AI Workflow Automation for Document Processing

Raj Pathak
Solutions Architect, Amazon Web Services

The AWS ML stack

Broadest and most complete set of machine learning capabilities

AI SERVICES

Vision
Rekognition

Chatbots
Lex

Business tools
Personalize Forecast
Fraud Detector
Lookout for Metrics

Search
Kendra

Healthcare
HealthLake
Comprehend Medical
Transcribe Medical

Speech
Polly
Transcribe

Text
Comprehend
Translate
Textract

Contact centers
Contact Lens
Connect Voice ID

Code + DevOps
CodeGuru
DevOps Guru

Industrial
Panorama Appliance and SDK, Monitron, Lookout for Equipment, Lookout for Vision

AMAZON SAGEMAKER

Label data

SAGEMAKER STUDIO IDE

Data collection prep
Store features
Detect bias and explain predictions
Visualize in notebooks
Pick algorithm
Train models faster
Tune parameters
Deploy in production
Manage and monitor

CI/CD

Manage edge devices

ML FRAMEWORKS AND INFRASTRUCTURE

TensorFlow, PyTorch, Apache MXNet
Deep learning AMIs and containers
GPUs
Inferentia
Elastic inference
FPGA

Documents are a primary business tool across industries

- Finance
- Insurance
- Real estate
- Accounting
- Tax management
- Business operations
- Medical
- Legal
- Education
- And many more...

How documents are processed today

Manual processing

- Expensive
- Error prone
- Time consuming

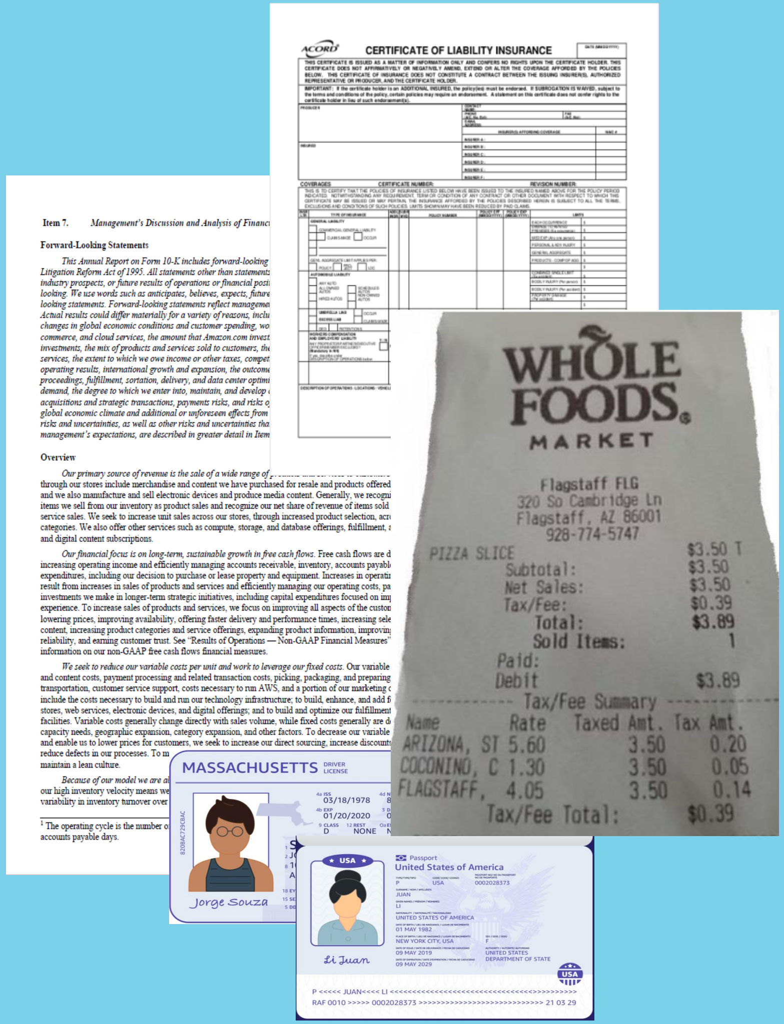
Traditional Optical Character Recognition (OCR)

- Simple documents only
- Error prone
- Dump of text

Rules and template-based extraction

- Limited by OCR accuracy
- Development and management overhead
- Templates are brittle

