Cloud & Serverless Computing

Project Report

Extract text, structured data and detect signature from images using Amazon textract , Lambda, DynamoDB services

Name: C.Bhavath Ram

ID:2000032221

Sec:15

Introduction

Amazon Textract is a service provided by Amazon Web Services (AWS) that enables users to automatically extract text and structured data from documents. It uses machine learning algorithms to identify and extract key information from a variety of document formats, including PDFs, images, and scanned documents. With Amazon Textract, users can extract data such as text, tables, and form data from documents and store it in a structured format that can be easily analyzed and processed.

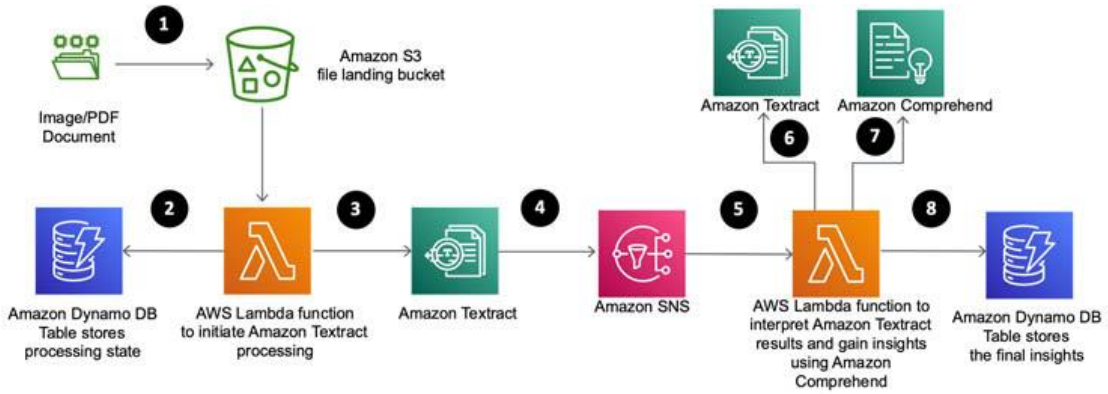
The service supports a variety of languages, including English, Spanish, French, German, Italian, Portuguese, and Japanese. To use Amazon Textract, users can upload their documents to the AWS platform or integrate the service into their existing applications using the AWS API.

Once the documents have been processed, the extracted data can be exported in a variety of formats, including JSON, CSV, and Excel. Some common use cases for Amazon Textract include invoice processing, receipt processing, and document digitization.

By automating the process of data extraction, businesses can save time and improve accuracy, while also freeing up employees to focus on more high-value tasks.

It is important to note that Amazon Textract is a pay-per-use service, and the cost of using the service depends on factors such as the number and complexity of documents being processed.

Block Diagram of the project:



- Amazon S3 - This is a storage service that can be used to store the documents that you want to extract data from. You can configure Amazon Textract to process the documents stored in your S3 bucket.

- AWS Lambda - This is a compute service that can be used to run code in response to events, such as the completion of an Amazon Textract job. You can configure a Lambda function to process the output of an Amazon Textract job and perform further processing or analysis.

- Amazon DynamoDB - This is a NoSQL database service that can be used to store the extracted data from documents. You can configure Amazon Textract to save the extracted data to DynamoDB.

