[Read Me] Deriving Relevant Sentences for Patient Summaries in Medical Transcriptions Using BERT

Introduction

In many different applications, it can be useful to determine whether a particular statement is accurately reflected in a larger corresponding document. Throughout this blog post, we will be considering the benefits of this type of application within the medical field. Often times, doctors are tasked with writing a quick summary statement that will summarize their patient's entire visit. Later on, if that patient revisits the doctor or transfers to a new doctor, then those summary statements will be reviewed by their new doctor in order to get a better understanding of the patient's medical history. That being said, the accuracy of these summary statements can have direct impact on the patient's treatment in the future. This application will provide a way to check that the summary statement is an accurate representation of the information from the corresponding document (aka the full details from the patient visit).

Example
If the summary statement is “The patient is healthy” and the supporting documentation is “The patient's health is good. The patient does not have a fever. The patient is a 28 year old female.” the most accurate summary statement would be “The patient's health is good.”

Note: From this point forward, we will be referring to the summary statement as a "restatement" to avoid any confusion with the corresponding document that summarizes the patient visit. That being said, the restatement may be a summary statement of the document, a high level description, an exact quote from the document, or bear some other relationship to the document.

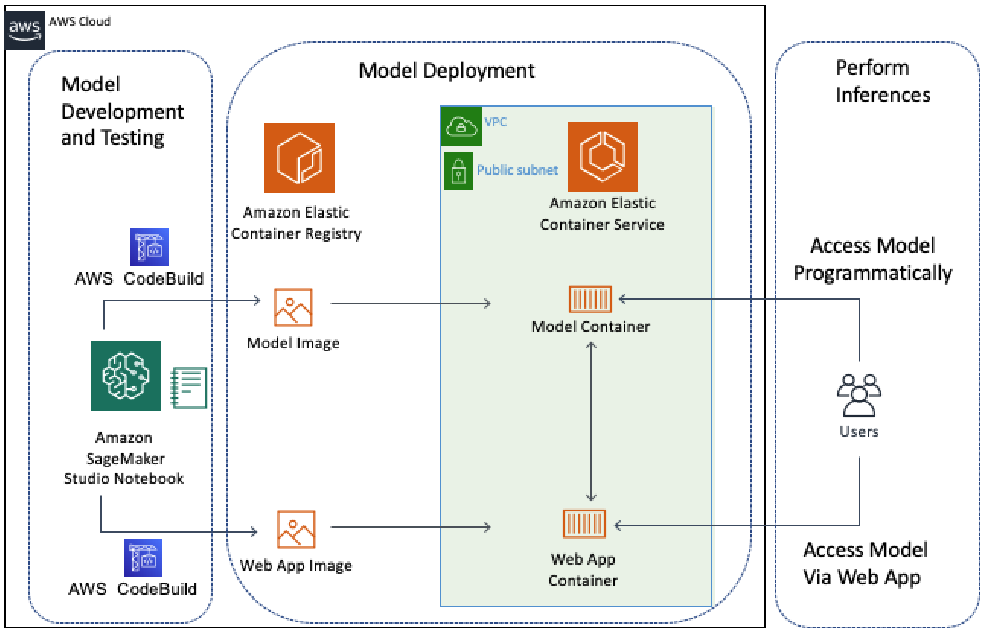
Approach

As briefly mentioned above, we will deploy a solution that takes in a restatement and a corresponding document. The corresponding document will first be broken up into individual sentences, and then Bio_ClinicalBERT will be used to embed each of those sentences as well as the restatement sentence itself. The script will then compute the distances (using cosine similarity) between the restatement and each of the individual sentences from the corresponding document. The top 5 sentences (i.e. the ones with the lowest distances) will be returned along with their respective scores.

Machine Learning Method

This core machine learning method used relies on BERT, a language/word embedding model published by Google (the original paper can be found here). BERT is a context aware embedder that can be used for a number of Natural Language Processing tasks (a brief overview of BERT can be found here). Please note that for this specific use case, we are using a pre-trained BERT model, called Bio_ClinicalBERT, that was developed specifically for medical/clinical tests. This pre-trained model was developed by `Alsentzer et al`, and the full paper published by that group can be found here.