Why do these challenges exist



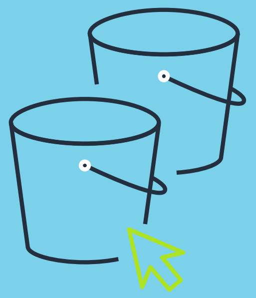
Documents are varied, and requirements for data are unique

Simply extracting data has become table stakes, adding insights and structure is what will provide organizations value from their documents

Solving these challenges with Intelligent Document Processing

Let’s start by modelling the process into there unique phases of the document processing lifecycle

Ingestion



Ingest documents into centralized document repository from different sources (email, upload, fax, scan etc.)

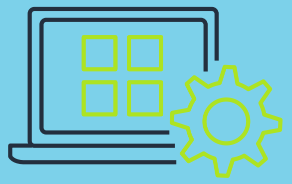
Extraction & Classification

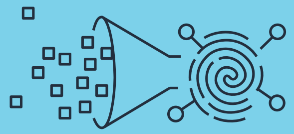
Extract data and classify documents

Post-Processing and Transformation

Validate and transform results from extraction into system ready requests

Storage and Workflow Automation





Ingest data into backend systems for storage or workflow automation

Ingestion

The first step in the IDP lifecycle, ingestion allows us to store the document, and tag it with the appropriate metadata preparing it for processing

Using Amazon S3 as a landing zone for documents allows us to

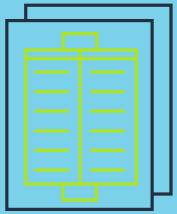
- Tag documents with important metadata (timestamp, sender, document format) with Amazon S3 Object Tagging
- Allows for versioning and encryption of documents
- Provides low cost of storage, and storage tiering for archival data
- 11 9's of data durability
- WORM configurations can be applied to documents
- Built in integrations to AWS Services

Extraction & Classification

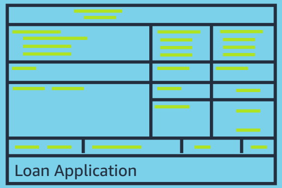
The next step in our IDP workflow, Extraction and Classification, we will extract data off different documents and classify our results

Using Amazon Textract will allow **for extraction of data from complex documents** and Amazon Comprehend will allow for **granular classification and insight generation** from extracted data