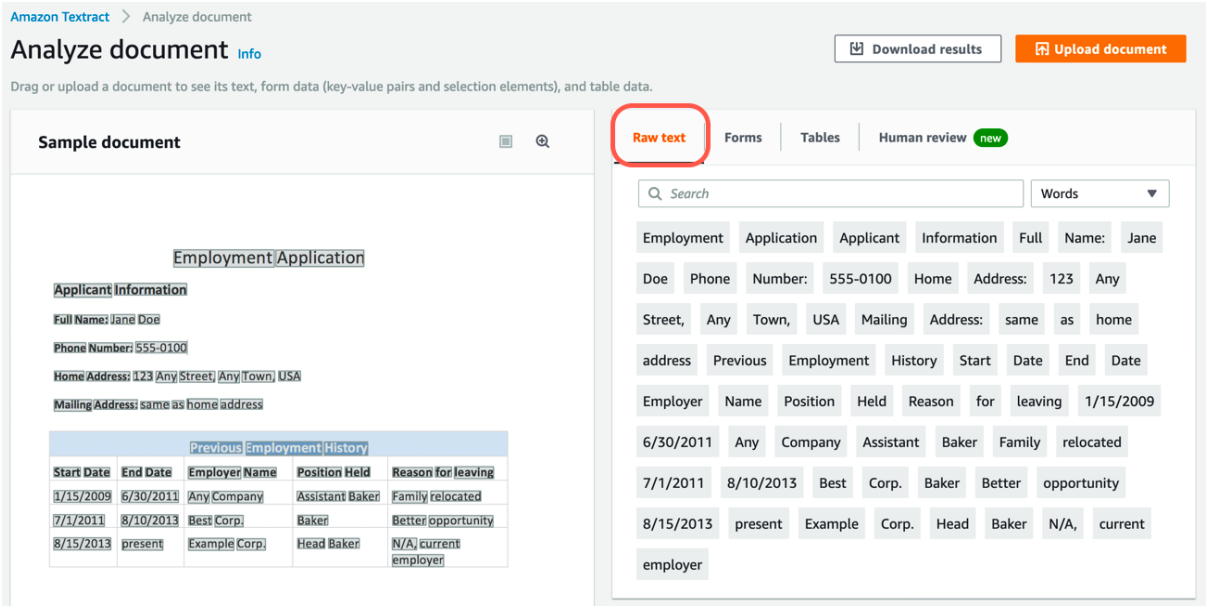
- Amazon Comprehend: You can use Amazon Comprehend to analyze the extracted text and identify entities, key phrases, sentiment, and language.

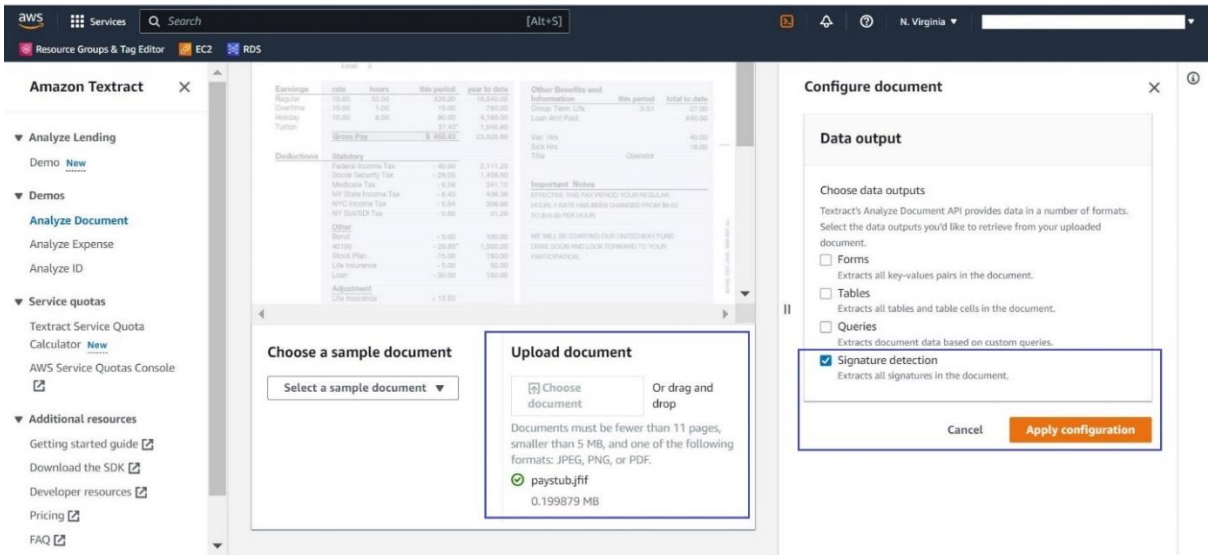
- Amazon SNS - to notify other AWS services or applications about the availability of this data. This can be done by configuring Amazon SNS to publish a notification message to a specific topic when a new document is processed by Amazon Textract

How AnalyzeDocument Signatures detects signatures in documents

The AnalyzeDocument API has four feature types: Forms, Tables, Queries, and Signatures. When Amazon Textract processes documents, the results are returned in an array of Block objects. The Signatures feature can be used by itself or in combination with other feature types. When used by itself, the Signatures feature type provides a JSON response that includes the location and confidence scores of the detected signatures and raw text (words and lines) from the documents.

The Signatures feature combined with other feature types, such as Forms and Tables, can help draw useful insights. In cases where the feature is used with Forms and Tables, the response shows the signature as part of key value pair or a table cell. For example, the response for the following form contains the key as Signature of Lender and the value as the Block object.

