Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

EATON
Powering Business Worldwide

Enabling Intelligent Business Process Automation

Guided By: Professor Neelam Dwivedi
Team: Woojin Park, Ai Chen, Soham Chakraborti, Youchun Ni, Harita DL

Eaton Team

Current interactions with the IT Team:

Viswanathan Krishnamurthy	Vice President
Jason Pappas	Senior Manager
Abdul Razak	Senior Manager
Abhinay Dande	Solution Architect
Bill Overton	Innovation Engineer

Powering Business Worldwide

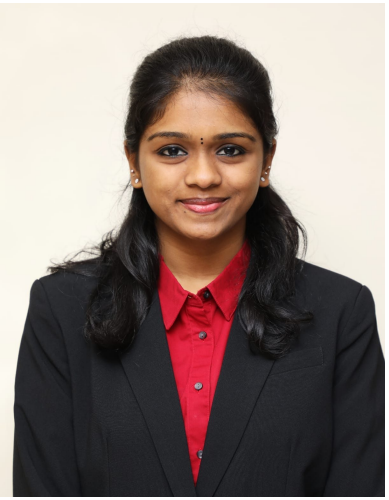
Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Eaton Team

Interactions with the Global Financial Shared Services (GFSS) Team:

Daniel Hopgood	Senior Vice President
Troy Huth	Global Process Leader - Purchase to Pay
Sachit Nayak	Finance Director - India
Vikram Chitale	Process Leader - Purchase to Pay

CMU Team

				
Youchun Ni MISM-16 Financial Manager / Cloud Architecture	**Ai Chen** MISM -16 Cloud Architecture	**Soham Chakraborti** BIDA -16 Product Development / Cloud Architecture	**Woojin Park** BIDA-16 Systems Administrator / ML Development	**Harita DL** MISM-16 Project Manager

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Agenda

- About Eaton
- Project Background
- Problem Statement & Analysis
- Approaches Explored
- Our Solution
- Demo
- Lessons Learned
- Next Steps for Eaton

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

About Eaton

- Power management company with operational headquarters in Ohio
- Has about 97,000 employees and operates in 175 countries
- Building energy-efficient products and services that help customers manage electrical, hydraulic and mechanical power
- Focus on Reliability, Efficiency, Safety and Sustainability



Air Power Circuit Breakers



Glass & Ceramic Tube Fuses



Interior Toggle Switch



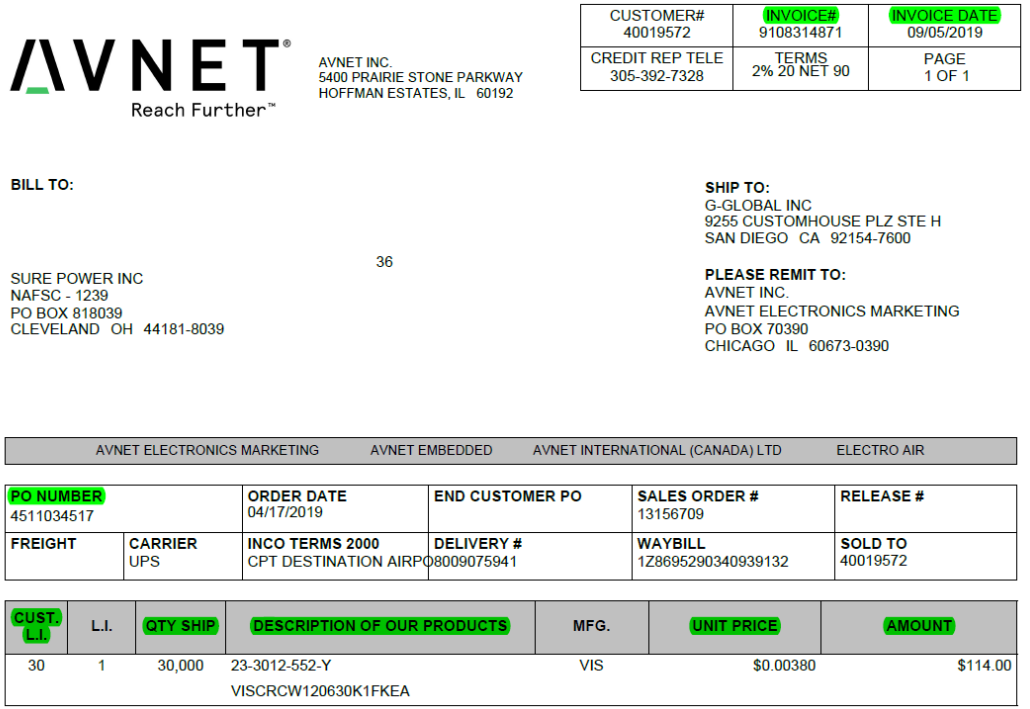
Hazardous area device box and covers

Source: https://www.eaton.com/

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Project Background

- Key customer - GFSS
- **50000 vendors** each with different formats of invoices
- **2500 invoices** processed **per day** manually at one of GFSS Center



AVNET
Reach Further™

AVNET INC.
5400 PRAIRIE STONE PARKWAY
HOFFMAN ESTATES, IL 60192

CUSTOMER# 40019572	INVOICE# 9108314871	INVOICE DATE 09/05/2019
CREDIT REP TELE 305-392-7328	TERMS 2% 20 NET 90	PAGE 1 OF 1

BILL TO:

SURE POWER INC
NAFSC - 1239
PO BOX 818039
CLEVELAND OH 44181-8039

36

SHIP TO:
G-GLOBAL INC
9255 CUSTOMHOUSE PLZ STE H
SAN DIEGO CA 92154-7600

PLEASE REMIT TO:
AVNET INC.
AVNET ELECTRONICS MARKETING
PO BOX 70390
CHICAGO IL 60673-0390

AVNET ELECTRONICS MARKETING | AVNET EMBEDDED | AVNET INTERNATIONAL (CANADA) LTD | ELECTRO AIR

PO NUMBER 4511034517		ORDER DATE 04/17/2019	END CUSTOMER PO	SALES ORDER # 13156709	RELEASE #
FREIGHT	CARRIER UPS	INCO TERMS 2000 CPT DESTINATION AIRPO	DELIVERY # 8009075941	WAYBILL 1Z8695290340939132	SOLD TO 40019572

CUST. L.I.	L.I.	QTY SHIP	DESCRIPTION OF OUR PRODUCTS	MFG.	UNIT PRICE	AMOUNT
30	1	30,000	23-3012-552-Y VISCRCW120630K1FKEA	VIS	$0.00380	$114.00

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Problem Statement

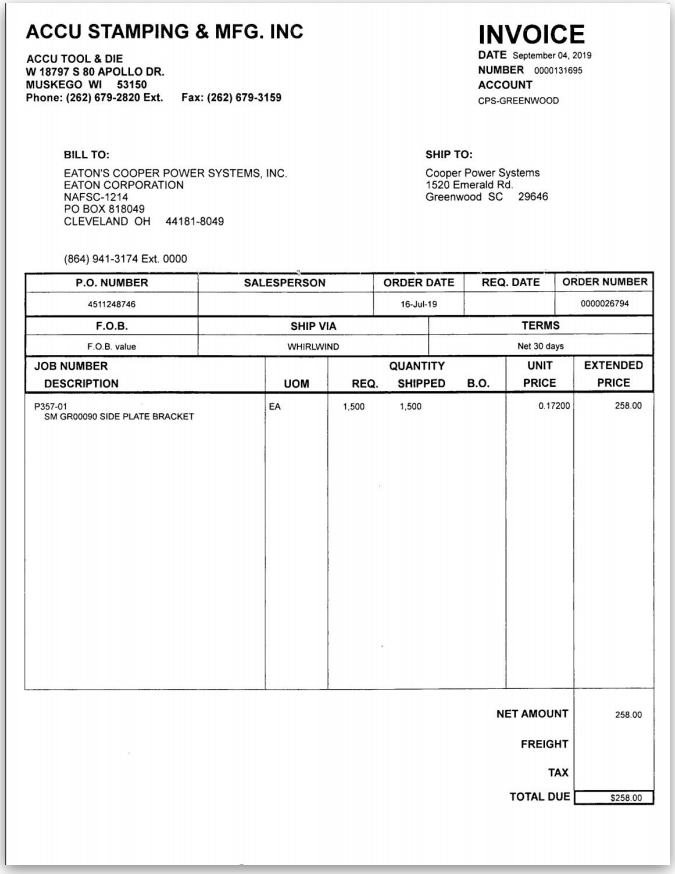
Manually indexing to retrieve certain fields is -