Amazon Textract capabilities



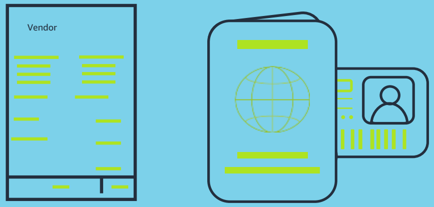
Text



Forms

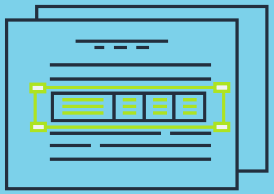
Invoices and receipts

Identity documents



Specialized documents

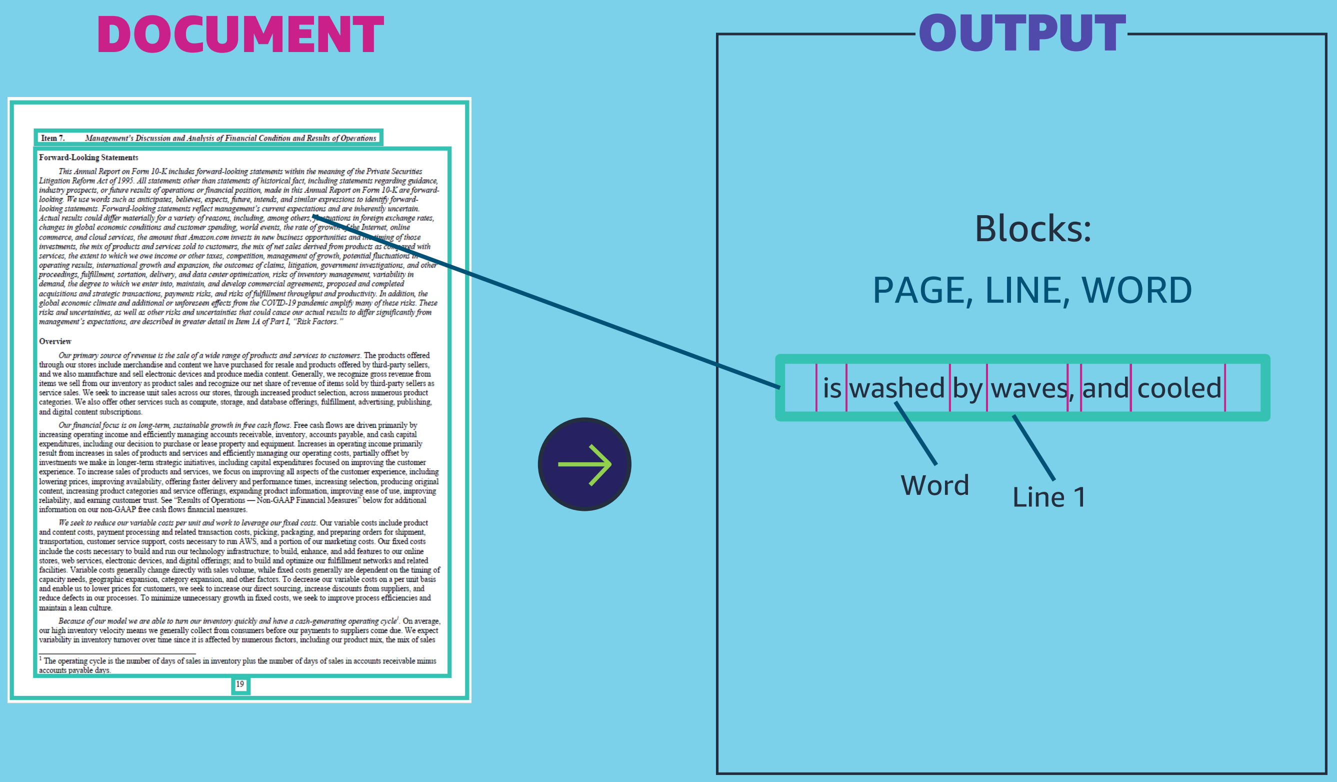
Handwriting



Tables

Text extraction

Optimal for dense text extraction with industry leading OCR accuracy



- Outputs detected text in 3 hierarchy blocks: Page, Lines and Word
- Bounding box for each Line and Word provide visual cues for post-processing
- Included Confidence scores enable informed decision making for your workflows

Table extraction

Extract tables from documents while preserving data structure and relationships

DOCUMENT

Previous employment history				
Start date	End date	Employer name	Position held	Reason for leaving
1/15/2009	6/30/2013	Any company	Head Baker	Family relocated
8/15/2013	Present	Example corp.	Baker	N/A, current employer

OUTPUT

Blocks:

PAGE, TABLE, CELL

For each 'Block' you get:

TEXT

CONFIDENCE SCORE

BLOCK RELATIONSHIPS
(for example, cells within a table)

Page
Table
Cell
Word

Outputs recognized tables with relationships data intact

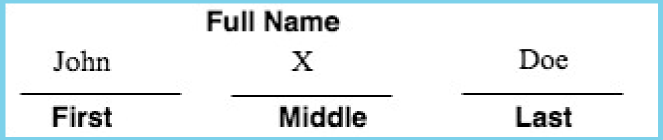
Intelligently groups cells within tables and words within each cell

Output also includes confidence scores, geometry info, and row/column indexes

Form extraction

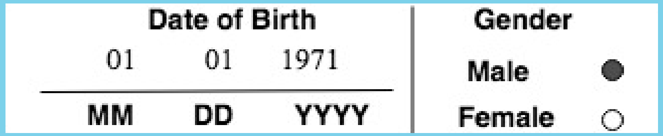
Extract form data from documents as key-value pairs to preserve document structure

DOCUMENT



Full Name

First	Middle	Last
John	X	Doe



Date of Birth

MM	DD	YYYY
01	01	1971

Gender

Male ●

Female ○

OUTPUT

Blocks:

PAGE, KEY_VALUE_SET

Example Output:

First: John

Middle: X

Last: Doe

MM: 01
DD: 01
YYYY: 1971

Male: True
Female: False

- Outputs form field name (Key) and field value name (Value) with relationship data intact
- Captures logical groupings, relationships, and glyphs
- Output also includes confidence scores, and geometry info