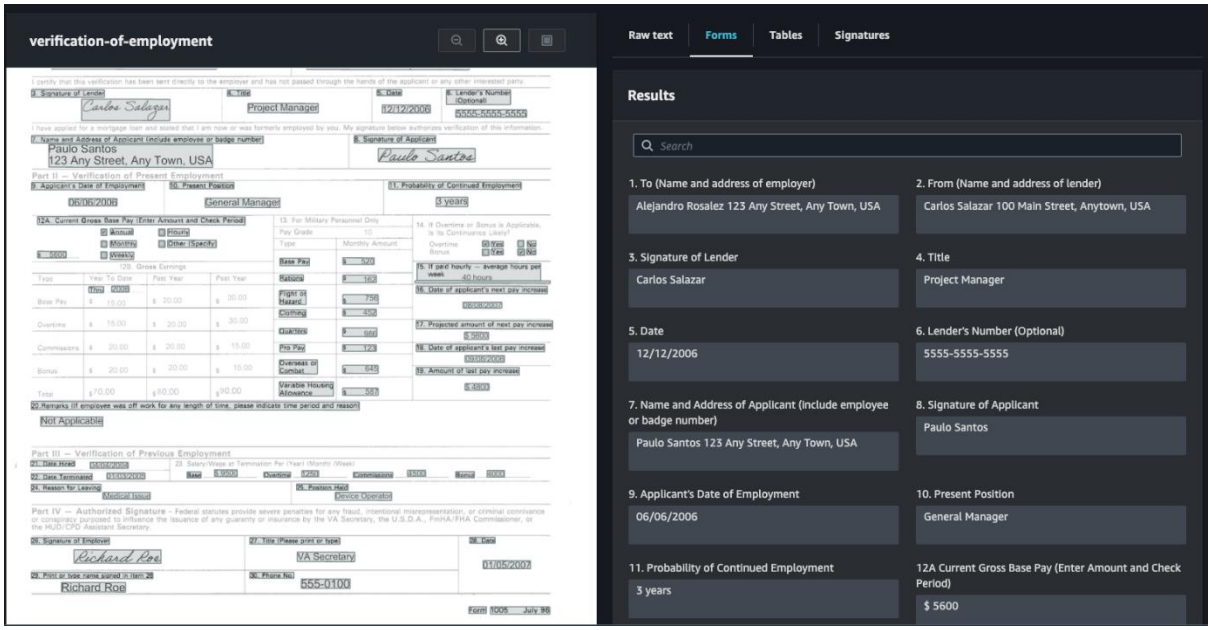


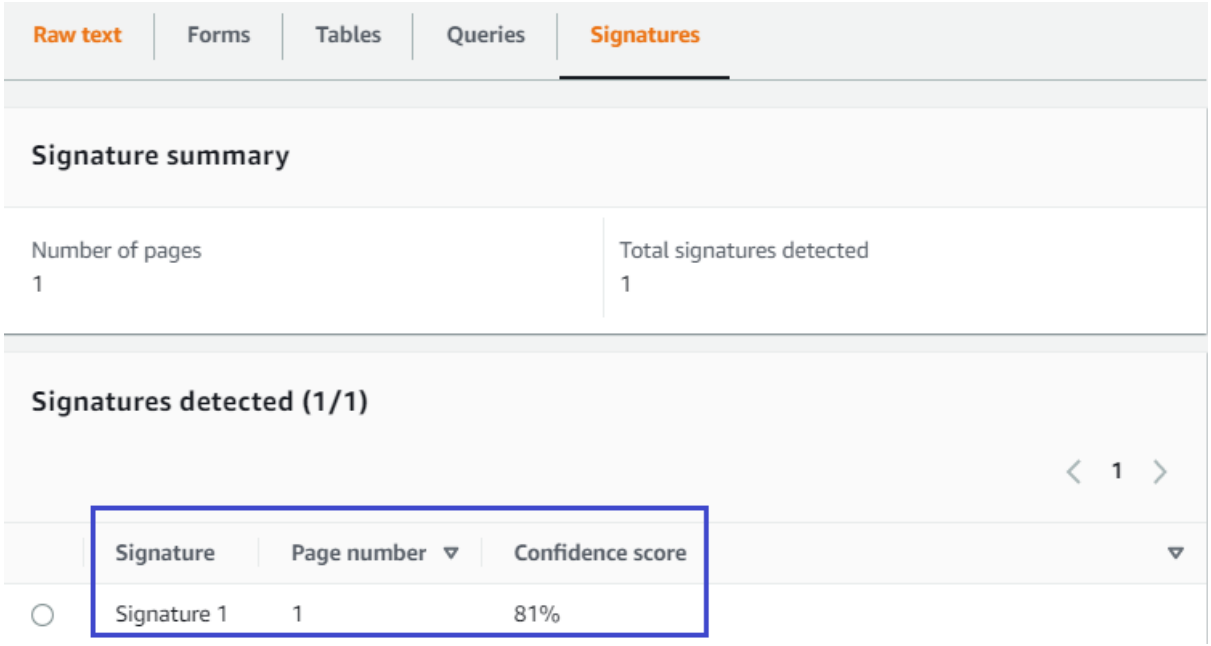


How to use the Signatures feature on the Amazon Textract console

Before we get started with the API and code samples, let's review the Amazon Textract console. After you upload the document to the Amazon Textract console, select Signature detection in the Configure document section and choose Apply configuration.

The following screenshot shows an example of a paystub on the Signatures tab for the Analyze Document API on the Amazon Textract console.





The feature detects and presents the signature with its corresponding page and confidence score.

We use the following sample Python code:

```
import boto3

import json



#create a Textract Client

textract = boto3.client('textract')

#Document

documentName = image_filename



response = None
```

```
with open(image_filename, 'rb') as document:
    imageBytes = bytearray(document.read())



# Call Textract AnalyzeDocument by passing a document from local disk
response = textract.analyze_document(
    Document={'Bytes': imageBytes},
    FeatureTypes=["FORMS",'SIGNATURES']

    )
```

Let's analyze the response we get from the AnalyzeDocument API. The following response has been trimmed to only show the relevant parts. The response has a `BlockType` of `SIGNATURE` that shows the confidence score, ID for the block, and bounding box details:

```
'BlockType': 'SIGNATURE',
  'Confidence': 38.468597412109375,
  'Geometry': {'BoundingBox': {'Width': 0.15083004534244537,
    'Height': 0.019236255437135696,
    'Left': 0.11393339931964874,
    'Top': 0.8885205388069153},
   'Polygon': [{'X': 0.11394496262073517, 'Y': 0.8885205388069153},
    {'X': 0.2647634446620941, 'Y': 0.8887625932693481},
```

{'X': 0.264753133058548, 'Y': 0.9077568054199219},

