Manual for Deploying NLP Model on Azure

1. Set Up Your Azure Resources

1.1 Log in to Azure CLI

To begin, log in to your Azure account using the Azure CLI:

```
az login
```

1.2 Create a Resource Group

Create a resource group to manage your Azure resources:

```
az group create --name myResourceGroup --location westus
```

1.3 Create an App Service Plan

Set up an App Service Plan where your web app will be hosted:

```
az appservice plan create --name myAppServicePlan --resource-group
myResourceGroup --sku FREE
```

1.4 Create a Web App

Create a web app within your App Service Plan:

```
az webapp create --resource-group myResourceGroup --plan myAppServicePlan --name
myWebApp --runtime "PYTHON|3.8"
```

2. Prepare Your Project

Ensure your project directory is structured as follows:

```
/your_project
├── app.py
├── requirements.txt
├── tokenizer.pkl
├── disaster_tweet_lstm_model.h5
├── templates
|   └── index.html
├── static
|   ├── css
|   |   └── reset.css
|   |   └── screen.css
|   ├── js
```

```
|   |   └── script.js
|   └── swagger.json
└── .github
    └── workflows
        └── azure_deploy.yml
```

3. Add GitHub Actions for CI/CD

3.1 Initialize a Git Repository

Navigate to your project directory and initialize a Git repository:

```
cd your_project
git init
git add .
git commit -m "Initial commit"
```

3.2 Create a GitHub Repository

1. Create a new repository on GitHub.
2. Push your local repository to GitHub:

```
git remote add origin https://github.com/yourusername/your-repo.git
git branch -M main
git push -u origin main
```

3.3 Add GitHub Secrets

1. In your GitHub repository, navigate to `Settings > Secrets > Actions`.
2. Add a new secret named `AZURE_WEBAPP_PUBLISH_PROFILE`.
3. In the Azure Portal, navigate to your Web App, and in the "Overview" section, download the publish profile.
4. Copy the content of the publish profile and paste it into the GitHub secret.

3.4 Create GitHub Actions Workflow

Create a `.github/workflows/azure_deploy.yml` file with the following content:

```
name: Azure Web App Deploy

on:
  push:
    branches:
      - main

jobs:
  build-and-deploy:
    runs-on: ubuntu-latest

    steps:
    - name: Checkout code
```

```
      uses: actions/checkout@v2

    - name: Set up Python
      uses: actions/setup-python@v2
      with:
        python-version: '3.8'

    - name: Install dependencies
      run: |
        python -m pip install --upgrade pip
        pip install -r requirements.txt

    - name: Create ZIP package
      run: |
        zip -r source_code.zip .

    - name: Deploy to Azure Web App
      uses: azure/webapps-deploy@v2
      with:
        app-name: 'myWebApp'
        package: './source_code.zip'
        publish-profile: ${{ secrets.AZURE_WEBAPP_PUBLISH_PROFILE }}
```

4. Push Your Code to GitHub

4.1 Push Code to GitHub

Commit and push your changes to the `main` branch:

```
git add .
git commit -m "Add deployment configuration"
git push origin main
```

5. Monitor GitHub Actions

5.1 Monitor the Workflow

1. Go to the `Actions` tab of your GitHub repository.
2. Ensure the workflow runs successfully and deploys your application to Azure.

6. Access Your Deployed Application

6.1 Open Your Application

Once the deployment is successful, your application will be available at:

```
http://<your-web-app-name>.azurewebsites.net
```

6.2 Access Swagger UI

The Swagger UI for API documentation and testing will be available at:

```
http://<your-web-app-name>.azurewebsites.net/swagger
```
