

Code Comment Generator

Comments in coding are lines within the source code that are not executed by the compiler but serve as human-readable explanations or documentation for the code. They play a crucial role in enhancing code comprehension, which helps the developers understand the purpose, logic and intricacies of a particular piece of code.

The need for comment arises from the inherent complexity of software development, where codebases can become complex and challenging to understand, especially for collaborators or future maintainers. Comments contribute to the overall readability and maintainability of code, facilitating collaboration, easing the onboarding process for new developers, and streamlining the code review and debugging phases.

Despite the obvious importance of code comments, many developers do not consistently comment on their code for various reasons,

- Sometimes, tight deliverable deadlines might lead developers to prioritize coding speed over documentation.
- Assuming that the code is self-explanatory or will only be maintained by the original author can contribute to insufficient commenting.
- Sometimes, developers might find it challenging to strike a balance between the number of words needed for a comment.

Ultimately, the reasons for inadequate code commenting often stem from a combination of time constraints, communication gaps, and varying individual perspectives on the perceived value of thorough documentation.

A code comment generator helps create relevant comments for a given code, which helps the developers maintain consistent and well-commented codes throughout the codebase. This tool offers benefits such as improved documentation quality, reduced developer workload, enhanced collaboration, and adherence to coding standards, ultimately leading to more efficient and sustainable software development processes.

The steps to build the app are:

Step 1: Create prompts

Step 2: Integrate the prompts with the Cohere generate model

Step 3: Build basic Streamlit components for UI

Step 4: Run your application and deploy

Read on for more details on each of these steps.

Building a Code Comment Generator

Step 1: Install and import Cohere and Streamlit

If you are using Python, we can install Cohere and Streamlit using the following pip commands

```
pip install cohere  
pip install streamlit
```

Import them into your app.py file

```
import cohere  
import streamlit as st
```

Step 2: Set your Cohere API Key in app.py or you can also use the env file to save your key

```
co = cohere.Client('Enter your COHERE API KEY')
```

Step 3: Get the code as a user input

```
user_input = st.text_area("Enter your code here")
```

The code that the user enters here will be the input code for which a comment will be generated.

Step 4: Add information about comments along with the user input code so that the cohere model can understand comments and give out a proper output

```
add_info = '''  
In computer programming, a comment is a programmer-readable  
explanation or annotation in the source code of a computer program.  
They are added with the purpose of making the source code easier for  
humans to understand, and are generally ignored by compilers and  
interpreters.  
  
give a comment line for this code and nothing more  
'''  
prompt = user_input + add_info
```

Step 5: Give prompt as input to the cohere command model

```
if st.button('Generate Comment'):  
    response = co.generate(  
        model='command',  
        prompt= prompt,  
        max_tokens=1376,  
        temperature=0.9,  
        k=0,  
        stop_sequences=[],  
        return_likelihoods='NONE'  
    )
```

Step 6: Display the Comment generated

```
st.text('Comment: {}'.format(response.generations[0].text))
```

To run this code on your local machine use the command in your terminal

```
PS C:\project path\project folder> streamlit run app.py
```

You can now view your Streamlit app in your browser using the following Local URL:

<http://localhost:8501>

