Topic 2: Assignment 1: Hugging Face Model Exploration

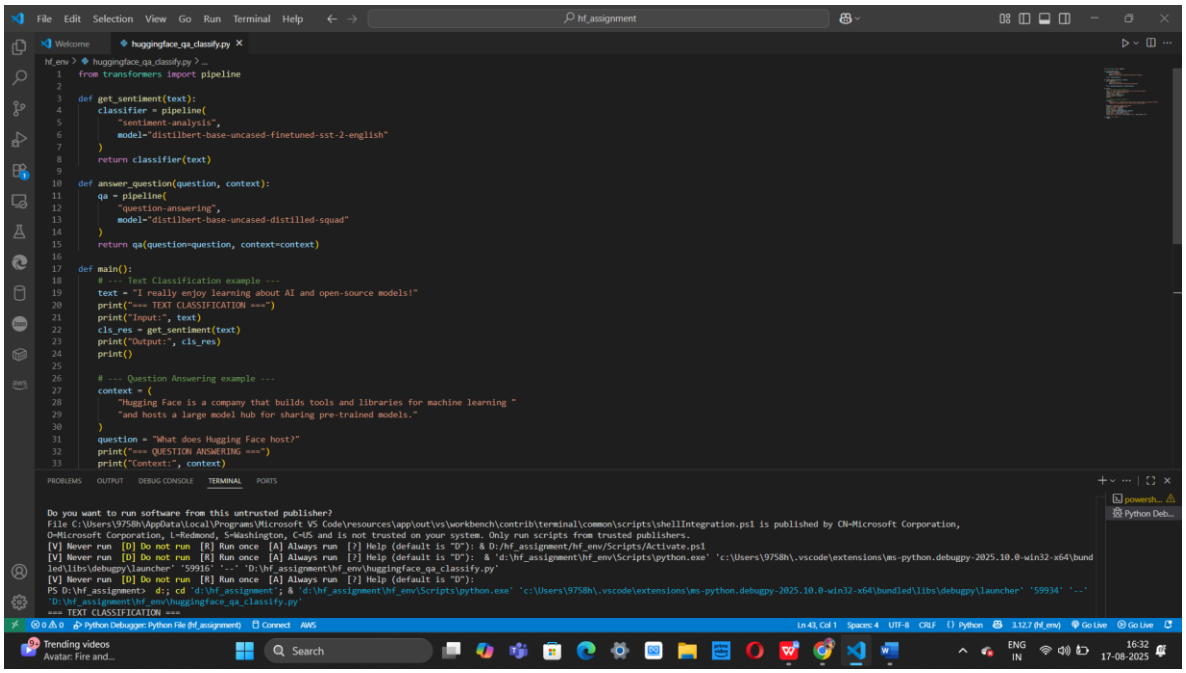
1. Introduction

The objective of this assignment is to explore an open-source large language model (LLM) from the Hugging Face Hub, load it locally using Python, and perform a simple NLP task such as text classification or question answering.

2. Environment Setup

The following commands were used to create a Python virtual environment and install the required libraries:

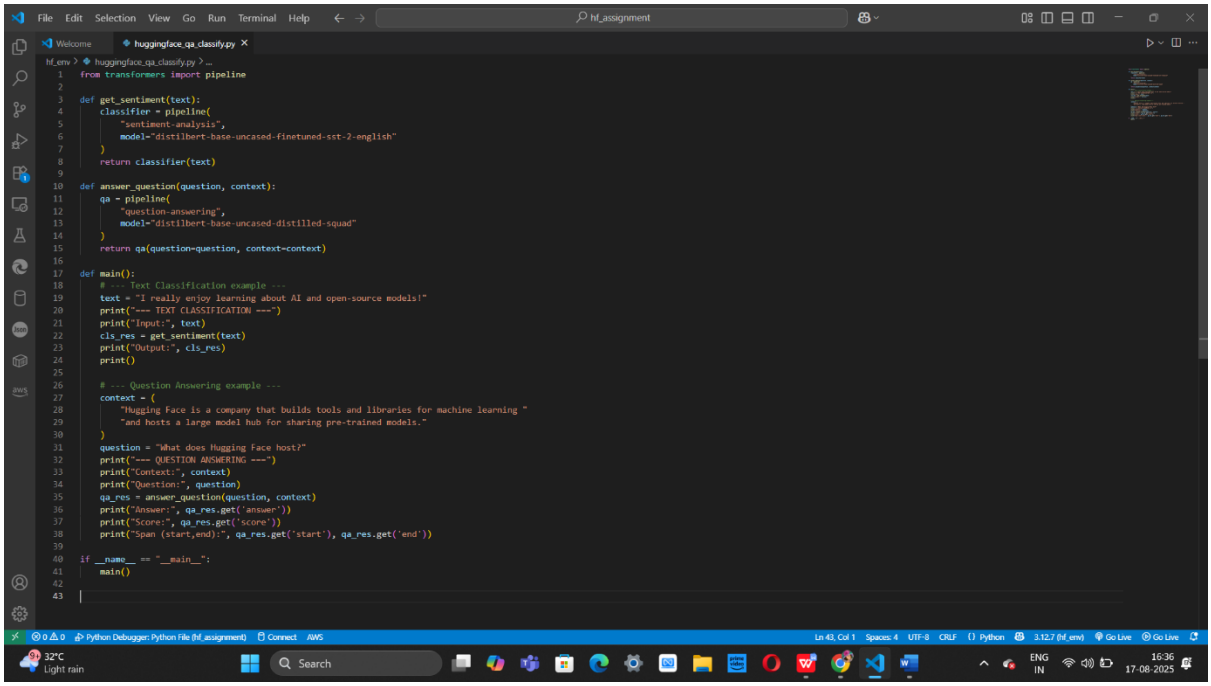
```
# Create a virtual environment

python -m venv hf_env



# Activate the environment

# Windows PowerShell

.\hf_env\Scripts\Activate.ps1

# Windows CMD

.\hf_env\Scripts\activate.bat
```



3. Model Selection

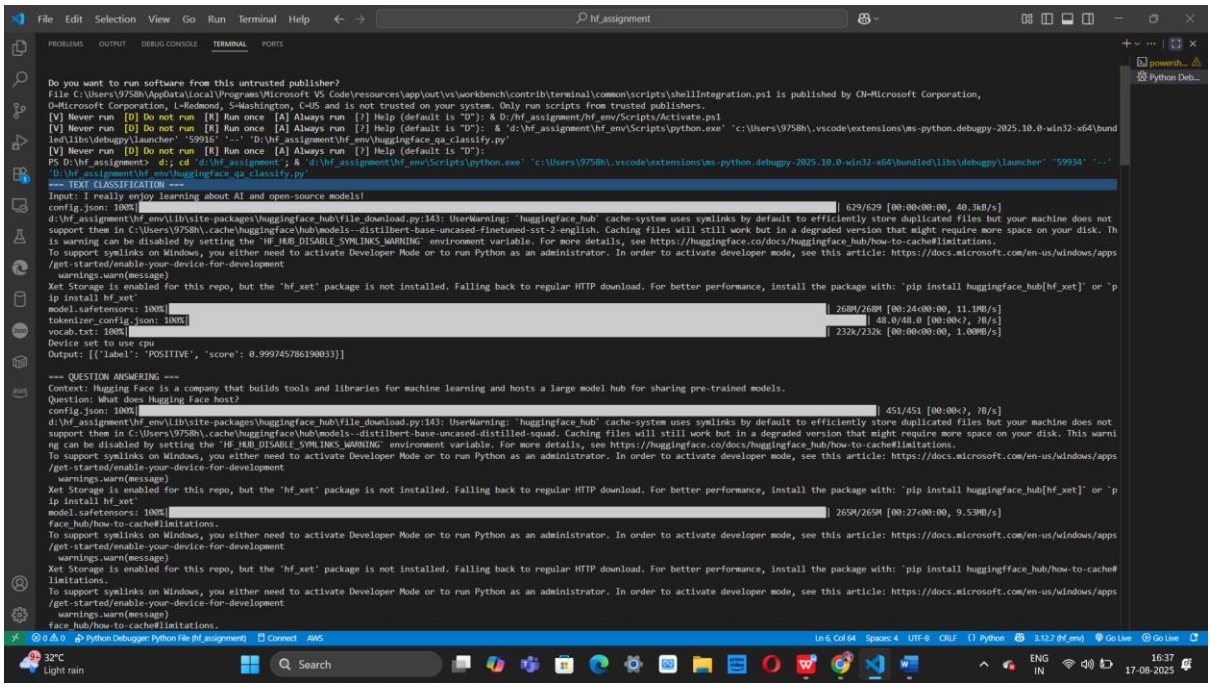
For this assignment, the “distilbert-base-uncased” model was chosen because it is a lightweight, fast version of BERT suitable for tasks like text classification and question answering.