Invoices and receipts

Specialized support to process invoices and receipts at scale

DOCUMENT

OUTPUT

Summary Fields:
Vendor Name: WHOLE FOODS MARKET
Subtotal (SUBTOTAL): $3.50
Net Sales (OTHER): $3.50
Tax/Fee (TAX): $0.39
Sold Items (OTHER): 1
Paid (OTHER):
Debit (OTHER): $3.89
Tax/Fee Total (TAX): $0.39
Total (TOTAL): $3.89

Line Items:
ITEM: Pizza Slice
PRICE: $3.50

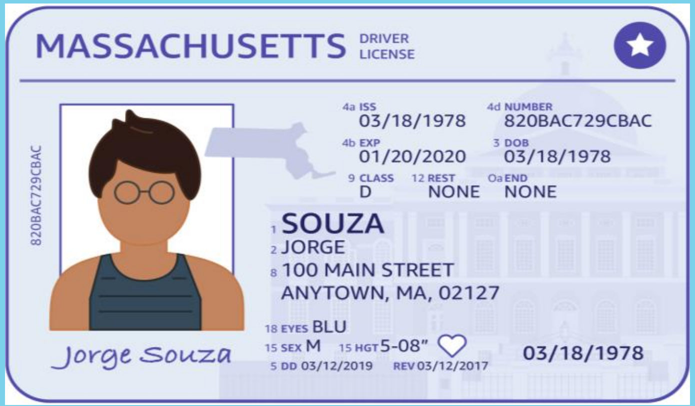
- Outputs headline amounts, line item details and inferred fields (like Vendor Name)
- Supports any style of invoice or receipt
- No templates or configuration required

Identity documents

NEW!

Specialized support for identity documents

DOCUMENT



OUTPUT

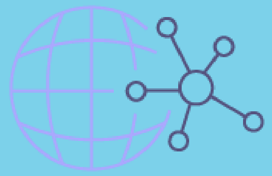
First Name: JORGE
Last Name: SOUZA
Middle Name:
Address Line1: 100 MAIN STREET
Address Line 2:
City: ANYTOWN
State: MA
Document Number: 820BAC729CBAC
Expiration Date: 01/20/2020
Date of Birth: 03/18/1978
ID Type: Driver License
Date of Issue: 03/18/1978
Issued By: MASSACHUSETTS
Class: D
Restrictions: NONE
Endorsements: NONE

95%+ accuracy for US driver licenses and passports

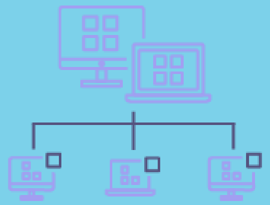
No templates or configuration required

Outputs normalized field names and supports implied elements

Amazon Comprehend IDP capabilities



Named Entity Recognition (NER)



Document Classification

PII Detection and Redaction

Entity Detection

Amazon.com, Inc. is located in Seattle, WA and was founded July 5th, 1994 by Jeff Bezos. Known to the most customer obsessed organization, it welcomes thousands of customers and partners to one of its flagship events AWS re:Invent every year.

Amazon.com, Inc.	Entity: ORGANIZATION
Seattle, WA	Entity: LOCATION
July 5th, 1994	Entity: DATE
Jeff Bezos	Entity: PERSON
thousands of customers	Entity: QUANTITY
re:Invent	Entity: EVENT

Entities detected: Person, Organization, Location, Date, Quantity, Title, Commercial Item, Event, Other

Custom Entity Detection

Customize entity detection to your specific requirements by training an AutoNLP model

"I never received the shipment for part number **XT1764PY98**"

XT1764PY98: PART_NUMBER

"Trading of **ETFs** was halted today"