Architecture Diagram

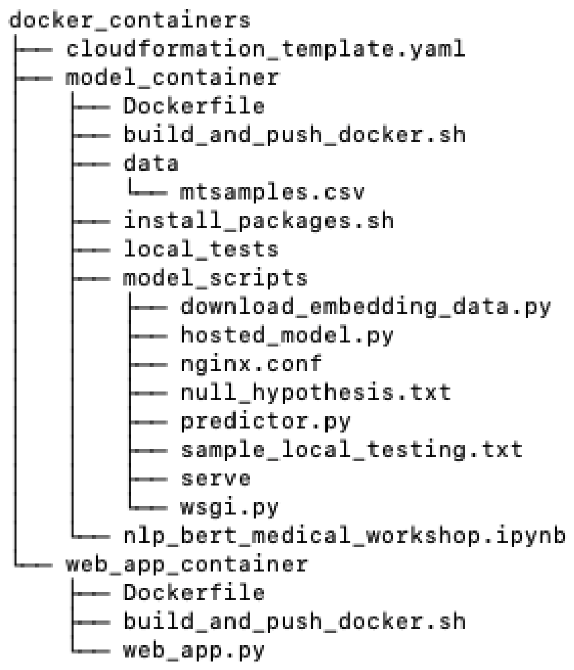


As you can see, there are three sections.

1. In the first section `Model Development and Testing` we will use the SageMaker Studio Integrated Development Environment (IDE) for building the application. SageMaker Studio is an IDE developed specifically for machine learning.
2. In the second section `Model Deployment` we will deploy both the model and a web app (powered by Streamlit for querying the model. Both the model and the web app will be packaged as Docker containers using AWS CodeBuild .They will be first registered with Amazon Elastic Container Registry (ECR) and then deployed on Amazon Elastic Container Service(ECS) using AWS Fargate. For a more in-depth tutorial on deploying Streamlit apps to ECS; checkout out this blog post.
3. In the third section `Perform Inferences` we will perform some inferences using our model and web app.

Contents

The content of the workshop is as follows:



```
docker_containers
├── cloudformation_template.yaml
├── model_container
│   ├── Dockerfile
│   ├── build_and_push_docker.sh
│   ├── data
│   │   └── mtsamples.csv
│   ├── install_packages.sh
│   ├── local_tests
│   ├── model_scripts
│   │   ├── download_embedding_data.py
│   │   ├── hosted_model.py
│   │   ├── nginx.conf
│   │   ├── null_hypothesis.txt
│   │   ├── predictor.py
│   │   ├── sample_local_testing.txt
│   │   ├── serve
│   │   └── wsgi.py
│   └── nlp_bert_medical_workshop.ipynb
└── web_app_container
    ├── Dockerfile
    ├── build_and_push_docker.sh
    └── web_app.py
```

Workshop Requirements

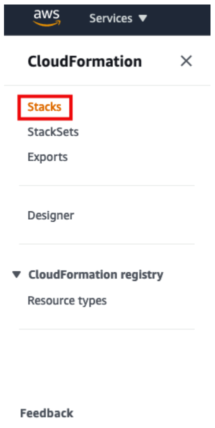
You will need an AWS account to use this solution. Sign up for an account here.

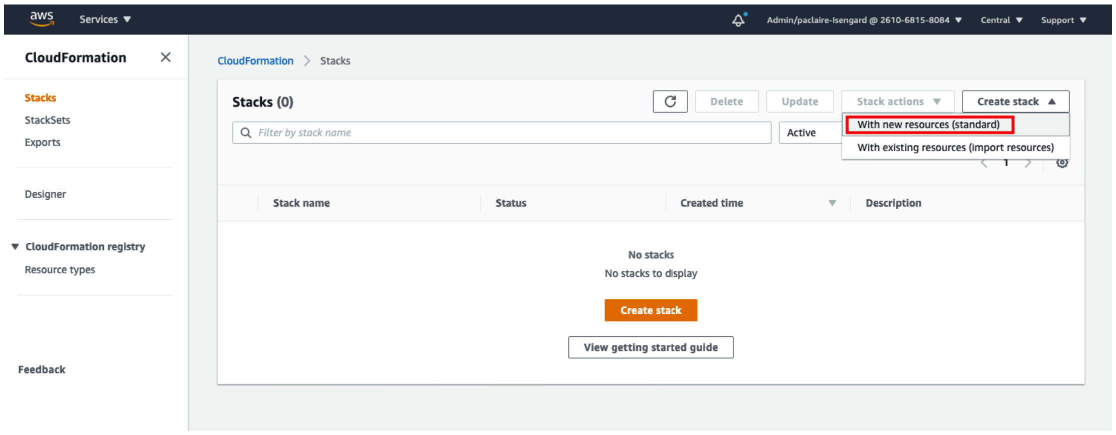
You will also need to have permission to use AWS CloudFormation and to create all the resources detailed in the architecture section. All AWS permissions can be managed through AWS IAM. Admin users will have the required permissions, but please contact your account's AWS administrator if your user account doesn't have the required permissions.

