Team Rubeosaurus:

Original Proposal

Abstract

Cat Meme Generator!! Our application will allow you to turn your cat (or any other cat) into a meme. It has two ML components. First, a classifier to classify the cat image as 'happy', 'sad' , 'sleepy' etc. And second, sentiment analysis of the meme text. The application would be built for the Android environment. The user will be prompted for an input. This will work both ways. The input can be an image. In that case, a meme will be created by classifying the image and then choosing from a bunch of meme text we already have based on the classification. But the user can also input a String. In that case, we'll carry out sentiment analysis of the text and choose a Cat Image we already have that would go well with the sentiments of the text. In both cases, the user gets their cat meme.

ML models

For the image classification task, we'll be using ResNet18 to follow a very similar approach to the one in Assignment 2. For the text classification part, we'll use torch embeddings and classify text based on sentiments. We may have to merge similar sentiments together but that's a design choice for later.

Data / Sources

For the image classification task, we'll be constructing a dataset using Selenium. Selenium is an easy way to extract images from google. We may have to manually remove some 'bad' data. We expect we'll need about 300 images for every category and augment if required. For the text classification task, we'll be using this dataset on Kaggle : https://www.kaggle.com/praveengovi/emotions-dataset-for-nlp

For the captions we have 2 datasets here and here, which we need to filter to fit our project and classify using our model.

Non-ML Component

For the Non-ML component, we will be using Android Studio. Android Studio is a free and open source IDE developed by Google for developing Android mobile apps and other Android-based multitouch application software with a natural user interface. It allows for easy ML model integration and better performance using dedicated tools for mobile development like PyTorch Mobile. The IDE also provides us with tools for Debugging the App in real time which is a huge plus.

Related Work

A google search for "cat meme generator" or "meme generator" shows me a bunch of image editing tools. So, maybe this is the first ML based meme generator?

Work Distribution

Since each problem has its own challenges, we feel it would be better if two people work on a task at a time and then move on to the next. We will distribute duties in a way that allows everyone to work in pairs on the four components :

1. Image classifier
2. Text Classifier
3. Generating meme output
4. Creating the user interface.

Milestones

Milestone 1 : Getting decent results for the Image classifier + Creating a basic app using kivy
Milestone 2 : Getting decent results for the Text Classifier and generating meme output from input
Milestone 3 : Getting somewhat functional user interface
Milestone 4 : The complete project

Risks / Backup

- There are no restrictions for text inputs. This means our classifier may not work very well with the input if it has words outside our vocabulary. Anyway, we'll get some meme output that'll make us happy.
- If the Android App does not seem to integrate well with the PyTorch model, we can create a web-app.