ETF: INVESTMENT_VEHICLE

"I liked the **shwarma** but loved the **hummus**"

shwarma, humus: MENU_ITEMS

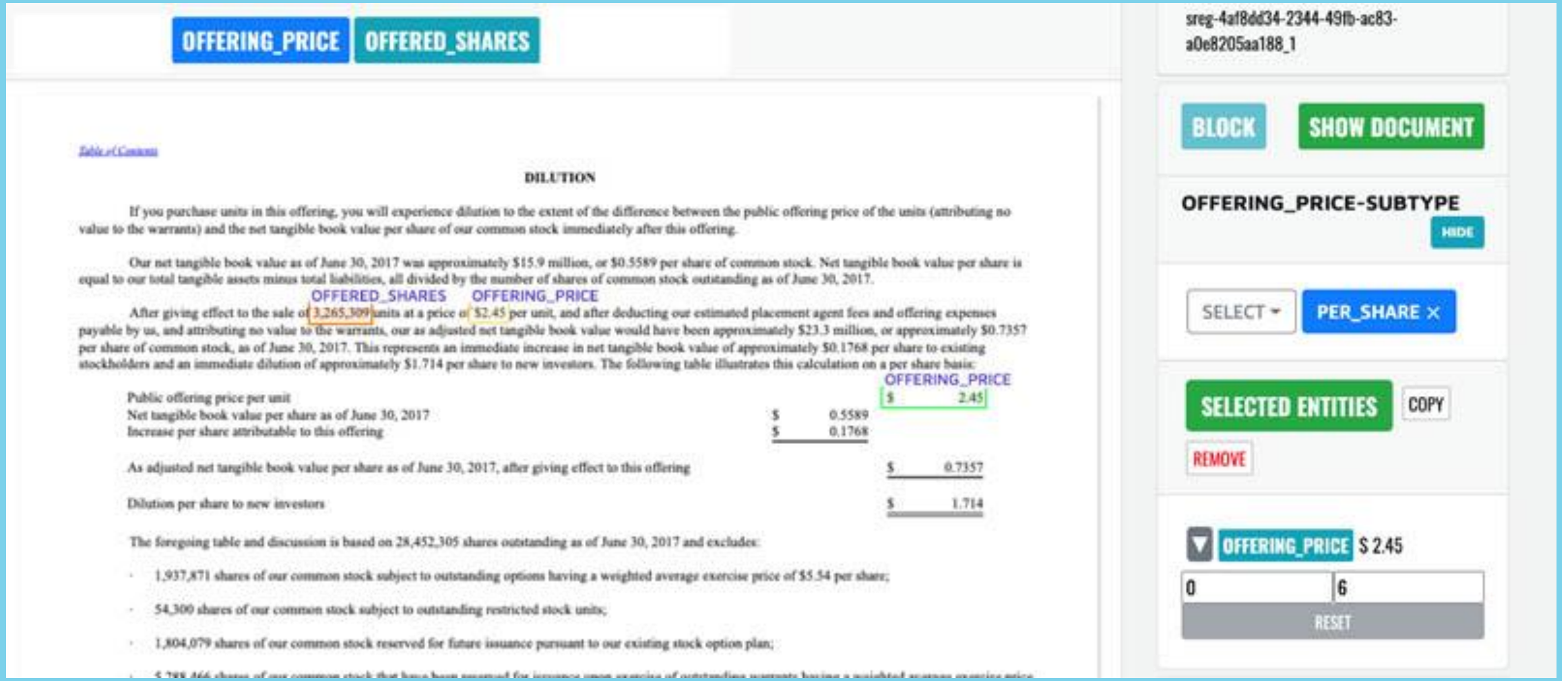
Custom Entity Detection with spatial information

NEW!

Custom NER enhancements + document format support

Comprehend entity detection support for semi-structured documents

- Use positional and structural context in addition to natural language context to build NER Models
- Bullets, tables, forms, and more
- Don’t have to develop ‘flattening’ strategies
- Pre-integrates Amazon Textract and Amazon Comprehend



Custom Classification

Classify documents to your specific requirements by training an AutoNLP classification model

Multi-class

- Classes are mutually exclusive
- Classification into one class
- Each model allows 1000 classes

Multi-label

- Classes are not mutually exclusive
- Classification into one or more classes
- Each model allows 100 unique classes

PII Detection

Hi, my name is John Doe. For verification, the last 4 digits of my social are 6789 and my dob is 01/01.

I paid for my credit card 1111-0000-1111-0000 last week from my bank account XXXXXX1111 with the routing number XXXXX0000. The check was mailed from 100 Main Street, Anytown, WA 98121.

Please confirm receipt by calling me at 206-555-0199 or emailing at john.doe@anycompany.com.