- Repetitive
- Time-consuming task
- Error prone



Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Sample Invoices



ACCU STAMPING & MFG. INC

ACCU TOOL & DIE
W 18797 S 80 APOLLO DR.
MUSKEGO WI 53150
Phone: (262) 679-2820 Ext. Fax: (262) 679-3159

INVOICE
DATE September 04, 2019
NUMBER 0000131695
ACCOUNT
CPS-GREENWOOD

BILL TO:
EATON'S COOPER POWER SYSTEMS, INC.
EATON CORPORATION
NAFSC-1214
PO BOX 818049
CLEVELAND OH 44181-8049

SHIP TO:
Cooper Power Systems
1520 Emerald Rd
Greenwood SC 29646

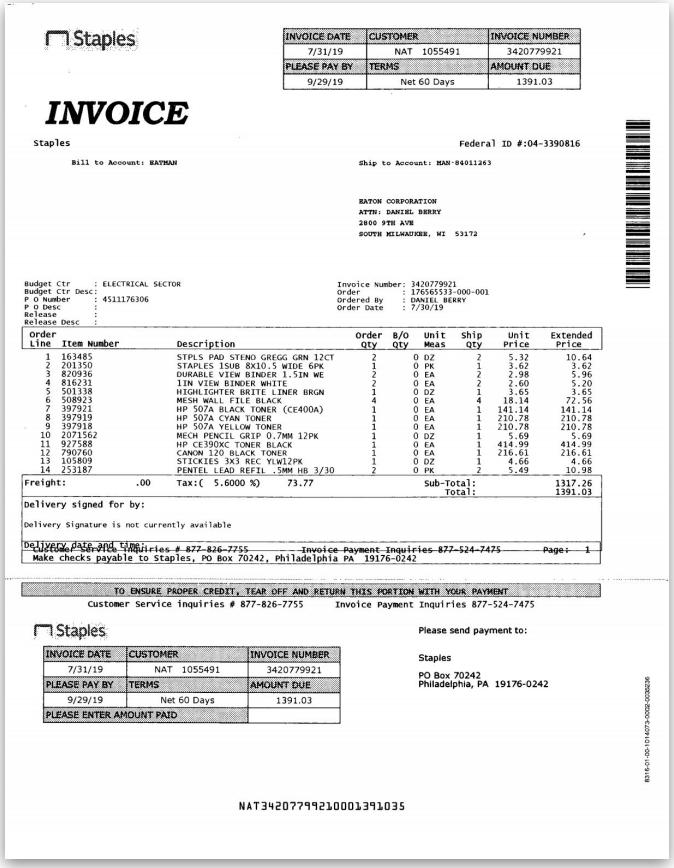
(864) 941-3174 Ext. 0000

P.O. NUMBER	SALESPERSON	ORDER DATE	REQ. DATE	ORDER NUMBER
4511248746		16-Jul-19		0000026794

F.O.B.	SHIP VIA	TERMS
F.O.B. value	WHIRLWIND	Net 30 days

JOB NUMBER / DESCRIPTION	UOM	QUANTITY REQ.	QUANTITY SHIPPED	B.O.	UNIT PRICE	EXTENDED PRICE
P357-01 SM GR00090 SIDE PLATE BRACKET	EA	1,500	1,500		0.17200	258.00

NET AMOUNT	258.00
FREIGHT	
TAX	
TOTAL DUE	$258.00



Staples

INVOICE

INVOICE DATE	CUSTOMER	INVOICE NUMBER
7/31/19	NAT 1055491	3420779921
PLEASE PAY BY	TERMS	AMOUNT DUE
9/29/19	Net 60 Days	1391.03

Staples
Federal ID #:04-3390816

Bill to Account: EATMAN

Ship to Account: MAN-84011263

EATON CORPORATION
ATTN: DANIEL BERRY
2800 9TH AVE
SOUTH MILWAUKEE, WI 53172

Budget Ctr : ELECTRICAL SECTOR
Budget Ctr Desc:
P O Number : 4511176306
P O Desc :
Release :
Release Desc :

Invoice Number: 3420779921
Order : 176565533-000-001
Ordered By : DANIEL BERRY
Order Date : 7/30/19

Order Line	Item Number	Description	Order Qty	B/O Qty	Unit Meas	Ship Qty	Unit Price	Extended Price
1	163485	STPLS PAD STENO GREGG GRN 12CT	2	0	DZ	2	5.32	10.64
2	201350	STAPLES 1SUB 8X10.5 WIDE 6PK	1	0	PK	1	3.62	3.62
3	820936	DURABLE VIEW BINDER 1.5IN WE	2	0	EA	2	2.98	5.96
4	816231	1IN VIEW BINDER WHITE	2	0	EA	2	2.60	5.20
5	501338	HIGHLIGHTER BRITE LINER BRGN	1	0	DZ	1	3.65	3.65
6	508923	MESH WALL FILE BLACK	4	0	EA	4	18.14	72.56
7	397921	HP 507A BLACK TONER (CE400A)	1	0	EA	1	141.14	141.14
8	397919	HP 507A CYAN TONER	1	0	EA	1	210.78	210.78
9	397918	HP 507A YELLOW TONER	1	0	EA	1	210.78	210.78
10	2071562	MECH PENCIL GRIP 0.7MM 12PK	1	0	DZ	1	5.69	5.69
11	927588	HP CE390XC TONER BLACK	1	0	EA	1	414.99	414.99
12	790760	CANON 120 BLACK TONER	1	0	EA	1	216.61	216.61
13	105809	STICKIES 3X3 REC YLW12PK	1	0	DZ	1	4.66	4.66
14	253187	PENTEL LEAD REFIL .5MM HB 3/30	2	0	PK	2	5.49	10.98

Freight: .00 Tax:(5.6000 %) 73.77 Sub-Total: 1317.26
Total: 1391.03

Delivery signed for by:

Delivery Signature is not currently available

Delivery date and time:

Customer Service Inquiries # 877-826-7755 Invoice Payment Inquiries 877-524-7475 Page: 1
Make checks payable to Staples, PO Box 70242, Philadelphia PA 19176-0242

TO ENSURE PROPER CREDIT, TEAR OFF AND RETURN THIS PORTION WITH YOUR PAYMENT
Customer Service Inquiries # 877-826-7755 Invoice Payment Inquiries 877-524-7475

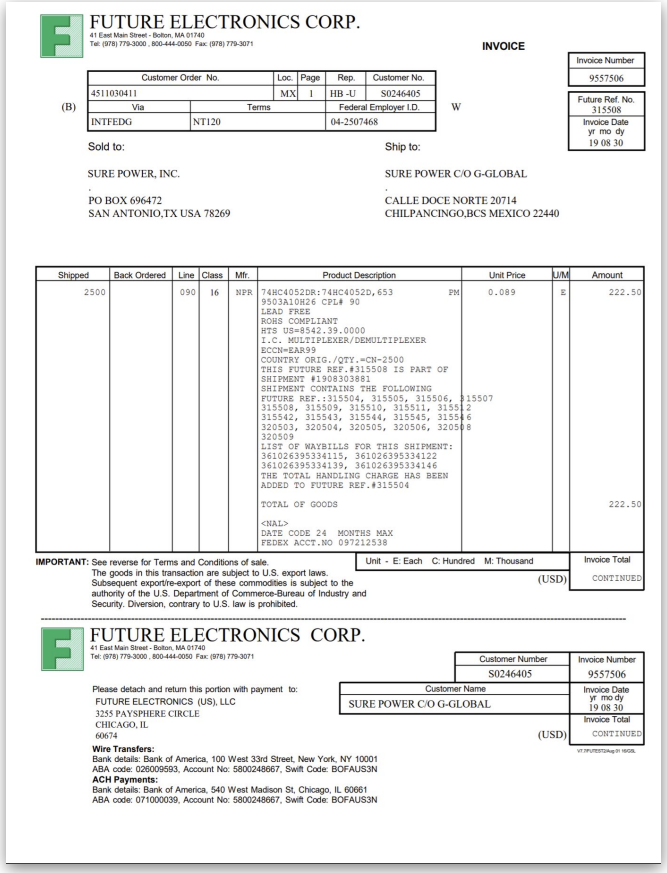
Staples

INVOICE DATE	CUSTOMER	INVOICE NUMBER
7/31/19	NAT 1055491	3420779921
PLEASE PAY BY	TERMS	AMOUNT DUE
9/29/19	Net 60 Days	1391.03
PLEASE ENTER AMOUNT PAID		

Please send payment to:

Staples
PO Box 70242
Philadelphia, PA 19176-0242

NAT34207799210001391035



FUTURE ELECTRONICS CORP.
41 East Main Street - Bolton, MA 01740
Tel: (978) 779-3000 , 800-444-0050 Fax: (978) 779-3071

INVOICE

Invoice Number: 9557506
Future Ref. No.: 315508
Invoice Date yr mo dy: 19 08 30

Customer Order No.		Loc.	Page	Rep.	Customer No.
4511030411		MX	1	HB -U	S0246405
Via	Terms			Federal Employer I.D.	
INTFEDG	NT120			04-2507468	

(B) W

Sold to:
SURE POWER, INC.
.
PO BOX 696472
SAN ANTONIO,TX USA 78269

Ship to:
SURE POWER C/O G-GLOBAL
.
CALLE DOCE NORTE 20714
CHILPANCINGO,BCS MEXICO 22440

Shipped	Back Ordered	Line	Class	Mfr.	Product Description	Unit Price	U/M	Amount
2500		090	16	NPR	74HC4052DR:74HC4052D,653 PM 9503A10N26 CPL# 90 LEAD FREE ROHS COMPLIANT HTS US=8542.39.0000 I.C. MULTIPLEXER/DEMULTIPLEXER ECCN=EAR99 COUNTRY ORIG./QTY.=CN-2500 THIS FUTURE REF.#315508 IS PART OF SHIPMENT #1908303881 SHIPMENT CONTAINS THE FOLLOWING FUTURE REF.:315504, 315505, 315506, 315507 315508, 315509, 315510, 315511, 315512 315542, 315543, 315544, 315545, 315546 320503, 320504, 320505, 320506, 320508 320509 LIST OF WAYBILLS FOR THIS SHIPMENT: 361026395334115, 361026395334122 361026395334139, 361026395334146 THE TOTAL HANDLING CHARGE HAS BEEN ADDED TO FUTURE REF.#315504 TOTAL OF GOODS	0.089	E	222.50 222.50
					<NAL> DATE CODE 24 MONTHS MAX FEDEX ACCT.NO 097212538			

IMPORTANT: See reverse for Terms and Conditions of sale.
The goods in this transaction are subject to U.S. export laws. Subsequent export/re-export of these commodities is subject to the authority of the U.S. Department of Commerce-Bureau of Industry and Security. Diversion, contrary to U.S. law is prohibited.

Unit - E: Each C: Hundred M: Thousand
(USD)
Invoice Total
CONTINUED

FUTURE ELECTRONICS CORP.
41 East Main Street - Bolton, MA 01740
Tel: (978) 779-3000 , 800-444-0050 Fax: (978) 779-3071

Please detach and return this portion with payment to:
FUTURE ELECTRONICS (US), LLC
3255 PAYSPHERE CIRCLE
CHICAGO, IL
60674

Customer Number	Invoice Number
S0246405	9557506
Customer Name	Invoice Date yr mo dy: 19 08 30
SURE POWER C/O G-GLOBAL	Invoice Total
	CONTINUED

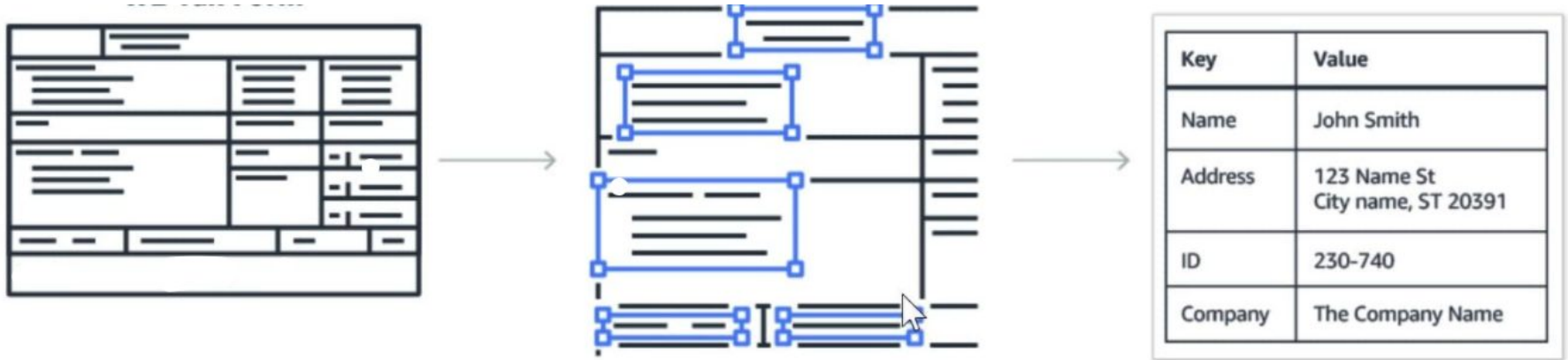
(USD)

Wire Transfers:
Bank details: Bank of America, 100 West 33rd Street, New York, NY 10001
ABA code: 026009593, Account No: 5800248667, Swift Code: BOFAUS3N
ACH Payments:
Bank details: Bank of America, 540 West Madison St, Chicago, IL 60661
ABA code: 071000039, Account No: 5800248667, Swift Code: BOFAUS3N

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Vision

- To build an **AI** based automation that can **interpret and extract** various **data fields** from **varyingly formatted** supplier invoices, reducing manual indexing work and possible manual errors.



Key	Value
Name	John Smith
Address	123 Name St City name, ST 20391
ID	230-740
Company	The Company Name

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Current Scope

- Parsing digital invoices, including scanned/ searchable text PDF’s
- Retrieving the 14 mandatory fields
- Parsing invoices in English Language

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Business Impact



Approximately 55,000 invoices/month will be automated



Saves about 7 FTE manual work



Improve productivity by generating results in real-time



Reduce error from manual entry

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Approaches Considered

Cloud Based Services

- Amazon AWS (Textract)
- Microsoft Azure (Form Recognition)
- Google DocAI

Python Library Based Services

- Google Tesseract
- Pdf2text
- Other openCV based libraries











Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Project Pipeline

Input Data
(PDF format Invoice)

Upload to S3 bucket
& Develop solution in
AWS SageMaker

Text Extraction Technology

1.Google DocumentAI
(Invoice Processor)

2.AWS Textract

Output Data (CSV format)

Generate Key-Value Pairs,
process Alias matching,
and Store CSV file

Generate Key-Value Pairs,
process Alias matching,
and Store CSV file

Applying Matching
Function for Analysis

'Highlight the
Mismatched entries
& Extraction Entity
Score(EES)'

Entity Extraction Score

Bucket 1 : 80 ~ 100 %

Bucket 2 : 60 ~ 80 %

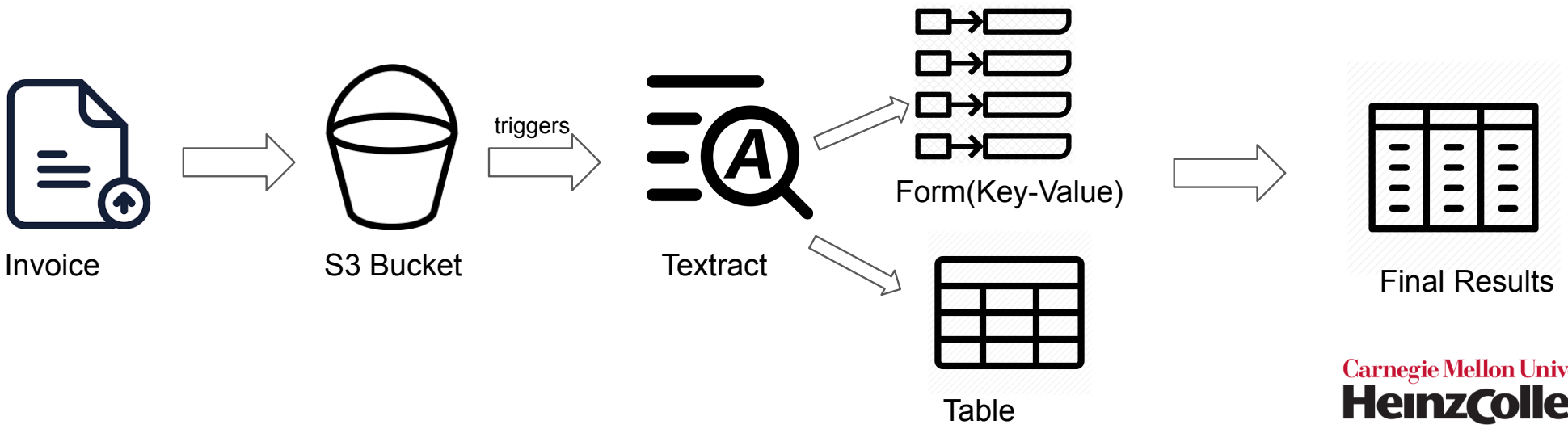
Bucket 3 : < 60 %

Bucketize the Invoices
based on 'EES'

Analytics Purpose
Output Data (CSV format)

AWS Approach

- Built an automatic **pipeline** in AWS that can detect and run textract on batch invoices in real-time
- Save **text-extraction** output in AWS S3 as csv files, including forms and tables.
- Process the forms and tables into structured csv files.



Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Invoice to Form

BILL TO:

EATON'S COOPER POWER SYSTEMS, INC.
EATON CORPORATION
NAFSC-1214
PO BOX 818049
CLEVELAND OH 44181-8049

SHIP TO:

SO. MILWAUKEE, WI (CPS)
COOPER POWER SYSTEMS
2800 NINTH AVE.
SO. MILWAUKEE WI 53172

(414) 768-0000 Ext.

P.O. NUMBER	SALESPERSON	ORDER DATE	REQ. DATE	ORDER NUMBER
4511232718		24-Jun-19		0000026691

F.O.B.	SHIP VIA	TERMS
F.O.B. value	UPS/FRT.COLL.	Net 30 days

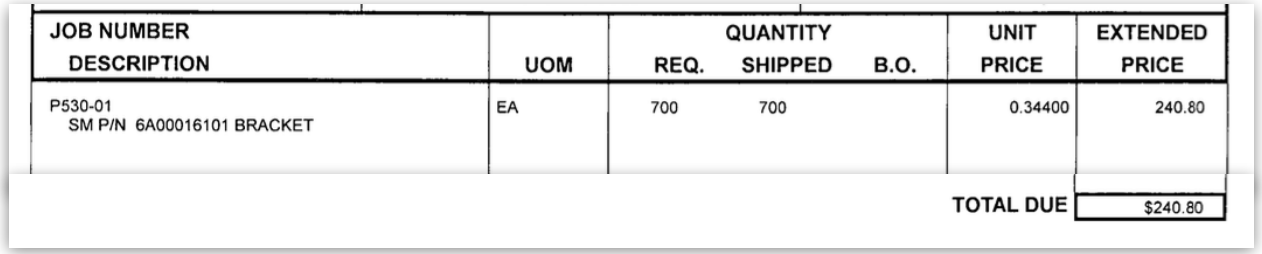
Invoice.pdf

Key	Value
ORDER NUMBER	26691
ORDER DATE	24-Jun-19
REQ. DATE	
SHIP TO:	SO. MILWAUKEE, WI (CPS) COOPER POWER SYSTEMS 2800 NINTH AVE. SO. MILWAUKEE WI 53172
SHIP VIA	UPS/FRT.COLL.
TERMS	Net 30 days
TOTAL DUE	$240.80
SALESPERSON	
NUMBER	131472
BILL TO:	EATON'S COOPER POWER SYSTEMS, INC. EATON CORPORATION NAFSC-1214 PO BOX 818049 CLEVELAND OH 44181-8049
Phone:	(262) 679-2820 Ext.
Fax:	(262) 679-3159
TAX	
DATE July	23, 2019
JOB NUMBER	DESCRIPTION
P.O. NUMBER	4511232718

form.csv

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Invoice to Table



JOB NUMBER DESCRIPTION	UOM	QUANTITY REQ.	QUANTITY SHIPPED	QUANTITY B.O.	UNIT PRICE	EXTENDED PRICE
P530-01 SM P/N 6A00016101 BRACKET	EA	700	700		0.34400	240.80
					TOTAL DUE	$240.80

Invoice.pdf

JOB NUMBER DESCRIPTION	UOM	REQ.	QUANTITY SHIPPED B.O.	UNIT PRICE	EXTENDED PRICE
P530-01 SM P/N 6A00016101 BRACKET	EA	700	700	0.344	240.8
				TOTAL DUE	$240.80

table.csv

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Input - Alias Table

	URBAN MANUFACTURING CO INC	GENERAL METALS	ENDRIES INTERNATIONAL	ACCU FAB
Invoice_Number	invoice	Invoice #	invoice #	INVOICE ID
Invoice_Date	date	Date	invoice date	Date
Invoice_Total	total	total	Invoice Total	TOTAL AMOUNT DUE
Vendor_Name *(Apply string	PDF File Name	PDF File Name	PDF File Name	PDF File Name
Currency	(with invoice_total as $)	(with invoice_total as $)	Currency	(with invoice_total as $)
Taxes	Total Tax	\	\	\
Subtotal	\	\	\	SUB TOTAL
Purchase_Order_Number	PO Number	P.O. Number	Customer PO	CUSTOMER ID
Document_Type	\	\	\	\
Item_Description	Descrption	Description	Description	Description
Quantity	Quantity	Quantity	quantity ordered	QUANTITY
Item_Total	Ext Price	Amount	net amount	EXTENDED PRICE
Item_Number	Part Number	Item Code	\	\
Unit_Price	Unit Price	Price Each	unit price	UNIT PRICE

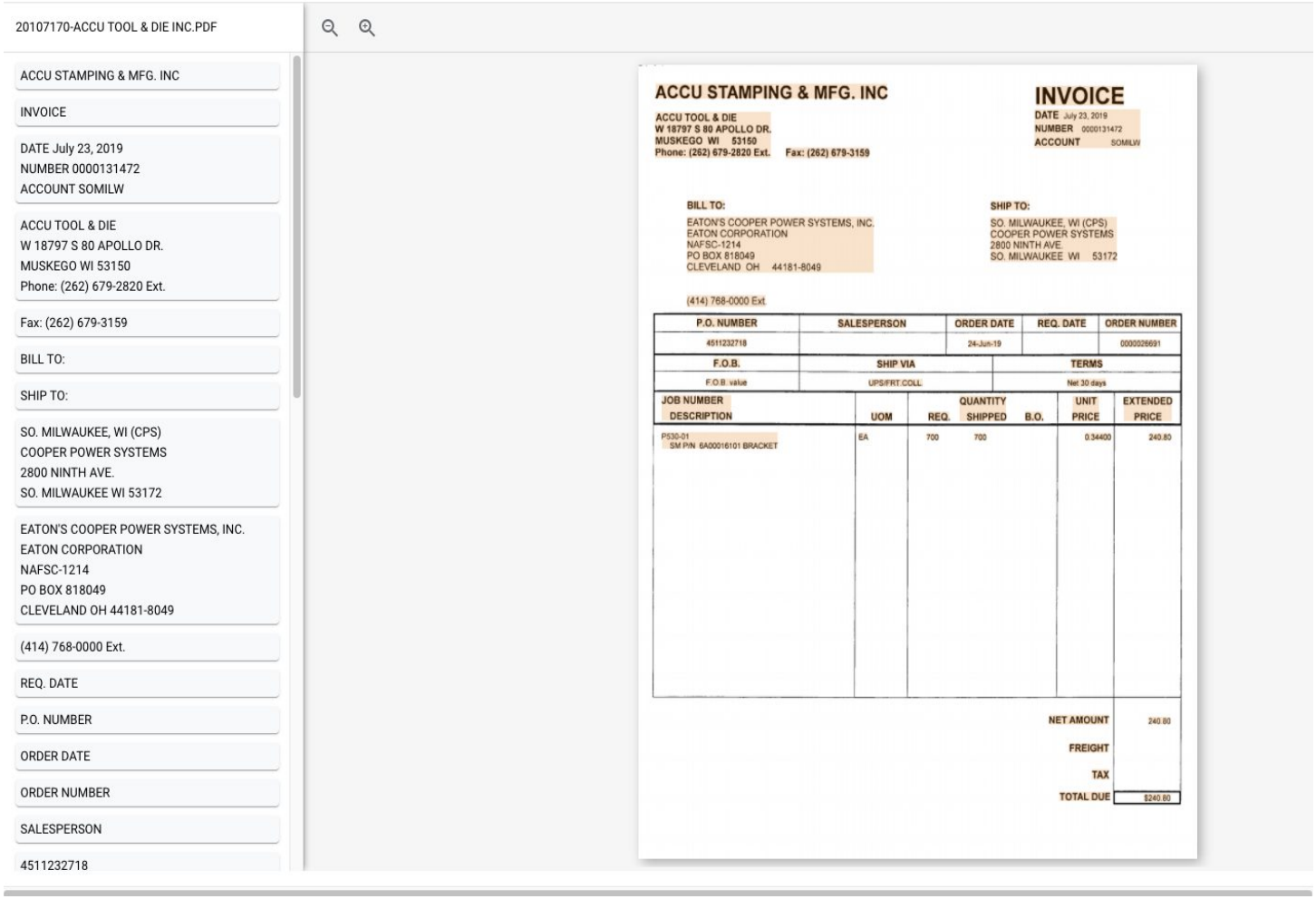
Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Output - Structured CSV

