Microsoft

Develop solutions with Azure AI Document Intelligence

Agenda

- Use prebuilt Document Intelligence models
- Train a custom Document Intelligence model

Develop a Document Intelligence solution

Learning Objectives

After completing this module, you will be able to:

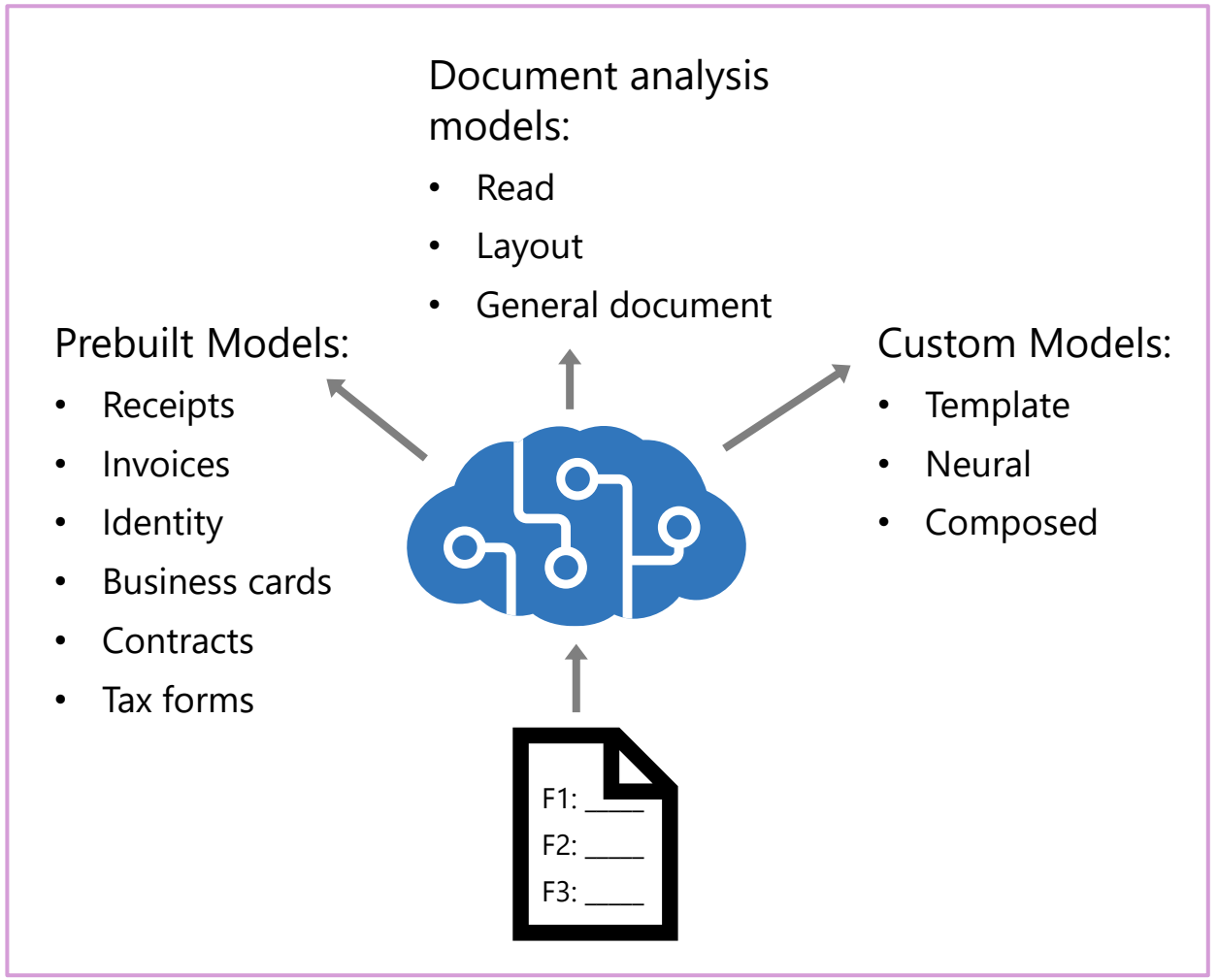
1. Understand models in Azure AI Document Intelligence
2. Train a custom Document Intelligence model
3. Connect an app to Document Intelligence APIs

The Document Intelligence Service

Data extraction from forms and documents:

- Document analysis from general documents
  - Read: OCR for printed and written text
  - Layout: Extract text and structure
  - General document: Extract text, structure, and key-value pairs
- Prebuilt models for common form types
- Train custom models for your own forms
  - Custom template: Extract data from static layouts
  - Custom neural: Extract data from mixed-type documents
  - Custom composed: Collection of multiple models assigned to a single model