John Doe	Entity: NAME
6789	Entity: SSN
01/01	Entity: DATE_TIME
1111-0000-1111-0000	Entity: CREDIT_DEBIT_NUMBER
XXXXXX1111	Entity: BANK_ACCOUNT_NUMBER
XXXXX0000	Entity: BANK_ROUTING
100 Main Street, Anytown, WA 98121	Entity: ADDRESS
206-555-0199	Entity: PHONE
john.doe@anycompany.com	Entity: EMAIL

PII handing options: Detect, Redact, Mask

Detect

Hi, my name is John Doe. For verification, the last 4 digits of my social are 6789 and my dob is 01/01.

I paid for my credit card 1111-0000-1111-0000 last week from my bank account XXXXXX1111 with the routing number XXXXX0000. The check was mailed from 100 Main Street, Anytown, WA 98121.

Please confirm receipt by calling me at 206-555-0199 or emailing at john.doe@anycompany.com.

Redact

Hi, my name is [NAME]. For verification, the last 4 digits of my social are [SSN] and my dob is [DATE_TIME].

I paid for my credit card [CREDIT_DEBIT_NUMBER] last week from my bank account [BANK_ACCOUNT_NUMBER] with the routing number [BANK_ROUTING]. The check was mailed from [ADDRESS].

Please confirm receipt by calling me at [PHONE] or emailing at [EMAIL].

Mask

Hi, my name is ********. For verification, the last 4 digits of my social are **** and my dob is *****.

I paid for my credit card ******************* last week from my bank account ********** with the routing number **********. The check was mailed from *************************.

Please confirm receipt by calling me at ************ or emailing at ***********************.

Putting it together

Let’s understand where each API can be used as apart of our extraction and classification workflow

If the extracted data from Amazon Textract is sufficient for the required use case send results to system of record storage and/or post-processing

Amazon Textract

Identify based on document type which Textract extraction API is most appropriate (Text Detection, Forms, Tables, Invoices, ID etc.)