Invoice_Number	Invoice_Date	Invoice_Total	Vendor_Name	Taxes	Subtotal	Purchase_Order_Number
0000131472		$240.80	ACCU TOOL & DIE INC			4511232718
0000131476	July 23, 2019	$902.00	ACCU TOOL & DIE INC			
0000131479	July 23, 2019	$258.00	ACCU TOOL & DIE INC			4511213241
0000131461		$540.00	ACCU TOOL & DIE INC			
0000131419	July 15, 2019	$2,281.54	ACCU TOOL & DIE INC			
0000131456	July 22, 2019	$642.60	ACCU TOOL & DIE INC			
0000131449	July 19, 2019	$5,639.50	ACCU TOOL & DIE INC		5,639.50	4511198679
0000131454	July 22, 2019	$1,141.58	ACCU TOOL & DIE INC			4511265635
		$1,926.00	ACCU TOOL & DIE INC			
0000131467		$12,268.82	ACCU TOOL & DIE INC			4511199078
25485	8/23/2019		ATLAS METAL PARTS AND FABRICATING	.00	723.24	4511197092
25499	8/28/2019		ATLAS METAL PARTS AND FABRICATING	.00	180.00	4511280342
25521	8/28/2019		ATLAS METAL PARTS AND FABRICATING	.00	1,392.00	4511360970
25501	8/29/2019		ATLAS METAL PARTS AND FABRICATING	.00	1,386.96	4511341932
25514	8/22/2019		ATLAS METAL PARTS AND FABRICATING	.00	1,533.00	4511372712
25511	8/23/2019		ATLAS METAL PARTS AND FABRICATING	.00	180.00	4511280373
25492	8/26/2019		ATLAS METAL PARTS AND FABRICATING	.00	1,063.00	4511213620
25495	8/22/2019		ATLAS METAL PARTS AND FABRICATING	.00	451.30	4511246987
25520	8/26/2019		ATLAS METAL PARTS AND FABRICATING	.00	54.52	4511335852
25512	8/26/2019		ATLAS METAL PARTS AND FABRICATING	.00	180.00	4511280373

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Google DocAI Output



Google Cloud

invoice_date

Invoice

DATE: 01/01/1970

INVOICE: NO. 001

supplier_name

invoice_id

FROM: Company ABC user@companyabc.com ADDRESS: 111 Main Street Anytown, USA	TO: John Doe johndoe@email.com ADDRESS: 222 Main Street Anytown, USA

payment_terms

due_date

TERMS: 6 month contract

DUE: 01/01/2025

line_item/quantity line_item/unit_price

Item Description	Quantity	Price	Amount
Tool A	500	$1.00	$500.00
Service B	1	$900.00	$900.00
Resource C	50	$12.00	$600.00
		Subtotal	$2000.00
		Tax	$140.00
		BALANCE DUE	**$2140.00**

amount_due

total_tax_amount

NOTES:

Supplies used for Project Q.

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Google DocAI Field Names

Field	Annotations		
`entities`	• carrier • currency • customer_tax_id • delivery_date • due_date • invoice_date • invoice_id	• net_amount • payment_terms • purchase_order • receiver_address • receiver_name • remit_to_address • remit_to_name	• ship_to_address • ship_to_name • supplier_address • supplier_name • supplier_tax_id • total_amount • total_tax_amount
nested `entities`	• line_item/amount • line_item/unit_price • line_item/quantity • line_item/unit • line_item/description • line_item/product_code		

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Test Results

Number of Invoices Provided	1000
Number of Invoices Tested	388
Total Processing Time	26’ 17’’
Entity Extraction Score*	91% for common fields, 69% for table fields
OCR Accuracy*	99% for common fields

*Entity Extraction Score - Average percentage of entities extracted among the mandatory fields

*OCR Accuracy - Accuracy of extraction of each field by AWS and Google DocAI

*Table fields - Line Item Number, Description, Quantity Ordered/Shipped, Item Amount

*Common fields - Invoice Number, Invoice Date, Purchase Order Number, Sub-Total, Taxes, Total Amount

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Comparison and Integration

Google DocAI:

- Structured output: Extracts tables as key-value pairs
- Standard set of keys: No need to do Alias matching
- Higher OCR accuracy: Have comparisons later

AWS:

- Have a template for end-to-end service, can run batch processing easily
- S3(Storage), SQS(Message Queue), SNS(Pub-Sub and Notification), Lambda Functions(Run Services), Cloud Formation(Monitor and Manage), Secrets Manager, ...

Decision: Integrate Google DocAI Service into the AWS pipeline

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Demo Video



Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Project Pipeline - Revisited



Input Data
(PDF format Invoice)





Upload to S3 bucket
& Develop solution in
AWS SageMaker

Text Extraction Technology

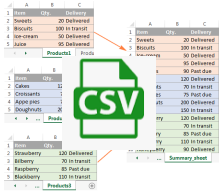


1.Google DocumentAI
(Invoice Processor)

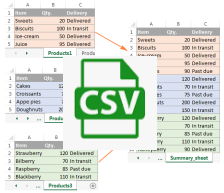


2.AWS Textract

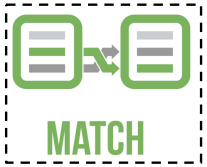
Output Data (CSV format)



Generate Key-Value Pairs,
process Alias matching,
and Store CSV file



Generate Key-Value Pairs,
process Alias matching,
and Store CSV file



Applying Matching
Function for Analysis

'Highlight the
Mismatched entries
& Extraction Entity
Score(EES)'

Entity Extraction Score



Bucket 1 : 80 ~ 100 %



Bucket 2 : 60 ~ 80 %

target



Bucket 3 : < 60 %

Bucketize the Invoices
based on 'EES'



Analytics Purpose
Output Data (CSV format)

Compare Output from Two Providers

- Analytics purpose program integrates two csv outputs from Google DocAI and AWS Textract
- Highlights the Mismatched Entries & Entity Extraction Score (EES)* Bucket
- User can extract the target population (Bucketized group) to further data extraction and exploration.

*Entity Extraction Score - Average number of entities extracted among the mandatory fields

	B	C	D	E	F	G
	DocAi_Extraction_score_bucket	DocAi_Extraction_Score	AWS_Extraction_Score	Invoice_Number_AWS	Invoice_Number_DocAi	Invoice_Number
0	2	0.857	0.714	131472	131472	Matched
1	2	0.857	0.714	131476	131476	Matched
2	2	0.857	0.857	131479	131479	Matched
3	2	0.857	0.571	131461	131461	Matched
4	2	0.857	0.714	131419	181180	Not Matched

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

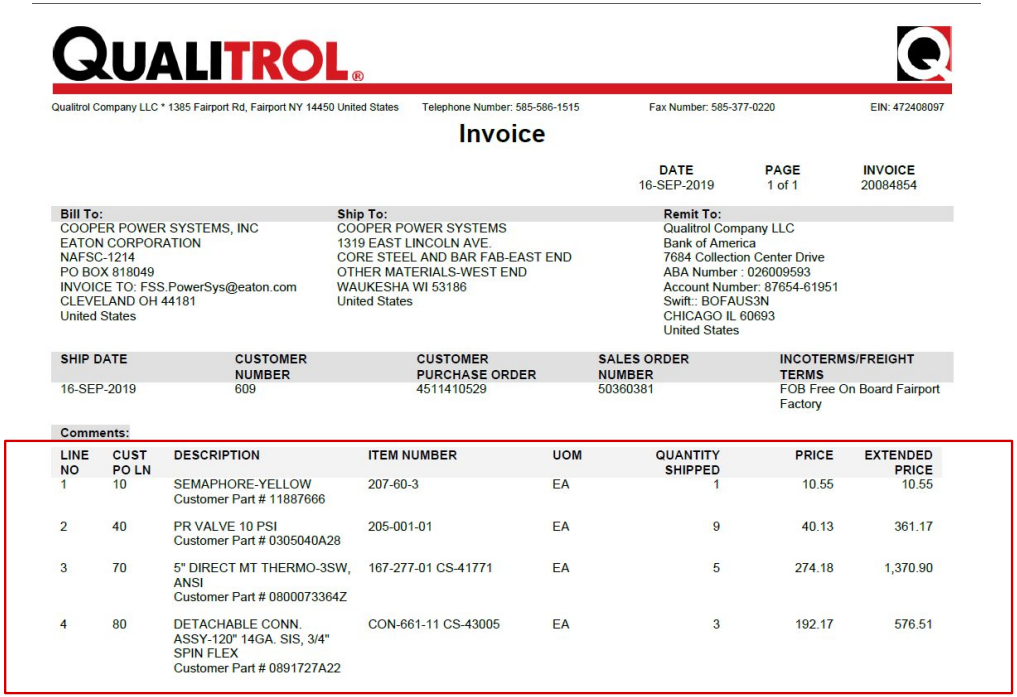
Lessons Learned

- Amazon Web Services (AWS)
  - Provide an end-to-end workflow framework from file uploading to text extraction
  - The raw output of AWS Textract can be very inaccurate, which requires heavy post-processing work
- Google Cloud Provider (GCP)
  - Provide a models and technology trained to work on invoices, which improve the extraction performance

