AWS Rekognition

Friday, November 22, 2024

10:01 AM

Amazon Rekognition is a service that makes it easy to add image and video analysis to your applications. It can identify objects, people, text, scenes, and activities in images and videos, as well as detect any inappropriate content. Amazon Rekognition also provides highly accurate facial analysis and facial recognition capabilities.

Here are some of the key features of Amazon Rekognition:

- \*\*Object and Scene Detection\*\*: Detects objects, scenes, and activities in images and videos.

- \*\*Facial Analysis\*\*: Detects faces in images and videos and analyzes attributes such as age range, gender, emotions, and more.

- \*\*Facial Recognition\*\*: Matches faces in images and videos against a collection of faces.

- \*\*Text in Image\*\*: Detects and extracts text from images.

- \*\*Moderation\*\*: Detects inappropriate content in images and videos.

- \*\*Celebrity Recognition\*\*: Recognizes celebrities in images and videos.

Below is a sample Java code that demonstrates how to use Amazon Rekognition to detect labels in an image. This example assumes you have the AWS SDK for Java set up in your project.

First, ensure you have the AWS SDK for Java dependencies in your

pom.xml

if you're using Maven:

```xml

<dependencies>

<dependency>

<groupId>software.amazon.awssdk</groupId>

<artifactId>rekognition</artifactId>

<version>2.17.89</version>

</dependency>

</dependencies>

```

Here's a sample Java code that uses Amazon Rekognition to detect labels in an image:

```java

import software.amazon.awssdk.auth.credentials.ProfileCredentialsProvider;

import software.amazon.awssdk.regions.Region;

import software.amazon.awssdk.services.rekognition.RekognitionClient;

import software.amazon.awssdk.services.rekognition.model.DetectLabelsRequest;

import software.amazon.awssdk.services.rekognition.model.DetectLabelsResponse;

import software.amazon.awssdk.services.rekognition.model.Image;

import software.amazon.awssdk.services.rekognition.model.Label;

import software.amazon.awssdk.services.rekognition.model.S3Object;

import java.util.List;

public class RekognitionExample {

public static void main(String[] args) {

Region region = Region.US\_EAST\_1;

RekognitionClient rekognitionClient = RekognitionClient.builder()

.region(region)

.credentialsProvider(ProfileCredentialsProvider.create())

.build();

String bucketName = "my-sample-bucket";

String imageName = "sample-image.jpg";

detectLabelsInImage(rekognitionClient, bucketName, imageName);

rekognitionClient.close();

}

private static void detectLabelsInImage(RekognitionClient rekognitionClient, String bucketName, String imageName) {

S3Object s3Object = S3Object.builder()

.bucket(bucketName)

.name(imageName)

.build();

Image image = Image.builder()

.s3Object(s3Object)

.build();

DetectLabelsRequest request = DetectLabelsRequest.builder()

.image(image)

.maxLabels(10)

.build();

DetectLabelsResponse response = rekognitionClient.detectLabels(request);

List<Label> labels = response.labels();

System.out.println("Detected labels for image " + imageName + ":");

for (Label label : labels) {

System.out.println(label.name() + ": " + label.confidence().toString() + "%");

}

}

}

```

In this example:

- The `RekognitionClient` is created to interact with the Amazon Rekognition service.

- The `detectLabelsInImage` method detects labels in an image stored in an S3 bucket.

- The `DetectLabelsRequest` is created with the S3 bucket and image name, and the maximum number of labels to detect.

- The `DetectLabelsResponse` contains the detected labels, which are printed to the console.

Make sure to replace `"my-sample-bucket"` and `"sample-image.jpg"` with your actual S3 bucket name and image name. This code demonstrates how to use Amazon Rekognition to analyze an image and detect labels.