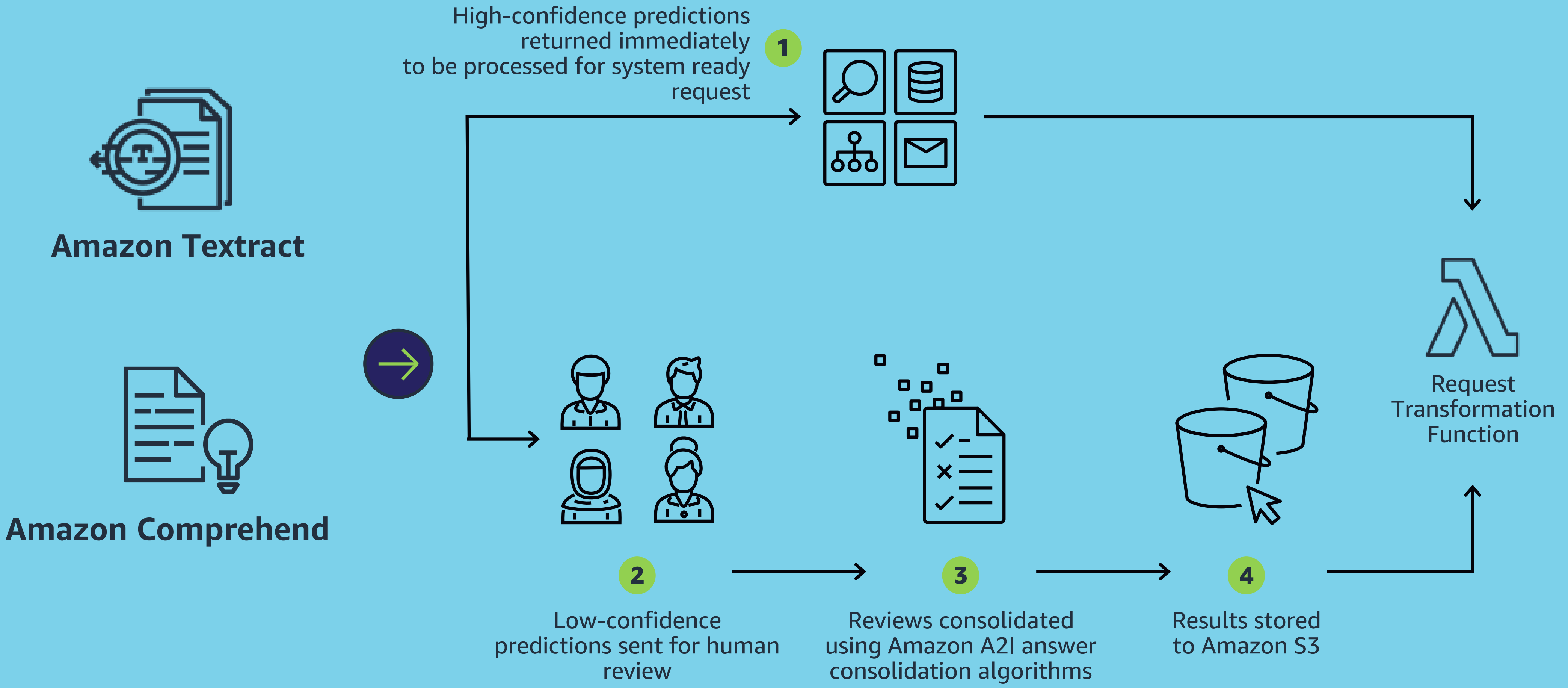
Amazon Comprehend

If further granularity is required from the documents such as identifying document class, identifying specific entities or redacting PII use the appropriate Amazon Comprehend API on extracted data from Amazon Textract

System of Record Storage

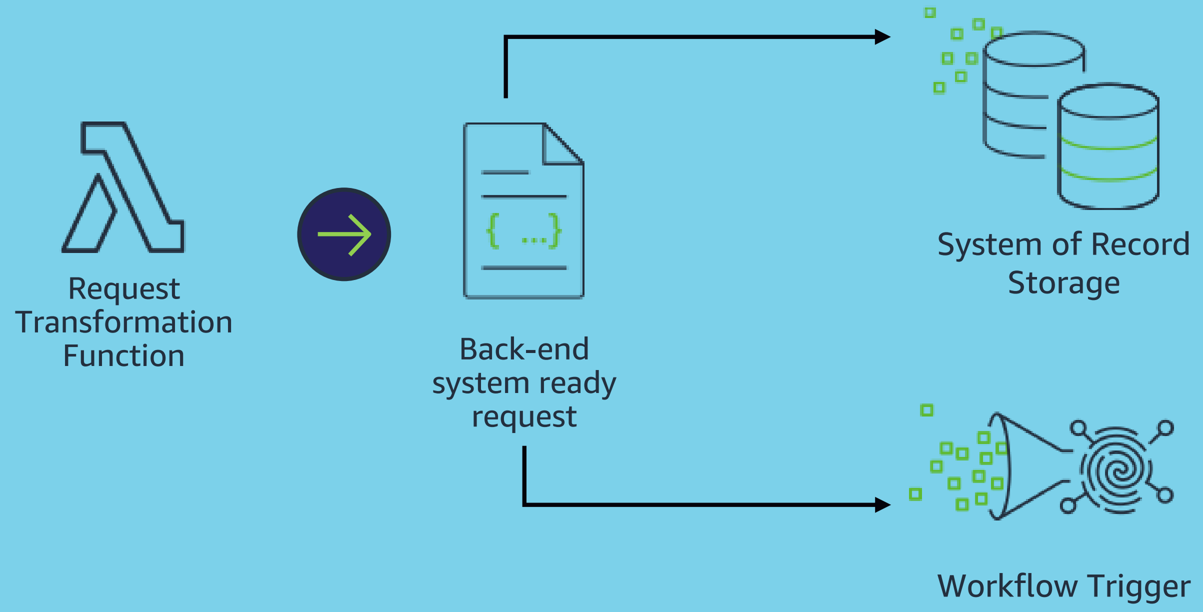
Post-Processing and Transformation

The next step in our IDP workflow, Post-Processing and Transformation, ensures our output data is of high quality and is ready to ingest into our back-end storage and workflows



Storage and Workflow Automation

The final step in our IDP workflow, Storage and Workflow Automation, acts as the final stop for our document data, it is used to store the document in a system of record, or used to trigger back-end automation



An end to end IDP example visualized – Insurance Policy Servicing

1

Broker fills out an semi-structured insurance policy servicing document and emails document to Insurance carrier