Provision as single-service **Document Intelligence** resource or multi-service **Azure AI Services** resource



Prebuilt models

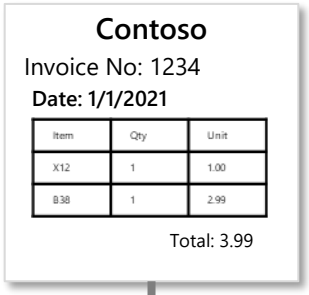
Receipt

Analyze Receipt

Get Analyze Receipt Result

```
{
  "MerchantName": "Fourth Coffee",
  "TransactionDate": 2021-01-01,
  "TransactionTime": 09:34,
  "Items" [
    ...],
  "Total": 3.75
}
```

Invoice



Analyze Invoice

Get Analyze Invoice Result

```
{
  "VendorName": "Contoso",
  "InvoiceNumber": 1234,
  "InvoiceDate": 2021-01-01
  "Tables" [
    ...],
  "TotalInvoiceAmount": 3.99
}
```

Business Card

Analyze Business Card

Get Analyze Business Card Result

```
{
  "ContactNames": [
    {
      "FirstName": "Hank",
      "LastName": "Zoeng"
    }],
    ...
}
```

Calling the API

- Each request is configured with your resource endpoint and needs your resource key
- Send the request, which when successful returns a poller to get the results
  - REST returns it in `Operation-Location` header
  - SDKs return an object from the request
- Query the poller received for the extracted data

REST

```
Request POST:
{endpoint}/documentintelligence/documentModels/prebuilt-layout:analyze?api-version={version}

Operation-Location:
{endpoint}/documentintelligence/documentModels/prebuilt-layout/analyzeResults/ab12345c-12ab-23cd-b19c-2322a7f11034?api-version={version}
```

C#

```
AnalyzeDocumentOperation operation = await client.AnalyzeDocumentFromUriAsync(WaitUntil.Completed, "prebuilt-layout", fileUri);

AnalyzeResult result = operation.Value;
```

Python

```
poller=document_analysis_client.begin_analyze_document_from_url("prebuilt-document", docUrl)

result = poller.result()
```

API response

- Response it broken down by page, lines, and words
- Subset of REST response included here
- SDK response objects have similar structure, broken down similarly
- Additional data about detected text or selection marks, such as bounding box and handwritten style

```
{
    "analyzeResult": {
        "apiVersion": “{version}",
        "modelId": "prebuilt-invoice",
        ...
        "pages": [{
            "pageNumber": 1,
            "angle": 0,
            "width": 8.5,
            "height": 11,
            "unit": "inch",
            "words": [{
                "content": “Margie’s",
                "boundingBox": [
                    0.5911,
                    0.6857,
                    1.7451,
                    0.6857,
                    1.7451,
                    ...
                ],
                "confidence": 1,
                "span": {...}
            }],
        }]
    }
}
```