Steps for deploying CloudFormation Template

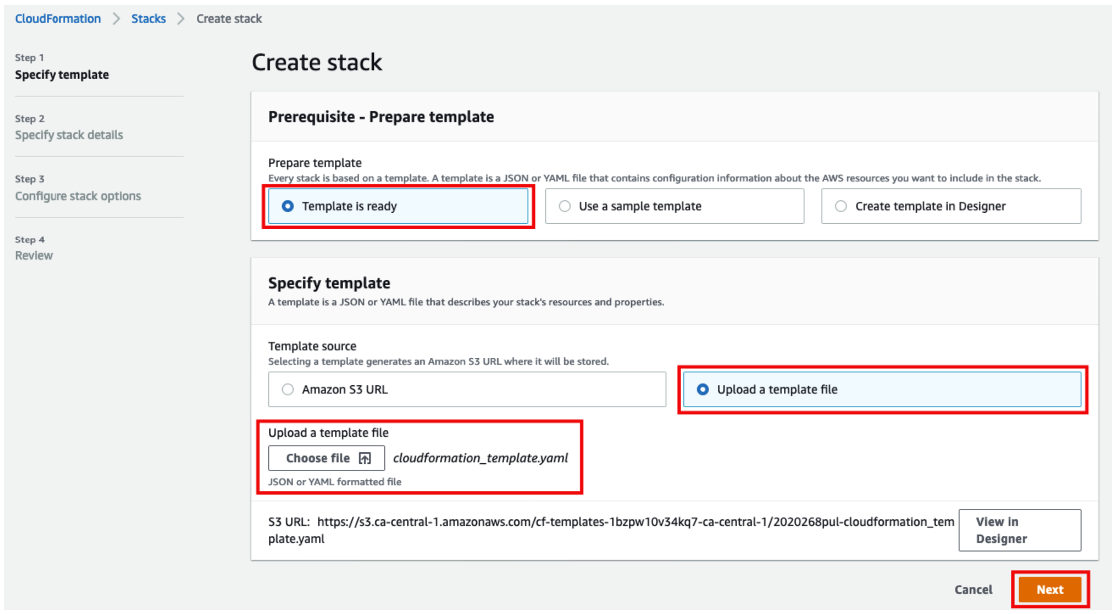
To run this workshop, you will need to first deploy the CloudFormation Template

- Open AWS Console
- Select "CloudFormation" from the list of services
- Select “Stacks” from the sidebar

- 

- Select "Create stack" and choose "With new resources (standard)"
  - 

- Step 1: Specify template
  - Under the Prepare template section select "Template is ready"
  - Then under the Specify template section select "Upload a template file"
  - Click the "Choose file" button and upload your cloudformation template
  - Then select "Next"

- 

  CloudFormation > Stacks > Create stack

  Step 1 Specify template
  Step 2 Specify stack details
  Step 3 Configure stack options
  Step 4 Review

  Create stack

  Prerequisite - Prepare template

  Prepare template
  Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.

  [✓ Template is ready] Use a sample template | Create template in Designer

  Specify template
  A template is a JSON or YAML file that describes your stack's resources and properties.

  Template source
  Selecting a template generates an Amazon S3 URL where it will be stored.

