from the hidden layer at any time (***t***) to the hidden layer at the next time (***t*** **+ *1***). As a result, the hidden layer at the time t receives the input from the hidden layer at the previous time ***t−1*** in addition to the input from the input layer at the same time ***t***. The combination of CNN and RNN is CRNN. In this study, we built CRNN by taking the images from the last pooling layer in time order and input them into the recurrent layer.

2.3 OpenCV and TensorFlow

The Open Source Computer Vision Library (OpenCV) [3] is an open source library that summarizes the functions for processing images and movies. This supports a wide variety of programming languages such as C, C++, Python, and Java. This is used for video and image processing in this research. TensorFlow [4] is also an open source software library used in machine learning developed by Google. This supports the programming languages, C, C++, and Python. This is used for neural network constructions in this research.

3 Experiments

3.1 Original Experiment

3.1.1 Preparation

In order to train CNN and CRNN, we have so far collected 133 of various television commercials. We use only 15 seconds of television commercials in this study. These CMs are labeled as a category with "food," "car", "cosmetic" and "other". After this task, we obtained training videos of food (44), car (16), cosmetic (17), and other (56). In the same way, in order to test CNN and CRNN, we have so far collected 120 of various television commercials and labeled according to categories food (35), car (16), cosmetic (16), and other (53).

3.1.2 Structure of CRNN

We developed CNN (Fig. 2) and CRNN (Fig. 3)[5] that categorizes television commercials into four categories ("food," "car", "cosmetic", "other"). CNN consists of input layer, three 3D convolutional layers, two fully connected layers and output layer. If the number of fully connected layers are reduced from two to one and a recurrent layer is added, the structure becomes CRNN. In the input layer, we have used 30 images as input that were taken out from one television commercial (Sampling every 0.5 second). All of CMs are originally recorded as 1920 x 1080 resolution (It depends on the broadcast media of CM such as terrestrial broadcasting, BS broadcasting and CS broadcasting), but for the computer source limitation, we have resized them to 80 x 45. The CNN and CRNN have four 3D convolution. Behind each 3D convolution layer, a pooling layer is applied (Fig. 3) We employed max pooling in pooling layer. We employed `BasicRNNCell` and `dynamic_rnn` in recurrent layer. Receive the output from the last pooling layer. Sort it by time. Enter them in order. Let the last output of the recurrent layer be the output on CRNN system. In this research, we evaluate recurrent layer by comparing CNN and CRNN and changing number of convolutional layer and pooling layer. The output layer gives the decision of the classification into four categories. For activation functions, the softmax function is employed on the output layer. Learning rate set to 0.00001. Training is terminated when training accuracy reaches 90%.

3.1.3 Result and Discussion

We trained and tested CNN and CRNN. Tables 1 and 2 show the results. As a result, with two convolutional layers, the number of epochs was 36 for CNN and 160 for CRNN. Accuracy was 41.96% for CNN and 31.45% for CRNN. At this time, the difference between CNN and CRNN is 10.51%. With three convolutional layers, the number of epochs was 47 for CNN and 132 for CRNN. Accuracy was 42.74% for CNN and 39.52% for CRNN. At this time, the difference of accuracy between CNN and CRNN is 3.22%. With four convolutional layers, the number of epochs was 63 for CNN and 103 for CRNN. Accuracy was 30.64% for CNN and 34.68% for CRNN. At this time, the difference of accuracy between CNN and CRNN is 4.04%. When we use four convolutional layers, CRNN is better than CNN. But when we use two convolutional layers and three convolutional layers, CNN is better than CRNN. From this, it is found that CRNN showed little improvement in performance compared to CNN. We think that this is because the CM classification is not categorized by time series. Therefore, we try to do additional experiments with datasets that are categorized in time-dependent categories.

