## **Capstone Project Deployment Report**

**Project Title:** NLP Workflows in the Cloud Using Azure  
 **Platform:** Microsoft Azure (Web App Deployment via GitHub Actions)  
 **Backup Deployment Strategy:** Docker (Local only)  
 **Model Deployment:** Azure AI Foundry – GPT-3.5 Turbo

### **Project Overview**

This project demonstrates how to build, evaluate, and deploy real-time NLP applications using **LLMs (Large Language Models)** in a cloud-native architecture. We focused on **sentiment analysis** and **document summarization** through REST APIs using Flask, Azure OpenAI integration, ROUGE evaluation, and GitHub Actions CI/CD workflows.

### **Learning Outcomes**

* Build scalable NLP pipelines using GPT models on Azure
* Deploy RESTful applications using GitHub Actions and Azure Web App
* Evaluate summaries using ROUGE scoring
* Securely integrate with Azure AI Studio for model inference
* Explore containerization using Docker for portability
* Ensure ethical and cost-effective usage of LLMs

### **Application Features**

|  |  |
| --- | --- |
| **Endpoint** | **Function** |
| /sentiment | Analyzes sentiment of input text using GPT-3.5-Turbo |
| /summarize | Summarizes input and calculates ROUGE-1, ROUGE-2, ROUGE-L evaluation |

### **What We Tried (Deployment Journey)**

#### **1. Azure OpenAI Model Integration**

* Used **Azure AI Foundry** to deploy gpt-35-turbo model
* Provisioning state: *Succeeded*
* Rate Limit: 1,000 tokens/min and 1 requests/min
* Configured **endpoint and key** in app.py to access the deployed model  
   Verified working through real-time requests via REST APIs.

#### **2. CI/CD Pipeline via GitHub Actions**

* Created main\_azurenlp.yml workflow to automate:
  + Python environment setup
  + Virtualenv and dependency installation
  + Packaging and deployment to Azure Web App
* Secrets securely managed using GitHub’s encrypted variables  
   Final CI/CD run completed successfully showed in resuts

#### **3. Testing Sentiment and Summarization APIs**

* Used PowerShell and Python scripts to validate endpoints
* Example sentiment result:  
   *“The user expresses love for using Azure AI…”*
* Example summarization ROUGE result:

"rouge1": [1.0, 0.3125, 0.47]  
 "rouge2": [0.5, 0.13, 0.21]

Confirmed model correctness and evaluation logic showed in results

#### **🔹 4. Docker Containerization.**

* Built Docker image with Dockerfile
* Successfully ran locally with:

docker build -t \*\*\*\*\*\*-app .  
 docker run -p 8000:8000 \*\*\*\*\*-app

* Verified endpoints worked locally
* Deployment to Azure Container Registry was attempted but **not finalized** due to registry configuration constraints

### **📈 Summary of Achievements**

|  |  |
| --- | --- |
| **Completed** | **Description** |
| GitHub Actions CI/CD | Deployed to Azure Web App automatically |
| OpenAI Integration | Configured and consumed GPT-3.5-Turbo via Azure |
| API Testing | Verified endpoints with sample inputs |
| Summarization Evaluation | Implemented ROUGE metrics using rouge-score |
| Dockerization (Local) | App ran successfully in containerized format |
| Secure Secrets Handling | Managed via .env and GitHub secrets |

After investigating and testing multiple deployment configurations, it was observed that the issue might be related to **network restrictions or port binding conflicts**, which could be affecting the application’s accessibility or response during deployment.

**Thank you for giving me this wonderful opportunity. Through these projects, I was able to explore new concepts, overcome real-world deployment challenges, and gain hands-on experience with cloud-based NLP workflows. It has been a valuable and enriching learning experience.**