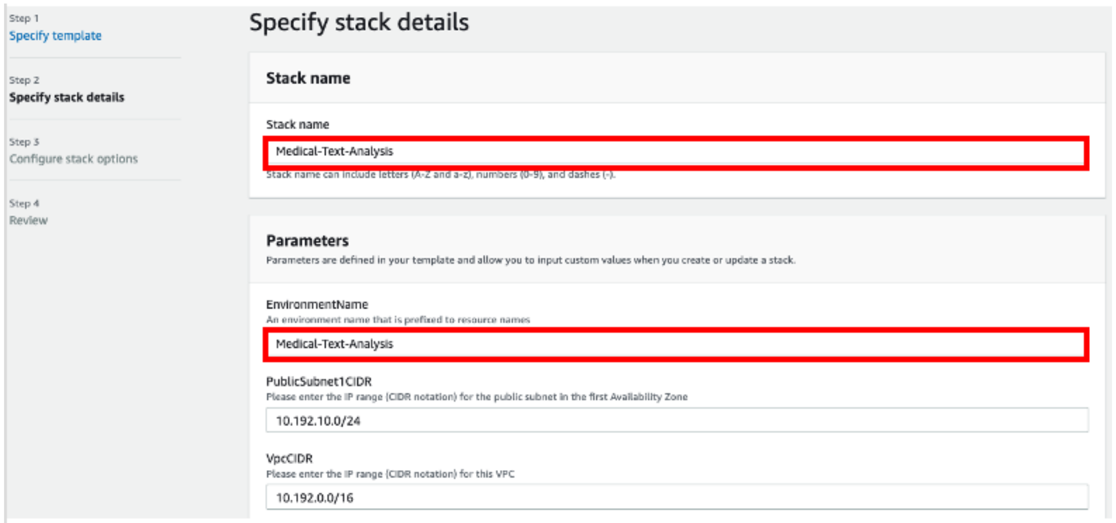
  Amazon S3 URL [✓ Upload a template file]

  Upload a template file
  Choose file *cloudformation_template.yaml*
  JSON or YAML formatted file

  S3 URL: https://s3.ca-central-1.amazonaws.com/cf-templates-1bzpw10v34kq7-ca-central-1/2020268pul-cloudformation_template.yaml

  View in Designer

  Cancel Next

- Step 2: Specify Stack Details
  - Enter Medical-Text-Analysis as the "Stack name" and as the "EnvironmentName"
  - Click "Next"
  - 

    Step 1 Specify template
    Step 2 Specify stack details
    Step 3 Configure stack options
    Step 4 Review

    Specify stack details

    Stack name

    Stack name
    Medical-Text-Analysis
    Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).

    Parameters
    Parameters are defined in your template and allow you to input custom values when you create or update a stack.

    EnvironmentName
    An environment name that is prefixed to resource names
    Medical-Text-Analysis

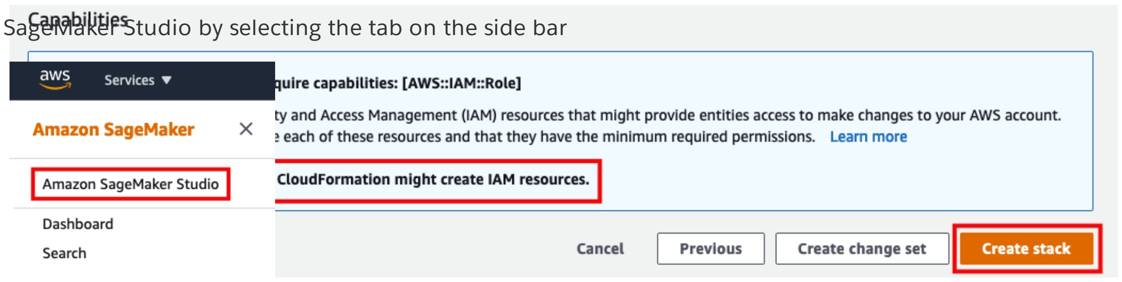
    PublicSubnet1CIDR
    Please enter the IP range (CIDR notation) for the public subnet in the first Availability Zone
    10.192.10.0/24

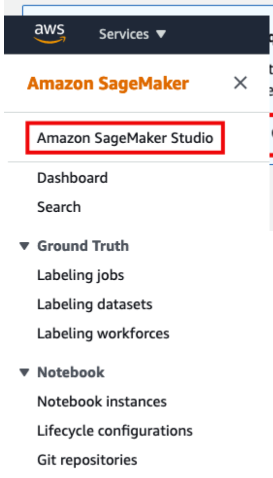
    VpcCIDR
    Please enter the IP range (CIDR notation) for this VPC
    10.192.0.0/16