{'X': 0.11393339931964874, 'Y': 0.907513439655304}]},

'Id': '609f749c-5e79-4dd4-abcc-ad47c6ebf777'}]

Form and table extraction and processing

Amazon Textract can provide the inputs required to automatically process forms and tables without human intervention. For example, a bank could write code to read PDFs of loan applications.

The information contained in the document could be used to initiate all the necessary background and credit checks to approve the loan so that customers can get instant results for their application rather than having to wait several days for manual review and validation.

The following image is an employment application with form fields, check boxes, and a table.

```
|------------------|-----------------------------|
| Key              | Value                       |
| Website          | NOT_SELECTED                |
| Full Name:       | Jane Doe                    |
| Job fair         | SELECTED                    |
| Company Employee | NOT_SELECTED                |
| Mailing Address: | same as above               |
| Home Address:    | 23 Any Street, Any Town. USA |
| Phone Number:    | 555-0100                    |

|------------|-----------|---------------|--------------------|--------------------|
|            |           | Previous      | Employment History |                    |
| Start Date | End Date  | Employer Name | Position Held      | Reason for leaving |
| 1/15/2009  | 6/30/2011 | Any Company   | Assistant baker    | relocated          |
| 7/1/2011   | 8/10/2013 | Example Corp. | Baker              | better opp.        |
| 8/15/2013  | Present   | AnyCompany    | head baker         | N/A, current       |
```

The key-value pairs from the `FORMS` output are rendered as a table with `Key` and `Value` headlines to allow for easier processing.