2

Document is sent to **Amazon S3** for storage and is tagged with sender, timestamp metadata

3

Text is extracted from the document, along with any tables or key-value pairs using **Amazon Textract**

4

Document is classified as a Policy servicing document and tagged, and required servicing action is identified using **Amazon Comprehend**

5

The extracted data from **Amazon Textract** and **Amazon Comprehend** is transformed into a policy servicing API request

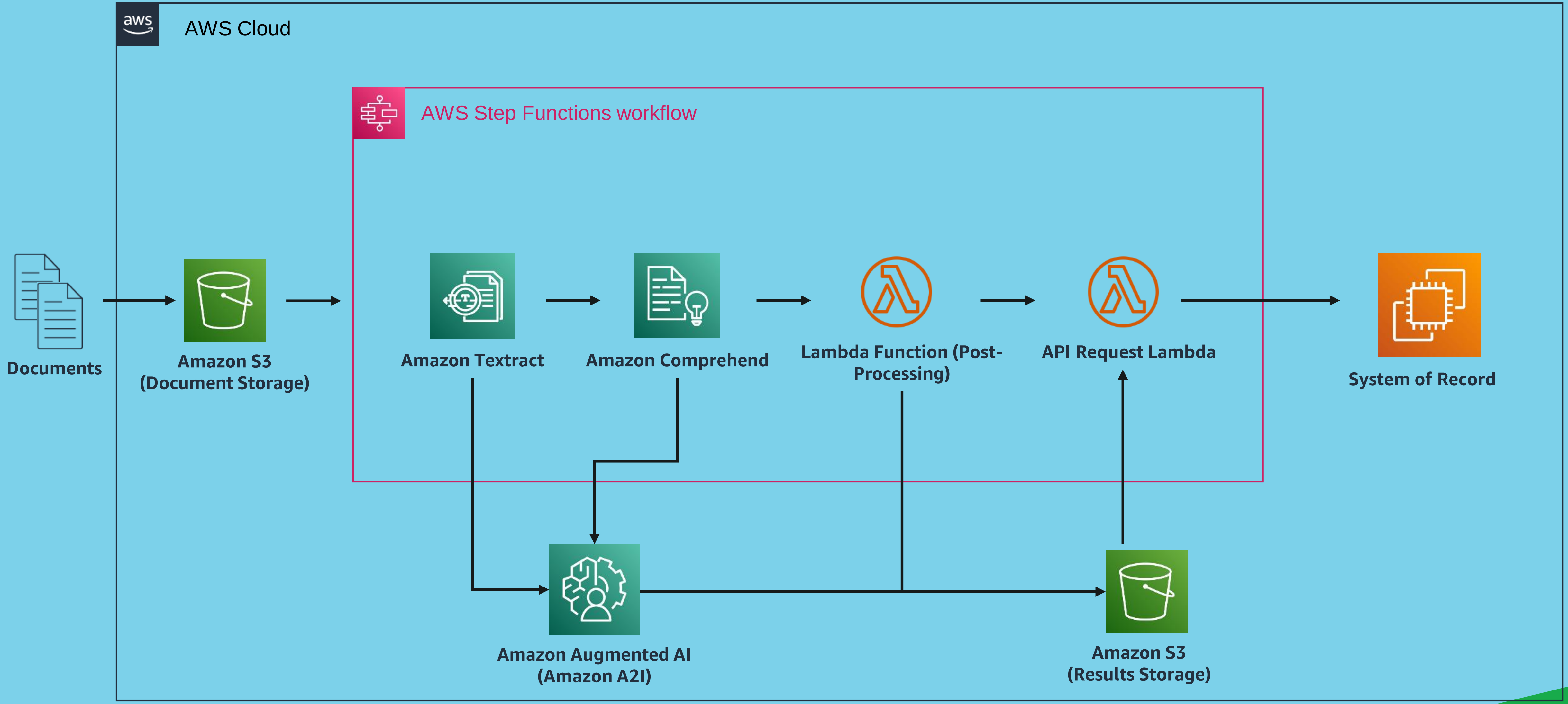
6

The servicing request is sent to policy management system API to trigger a policy servicing action

7

The changed policy is stored in the policy admin system

AWS Reference Architecture IDP



Thank you!