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Challenges Faced

- Developing a generic model that works for varying invoices
- Extraction of multiple lines of records from the Table section of the invoice



QUALITROL®

Qualitrol Company LLC * 1385 Fairport Rd, Fairport NY 14450 United States Telephone Number: 585-586-1515 Fax Number: 585-377-0220 EIN: 472408097

Invoice

DATE	PAGE	INVOICE
16-SEP-2019	1 of 1	20084854

Bill To:
COOPER POWER SYSTEMS, INC
EATON CORPORATION
NAFSC-1214
PO BOX 818049
INVOICE TO: FSS.PowerSys@eaton.com
CLEVELAND OH 44181
United States

Ship To:
COOPER POWER SYSTEMS
1319 EAST LINCOLN AVE.
CORE STEEL AND BAR FAB-EAST END
OTHER MATERIALS-WEST END
WAUKESHA WI 53186
United States

Remit To:
Qualitrol Company LLC
Bank of America
7684 Collection Center Drive
ABA Number : 026009593
Account Number: 87654-61951
Swift:: BOFAUS3N
CHICAGO IL 60693
United States

SHIP DATE	CUSTOMER NUMBER	CUSTOMER PURCHASE ORDER	SALES ORDER NUMBER	INCOTERMS/FREIGHT TERMS
16-SEP-2019	609	4511410529	50360381	FOB Free On Board Fairport Factory

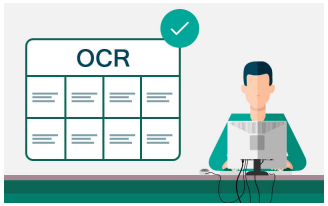
Comments:

LINE NO	CUST PO LN	DESCRIPTION	ITEM NUMBER	UOM	QUANTITY SHIPPED	PRICE	EXTENDED PRICE
1	10	SEMAPHORE-YELLOW Customer Part # 11887666	207-60-3	EA	1	10.55	10.55
2	40	PR VALVE 10 PSI Customer Part # 0305040A28	205-001-01	EA	9	40.13	361.17
3	70	5" DIRECT MT THERMO-3SW, ANSI Customer Part # 0800073364Z	167-277-01 CS-41771	EA	5	274.18	1,370.90
4	80	DETACHABLE CONN. ASSY-120" 14GA. SIS, 3/4" SPIN FLEX Customer Part # 0891727A22	CON-661-11 CS-43005	EA	3	192.17	576.51

- Converting extracted data into Key-Value pairs format
- Integrate services from multiple cloud providers into one pipeline system

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

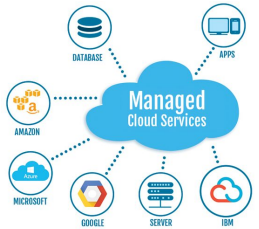
Key Experiences



Exploration of various OCR text extraction and detection technologies

2-Day Training Workshop by AWS Team

Interaction with Google Representatives regarding Google DocAI - Invoice Processor features



Exploring various cloud services + Building overall pipeline

Constant weekly interaction with Eaton Team, Faculty Advisor



Presenting our solution to the GFSS Team at Eaton

Planning tasks, Scheduling timeline, Coordinating with team

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Next Steps

- Identify other approaches to retrieve the missed out mandatory fields
- Analyze the output of AWS text-extraction to identify the optional fields we need
- Parsing invoices in languages other than English
- Man in the loop approach, were we have manual interventions for invoices with low entity extraction score
- Improvement in text detection and extraction technologies over time and by other providers

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Thank You

Carnegie Mellon University
HeinzCollege
INFORMATION SYSTEMS • PUBLIC POLICY • MANAGEMENT

Technical Documentation for Eaton

Enabling Intelligent Business Process Automation

CMU Capstone Project

Prepared By - Soham Chakraborti, Youchun Ni, Ai Chen, Woojin Park, Harita DL

- Overview
- Current Scope
- Approaches
- Python Libraries
    - Google Tesseract
    - Xpdf
    - Other openCV based libraries
- AWS Textextract
- Pricing
- Google Doc AI (Invoice Parser)
- Fields extracted
- Pricing
- Matching Function
- Sample Example Invoice
- Pipeline
    - Description
- Learnings
- Future Scope
- Conclusion

Overview

Eaton receives many documents from suppliers, customers, and governments globally. In only last year, Eaton's Global Financial Shared Services center (GFSS) received over 1.2 million supplier invoice pages from their 54000+ registered suppliers. Each invoice follows a unique supplier format. This information is received through multiple channels including postal mail, fax, email, etc. Later, each physical document is scanned into an electronic format for processing. Eaton has already leveraged Robotic Process Automation(RPA) for processing some of these documents that have helped save more than 16000 work hours. However, with so many different invoice formats and varying document quality, RPA has its limitations and much of the documents require human review. The manual review comes with the disadvantage of being a monotonous, repetitive, and time-consuming task.
With this project given to CMU, Eaton aims to increase automation capabilities by leveraging AI through cognitive services provided by technology leaders like Google, Amazon, and Microsoft. The overall idea of the project is to extract certain mandatory and optional fields from each of these invoices like Invoice Number, Invoice Date, Invoice Total Amount, Vendor Name, Purchase Order Number, each Item's number, description, quantity and amount, etc. into a consistently formatted CSV file. Having a standard formatted file with the essential details from the varyingly formatted invoices would allow Eaton to use this as an input for any other RPA process that benefits them. Considering the varying formats, the initial scope of the project was to only fetch the mandatory fields from these invoices.

Eaton, receives around 50000 invoices per day from more than 5000+ different vendors each with different formats of invoice. The sheer scale and variety of invoices makes it a very difficult parsing task for traditional Robotic Process Automation tools, hence Eaton needs a more effective solution to extract meaningful data from these invoices.

A better solution will have many benefits including the following:

- Less time consuming
- Less prone to error, since manual intervention can be limited
- More efficient and cost effective.

We explored various existing technologies to develop a customized tool for Eaton to use. during

Current Scope

Our current scope of work includes extracting the mandatory fields from invoice PDFS and been restricted to just English language based PDF's.

The mandatory fields are :

- Invoice_id
- Company_name

- File_number
- Currency
- Invoice_date
- Net_amount
- Purchase_order
- Total_amount
- total_tax_amount

Approaches

PDF or Portable Document File format is one of the most common file formats in use today. It is widely used across enterprises, in government offices, healthcare and other industries. As a result, there is a large body of unstructured data that exists in PDF format and to extract and analyse this data to generate meaningful insights is a common task among data scientists. In case of Eaton, successful parsing of PDF data will enable them to seamlessly integrate with their CRM tool.

Python Libraries

Google Tesseract

Python-tesseract is an optical character recognition (OCR) tool for python. That is, it will recognize and "read" the text embedded in images.
Python-tesseract is a wrapper for Google's Tesseract-OCR Engine. It is also useful as a stand-alone invocation script to tesseract, as it can read all image types supported by the Pillow and Leptonica imaging libraries, including jpeg, png, gif, bmp, tiff, and others. Additionally, if used as a script, Python-tesseract will print the recognized text instead of writing it to a file. [https://pypi.org/project/pytesseract/]

Xpdf

The Xpdf open source project includes a PDF viewer (XpdfViewer only available on Linux and Windows) along with a collection of command line tools which perform various functions on PDF files like *pdftotext, pdftopng or pdftohtml.* The source code is written in C++.We can get Xpdf's CLI on Python thanks to the free software utility library Poppler. Poppler is a fork of Xpdf-3.0.

