## **INTEGRATING HOSTED SERIVCE APPLICATIONS WITH AWS**

## 

## **Setting up the Integration**

* Creating an AWS account and setting up necessary services
* Configuring AWS Config management rules for compliance monitoring
* Setting up AWS API Gateway and Lambda function for ChatGPT integration

## Setting up the Integration

To integrate OpenAI’s ChatGPT with AWS services, follow these steps:

### **1. Creating an AWS account and setting up necessary services**

**1.1** create a new account on the AWS website.

Once you have an AWS account, set up the necessary services for the integration. This may include:

* Creating an Amazon S3 bucket to store any required files or data
* Creating an AWS Lambda function to execute the ChatGPT model
* Setting up AWS API Gateway to create the API endpoint for communication

### **2. Configuring AWS Config management rules for compliance monitoring**

**2.1** Setting up and configuring these rules helps maintain the integrity and security of your integration.

### **3. Setting up AWS API Gateway and Lambda function for ChatGPT integration**

**3.1** Now, set up AWS API Gateway and Lambda function to integrate ChatGPT.

**3.2** Start by creating an API Gateway, which serves as the API endpoint for communication with ChatGPT.

**3.3** Configure the necessary settings, security measures, and request/response formats.

**3.4** Next, create an AWS Lambda function to execute the ChatGPT model.

**3.5** write the code for the Lambda function using Python or any other supported programming language.

**3.6** This function will communicate with the API Gateway and receive the incoming requests, process them using ChatGPT, and provide the appropriate responses.

**3.7** Once both the API Gateway and Lambda function are set up, test the integration to ensure everything is working correctly.

## **Implementing ChatGPT with AWS Lambda and API Gateway**

* **1 . Creating an AWS Lambda function for ChatGPT implementation:**
  + **1.1** Using the serverless framework: The serverless framework allows you to easily create and deploy AWS Lambda functions. You can use it to define your ChatGPT Lambda function and configure the necessary permissions and resources.
  + **1.2** API key storage: It is recommended to use AWS Secrets Manager to securely store your API key. You can use the AWS console to create a secret and retrieve its value in your Lambda function.
  + **1.3** Calling the ChatGPT API: In your Lambda function, you will need to make HTTP requests to the ChatGPT API. You can use the `requests` library in Python to send the requests and process the responses.
* **2. Configuring AWS API Gateway to trigger the Lambda function:**
  + **2.1** Creating a new API: In the AWS API Gateway console, you can create a new API and configure its settings, including the endpoints and integration with your Lambda function.
  + **2.2** Setting up method integrations: For each endpoint in your API, you can configure the integration type as AWS Lambda and select your ChatGPT Lambda function. This will enable API Gateway to trigger your Lambda function when the endpoint is called.
  + 2.3 API security: API Gateway provides various methods to secure access to your API, including API keys, IAM roles, and custom authorizers. You can choose the appropriate method based on your requirements.
* **3. Testing and troubleshooting the integration:**
  + **3.1** Testing with the API Gateway console: You can test your API using the built-in testing capabilities of the API Gateway console. This allows you to send sample requests to your API and inspect the responses.
  + **3.2** Debugging Lambda function errors: If your Lambda function encounters errors, you can use the AWS CloudWatch logs to investigate and debug the issues. The logs provide detailed information about the execution of your Lambda function.
  + **3.3** Monitoring and scaling: AWS provides various monitoring and scaling features for Lambda functions and API Gateway, such as AWS CloudWatch metrics, alarms, and auto-scaling. These features help you ensure the performance and availability of your ChatGPT integration.

**METRICS & EVALUATION:**

ROUGE (Recall-Oriented Understudy)

* ROUGE is the F1 score from the n-gram precision and recall
* Precision in the context of ROUGE reflects the fraction of the n-grams in the prediction that are also in the reference
* the recall is the fraction of reference n-grams that also appear in the model prediction

EX:

**System Summary (what the machine produced):**

the cat was found under the bed

**Reference Summary (gold standard — usually by humans):**

the cat was under the bed

0*DttHG7bvoji8eQY3

0*lZmp28lCclAPTqJ-

Recall = 6/6=1

**BLEU SCORE:**

BLEU stands for **Bilingual Evaluation Understudy**.

It compares your predicted output with a reference text and calculates a score for the same.

* A unigram (n=1) would be a single word: "I," "love," "to," "eat," "ice," or "cream."
* A bigram (n=2) would be a group of two words: "I love," "love to," "to eat," "eat ice," or "ice cream."
* A trigram (n=3) would be a group of three words: "I love to," "love to eat," "to eat ice," and "eat ice cream."

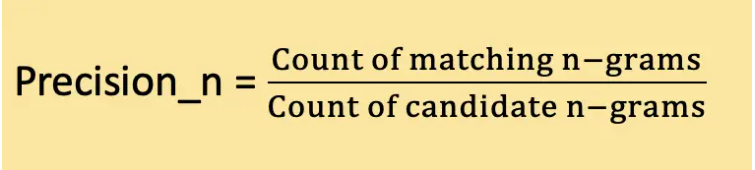
**N-grams** help us understand patterns and relationships between words in a sentence or a text.

They can be used to predict what word might come next, to generate new text, or to analyze the frequency of certain word combinations.

## **Calculate the BLEU Score using n-grams**

N-gram precision measures how well the candidate sentence matches the reference sentences in terms of n-gram sequences. It considers the count of matching n-grams in the candidate sentence and the maximum count of those n-grams in the reference sentences.

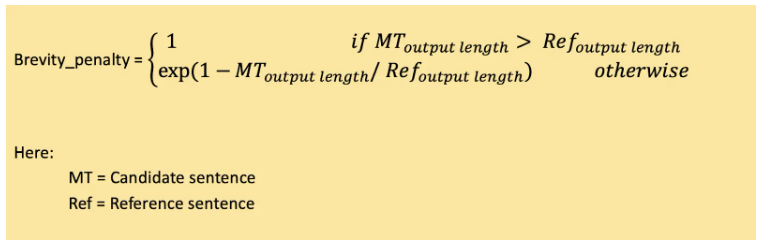
The formula to calculate precision for a particular n-gram size is:

****

### **Calculate the Brevity Penalty**

The brevity penalty addresses the issue of shorter candidate sentences receiving higher scores. It penalizes the BLEU score if the candidate sentence is significantly shorter than the reference sentences.

The brevity penalty is calculated as:



### **Calculate the n-gram Weights**

N-gram weights assign different levels of importance to different n-gram sizes. Typically, equal weights are assigned to each n-gram size. For example, for 2-gram, the weights can be (0.5, 0.5), and for 3-gram, the weights can be (0.33, 0.33, 0.33).

### **Calculation of the Final BLEU Score**

The BLEU score combines the n-gram precisions with the brevity penalty to get the final score. The precisions are weighted by their respective weights, and the brevity penalty is applied:

