

Large Language Model (LLM) for Sentiment Analyzing and Image Reasoning in New York City street view

What is LLMs?

Large Language Models (LLMs) are advanced artificial intelligence systems designed to understand, generate, and manipulate human language on a large scale.

Using LLMs, we can analyze how people feel about different street views in New York City, linking emotions to urban space design.

Top 10 applications of large language models



1. Content generation
2. Translation and localization
3. Search and recommendation
4. Virtual assistants
5. Code development
6. Sentiment analysis
7. Question answering
8. Market research
9. Education
10. Classification

Project summary

- Build a lightweight system using an LLM (ChatGPT)
- Implement two key functionalities:
 - Image-based reasoning through textual description
 - Sentiment analysis of text inputs

To obtain the API key, navigate to <https://platform.openai.com/account/api-keys>.

- free usage limit of the OpenAI API was not sufficient for this project, so I spent around \$20 to access additional API credits and complete the assignment.
- Access to the OpenAI API may be blocked on certain restricted networks (e.g., school Wi-Fi), requiring workarounds like VPNs or mobile hotspots.

Task 1: Image-Related Reasoning

- Instead of processing Street View image directly, publicly available image URLs are used.
- Text prompts describing the Street view image URLs are fed into the LLM.
- The model generates descriptive text based on the imagined content of the images, simulating a simple form of visual reasoning.

Ask the model to describe what it sees

```
async_resp = await openai_llm.astream_chat(messages=[msg])
async for delta in async_resp:
    print(delta.delta, end="")
```

1. A narrow city street lined with tall buildings on both sides. The road has "STOP" painted on it, and there are a few pedestrians walking on the sidewalk. Several cars are parked along the street, and American flags are visible on the right side.

2. A wide city street with tall brick buildings on both sides. The street is busy with cars and pedestrians, and the sky is overcast with clouds. The buildings have multiple stories, and the street is lined with trees and traffic lights.

Task 2: Sentiment Analysis

- The system takes a text input about description of street view images from the user.
- The text is sent to the LLM with a carefully designed prompt asking the model to analyze its sentiment.
- The model returns a result classifying the input as positive, negative, or neutral, along with a brief summary.


```
input_text = "This street feels narrow, busy, a bit stressful with tall buildings surrounding the pedestrians"  
result = generate_summary_and_sentiment(input_text, api_key)  
print(result)
```

```
{'summary': 'The street is narrow and busy with tall buildings creating a  
stressful environment for pedestrians.', 'sentiment': 'Overall, the  
sentiment of the text is negative. Words like "narrow," "busy,"  
"stressful," and "surrounding" evoke a sense of discomfort and unease. The  
mention of tall buildings further adds to the feeling of being'}
```

```
text = "This avenue feels open, bright, and lively with wide sidewalks and beautiful views of the sky."
```

```
sentiment = response.choices[0].message.content  
print(sentiment)  
if sentiment == "Positive":  
    print("😊")  
elif sentiment == "Negative":  
    print("😞")  
else:  
    print("😐")
```

Positive 😊