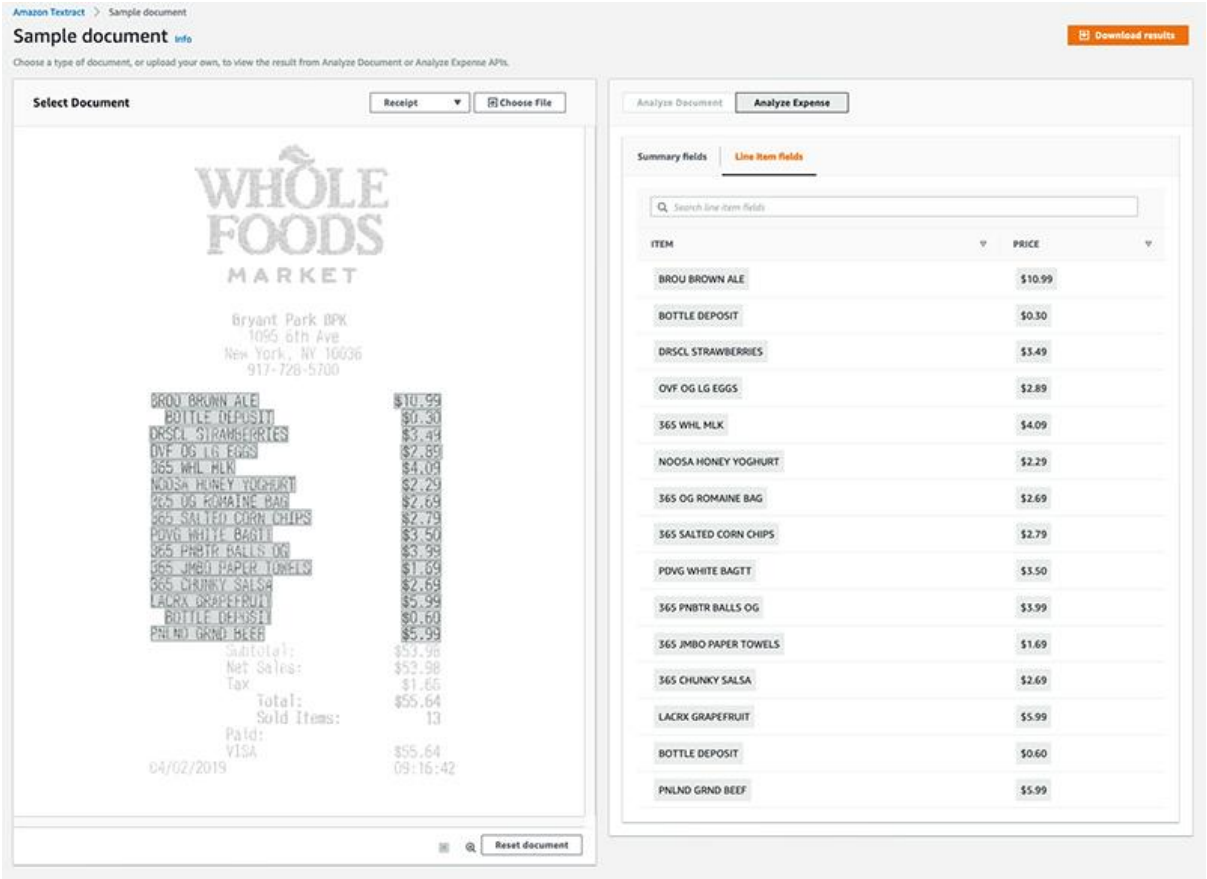
For example, changing the output format by including `—pretty-print-table-format=csv parameter` outputs the data in CSV format (check `amazon-textract —help` for a list of other formats

Extract information from invoices and receipts

Invoices and receipts are difficult to process at scale because they follow no set design rules, yet any individual customer encounters thousands of distinct types of these documents. The Amazon Textract AnalyzeExpense action identifies standard fields and line-item details for these document types.

The standard fields supported include “Vendor Name”, “Total”, “Receiver Address”, “Invoice/Receipt Date”, “Invoice/Receipt ID”, “Payment Terms”, “Subtotal”, “Due Date”, “Tax”, “Invoice Tax Payer ID”, “Item Name”, “Item Price”, “Item Quantity” plus line-item details. For a complete list check the Analyzing Invoices and Receipts documentation.

The AWS Management Console offers options to test the AnalyzeExpense action through the “Select Document” options “Receipt” (image below) or “Invoice” or by “Choose File” option. The latter allows uploading of a document and subsequent selection of “Analyze Expense” in the output tab on the right side. Through “Download results” a zip file including the line-item fields and summary fields can be received.



Conclusion

Use the confidence scores provided with the detected signatures to route the documents for human review when the scores don’t meet your required threshold. The confidence score is not a measure of accuracy, but an estimate of the model’s confidence in its prediction. You should select a confidence score that makes the most sense for your use case.

For real-time responses, use the synchronous operation of the AnalyzeDocument API. For use cases where you don’t need the response in real time, such as batch processing, we suggest using the asynchronous operation of the API.

The Signatures feature works best when there are up to three signatures on a page. When there are more than three signatures on a page, it’s best to split the page into sections and feed each of the sections separately to the API.