Detecting Mental Health Incidents & Suicidal Thoughts from Text

DATA 607
August 20th, 2024
By: Richie Rivera

Summary & Motivation

According to Harvard's News and Research, it is expected that half of the world will experience a mental health disorder[1]. As these disorders become more complex and difficult to diagnose, it's apparent that a way to help expedite the process of diagnosis and treatment of people afflicted.

In this project, I attempt to build one such engine using fastai, a high-level Python library that simplifies building and training deep learning models, offering user-friendly tools and advanced features on top of PyTorch.

[1] https://hms.harvard.edu/news/half-worlds-population-will-experience-mental-health-disorder

Data Source

To begin this project, I have used the Mental Health Corpus[2], a labeled dataset which labels sentences scraped from comments as “poisonous” where poisonous is defined as indicative of anxiety, depression, and other mental health issues.

[2] https://www.kaggle.com/datasets/reihanenamdari/mental-health-corpus

Procedure

1. Firstly, I installed the Nvidia's CUDA toolkit. This toolkit provides the ability to use my local GPU with PyTorch to accelerate the task of using fastai to train a deep learning model.
2. Using fastai, I was able to use the TextDataLoader to build a model that split the data into an 80% training and 20% holdout testing dataset[3].
3. Iterating through multiple epoch numbers, I was able to settle on 3 epochs to train the model[3].
4. Exporting the model, I've used gradio to build an online front end which takes in any text and provides a score of how "poisonous" it is.
5. With the model built, a gradio front end was built and deployed onto a hugging face space[4] where it is possible to utilize the model online via api or web frontend with novel text.

[3] https://github.com/riverar9/cuny-msds/blob/main/data607/projects/final-project/build_mental_health_model.ipynb

[4] https://huggingface.co/spaces/riverar9/minima

Results

A model was built using fastai and deployed onto a hugging face space. This model's purpose is to categorize any input text as "poisonous" or "non-poisonous".

Bonus Project

Scraping Restaurant data for NYC's Restaurant Week

Summary & Motivation

In New York City, there is a time called Restaurant week where participating restaurants will create a price fixe menu which typically includes 2-3 courses that sample some of their highest esteemed dishes. For this, it's possible to use the nyctourism[5] website to find restaurants but the limitations on filtering makes it difficult to find suitable entries.

Because of that, I took it upon myself to try to do it better which started with simple data collection.

[5] https://www.nyctourism.com/restaurant-week/

Data Source

For this project, I used the nyctourism[5] website and scraped data from web pages.

[5] https://www.nyctourism.com/restaurant-week/

Procedure

1. To scrape this data, I began by navigating the pages manually to identify patterns.
2. With these patterns isolated and documented, I installed selenium and GetChromeDriver python packages
3. Using GetChromeDriver, I was able to then launch selenium and open all of the restaurant pages.
4. With these capabilities, I've chained together a procedure to open the root page on the website and then iterate over each restaurant on that page. Finally, scraping the restaurant information from each restaurant page and saving the information in to a json file[6].

[5] https://github.com/riverar9/2024-restaurant-week

Results

A json file under the root directory was created and it contains the below attributes over each restaurant:

- Name
- Address
- Description
- Restaurant Week URL
- Restaurant URL
- Deal Types
- Participating Weeks