Reproducing

“Using millions of emoji occurrences to learn any-domain representations for detecting sentiment, emotion and sarcasm”

Members:

Ajit Kumar Chaudhary

Avinash Pandey

Moiz Ahmed Khan

Saroj Bogati

Table of Contents

[Abstract 3](#_Toc55540507)

[Brief description of the source paper and justification 3](#_Toc55540508)

[Evaluation Framework 3](#_Toc55540509)

[Justification 3](#_Toc55540510)

[Description of original dataset 4](#_Toc55540511)

[Replication of original work 4](#_Toc55540512)

[Issues of original work and resolving those 5](#_Toc55540513)

[Construction of new data 6](#_Toc55540514)

[Twitter API 6](#_Toc55540515)

[Results on new data 7](#_Toc55540516)

[Any other reflections 7](#_Toc55540517)

[References 7](#_Toc55540518)

# Abstract

Reproducibility, a similar act as replicability, performs as a repetitive method of proving correctness, or reaching to similar suppositions as the expected results. When a published result is replicated with the usage of data, codes as well as papers delivered by the authors leading to expected results or similar, a research is said to be reproducible. Not just as a justification, but it also facilitates us as a revelation of research roadmaps as well as the transmission of knowledge. Here in this report, we will explain about our reproducible research based on a paper that used millions of emoji occurrences to study any domain illustrations in order to detect sentiment, emotion and sarcasm underlying.

# Brief description of the source paper and justification

Breaking down the probable aspects taken into consideration for the reproduction of research paper, our reproducible research doesn’t seem to be much intricate or unmanageable. While going through the process of research paper selection, there is a necessity of proper consideration of its impression, overall arrangement that clarifies work flow, management of data and lastly the specific phases for reproduction as well as essential aspects.

<<<<<<<<<<<START>>>>>>

It is published at EMNLP and globally ranked 1st in Sarcasm detection. It uses 1246 million tweets containing about 64 common emojis on 8 benchmark datasets to predict sentiment, emotion, and sarcasm.

<<<<<<<<<<<<<<<<<<<<<<<<<<<PICTURE>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<AROUND 150 WORDS>>>>>>

## Evaluation Framework

<<<<<<<<<<<<AROUND 200 WORDS>>>>>>>>

## Justification

<<<<<<<<<<<AROUND 150 WORDS>>>>>>>>>

# Description of original dataset

<<<<<<<<<<<<<<<<<<<<<<<<<<<PICTURE>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

<<<<<<<<<<<<<<<<<<AROUND 200 WORDS>>>>>>>>>>>>>>>>>>

In the paper, there are 8 benchmark datasets.

Olympic: Negative and high control, positive and high control,

negative and low control, positive and high control.

PsychExp: joy, fear, anger, sadness, disgust, shame, guilt

SCv1: not sarcastic, sarcastic

SCv2-GEN: not sarcastic, sarcastic

SE0714: fear, joy, sadness

SS-Twitter: negative, positive

SS-Youtube: negative, positive

kaggle-insults: neutral, insult

# Replication of original work

The concept of empirical generalization greatly implies on any research being able to replicate. For our approach of replication, duplication or even extending the original work, there are sufficient tools available that facilitate us on the process. However, prior to the initiation, a good plan is needed that ease us with rectifying and evaluating necessary changes on the original work. One of our major goals in this project is to properly replicate TorchMoji model and ensure it runs properly as done in the original work. This model will help us to understand how language has a usage of expressing emotions. Likewise, with the usage of transfer learning, the model will help us to obtain state of art performance on many text modeling tasks related to emotions. With reference to original work, we were able to successfully install and run source code using torchmoji model. Following to that, TorchMoji model is trained and tested among various tweet texts that we fetched as a new dataset.