4. Python Code – Model Loading & Task Execution



```python
from transformers import pipeline

def get_sentiment(text):
    classifier = pipeline(
        "sentiment-analysis",
        model="distilbert-base-uncased-finetuned-sst-2-english"
    )
    return classifier(text)

def answer_question(question, context):
    qa = pipeline(
        "question-answering",
        model="distilbert-base-uncased-distilled-squad"
    )
    return qa(question=question, context=context)

def main():
    # --- Text Classification example ---
    text = "I really enjoy learning about AI and open-source models!"
    print("--- TEXT CLASSIFICATION ---")
    print("Input:", text)
    cls_res = get_sentiment(text)
    print("Output:", cls_res)
    print()

    # --- Question Answering example ---
    context = (
        "Hugging Face is a company that builds tools and libraries for machine learning "
        "and hosts a large model hub for sharing pre-trained models."
    )
    question = "What does Hugging Face host?"
    print("--- QUESTION ANSWERING ---")
    print("Context:", context)
    print("Question:", question)
    qa_res = answer_question(question, context)
    print("Answer:", qa_res.get('answer'))
    print("Score:", qa_res.get('score'))
    print("Span (start,end):", qa_res.get('start'), qa_res.get('end'))

if __name__ == "__main__":
    main()
```

5. Sample Output



```
Do you want to run software from this untrusted publisher?
File C:\Users\9758h\AppData\Local\Programs\Microsoft VS Code\resources\app\out\vs\workbench\contrib\terminal\common\scripts\shellIntegration.ps1 is published by CN=Microsoft Corporation,
O=Microsoft Corporation, L=Redmond, S=Washington, C=US and is not trusted on your system. Only run scripts from trusted publishers.
[V] Never run  [D] Do not run  [R] Run once  [A] Always run  [?] Help (default is "D"): & D:/hf_assignment/hf_env/Scripts/Activate.ps1
[V] Never run  [D] Do not run  [R] Run once  [A] Always run  [?] Help (default is "D"):  & 'd:\hf_assignment\hf_env\Scripts\python.exe' 'c:\Users\9758h\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher' '59916' '--' 'D:\hf_assignment\hf_env\huggingface_qa_classify.py'
[V] Never run  [D] Do not run  [R] Run once  [A] Always run  [?] Help (default is "D"):
PS D:\hf_assignment>  d:; cd 'd:\hf_assignment'; & 'd:\hf_assignment\hf_env\Scripts\python.exe' 'c:\Users\9758h\.vscode\extensions\ms-python.debugpy-2025.10.0-win32-x64\bundled\libs\debugpy\launcher' '59934' '--'
'D:\hf_assignment\hf_env\huggingface_qa_classify.py'
--- TEXT CLASSIFICATION ---
Input: I really enjoy learning about AI and open-source models!
config.json: 100%|| 629/629 [00:00<00:00, 40.3kB/s]
d:\hf_assignment\hf_env\Lib\site-packages\huggingface_hub\file_download.py:143: UserWarning: `huggingface_hub` cache-system uses symlinks by default to efficiently store duplicated files but your machine does not
support them in C:\Users\9758h\.cache\huggingface\hub\models--distilbert-base-uncased-finetuned-sst-2-english. Caching files will still work but in a degraded version that might require more space on your disk. Th
is warning can be disabled by setting the `HF_HUB_DISABLE_SYMLINKS_WARNING` environment variable. For more details, see https://huggingface.co/docs/huggingface_hub/how-to-cache#limitations.
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/apps
/get-started/enable-your-device-for-development
  warnings.warn(message)
