Machine Learning Algorithm Predicts The Popularity Of A Meme

Robin Leman

MAIS 202 - Winter 2021 - Deliverable 1

Abstract

In this experiment, a machine learning algorithm will try to predict "how funny" a meme is. By training the algorithm on a dataset of memes from Reddit, the machine will be able to learn the popularity of a meme based on its number of upvotes. Then, it will be able to predict how funny an original meme is by returning the number of upvotes the machine thinks the meme could receive.

1 Introduction

Attributing human emotions to machines, and machines being able to read human emotions[2], has been a great focus in Machine Learning research. In this experiment, we will try to teach a machine learning algorithm a well known human emotion: humour.

2 Methods

To teach an algorithm humour, we need two types of data: funny instances, and not so funny instances. The best way to obtain this data is through the internet, and more specifically, through Reddit[5]. Reddit is a social media where, between others, users can post memes. A meme is the combination of an image and a caption that aims, and often fails, to be funny. On Reddit, users can upvote a meme if they find it funny, and downvote it if they don't. Thus, the popularity score of a meme on Reddit is given by #number of upvotes − #number of downvotes. We will consider the popularity score of a meme as a rating of how funny the meme is.

The model will thus train on a data set of tuples of an image and a caption, and a popularity score. Then, it will be able to predict from an input of an image and a caption an integer output representing its popularity score.

A frontend will be implemented to upload an original meme and compute its predicted popularity score. This score represents how funny the machine thinks the meme is. By giving a meme to the algorithm, and by returning this kind of score, we are in a way making a machine laugh.

3 Dataset

The dataset used is a combination of scrapped memes from Reddit with useful information like the author, the number of upvotes and downvotes, a link to the image, etc. The data set can be found on Kaggle at `https://www.kaggle.com/sayangoswami/reddit-memes-dataset`.

The dataset contains 3327 files labeled with their number of upvotes. It thus applies to our experiment, as our goal is not to predict the meme[3] but it's popularity score.

We will divide our data set between a training set and a validation set. If needed, and if time permits, more data could be added to our model by directly scrapping Reddit. This makes the scope of our dataset

almost infinite. We will also need to use processing techniques like optical character recognition to extract the meme caption from the image[1] and analyze both.

4 Model

The model used will be a convutional network using a combination of text analysis, for the meme caption, and computer vision, for the image itself. We are dealing with a regression problem, probably with the Mean Squared Error as the evaluation metric.

5 Discussion

Because Reddit counts billions of users, the meme communities can be quite large[4] and the popularity score gives us a significant average of how funny a meme is. However, the popularity score of a meme can depend on a lot of factors, such as time posted, user followers, subreddit, etc..[6]. Thus, our model will not be able to accurately predict the actual number of upvotes the meme would get if posted on the platform. Our model will also be biased since we did not take those parameters into account during training. An analysis must be done by comparing our method to existing research, and if time permits, correct it by normalizing our data to account for these factors.

Secondly, humour itself can be very subjective. The popularity score of the Internet might not be what each individual might find funny. In our prediction phase, we thus need to interpret our result as something that the internet might find funny and not as something that we find funny ourselves.

References

[1] Barnes et al. "Dank or Not? – Analyzing and Predicting the Popularity of Memes on Reddit". In: (2021).

[2] *Emotion AI, explained.* URL: https://mitsloan.mit.edu/ideas-made-to-matter/emotion-ai-explained.

[3] *Meme Text Generation with a Deep Convolutional Network in Keras Tensorflow.* URL: https://towardsdatascience.com/meme-text-generation-with-a-deep-convolutional-network-in-keras-tensorflow-a57c6f218e85.

[4] *r/memes, Reddit.* URL: https://www.reddit.com/r/memes/.

[5] *Reddit.* URL: https://en.wikipedia.org/wiki/Reddit.

[6] Fredrik Wigsnes. "Predicting popularity of Reddit posts using machine learning". In: (2019).