Other openCV based libraries

Firstly, we need to convert the pages of the PDF to images and then, use OCR (Optical Character Recognition) to read the content from the image and store it in a text file.

pip3 install PIL

pip3 install pytesseract

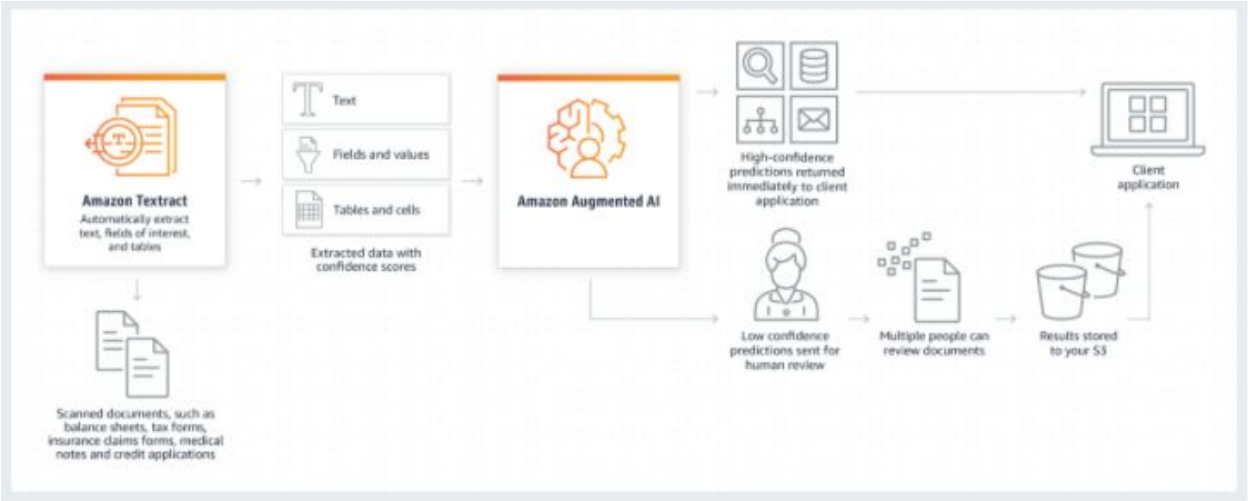
pip3 install pdf2image

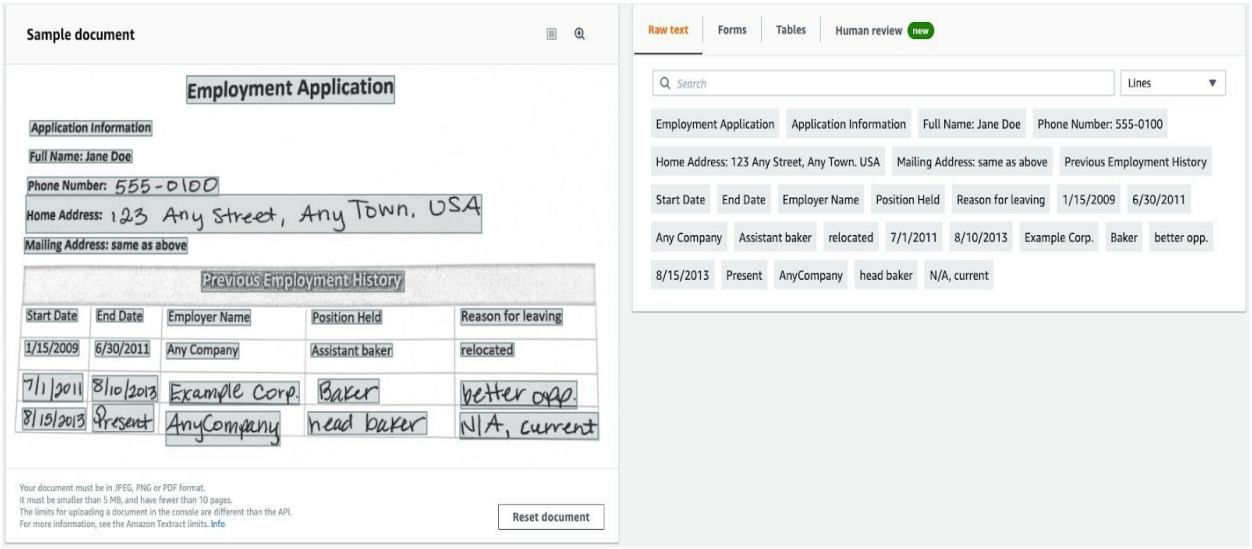
sudo apt-get install tesseract-ocr

Sample code can be found here
https://www.geeksforgeeks.org/python-reading-contents-of-pdf-using-ocr-optical-character-recognition/

AWS Textextract

Amazon Textract is a fully managed machine learning service that automatically extracts printed text, handwriting, and other data from scanned documents that goes beyond simple optical character recognition (OCR) to identify, understand, and extract data from forms and tables. Many companies today extract data from scanned documents, such as PDF's, tables and forms, through manual data entry (that is slow, expensive and prone to errors), or through simple OCR software that requires manual configuration which needs to be updated each time the form changes to be usable.





Pricing

Analyze Document API for pages with tables and forms

Region: US East (Ohio)

Monthly	OCR	Price per page	Effective Price per 1,000 Pages
First 1 Million pages	Included	$0.015 + $0.05	$15.00 + $50.00
Over 1 Million pages	Included	$0.01 + $0.04	$10.00 + $40.00

Sample Code

```
import boto3
from trp import Document
# Document
s3BucketName = "ki-textract-demo-docs"
documentName = "employmentapp.png"
# Amazon Textract client
textract = boto3.client('textract')
# Call Amazon Textract
response = textract.analyze_document(
    Document={
        'S3Object': {
            'Bucket': s3BucketName,
            'Name': documentName
        }
    },
    FeatureTypes=["FORMS"])

doc = Document(response)
for page in doc.pages:
    # Print fields
    print("Fields:")
    for field in page.form.fields:
        print("Key: {}, Value: {}".format(field.key, field.value))

    # Get field by key
    print("\nGet Field by Key:")
    key = "Phone Number:"
    field = page.form.getFieldByKey(key)
    if(field):
        print("Key: {}, Value: {}".format(field.key, field.value))

    # Search fields by key
    print("\nSearch Fields:")
    key = "address"
    fields = page.form.searchFieldsByKey(key)
    for field in fields:
        print("Key: {}, Value: {}".format(field.key, field.value))
```

Google Doc AI (Invoice Parser)

Document AI allows you to take invoices in a variety of formats and return structured data to reduce manual document processing.The invoice parser extracts both header and line item fields, such as invoice number, supplier name, invoice amount, tax amount, invoice date, due date, and line item amounts.

Fields extracted

The following is a list of the exemplar annotations returned in a Document response object's entities field.

This example list may be expanded in current or future API versions:

Field	Annotations
entities	● **carrier** ● **currency** ● **customer_tax_id** ● **delivery_date** ● **due_date** ● **invoice_date** ● **invoice_id** ● **net_amount** ● **payment_terms** ● **purchase_order** ● **receiver_address** ● **receiver_name** ● **remit_to_address** ● **remit_to_name** ● **ship_to_address** ● **ship_to_name** ● **supplier_address** ● **supplier_name** ● **supplier_tax_id**

	• **total_amount** • **total_tax_amount**
nested entities	• **line_item/amount** • **line_item/unit_price** • **line_item/quantity** • **line_item/unit** • **line_item/description** • **line_item/product_code**
nested entities.normalized_value	• **moneyValue (ISO 4217 text format)** • **dateValue (ISO 8601 text format)** • **datetimeValue (ISO 8601 text format)** • **addressValue**

Processor details

File types supported	PDF, TIFF, GIF
Maximum number of pages (online/synchronous)	5
Maximum number of pages (offline/asynchronous/batch)	10
Maximum file size	20Mb

Pricing

Form Parser & Document OCR Pricing overview

Your usage of Document AI Form Parser & Document OCR processor is calculated in terms of how many pages you send for annotation.

Processor	Price per 1000 pages	
	1 - 1,000,000 pages/month	1,000,001+ pages/month
Form Parser	$65 per 1000 pages	$50 per 1000 pages
Document OCR Processor	$1.50 per 1000 pages	$0.60 per 1000 pages

If you pay in a currency other than USD, the prices listed in your currency on Cloud Platform SKUs apply.