<<<<<<<<<<<<<<<<<<<<<<<<<<<FIGURE>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

The original work consisted of emoji as well as score prediction. Emoji prediction consists of predicting 5 most relevant emoji based on the sentences. Similarly, Score prediction consists of accuracy score of the emoji in relevant to the sentences. Likewise, the original work had 8 benchmark datasets and we could only replicate 3 datasets out of them. Because it took approximately 12 hours for us to replicate those 3 datasets. Even though, we tried evaluating for these 3 datasets named as Twitter, YouTube and Sarcasm dataset, our main aim was to check and implement this model for Twitter dataset, since we succeeded to create a new dataset from tweets fetched from twitter using python libraries. Correspondingly, we made usage of 1000 epochs along with each epoch having a sample size of 1000. The metrics used here is accuracy score along with the appropriate labels. Also, there is a usage of fine tune model under torchmoji to facilitate us with the execution part which all comes under last method. Among benchmark datasets that we executed; we got the following accuracy scores:

<<<<<<<<<<<<<<<<<<<<<<<<FIGURE>>>>>>>>>>>>>>>>>>>>

## Issues of original work and resolving those

The original work is implemented in Python programing language along with PyTorch, NumPy, and Scikit-learn as dependencies. To execute the python files, there was need of running command line arguments along with the adjustment of dataset size, location and other attributes. Overall work was performed on Google Colaboratory with the creation of Jupyter Notebook, since the datasets are relatively large and also won’t descend performance of the local machine while program execution. Another great advantage of Google Colaboratory is that, there is no need of installing any modules for running the code since modules are already installed within it. Using Google Colaboratory made it convenient for us to share the notebook file within team members. The below mentioned issues are addressed while execution:

* The model took quite long time to train (approximately 12 hours to train only 40% of the dataset). Due to this reason, we were only able to train 3 datasets.
* There were lots of bugs on the original code, which required manually fixes. Also, we referred to common issues and comments on the GitHub code repository to fix those.
* The accuracy score didn’t match with the paper as we were unable to run the complete dataset.
* The original data file was in .pickle, which consist of texts, labels, validation and testing data. To address this occurrence, we made some changes in the script to load texts and labels for performing evaluation.

<<<<<<<<<<<<<<<<<<<<FIGURE IF APPLICABLE>>>>>>>>>>>>>>>

# Construction of new data

Precisely, there is need of a reproducibility so as to aid the crisis of reproducibility. Replicability, can also facilitate to fulfill this gap, where there is a requirement of replication of original research paper along with the code, followed by the achievement of similar results, however on a changed set up. Likewise, the data can either be downloaded from any public source or be scrapped using certain tools and techniques. Here in this project work, we fetched our dataset through specific techniques and procedures of raw data extraction. Extracting raw tweets data from Twitter’s database has been our major source of forming a dataset.

## Twitter API

With known fact of Twitter’s massive database, it has always been one of the major hubs for data extraction. The process initiates with the usage of Twitter API and tweepy library, thanks to Python libraries that made us easier to access and analyze data from the twitter. To be more specific, we have extracted 3200 tweets of Bill Gates from his twitter account using these libraries. One of the prominent reasons to extract this dataset was due to its characteristics of consisting various tweet texts written by the same user. Likewise, the dataset will also ease us with variance and affirmation, since we can evaluate the predictions of 5 most relevant emojis based on each sentence extracted from various tweets.

<<<<<<<<<<<<<<<<<<<<<<<<<<<FIGURE>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

Following to that, there was a need of data cleaning process in order to get clean texts from raw tweets, excluding hashtags, mentions and other elements from the dataset. Similarly, we also divided tweet texts into each separate sentence with the purpose of getting more specific results on relevant emojis. Correspondingly, after cleaning the texts, we generated labels with positive and negative sentiments using TextBlob library. Besides, we also converted paragraphs of a tweet to several sentences, resulting to 5515 rows of dataset.

<<<<<<<<<<<<<<SHORT DESCRIPTION ON SUBJECTIVITY, POLARITY AND LABEL>>>>>>>>

# Results on new data

<<<<<<<<<<<<<<<<<<<<<<<<<<<FIGURE>>>>>>>>>>>>>>>>>>>>>>>>>>>>>

Evaluation of new data:

We have performed evaluation using finetune of torchmoji model.

We have received 92.20% accuracy on the new data.

# Any other reflections

# References