NAME : NEEL PATEL

[Nap73@njit.edu](mailto:Nap73@njit.edu)

CLOUD COMPUTING : Programming Assignment 1

***ImageRecognition.java***

package com.amazonaws.samples;

import com.amazonaws.AmazonClientException;

import com.amazonaws.auth.AWSStaticCredentialsProvider;

import com.amazonaws.auth.BasicSessionCredentials;

import com.amazonaws.auth.profile.ProfileCredentialsProvider;

import com.amazonaws.regions.Regions;

import com.amazonaws.services.rekognition.AmazonRekognition;

import com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder;

import com.amazonaws.services.rekognition.model.DetectLabelsRequest;

import com.amazonaws.services.rekognition.model.DetectLabelsResult;

import com.amazonaws.services.rekognition.model.Image;

import com.amazonaws.services.rekognition.model.Label;

import com.amazonaws.services.rekognition.model.S3Object;

import com.amazonaws.services.s3.AmazonS3;

import com.amazonaws.services.s3.AmazonS3ClientBuilder;

import com.amazonaws.services.s3.model.\*;

import com.amazonaws.services.sqs.\*;

import com.amazonaws.services.sqs.model.CreateQueueRequest;

import com.amazonaws.services.sqs.model.SendMessageRequest;

import java.util.List;

public class ImageRecognition {

public static void main(String[] args) throws Exception {

String url = "https://sqs.us-east-1.amazonaws.com/442252585959/MyQueue" ;

BasicSessionCredentials sessionCredentials = new BasicSessionCredentials(

ses\_input.getAccessKeyId(), ses\_input.getSecretAccessKey(),

ses\_input.getSessionToken());

final AmazonSQS sqs = AmazonSQSClientBuilder.standard().withCredentials(new AWSStaticCredentialsProvider(sessionCredentials)).withRegion("us-east-1").build();

String bucket = "njit-cs-643";

String ResImage = "";

AmazonRekognition rekognitionClient = AmazonRekognitionClientBuilder.standard()

.withRegion(Regions.US\_EAST\_1)

.build();

final AmazonS3 s3 = AmazonS3ClientBuilder.standard().withRegion(Regions.US\_EAST\_1).build();

ListObjectsV2Result result1 = s3.listObjectsV2(bucket);

List<S3ObjectSummary> objects = result1.getObjectSummaries();

System.out.println("---------------------------------------------------------");

System.out.println("IMAGE RECOGNITION RESULT ");

System.out.println("---------------------------------------------------------");

System.out.println("Images of CAR in the bucket :");

for (S3ObjectSummary os : objects) {

DetectLabelsRequest request = new DetectLabelsRequest()

.withImage(new Image()

.withS3Object(new S3Object()

.withName(os.getKey()).withBucket(bucket)))

.withMaxLabels(10)

.withMinConfidence(90F);

try {

DetectLabelsResult result = rekognitionClient.detectLabels(request);

List <Label> labels = result.getLabels();

for (Label label: labels) {

if(label.getName().equalsIgnoreCase("car")) {

System.out.println(os.getKey() + " Confidence : " +label.getConfidence() );

ResImage=ResImage.concat(os.getKey()+",");

}

}

}

catch(Exception ee) {

}

// CreateQueueRequest createQueueRequest = new CreateQueueRequest("MyQueue");

// String myQueueUrl = sqs.createQueue(createQueueRequest).getQueueUrl();

// System.out.println(myQueueUrl);

ProfileCredentialsProvider credentialsProvider = new ProfileCredentialsProvider();

try {

credentialsProvider.getCredentials();

} catch (Exception e) {

throw new AmazonClientException(

"Cannot load the credentials from the credential profiles file. " +

"Please make sure that your credentials file is at the correct " +

"location (/Users/neel/.aws/credentials), and is in valid format.",

e);

}

System.out.println();

// System.out.println(ResImage);

sqs.sendMessage(new SendMessageRequest(url,ResImage));

}

sqs.sendMessage(new SendMessageRequest(url,ResImage));

// System.out.println(ResImage);

System.out.println("Image Keys successfully send for text recognition");

}

}

***textRecognotion.java***

**package** com.amazonaws.samples;

**import** java.io.BufferedWriter;

**import** java.io.File;

**import** java.io.FileNotFoundException;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** java.io.PrintStream;

**import** java.util.ArrayList;

**import** java.util.List;

**import** java.util.Map.Entry;

**import** com.amazonaws.AmazonClientException;

**import** com.amazonaws.auth.AWSStaticCredentialsProvider;

**import** com.amazonaws.auth.BasicSessionCredentials;

**import** com.amazonaws.auth.profile.ProfileCredentialsProvider;

**import** com.amazonaws.regions.Regions;

**import** com.amazonaws.services.rekognition.AmazonRekognition;

**import** com.amazonaws.services.rekognition.AmazonRekognitionClientBuilder;

**import** com.amazonaws.services.rekognition.model.AmazonRekognitionException;

**import** com.amazonaws.services.rekognition.model.DetectTextRequest;

**import** com.amazonaws.services.rekognition.model.DetectTextResult;

**import** com.amazonaws.services.rekognition.model.Image;

**import** com.amazonaws.services.rekognition.model.S3Object;

**import** com.amazonaws.services.rekognition.model.TextDetection;

**import** com.amazonaws.services.sqs.AmazonSQS;

**import** com.amazonaws.services.sqs.AmazonSQSClientBuilder;

**import** com.amazonaws.services.sqs.model.Message;

**import** com.amazonaws.services.sqs.model.ReceiveMessageRequest;

**public** **class** TextRecognotion {

**static** String *bucket* = "njit-cs-643";

**public** **static** **void** main(String a[]) **throws** IOException {

ProfileCredentialsProvider credentialsProvider = **new** ProfileCredentialsProvider();

