Module 25
Amazon Textract

In this module we will use another Machine Learning service for detecting text (even hand-written ones). We can think of many possible scenarios where something like this is useful (e.g., digitizing and making "old" hand-written documents searchable).

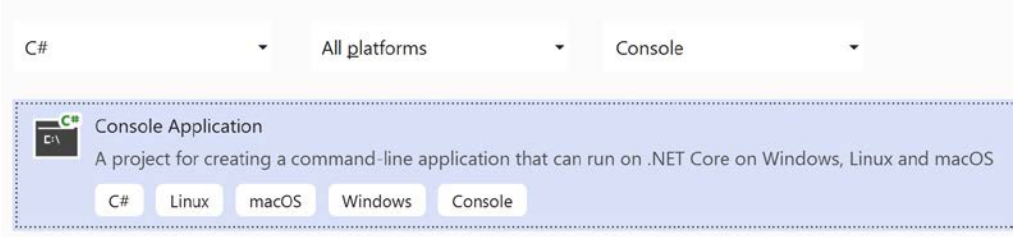
First read about the service here.

Using the Textract SDK to Detect Text

1. The enclosed "Documents" folder contains documents you can test with. One of the document is a hand-written note I copied from Bing images. You can also copy your own and try them out.

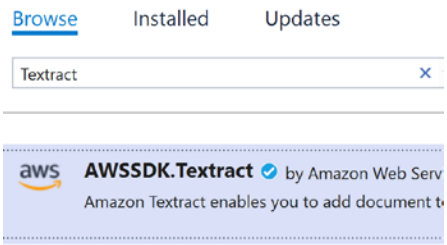
2. Create a Visual Studio solution and name it **TextractSolution**.

3. To the above solution, add a project of type Console Application. Give the project the name **AnalyzeDocument**.

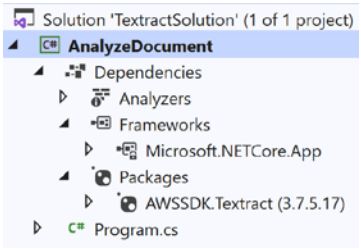


4. In the solution tree in Visual Studio, right-click on Dependencies and choose menu **Manage NuGet Packages…**

5. While Browse is selected, search for Textract. Select it and click the **Install** button on the pane shown on the right. Then click **OK** on the Preview Changes dialog to accept the DLL libraries to be added to the project (AWSSDK.Core and AWSSDK.Textract)



You project structure should now look like this:



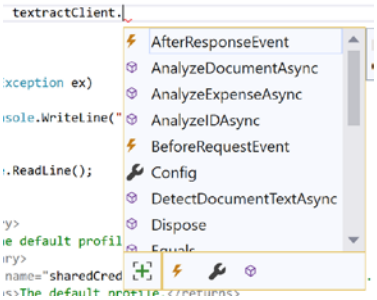
6. Add these namespaces at the top of Program.cs

```
using System;
using System.IO;
using System.Threading;
using System.Threading.Tasks;
using Amazon.Runtime;
using Amazon.Runtime.CredentialManagement;
using Amazon.Textract;
using Amazon.Textract.Model;
```

7. You can create an `AmazonTextractClient` object in a way similar to how we created client objects to other AWS services (the pattern is similar). The `AmazonTextractClient` is the object you need to access the functionalities of the Amazon Textract service. For example, you can do something like:

```
static void Main(string[] args)
{
    try
    {
        // Constructs a SharedCredentialsFile object from the default credentials file.
        SharedCredentialsFile sharedCredentialsFile = new SharedCredentialsFile();

        // Get the [default] profile from the credentials file.
        CredentialProfile defaultProfile = GetDefaultProfile(sharedCredentialsFile);

        if (defaultProfile != null)
        {
            // Get the credentials (access key, secret access key, etc.)
            AWSCredentials credentials = AWSCredentialsFactory.GetAWSCredentials(defaultProfile, new SharedCredentialsFile());

            AmazonTextractClient textractClient = new AmazonTextractClient(credentials, Amazon.RegionEndpoint.USEast1);


        }
    }
    catch(Exception ex)
    {
        Console.WriteLine("ERROR: {0}", ex.Message);
    }

    Console.ReadLine();
}
```

8. Now if you type `textractClient.`, you should see some of the methods that you can use.



9. Add the following utility function to the Program class.

```
private static byte[] FileToByteArray(string filePath)
{
    byte[] byteArray = null;

    using (FileStream fs = new FileStream(filePath, FileMode.Open, FileAccess.Read))
    {
        using (BinaryReader reader = new BinaryReader(fs))
        {
            byteArray = reader.ReadBytes((int)(new FileInfo(filePath).Length));
        }
    }

    return byteArray;
}
```

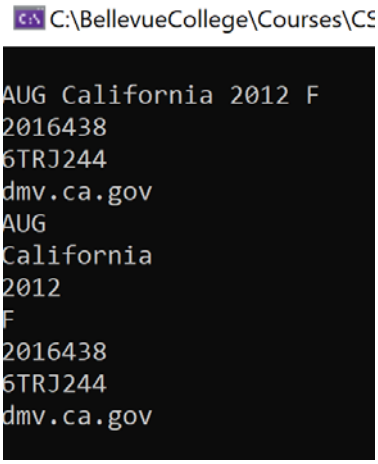
10. The above method helps with initializing a `DetectDocumentTextRequest` object. For example:

```
string filePath = @"C:\BellevueCollege\Courses\CS455\Modules\Module25-MachineLearning5\Documents\HandWrittenDocument.png";
byte[] docBytes = FileToByteArray(filePath);
Document doc = new Document();
doc.Bytes = new MemoryStream(docBytes);

DetectDocumentTextRequest req = new DetectDocumentTextRequest()
{
    Document = doc
};
```

11. Try this:

```
Task<DetectDocumentTextResponse> res = textractClient.DetectDocumentTextAsync(req, new System.Threading.CancellationToken());
while(true)
{
    if(res.Status == TaskStatus.RanToCompletion)
    {
        // See what's inside the document
        foreach(Block b in res.Result.Blocks)
        {
            Console.WriteLine(b.Text);
        }

        break;
    }

    Thread.Sleep(1000);
}
```

12. Experiment with the license plates. For example, this is what license plate LicensePlate1.jpg gave me:



You can use this service in Project3 (instead of AWS Rekognition).

What to Submit:

Nothing to submit for this module