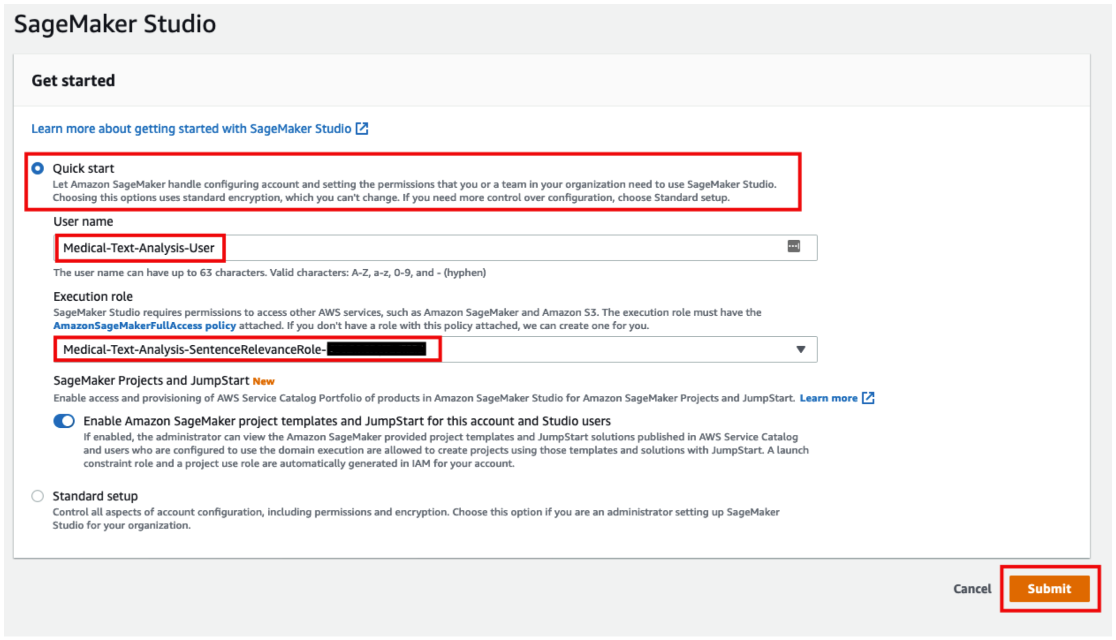
- Step 3: Configure Stack Options
  - Optional: Add any tags that you want to be applied to the resources in your stack
  - Leave everything else as default and select "Next"
- Step 4: Review
  - Review over all of the information and at the bottom of the page select the checkbox to acknowledge that this cloudformation template will be creating an IAM resource.
  - Then select "Create Stack"

Steps for opening SageMaker Studio

- Once the cloudformation template has finished deploying all of the resources, open SageMaker from the main AWS console
- We will be using SageMaker Studio for this workshop. Here's the list of regions that SageMaker Studio is available in.

- Go to SageMaker Studio by selecting the tab on the side bar
    - 
      Capabilities
      …quire capabilities: [AWS::IAM::Role]
      …ty and Access Management (IAM) resources that might provide entities access to make changes to your AWS account. …e each of these resources and that they have the minimum required permissions. Learn more
      …CloudFormation might create IAM resources.

    - 

- If this is your first time using the SageMaker Studio console, you will need to first set it up.
    - Choose the “Quick start" option
    - Set the “User name” to “Medical-Text-Analysis-User” (or any other user name will do!)
    - For the Execution role, select the IAM role that was created when the CloudFormation template was deployed. This will look something like “Medical-Text-Analysis-Sentence-RelevanceRole-XXXXXXXXXXXX”
    - Select Submit
    - 
      SageMaker Studio

      Get started

      Learn more about getting started with SageMaker Studio

      ● Quick start
      Let Amazon SageMaker handle configuring account and setting the permissions that you or a team in your organization need to use SageMaker Studio. Choosing this options uses standard encryption, which you can't change. If you need more control over configuration, choose Standard setup.

      User name
      Medical-Text-Analysis-User
      The user name can have up to 63 characters. Valid characters: A-Z, a-z, 0-9, and - (hyphen)