**try** {

credentialsProvider.getCredentials();

} **catch** (Exception e) {

**throw** **new** AmazonClientException(

"Cannot load the credentials from the credential profiles file. " +

"Please make sure that your credentials file is at the correct " +

"location (/Users/neel/.aws/credentials), and is in valid format.",

e);

}

AmazonSQS sqs = AmazonSQSClientBuilder.*standard*()

.withCredentials(credentialsProvider)

.withRegion(Regions.***US\_EAST\_1***)

.build();

System.***out***.println("===========================================");

System.***out***.println("Receiving with Amazon SQS");

System.***out***.println("===========================================\n");

// Receive messages

System.***out***.println("Receiving messages from MyQueue.\n");

String url = "https://sqs.us-east-1.amazonaws.com/442252585959/MyQueue";

ReceiveMessageRequest receiveMessageRequest = **new** ReceiveMessageRequest(url);

String result = "";

List<Message> messages = sqs.receiveMessage(receiveMessageRequest).getMessages();

System.***out***.println();

**for** (Message message : messages) {

System.***out***.println(" Message Received for Text Recognition");

System.***out***.println(" MessageId: " + message.getMessageId());

System.***out***.println(" ReceiptHandle: " + message.getReceiptHandle());

System.***out***.println(" MD5OfBody: " + message.getMD5OfBody());

System.***out***.println(" Body: " + message.getBody());

result = message.getBody();

**for** (Entry<String, String> entry : message.getAttributes().entrySet()) {

System.***out***.println(" Attribute");

System.***out***.println(" Name: " + entry.getKey());

System.***out***.println(" Value: " + entry.getValue());

}

}

System.***out***.println("---------------------------------------------------------");

System.***out***.println("TEXT DETECTION RESULT");

System.***out***.println("---------------------------------------------------------");

ArrayList<String> Images = **new** ArrayList<String>();

FileWriter write = **new** FileWriter("output.txt") ;

BufferedWriter bw = **new** BufferedWriter(write);

**char** ch = '\0';

String temp="";

**for** (**int** i = 0; i < result.length(); i++) {

ch = result.charAt(i);

**if**(ch != ',') {

temp = temp.concat(Character.*toString*(ch));

}**else** {

Images.add(temp);

temp="";

}

}

BasicSessionCredentials sessionCredentials = **new** BasicSessionCredentials(

ses\_input.*getAccessKeyId*(), ses\_input.*getSecretAccessKey*(),

ses\_input.*getSessionToken*());

AmazonRekognition rekognitionClient = AmazonRekognitionClientBuilder.*standard*()

.withCredentials(**new** AWSStaticCredentialsProvider(sessionCredentials))

.withRegion(Regions.***US\_EAST\_1***)

.build();

**for**(**int** j=0 ; j < Images.size() ; j++)

{

System.***out***.println(Images.get(j));

DetectTextRequest requestText = **new** DetectTextRequest()

.withImage(**new** Image()

.withS3Object(**new** S3Object()

.withName(Images.get(j))

.withBucket(*bucket*)));

**try** {

DetectTextResult resultText = rekognitionClient.detectText(requestText);

List<TextDetection> textDetections = resultText.getTextDetections();

**if**(textDetections.size()!=0) {

System.***out***.println("Detected Text for " +Images.get(j) );

bw.write("Detected Text for " +Images.get(j) );

}

**for** (TextDetection text: textDetections) {

System.***out***.println(" Text : " + text.getDetectedText());

bw.write("Text : " +text.getDetectedText() );

bw.newLine();

}

bw.write("--------------------------------");

} **catch**(AmazonRekognitionException e) {

e.printStackTrace();

}

System.***out***.println("---------------------------------------------------------");

}

bw.close();

}

}

***ses\_input.java***

**package** com.amazonaws.samples;

**public** **class** ses\_input {

**static** String

*AccessKeyId*="ASIAWN6CO5PT6GFJR6F4";

**static** String *SecretAccessKey*="pwO+1IZBNQJjF7NXtYKWvbcR8N/RTFuk246ma1Te";

**static** String *SessionToken*="FwoGZXIvYXdzEP3//////////wEaDBDfToLnK/lzvC3ErSK9AX7Fo8/HPY6hicdIoVMfdYGVxhll7WZGZ8q1wkVb2XSqknsrsLMThwx1Dz6UakawH0MEmKfVsFODfOeG7Tk+bqhrlMpckwLbE0GUOuy/+IdsauQm/l/ke/4YlKgdDlZXBwfcH8R5w0cljEIKdbb7gjnVLgvrBz0VbsZl2PYJ+LFJU+fnG4qyvbWfQvdsvDh8fivUS+4xFrr1i/EeiCHpW+lh0Tki2SAfxtdviiuBpi/jGqoKUUUsMqYsXxp6VSiQm5XzBTItcOsPjL/jjMNVB48S17wSyBfL0lng9sl0EdlS1DcIrqmYC2117AJD/MVjXxXg";

**public** **static** String getAccessKeyId() {

**return** *AccessKeyId*;

}

**public** **void** setAccessKeyId(String accessKeyId) {

*AccessKeyId* = accessKeyId;

}

**public** **static** String getSecretAccessKey() {

**return** *SecretAccessKey*;

}

**public** **void** setSecretAccessKey(String secretAccessKey) {

*SecretAccessKey* = secretAccessKey;

}

**public** **static** String getSessionToken() {

**return** *SessionToken*;

}

**public** **void** setSessionToken(String sessionToken) {

*SessionToken* = sessionToken;

}

}