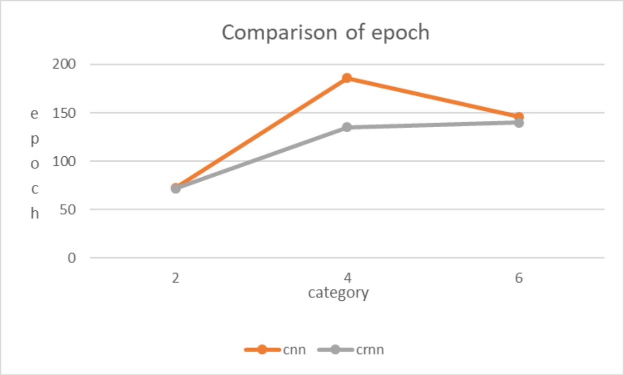

Table 1: Comparison of epoch

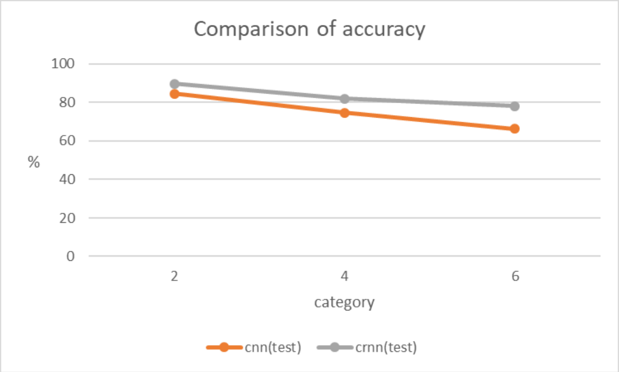

Table 2: Comparison of accuracy

3.2 Additional Experiments

3.2.1 Preparation

In order to training CRNN, we use 599 of various human's action in KTH dataset. KTH dataset is the most widely used dataset in motion recognition research [6][7]. In this study, we compare CRNN to CNN by using this data set. These CMs are labeled as a category with "walking," "running," "jogging," "boxing," "waving," and "clapping,". After this task, we got training videos of 80 videos per category. In the same way, in order to test CRNN and CNN, we tested 20 videos per category.

3.2.2 Structure of CRNN

We developed CNN (Fig. 4) and CRNN (Fig. 5) that categorizes human's motion into six categories ("walking," "jogging," "running," "waving," "hand clapping," "boxing"). It consists of input layer, 3D convolutional layers, recurrent layer, fully connected layer and output layer. In the input layer, we have used 30 images as input that were taken out from one kth-dataset (Sampling every 7 images). All of kth-dataset are 160x120 pixels. we have resized them to 80 x 60. 3D convolution layers consist of four layers, and each pooling layer is applied behind each 3D convolution layers. We employed max pooling in pooling layer. In this research, we evaluate the recurrent neural network by changing number of category of motions. The output layer gives the decision of the classification into six categories. For activation functions, the softmax function is employed on the output layer. Learning rate set to 0.00001. Training is terminated when training accuracy reaches 90%.

3.2.3 Result and Discussion

We trained and tested CNN and CRNN. Tables 3 and 4 show the results. As a result, in two categories, epoch of CNN is 73, on the other hand, epoch of CRNN is 72. It was found that CRNN is slightly better than CNN. In accuracy, the CNN is 84.62%, the CRNN is 89.74%. The difference is 5.12%. Next, in four categories, the number of epochs is 186 in CNN and 135 in CRNN. Accuracy is 74.68% for CNN and 81.92% for CRNN. The difference is 7.24%. Finally, when six categories, the number of epochs was 146 for CNN and 140 for CRNN. Accuracy is 66.39% for CNN. It is 78.15% for CRNN. The difference is 11.76%. From these results, it is found that Accuracy decreased for both CRNN and CNN as the category increased. It is also found that the difference between CNN and CRNN was large. From this, CRNN is considered to be effective for time-dependent classification.

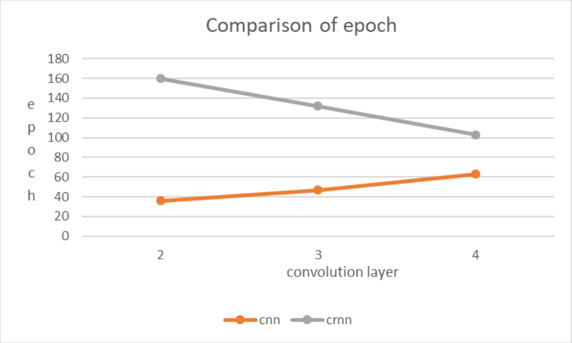

Table 3: Comparison of epoch

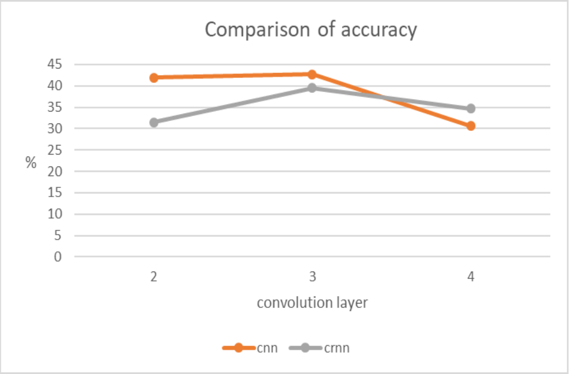

Table 4: Comparison of accuracy

4 Conclusion and Future Work

In this study, we developed CNN and CRNN system that classifies television commercial and evaluated the system by compare to CNN. The results showed that CNN was more efficient than CRNN in classifying television commercials. However, from the results of the additional experiments, it was found that CRNN gave better results than CNN when the category to be classified was time-dependent. Besides, as future works, it is necessary to change the category of the television commercials and the place where the recurrent layer is placed in order to obtain better results.