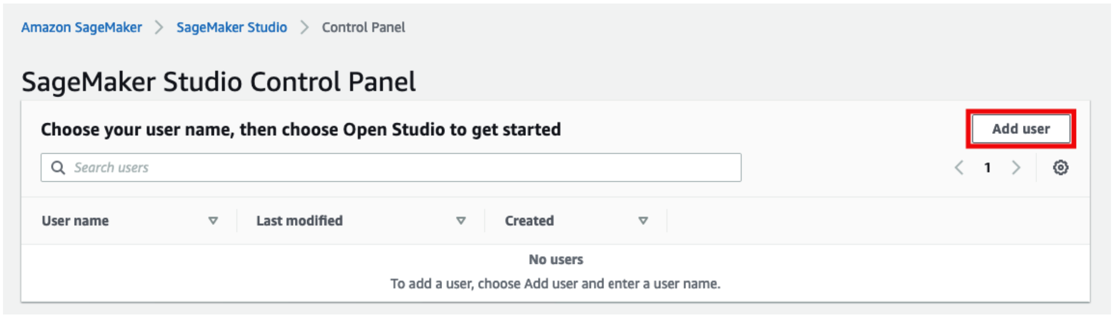
      Execution role
      SageMaker Studio requires permissions to access other AWS services, such as Amazon SageMaker and Amazon S3. The execution role must have the AmazonSageMakerFullAccess policy attached. If you don't have a role with this policy attached, we can create one for you.
      Medical-Text-Analysis-SentenceRelevanceRole-[redacted]

      SageMaker Projects and JumpStart New
      Enable access and provisioning of AWS Service Catalog Portfolio of products in Amazon SageMaker Studio for Amazon SageMaker Projects and JumpStart. Learn more

      Enable Amazon SageMaker project templates and JumpStart for this account and Studio users
      If enabled, the administrator can view the Amazon SageMaker provided project templates and JumpStart solutions published in AWS Service Catalog and users who are configured to use the domain execution are allowed to create projects using those templates and solutions with JumpStart. A launch constraint role and a project use role are automatically generated in IAM for your account.

      ○ Standard setup
      Control all aspects of account configuration, including permissions and encryption. Choose this option if you are an administrator setting up SageMaker Studio for your organization.

- If this is not your first time using SageMaker Studio, go ahead and just create a new user
    - Select “Add User”

- 

  Amazon SageMaker > SageMaker Studio > Control Panel

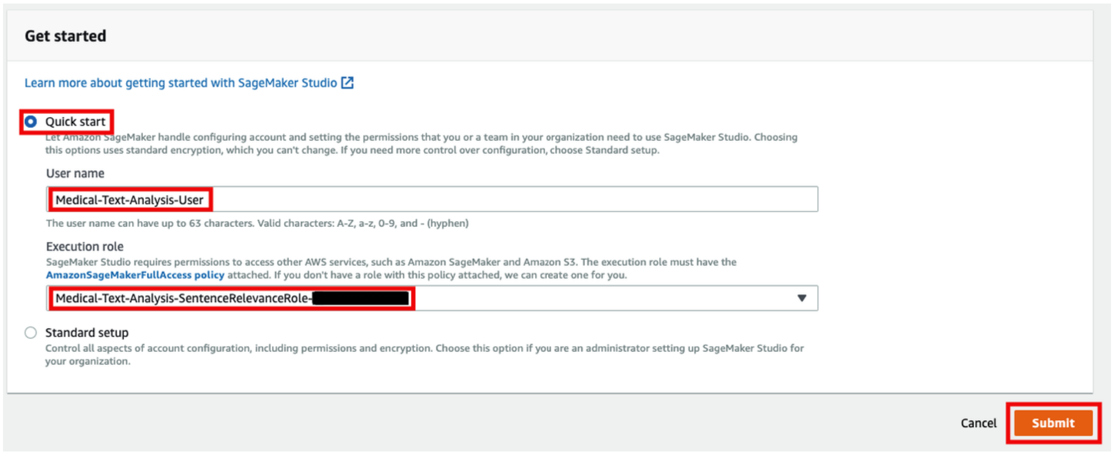
  SageMaker Studio Control Panel

  Choose your user name, then choose Open Studio to get started

User name	Last modified	Created

  No users

  To add a user, choose Add user and enter a user name.

- Select “Quick Start”
- Set the “User name” to “Medical-Text-Analysis-User” (or any other user name will do!)
- For the Execution role, select the IAM role that was created when the CloudFormation template was deployed. This will look something like “Medical-Text-Analysis-Sentence-RelevanceRole-XXXXXXXXXXXX”
- Select Submit
- 

  Get started

  Learn more about getting started with SageMaker Studio

  [✓ Quick start]
  Let Amazon SageMaker handle configuring account and setting the permissions that you or a team in your organization need to use SageMaker Studio. Choosing this options uses standard encryption, which you can't change. If you need more control over configuration, choose Standard setup.

