Configuration Manual- Memotion 2.0 - Sentiment Analysis and Emotion classification of Memes

MSc Research Project
Data Analytics

Nilam Bhapkar
Student ID: X20145331

School of Computing
National College of Ireland

Supervisor: Mohammed Hasanuzzaman

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Nilam Bhapkar
Student ID:	X20145331
Programme:	Msc Data Analytics
Year:	2021
Module:	MSc Research Project
Supervisor:	Mohammed Hasanuzzaman
Submission Due Date:	20/12/2018
Project Title:	Configuration Manual- Memotion 2.0 - Sentiment Analysis and Emotion classification of Memes
Word Count:	756
Page Count:	4

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Nilam Bhapkar
Date:	16th December 2021

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST:

Attach a completed copy of this sheet to each project (including multiple copies).	☐
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	☐
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	☐

Assignments that are submitted to the Programme Coordinator office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Nilam Bhapkar
X20145331

1 Introduction

An idea of a configuration handbook is a document that provides a detailed step-by-step guide for the development of the project 'Memotion 2.0 - Sentiment Analysis and Emotion classification of Memes', as described in the technical paper. The objective of this document is to guide you through every stage of the process in order to acquire the desired result, that can be provided inside the technical document. The entire project is constructed using a variety of library, software configurations.

1.0.1 Project Overview

The goal of the study is to categorize the emotions elicited by memes. The dataset, Memotion analysis 2.0, was created by the Defactify workshop and published by codelab. In this classification task, recurrent neural network models, convolutional neural network models, and multilayer perceptrons are all employed to solve the issue.

2 Project Pre-requisites

The below are the necessary conditions: The following sections detail the software and hardware configurations. The GPU (Graphics Processing Unit) is necessary in order to develop the model on such a massive number of image samples.

2.1 Hardware Requirement

Figure 1 shows a the required system configuration to implement this project.

Device specifications

Device name	DESKTOP-P4715LQ
Processor	Intel(R) Core(TM) i5-8250U CPU @ 1.60GHz 1.80 GHz
Installed RAM	8.00 GB
Device ID	288E9D8E-3DD8-43D8-87F4-FB063FC31045
Product ID	00327-35812-76115-AAOEM
System type	64-bit operating system, x64-based processor
Pen and touch	No pen or touch input is available for this display

Figure 1: Required System Configuration

A web browser, specifically Chrome Browser edition 87.0.4280.88, was utilized for this experiment. Previous versions of Google Chrome did not work with the colab notebook feature. Third, this project was constructed using the Google Colaboratory (colab) IDE Pro edition, a premium cloud platform product from Google that allows for the use of several Artificial Intelligence (AI) libraries, as well as a powerful GPU and TPU to accelerate computation.

3 Project Implementation

3.1 Accessibility of data set

```
from google.colab import drive
drive.mount('/content/gdrive', force_remount=True)
root_dir = "/content/gdrive/My Drive/"
base_dir = root_dir + 'memotion_colab/'
```

```
Mounted at /content/gdrive
```

Figure 2: Access to Google Drive

Figure 2 shows access to google drive. Google drive is placed over google colab so that you can retrieve the data which is stored inside. We have accessibility to all of the drive's publicly accessible folders.

The following credentials can be used to gain access to training pictures included within a zip file.
Password: memetaskaaai22
Password (for images): memeimageaaai22

3.2 Prepossessing

Following the academic report's guideline, an IPYNB file named **Preprocessing** can be used as a starting step in putting into action the project's findings. At this point, the data has been cleansed, preprocessed, and transformed in preparation for its subsequent use as shown in Figure 3. Comma separated files (.csv) will be used to hold the final cleaned dataset.

```
# Remove regular english words like am, is, are and ...
stop = stopwords.words('english')
pat = r'\b(?:{})\b'.format('|'.join(stop))
train_df['ocr_text'] = train_df['ocr_text'].str.replace(pat, '')
train_df['ocr_text'] = train_df['ocr_text'].str.replace(r'\s+', ' ')
train_df
```

	Id	Image url	ocr_text	
0	1	https://preview.redd.it/9jkzlvt8p4q31.jpg?widt...	enters wrong class teacher students	
1	2	https://i.pinimg.com/originals/fd/c8/e2/fdc8e2...	dj single make nooooisssssseeee heartbroken ass	
2	3	https://i.pinimg.com/originals/85/10/13/851013...	everyone sit like lady	
3	4	https://i.imgur.com/07ZcrjZ_d.webp?maxwidth=52...	youre watch 2 hour educational video sheep sh...	
4	5	https://i.imgur.com/MGddJxr_d.jpg?maxwidth=520...	government work home olympic swimmers nsdf	ver