5 References

[1] Hiromi Hirano,“ C でつくるニューラルネットワーク，パーソナルメディア株式会社, ”1991.
[2] 原田達也,“ 画像認識 ”, 講談社, 2017.
[3] “OpenCV documentation Index,” http://docs.opencv.org/
[4] “MNIST For ML Beginners,” Dec.2016, https://www.tensorflow.org/versions/r0.11/tutorials/mnist/beginners/index.html#the-mnist-data
[5] Goodfellow, Y. Bengio, A. Courville, Deep Learning, MIT Press, 2016.
[6] “Recognition of human actions” http://www.nada.kth.se/cvap/actions/
[7] “動画像認識のための 3 次元畳み込み RNN の提案” https://ipsj.ixsq.nii.ac.jp/ej/?action=repository_uri&item_id=157648&file_id=1&file_no=1

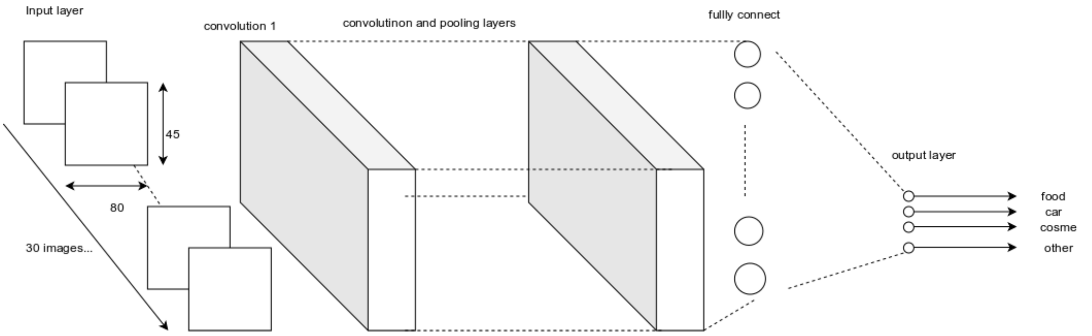

Fig. 2: Convolutional Neural Network

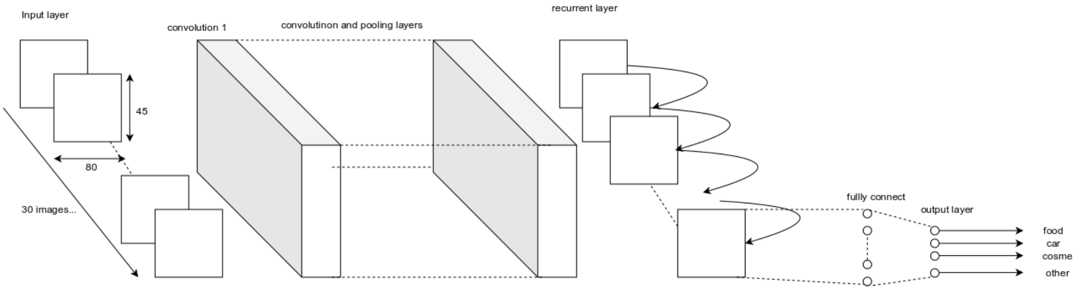

Fig. 3: Convolutional Recurrent Neural Network

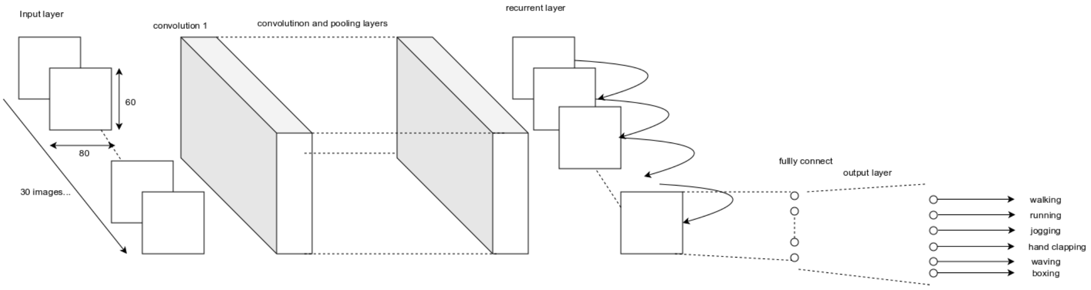

Fig. 4: Convolutional Neural Network

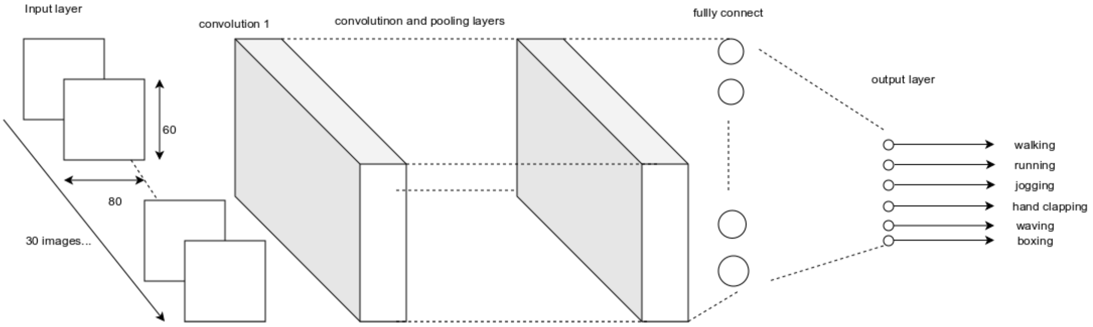

Fig. 5: Convolutional Recurrent Neural Network