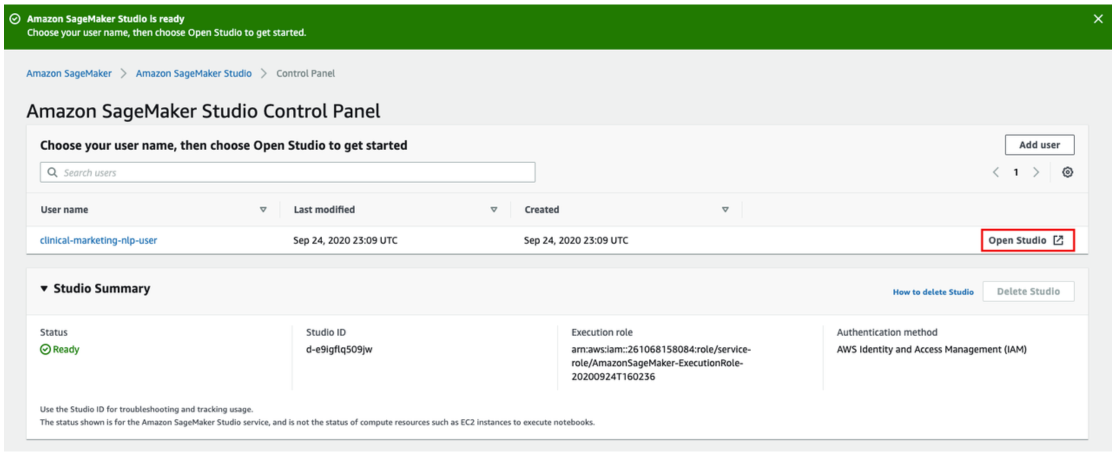
  User name: Medical-Text-Analysis-User

  The user name can have up to 63 characters. Valid characters: A-Z, a-z, 0-9, and - (hyphen)

  Execution role
  SageMaker Studio requires permissions to access other AWS services, such as Amazon SageMaker and Amazon S3. The execution role must have the **AmazonSageMakerFullAccess policy** attached. If you don't have a role with this policy attached, we can create one for you.

  Medical-Text-Analysis-SentenceRelevanceRole-[redacted]

  Standard setup
  Control all aspects of account configuration, including permissions and encryption. Choose this option if you are an administrator setting up SageMaker Studio for your organization.

* You will then be prompted to choose a VPC and Subnet you would like to use for SageMaker Studio. Select `sentence_relevance_VPC`. There is only one subnet in that VPC; select it.
* Once you choose your VPC and Subnet, you will then be brought to the Amazon SageMaker Studio Control Panel where you will see a bar at the top explaining that it is “Preparing Amazon SageMaker Studio”. This will take a few minutes to create.
* Once SageMaker studio is ready (this will take about 5 minutes), find your Username and select “Open Studio”
  - 

    Amazon SageMaker Studio is ready
    Choose your user name, then choose Open Studio to get started.

    Amazon SageMaker > Amazon SageMaker Studio > Control Panel

    Amazon SageMaker Studio Control Panel

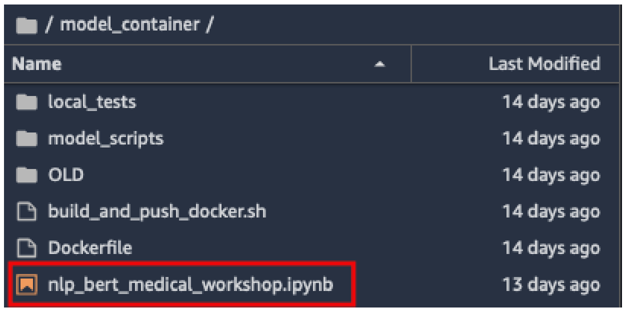
    Choose your user name, then choose Open Studio to get started

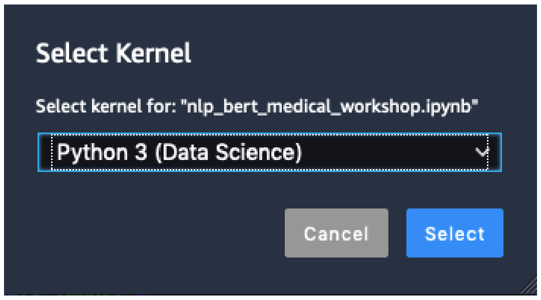
User name	Last modified	Created	
clinical-marketing-nlp-user	Sep 24, 2020 23:09 UTC	Sep 24, 2020 23:09 UTC	Open Studio

    Studio Summary

Status	Studio ID	Execution role	Authentication method
Ready	d-e9igflq509jw	arn:aws:iam::261068158084:role/service-role/AmazonSageMaker-ExecutionRole-20200924T160236	AWS Identity and Access Management (IAM)

    Use the Studio ID for troubleshooting and tracking usage.
    The status shown is for the Amazon SageMaker Studio service, and is not the status of compute resources such as EC2 instances to execute notebooks.