Figure 3: Data Prepossessing

3.3 EDA and Labeling

It is at this point Figure 4 that the file entitled **EDA_LABLING** will be created, which will accept as data the previously created comma separated file from the preceding phase. After putting the Google Drive in the notebook, we'll open the file with URLIB opener

```
# Create one unit csv file from all downloaded images


negatives_dataframe['overall_sentiment'] = 0
neutral_dataframe['overall_sentiment'] = 1
positive_dataframe['overall_sentiment'] = 2

memes_dataframe = pd.concat([negatives_dataframe, neutral_dataframe, positive_dataframe], axis=0, ignore_index=True)
memes_dataframe = memes_dataframe.sample(frac=1).reset_index(drop=True)
memes_dataframe = memes_dataframe.loc[:, ~memes_dataframe.columns.str.contains('^Unnamed')]
memes_dataframe
```

```
memes_dataframe.to_csv(Base_Path+'train.csv')
```

```
train_dataframe = pd.read_csv(Base_Path+'train.csv')
train_dataframe = train_dataframe.loc[:, ~train_dataframe.columns.str.contains('^Unnamed')]
train_dataframe = train_dataframe.dropna(subset=['text'])
train_dataframe = train_dataframe.sample(frac=1).reset_index(drop=True)
```

```
train_dataframe.to_csv(Base_Path+'train_dataframe.csv')
```

Figure 4: EDA and Labeling

to read. The following settings will allow us to download and collect the label of each picture. This method will collect and label each image, which will then be divided into three different folders, such as Negative, Neutral, and Positive. Following that, all of the data obtained is saved in a comma-separated file with the overall sentiment value (such as 0 for negative, 1 for neutral, and 2 for positive) disguising the overall sentiment value. A correlation matrix is created in order to verify the relationships between the features.

3.4 Model Developement

To begin model implementation, the Model Experiment file can be used as a starting point. After importing all of the necessary libraries and mounting Google Drive, feature extraction will be performed on the data that has been provided. The feature extraction

Dump data from drive

```
xs_images=''
with open('/content/gdrive/My Drive/memotion_colab/xs_images.pkl','rb') as f:
  xs_images= pkl.load(f)
xs_texts=''
with open('/content/gdrive/My Drive/memotion_colab/xs_texts.pkl','rb') as f:
  xs_texts= pkl.load(f)
xs_categoricals=''
with open('/content/gdrive/My Drive/memotion_colab/xs_categoricals.pkl','rb') as f:
  xs_categoricals= pkl.load(f)
ys=''
with open('/content/gdrive/My Drive/memotion_colab/ys.pkl','rb') as f:
  ys= pkl.load(f)
```

Figure 5: Data from pkl file

procedure can take several hours to complete. As a result, the pkl file created during the feature extraction phase as shown in Figure **??** can be used throughout the model building stage. A baseline network as shown in Figure 6 has been constructed using Keras functional APIs and a CNN that has been pre-trained using the VGG16 model. Model1 will be built on top of the architecture described in the text. In the same way, a different model has been developed and evaluated using evaluation metrics. Also plotted are graphs for each model's accuracy and loss score, which are both measured in percent.

Building up the model Now we create our baseline network using keras functional APIs and pre-trained CNN using VGG16 model. Model1 will base model with the given architecture.

```
from keras.layers import Input, Embedding, LSTM, Dense, Conv2D, MaxPool2D, Flatten, concatenate
from keras.models import Model


embedding_layer = Embedding(15364  + 1,
                            100,
                            weights=[embedding_matrix],
                            input_length=MAX_SEQUENCE_LENGTH,
                            trainable=False)

sequence_input = Input(shape=(MAX_SEQUENCE_LENGTH,), dtype='int32')
embedded_sequences = embedding_layer(sequence_input)
lstm = LSTM(300, return_sequences=True)(embedded_sequences)
lstm = LSTM(300, return_sequences=True)(lstm)
lstm = LSTM(300)(lstm)

categorical_input = Input(shape=(4,))
x = Dense(64, activation='relu')(categorical_input)
x = Dense(128, activation='relu')(x)
x = Dense(64, activation='relu')(x)
x = Dense(32, activation='relu')(x)
```

Figure 6: Model Building

References