[Optional] – Use prebuilt Document Intelligence models

Use the Read model

Use an app to use document analysis

Types of custom models

Custom classification

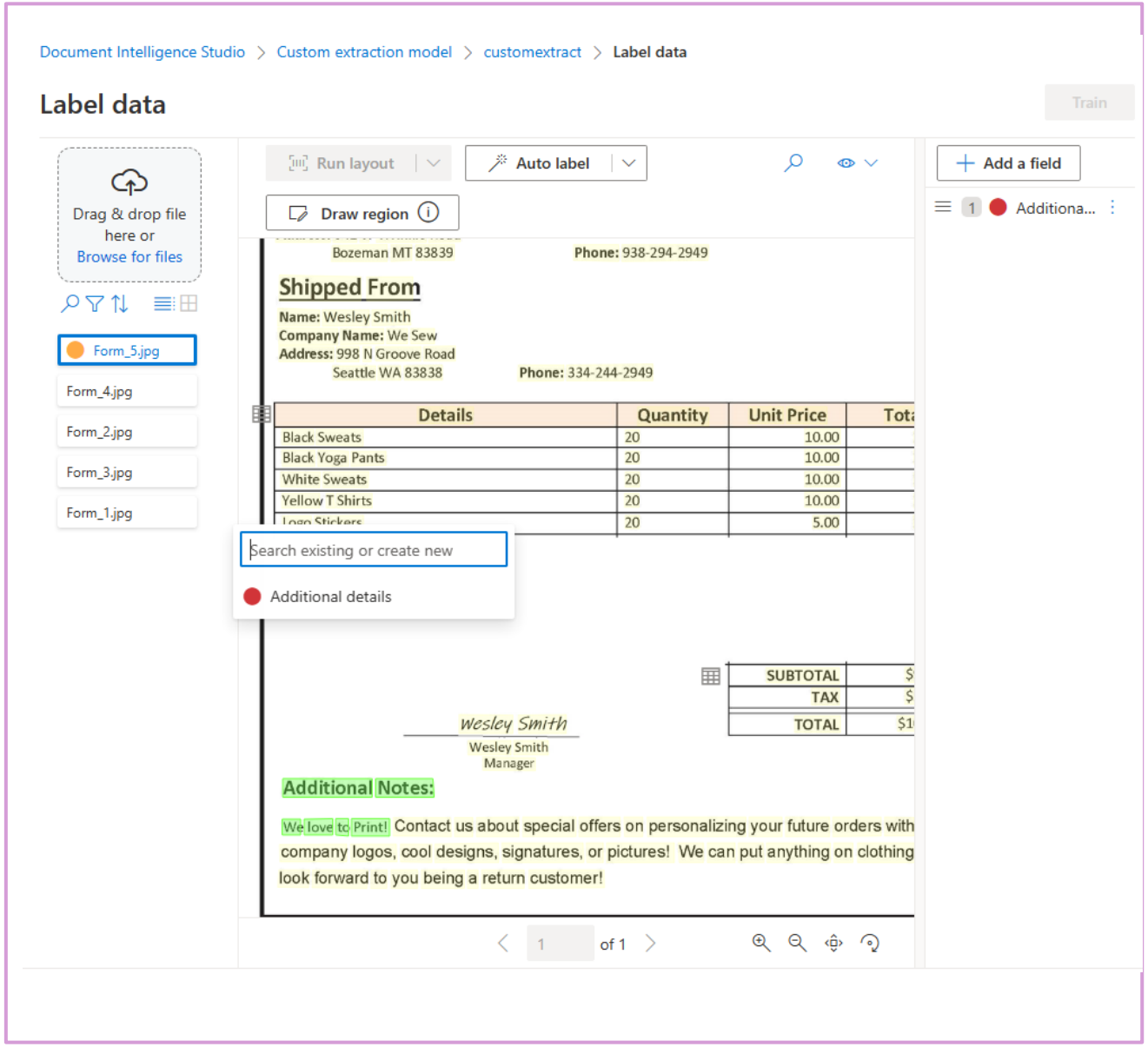
- Apply a label to the entire document
- Ideal for sorting large numbers of incoming documents into types
- Requires two different classes, and a minimum of five labeled documents per class
- One type of training model

Custom extraction

- Apply label to specific text
- Ideal for extracting custom labels from documents
- Requires five examples of the same document type
- Two training methods:
  - **Custom template (custom form)**
    - Training time: 1-5 minutes
    - Document structure: forms, templates, other structured documents
  - **Custom neural (custom document)**
    - Training time: 20-60 minutes
    - Document structure: structured and unstructured documents

