User Manual

Dependency / Package	Version / model
Python	3.9.11
Django	~=4.2.2
Django REST Framework	~=3.14.0
transformers[tf-cpu]	TensorFlow CPU version
Pre-trained model	nlptown/bert-base-multilingual-uncased-senti ment

Project Guideline:

- Clone/ download the repository from the following link: https://github.com/Mehedi-909/sentiment-analysis-hugging-face
- Open the project folder "sentiment_analysis" with a suitable IDE such as PyCharm/ VS Code
- 3. Open a new terminal window
- 4. Create a virtual environment with the following command:
 - # Windows OS
 - > py -m venv venv
 - # Unix/ Mac OS
 - > python -m venv venv
- 5. Activate the virtual environment using the following command for Windows OS:
 - # Windows OS
 - > venv/Scripts/activate
 - # Unix/ Mac OS
 - > source venv/bin/activate
- 6. Then install the necessary dependencies using the command: pip install -r requirements.txt
- 7. After the successful installation, run the following command to run the server: python manage.py runserver

Now the project is running on the localhost server.

You can now hit the url: http://localhost:8000/analyze in a browser to access the browsable API or test it through Postman in the following way:

Send a POST request with request body in JSON format s follows:

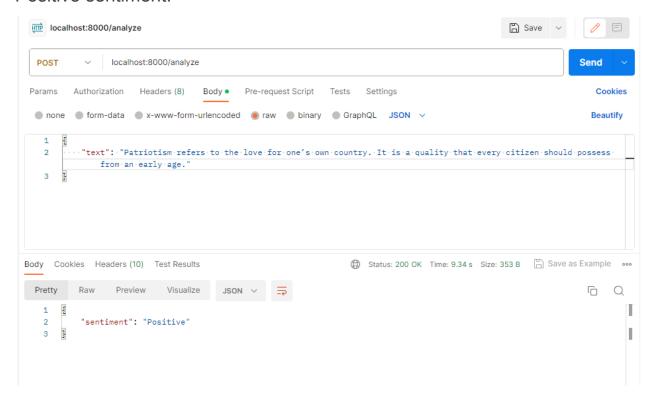
```
"text": "Your text to be analyzed"
}
```

2. Then wait for a while to get the response. The response will appear in the following format:

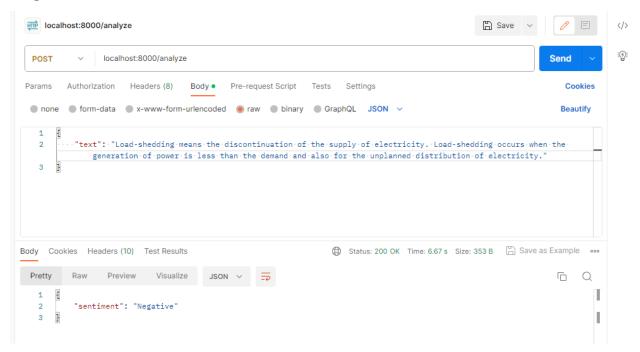
```
{
    "sentiment": "Positive / Negative / Neutral"
}
```

Here some examples are given both from Postman and browsable API:

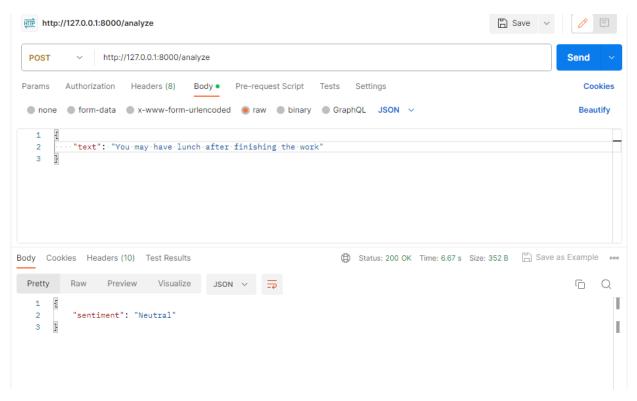
Positive sentiment:



Negative sentiment:



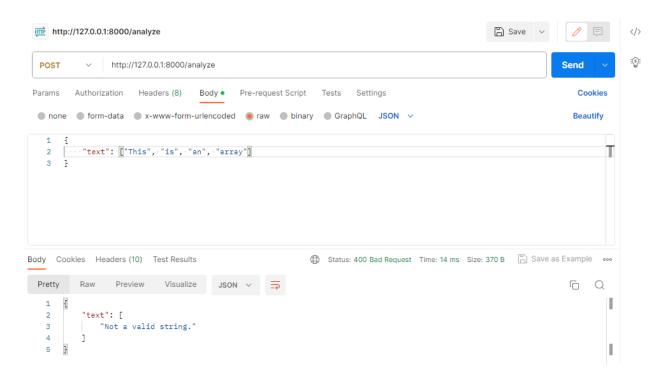
Neutral sentiment:



In my project, I have set a maximum limit of 500 characters for the input text. So the system gives an appropriate response if the text is above 500 characters.



It also provides a proper response if the input is not a valid text:

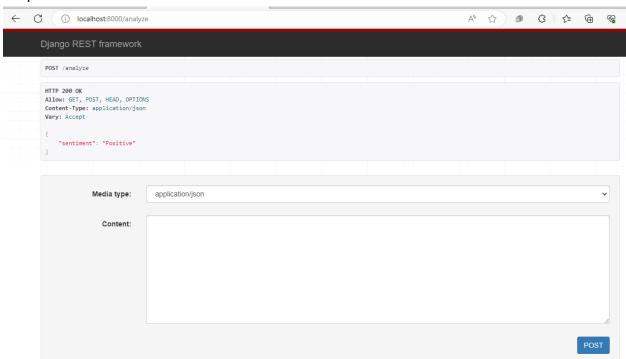


Here are some examples using the browsable API in web browser:

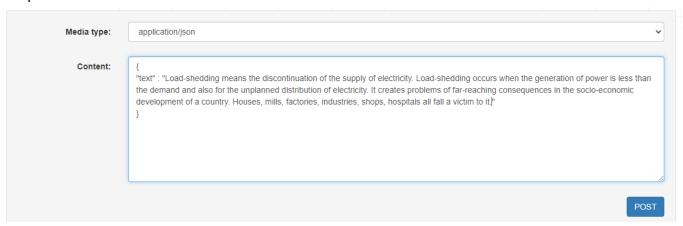
Request:



Response:



Request:



Response:

```
Api

POST /analyze

HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

{
    "sentiment": "Negative"
}
```