

COL774
Machine Learning
Assignment 4

Atif Anwer 2020EE10479
Kanishk Goel 2020EE10503

November 28, 2022

Contents

1	Convolutional Neural Network	3
1.1	Objective	3
1.2	Architecture	3
1.3	Results	3
2	Recurrent Neural Network	3
2.1	Objective	3
2.2	Architecture	4
2.3	Results	4
3	Competitive Model Architecture	4
3.1	Convolutional Neural Network - VGG11	4
3.2	Recurrent Neural Network - BERT (Kaggle Submission)	4

Libraries Used

- numpy
- nltk
- sys
- pandas
- os
- re
- time
- tqdm
- skimage
- torch
- torchtext
- torchvision
- sklearn
- copy
- warnings
- re
- matplotlib
- transformers
- tensorflow

1 Convolutional Neural Network

1.1 Objective

To implement and train a convolutional neural network on book cover images to predict the genre it belongs to among 30 different genres.

1.2 Architecture

The model consisted of the following layers implemented using the *Pytorch* library.

- CONV1: Kernel Size \rightarrow 5x5, Input Size \rightarrow 3, Output Size \rightarrow 32
- POOL1 : Kernel Size \rightarrow 2x2
- CONV2 : Kernel Size \rightarrow 5x5, Input Size \rightarrow 32, Output Size \rightarrow 64
- POOL2 : Kernel Size \rightarrow 2x2
- CONV3 : Kernel Size \rightarrow 5x5, Input Size \rightarrow 64, Output Size \rightarrow 128
- POOL3 : Kernel Size \rightarrow 2x2
- FC1 : Fully Connected Layer with 128 outputs
- FC2 : Fully Connected Layer with 30 outputs

1.3 Results

The model was trained on 80% of training data with the remaining 20% used for validation. *Adam* optimizer under cross-entropy loss was used to fit on the data. The results observed were as follows.

- Training Epochs : 10
- Time Taken : 16m 53s
- Training Accuracy : 96.33%
- Best Validation Accuracy : 14.2836% (at 4th Epoch)
- Test Accuracy : 10.9123%

2 Recurrent Neural Network

2.1 Objective

To implement and train a recurrent neural network on book titles and predict the the genre it belongs to among 30 different genres.

2.2 Architecture

Abidirectional RNN was implemented with the following architecture using the *Pytorch* library.

- Embedding Layer : Initialized with the vocabulary vectors from the pretrained *GloVe* embedding(`glove_6b_300d`).
- RNN layer : Hidden layer size \rightarrow 128, bidirectional \rightarrow True, batchfirst \rightarrow True
- MLP layers
 - FC1 : Fully Connected Layer with output 128
 - FC2 : Fully Connected Layer with input 128 and output 30 (number of classes)

2.3 Results

- Training Epochs : 30
- Time Taken : 115.04s
- Training Accuracy : 53.23%
- Test Accuracy : 45.1930%

3 Competitive Model Architecture

3.1 Convolutional Neural Network - VGG11

The pretrained VGG-11_BN network provided by Pytorch was used with last two fully connected layers left as trainable. The model was trained on 80% of training data with the remaining 20% used for validation.

Results

- Training Epochs : 25
- Time Taken : 58m 53s
- Training Accuracy : 80.01%
- Best Validation Accuracy : 26.8421%
- Test Accuracy(Non-Competitive) : 25.2982%

3.2 Recurrent Neural Network - BERT (Kaggle Submission)

BERT base model (uncased) was used, which is a pre-trained model on English language using a masked language modeling (MLM) objective. The model was fine tuned on the training data with the help of the Transformers Library by Hugging Face. For this task, the provided training and test set(non-competitive) were combined, used as the training data for the fine tuning process.

Results

- Training Epochs : 2
- Time Taken : 23m 59s
- Training Accuracy : 61.45%
- Test Accuracy(Competitive) : 61.146%