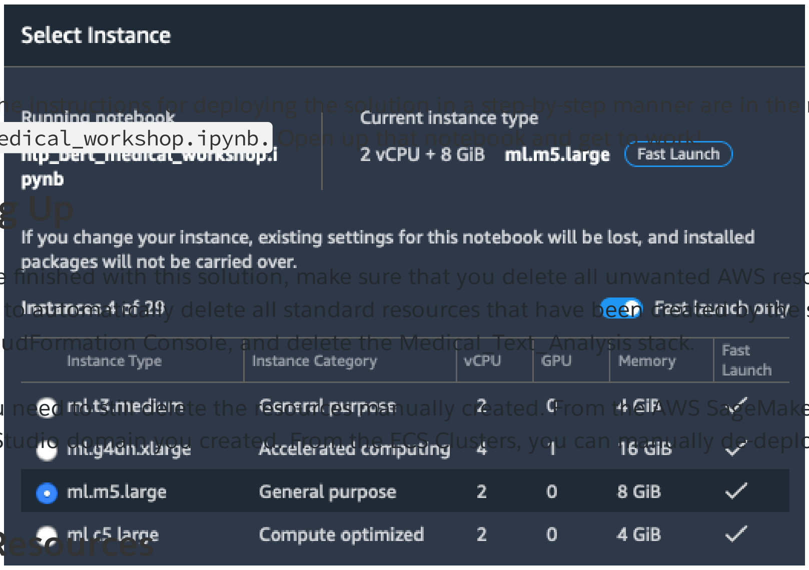
- Once the studio opens, clone this repo into the SageMaker Studio console with `git clone <URL> (note: you can also download this repository manually and upload it to SageMaker Studio).
- Click on the directory `Medical_Text_Analysis_Resources`, then `docker_containers` and finally the folder `model_container”`and double click on “nlp_bert_medical_workshop.ipynb” to open the notebook
    - 
      / model_container /

Name	Last Modified
local_tests	14 days ago
model_scripts	14 days ago
OLD	14 days ago
build_and_push_docker.sh	14 days ago
Dockerfile	14 days ago
nlp_bert_medical_workshop.ipynb	13 days ago
- Choose Python 3 (Data Science) as your preferred kernel
    - NOTE: This might take a few minutes to start up
    - 
      Select Kernel

      Select kernel for: "nlp_bert_medical_workshop.ipynb"

      Python 3 (Data Science)

- In the top right of the notebook click on “Unknown” and set to ml.m5.large



Select Instance

Running notebook
nlp_bert_medical_workshop.ipynb

Current instance type
2 vCPU + 8 GiB ml.m5.large Fast Launch

If you change your instance, existing settings for this notebook will be lost, and installed packages will not be carried over.

Instances 4 of 29 Fast launch only

	Instance Type	Instance Category	vCPU	GPU	Memory	Fast Launch
○	ml.t3.medium	General purpose	2	0	4 GiB	✓
○	ml.g4dn.xlarge	Accelerated computing	4	1	16 GiB	✓
●	ml.m5.large	General purpose	2	0	8 GiB	✓
○	ml.c5.large	Compute optimized	2	0	4 GiB	✓

The rest of the instructions for deploying the solution in a step-by-step manner are in the notebook `nlp_bert_medical_workshop.ipynb`. Open up that notebook and get to work!

Cleaning Up

When you've finished with this solution, make sure that you delete all unwanted AWS resources. AWS CloudFormation can be used to automatically delete all standard resources that have been created by the solution and notebook. Go to the AWS CloudFormation Console, and delete the Medical_Text_Analysis stack.

Caution: You need to still delete the resources manually created. From the AWS SageMaker Console, you can delete the SageMaker Studio domain you created. From the ECS Clusters, you can manually de-deploy the deployed containers.

Useful Resources

- CloudFormation User Guide
- Amazon SageMaker Developer Guide
- Amazon SageMaker Studio Guide

Credits

- Bio_ClinicalBERT
- MTSamples