DOCUMENTATION

Model Used: Pre-trained GPT2 model

Advantages of GPT2:

1. Generates text considering the sentiments of original text.
2. Capable of generating texts from headlines.
3. Easy to fine tune.
4. Reasonable size considering the number of parameters and free to use.

Reasons for not using a custom model:

1. Although I found a dataset of sentences, it was too huge to train the model.
2. Lack of access to GPU and sufficient memory for training.
3. I tried training it on a dataset of around 440k+ sentences but a single epoch took around 12 hours.

Basic Flow:

1. User inputs sentences on the interface along with the text length required. It is imperative to keep the input text length sufficiently high enough (3-4 sentences recommended) so that the model can predict the sentiment better.
2. Sentence is encoded and passed on to the model.
3. It takes some time for the model to load if you are using it for the first time.
4. Once loaded, the model processes the input sentence.
5. Output sentence is printed on the interface once the process is complete. (might take long depending on the input sentence and length required)

Files in the github repo:
App.py: Code for the basic streamlit interface for a user-friendly front end experience
Text.py: Implements the model in the backend
Getmodel.py: WIll help you download the GPT2 pre-trained model and save it in the directory.

Steps to run:

1. Open command prompt in the repo folder. First type “python getmodel.py”.
2. Once the model download is completed. Type “streamlit run app.py”.
3. The localhost would open in chrome and you will see the interface.
4. Then you can type in the inputs.
5. Click on generate.
6. Wait….
7. Your output will be ready.

Some dependencies/libraries:
Python - 3.11.4
Transformers
Streamlit
Tensorflow