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingface_hub[hf_xet]` or `p
ip install hf_xet`
model.safetensors: 100%|| 268M/268M [00:24<00:00, 11.1MB/s]
tokenizer_config.json: 100%|| 48.0/48.0 [00:00<?, ?B/s]
vocab.txt: 100%|| 232k/232k [00:00<00:00, 1.00MB/s]
Device set to use cpu
Output: [{'label': 'POSITIVE', 'score': 0.9997457861900330}]

--- QUESTION ANSWERING ---
Context: Hugging Face is a company that builds tools and libraries for machine learning and hosts a large model hub for sharing pre-trained models.
Question: What does Hugging Face host?
config.json: 100%|| 451/451 [00:00<?, ?B/s]
d:\hf_assignment\hf_env\Lib\site-packages\huggingface_hub\file_download.py:143: UserWarning: `huggingface_hub` cache-system uses symlinks by default to efficiently store duplicated files but your machine does not
support them in C:\Users\9758h\.cache\huggingface\hub\models--distilbert-base-uncased-distilled-squad. Caching files will still work but in a degraded version that might require more space on your disk. This warni
ng can be disabled by setting the `HF_HUB_DISABLE_SYMLINKS_WARNING` environment variable. For more details, see https://huggingface.co/docs/huggingface_hub/how-to-cache#limitations.
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/apps
/get-started/enable-your-device-for-development
  warnings.warn(message)
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingface_hub[hf_xet]` or `p
ip install hf_xet`
model.safetensors: 100%|| 265M/265M [00:27<00:00, 9.53MB/s]
face_hub/how-to-cache#limitations.
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/apps
/get-started/enable-your-device-for-development
  warnings.warn(message)
Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download. For better performance, install the package with: `pip install huggingfface_hub/how-to-cache#
limitations.
To support symlinks on Windows, you either need to activate Developer Mode or to run Python as an administrator. In order to activate developer mode, see this article: https://docs.microsoft.com/en-us/windows/apps
/get-started/enable-your-device-for-development
  warnings.warn(message)
face_hub/how-to-cache#limitations.
```

```
=== TEXT CLASSIFICATION ===
```

```
Input: I really enjoy learning about AI and open-source models!
100%|
| 268M/268M [00:24<00:00,
11.1MB/s]

tokenizer_config.json:
100%|
| 48.0/48.0 [00:00<?, ?B/s]

vocab.txt:
100%|
| 232k/232k
[00:00<00:00, 1.00MB/s]

Device set to use cpu

Output: [{'label': 'POSITIVE', 'score': 0.999745786190033}]

Question/Answering

question = "What does Hugging Face host?"

context = "Hugging Face is a company that builds tools and libraries for machine learning and hosts a large model hub for sharing
pre-trained models."

answer = "a large model hub"

score = 0.3878

start = 89

end = 106
```

6. Observations

- ✓ Text classification pipeline identified sentiment of text correctly.
- ✓ Question answering pipeline extracted accurate answers from given context.
- ✓ Hugging Face models can be loaded and executed locally without much setup.
- ✓ Both pipelines gave outputs quickly and efficiently.
- ✓ Easy to experiment with different open-source LLMs for NLP tasks.

Conclusion:

- Hugging Face library simplifies text classification and question answering tasks.
- Local execution of models is feasible and reliable.
- Open-source LLMs can be effectively used for NLP experiments.
- The activity helped understand practical usage of pipelines for real-world text tasks.