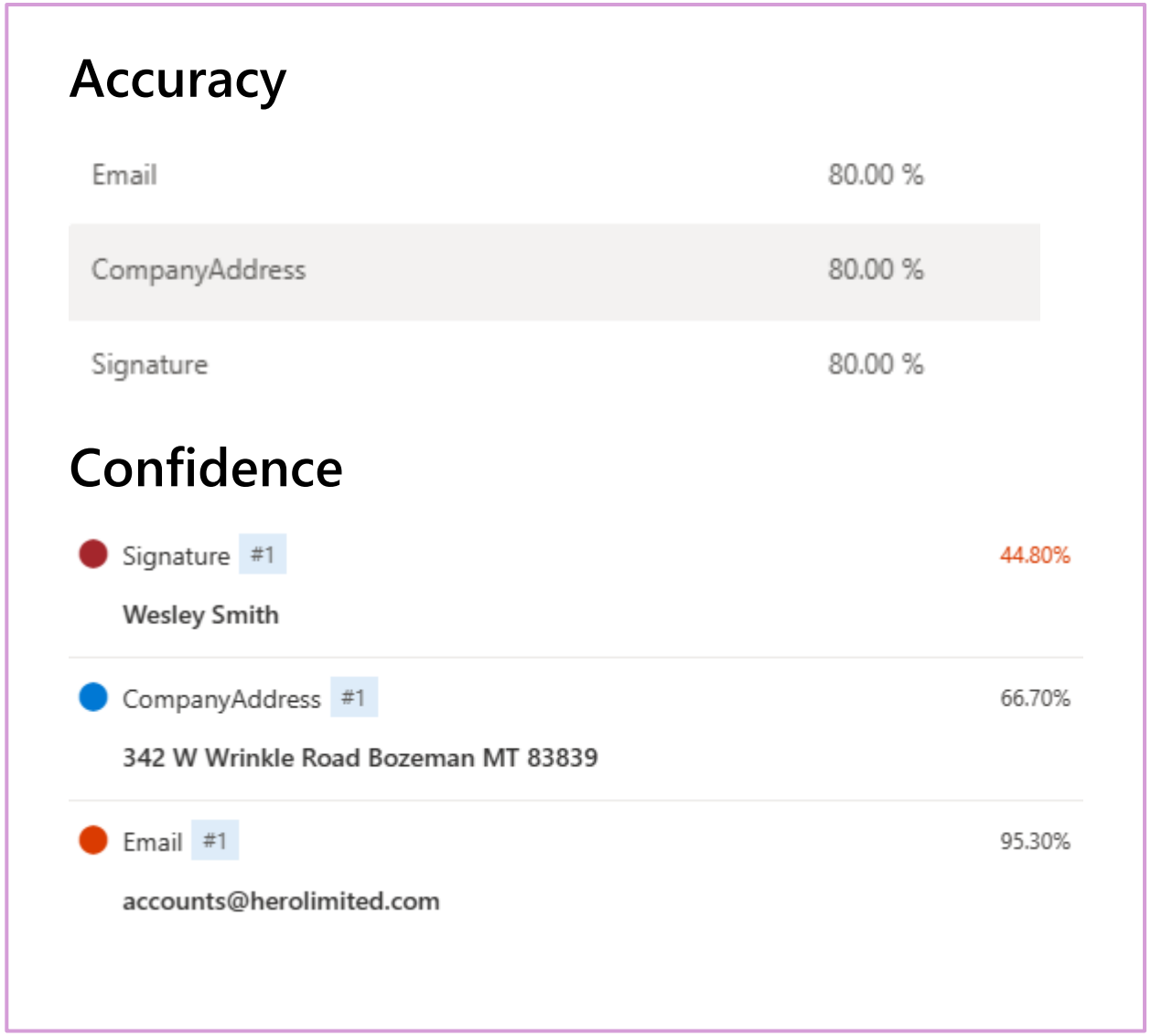
Training Custom Models

1. Create project and upload training files to your project, or connect to blob storage containing files
2. Add data type (such as field or signature) to start labeling your dataset
3. Select a word in the document, and assign one of the fields to label it
4. Repeat for all fields and files in your dataset
5. Layout and auto label (using a prebuilt model) can assist in this process
6. Train the model, providing a Model ID used in API requests



Accuracy and confidence scores

- After training, a custom model has an estimated accuracy score
- Score is calculated by running combinations of training data predictions against the labeled values
- Confidence score is the same as using prebuilt models, indicated how accurate the model thinks that specific prediction is
- Confidence scores are provided in the response from the model for each predicted label



Analyze document using custom model

- Requires endpoint and key from deployed resource, similar to prebuilt models
- Needs to also include the ID of your deployed custom model
- Query the poller received for the extracted data

C#

```
AnalyzeDocumentOperation operation = await
client.AnalyzeDocumentFromUriAsync(WaitUntil.Completed,
{modelId}, {fileUri});

AnalyzeResult result = operation.Value;
```

Python

```
poller = document_analysis_client.begin_analyze_document(
        model_id={modelId}, document={fileUri}

result = poller.result()
```

Lab – Extract Data from Forms

Train a custom model in Document Intelligence Studio

Test your custom Document Intelligence model

Knowledge check

1 **You have scanned a letter into PDF format and need to extract the text it contains. What should you do?**

- ☐ Use the Image Analysis feature in Azure AI Vision.
- ☑ Use the Read model in the Document Intelligence service.
- ☐ Use a custom model in the Document Intelligence service.

2 **You need to build an application that submits expense claims, extracting the merchant, date, and amount from scanned receipts. What's the best way to do this?**

- ☐ Use the general document model.
- ☐ Use the prebuilt Contract model.
- ☑ Use the prebuilt Receipt model.

3 **You need to extract only data from specific fields in cargo manifest forms using Document Intelligence. What should you do?**

- ☐ Use a prebuilt model.
- ☐ Build a custom composed model from several custom models.
- ☑ Train custom template model with labeled documents.

Learning Path Recap

In this learning path, we:

Explored available prebuilt models, and how to use them in Document Intelligence Studio

Trained and deployed a custom model

Connected an app to use Document Intelligence APIs