OCR
Optical Character Recognition

Taranpreet Kaur Bambrah

Introduction

- CRNN
- Handwritten words recognition
- IAM Handwritten Dataset

Model Architecture

Convolutional Recurrent Neural Network (CRNN) for OCR

Input: [32, 128, 1] (Image)

Conv Layers:

[Conv2D(64)] -> [MaxPool2D] -> [Conv2D(128)] -> [MaxPool2D] -> [Conv2D(256)] -> [Conv2D(256)] -> [MaxPool2D] -> [Conv2D(512)] -> [BatchNorm] -> [Conv2D(512)] -> [BatchNorm] -> [MaxPool2D] -> [Conv2D(512)]

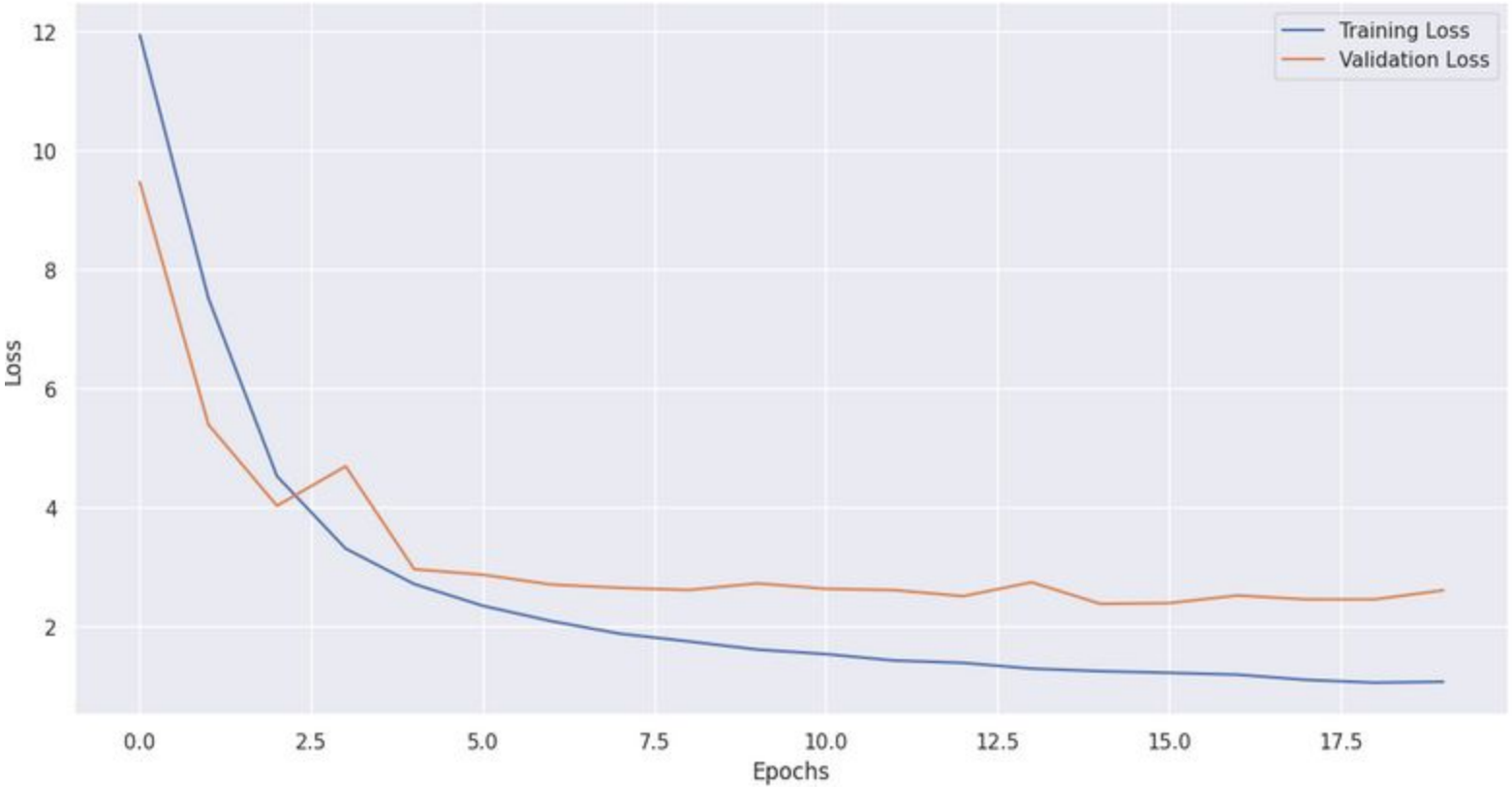
Bidirectional LSTM Layers:

[Bidirectional(LSTM(256, return_sequences=True, dropout=0.2))] -> [Bidirectional(LSTM(256, return_sequences=True, dropout=0.2))]

Dense Output Layer:

[Dense(len(char_set)+1, activation='softmax')]

Output: [Sequence of Predicted Characters]


Training Loss
Validation Loss
Loss
Epochs
12
10
8
6
4
2
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5

DEMO

Lessons Learned

- Loading the data

- data preprocessing

```
char_set = "!\"#&'()*+,-./0123456789:;?ABCDEFGHIJKLMNOPQRSTUVWXYZabcdefghijklmnopqrstuvwxyz"

print(char_set, len(char_set))
def encode_to_labels(txt):
    # encoding each output word into digits
    dig_lst = []
    for index, chara in enumerate(txt):
        dig_lst.append(char_set.index(chara))

    return dig_lst
```

Uploaded Image: Thank.png

Predicted Text: Thank

Uploaded Image: you.png

Predicted Text: you

Uploaded Image: exclamation.png

Predicted Text: !

PS: Everything was recognized correctly :)