Sample Code

```
import re
import csv

from google.cloud import documentai_v1beta3 as documentai

from os import walk
import glob

project_id= '';
location = 'us';
processor_id = '';
file_path = '19934117-GENERAL METALS INC.PDF';
dir_path = 'invoices/*.pdf'

def write_to_excel(rows):
     # field names
       fields = ['Name', 'Value']
       # name of csv file
       filename = "GM_google.csv"

       # writing to csv file
       with open(filename, 'w') as csvfile:
           # creating a csv writer object
           csvwriter = csv.writer(csvfile)
```

```
            # writing the fields
            csvwriter.writerow(fields)

            # writing the data rows
            csvwriter.writerows(rows)


def process_document_sample(
    project_id: str, location: str, processor_id: str, file_path: str
):
    # Instantiates a client
    client = documentai.DocumentProcessorServiceClient()

    # The full resource name of the processor, e.g.:
    # projects/project-id/locations/location/processor/processor-id
    # You must create new processors in the Cloud Console first
    name = f"projects/<project-id>/locations/us/processors/<processor-id>"

    with open(file_path, "rb") as image:
        image_content = image.read()

    # Read the file into memory
    document = {"content": image_content, "mime_type": "application/pdf"}

    # Configure the process request
    request = {"name": name, "document": document}

    # Recognizes text entities in the PDF document
    result = client.process_document(request=request)

    document = result.document

    print("Document processing complete.")

    document_entities = document.entities

    # Read the text recognition output from the processor
    rows=[]
    for entity in document_entities:
        rows.append([entity.type_,entity.mention_text])
    write_to_excel(rows)


# Extract shards from the text field
def get_text(doc_element: dict, document: dict):
    """
    Document AI identifies form fields by their offsets
```

```
    in document text. This function converts offsets
    to text snippets.
    """
    response = ""
    # If a text segment spans several lines, it will
    # be stored in different text segments.
    for segment in doc_element.text_anchor.text_segments:
        start_index = (
            int(segment.start_index)
            if segment.start_index in doc_element.text_anchor.text_segments
            else 0
        )
        end_index = int(segment.end_index)
        response += document.text[start_index:end_index]
    return response


process_document_sample(processor_id,location, processor_id, file_path)
```

Matching Function

Matching Function is the analytics purpose program which allows you to integrate two csv outputs from Google Docai and AWS Textract by applying customized code in your local system. It highlights the 'Mismatched' entries also at the same time, put another function layer in this step to simply classifying each vendor's invoice into different bucket based on the 'Entity Extraction Score (EES)' : Scale between 0 to 1, it indicates how many mandatory fields are extracted out of 7 mandatory fields by each cloud service. The EES will be applied to do the bucketization, you can extract the vendor population belonging to the Bucket groups which has extraction score lower than certain threshold like 60% to further data extraction and exploration.

Sample Code

In [19]:
```
## This is the program of Matching Function of 7 mandatory fields
# which are  which is 'Invoice_Number', Invoice_Date', 'Invoice_Total' ,
#'Vendor_Name', 'Currency', 'Taxes','Purchase_Order_Number'
# other mandatory fields need to be stored in different output csv spreadsheet
# which includes all the 'table output'fields of each invoice.

```

In [20]:
```
# import libraries to use for data analysis
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import numpy as np
from datetime import datetime
from os import listdir
```

In [21]:
```
## This is the function for accessing the s3 bucket includes google docai and awste

def load_dataset(dataset_dir):

    # define data locations for output matching
    # Iterate through all files in the folder to find two given csv files
        for filename in listdir(dataset_dir):
            if (filename == 'aws_output.xlsx'):
                aws_df = pd.read_excel(dataset_dir+'/'+filename)
                print('aws file found!')
            elif (filename == 'google_output_raw.xlsx') :
                google_df = pd.read_excel(dataset_dir+'/'+filename)
                print('googl file found!')
                print('\n')
            else :
                continue

        # Merge the two prediction output dataframe
        merged = aws_df.reset_index().merge(google_df.reset_index(), left_index=Tru
        print(merged.info()) # giving the merged dataframe info()
        print('\n')
        print(merged.head(2))# printing out the merged dataframe first 2 rows examp
        return merged

```

In [22]:
```
# Extraction Score (EES) : Scale between 0 to 1, it indicates how many mandatory
# fields are extracted out of 7 mandatory fields by each cloud service.
```

In [23]:
```
## Define function for Bucketizing the extraction score into 3 groups -
# user can modify the extraction score for each bucket

def bucketizing_extraction_score(row):
    if row["DocAi_Extraction_Score"] > 0.9:
        return 1
    elif row["DocAi_Extraction_Score"] >0.7:
        return 2
    else:
        return 3

```

In [24]:

```
## This is the function for modifying the order or rename the columns /
# generate new matched variables and calculated confidence score variable to add ne

def all_modify_dataset(merged):
    # define data locations for output matching
    merged = merged.iloc[:,[1,2,3,4,5,6,8,16,20,23,17,19,24,22]]

    # Calculating the confidence Score
    AWS_Extraction_Score= (1-(merged.iloc[:,0:7].isnull().sum(axis=1)/7)).round(3).
    DocAi_Extraction_Score = (1-(merged.iloc[:,7:].isnull().sum(axis=1)/7)).round(3

    # re-ordering the merged dataframe to match each cloud service's output columns
    merged= merged.iloc[:,[0,7,1,8,2,9,3,10,4,11,5,12,6,13]]

    # re-name the dataframe columns
    merged.columns = ['Invoice_Number_AWS', 'Invoice_Number_DocAi','Invoice_Date_AW
              'Invoice_Total_AWS','Invoice_Total_DocAi','Vendor_Name_AWS','Vendor_N
              'Taxes_AWS','Taxes_DocAi','Purchase_Order_Number_AWS','Purchase_Order_

    ## replace the nan value to Null
    merged = merged.replace(np.nan, 'NULL')

    ## This is the function for generating matched/mismatched variables
    # and also generate the confidence score for each cloud service output
    comparison_column1 = np.where(merged["Invoice_Number_AWS"] == merged["Invoice_N
    comparison_column2 = np.where(merged["Invoice_Date_AWS"] == merged["Invoice_Dat
    comparison_column3 = np.where(merged["Invoice_Total_AWS"] == merged["Invoice_To
    comparison_column4 = np.where(merged["Vendor_Name_AWS"] == merged["Vendor_Name_
    comparison_column5 = np.where(merged["Currency_AWS"] == merged["Currency_DocAi"
    comparison_column6 = np.where(merged["Taxes_AWS"] == merged["Taxes_DocAi"], 'Ma
    comparison_column7 = np.where(merged["Purchase_Order_Number_AWS"] == merged["Pu

    # generate mached output variables
    merged["Invoice_Number_matched"] = comparison_column1
    merged["Invoice_Date_matched"] = comparison_column2
    merged["Invoice_Total_matched"] = comparison_column3
    merged["Vendor_Name_matched"] = comparison_column4
    merged["Currency_matched"] = comparison_column5
    merged["Taxes_matched"] = comparison_column6
    merged["Purchase_Order_Number_matched"] = comparison_column7

    # re-arranging the column order
    merged=merged.iloc[:,[0,1,14,2,3,15,4,5,16,6,7,17,8,9,18,10,11,18,12,13,20]]

    # generate confidence score variable
    merged['AWS_Extraction_Score'] = AWS_Extraction_Score
    merged['DocAi_Extraction_Score'] = DocAi_Extraction_Score

    # bucketizing the invoices
    merged = merged.assign(DocAi_Extraction_score_bucket= merged.apply(bucketizing_

    merged = merged.iloc[:,[-1,-2,-3,0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18
    print(merged.info())
    print(merged.head(2))

    return merged
```

In [26]:
```
## Need to give the directory path of S3 BUCKET
dataset_dir = '/Users/Jay/Desktop/university_new/CMU/2020 Fall/Information Systems P
merged = load_dataset(dataset_dir) # Assigned first dataframe formatted merged file
final_df = all_modify_dataset(merged) #final ouput generated
```

```
aws file found!
googl file found!


<class 'pandas.core.frame.DataFrame'>
RangeIndex: 20 entries, 0 to 19
Data columns (total 25 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   index_x                20 non-null     int64
 1   Invoice_Number         19 non-null     float64
 2   Invoice_Date           16 non-null     object
 3   Invoice_Total          8 non-null      float64
 4   Vendor_Name            20 non-null     object
 5   Currency               0 non-null      float64
 6   Taxes                  18 non-null     float64
 7   Subtotal               11 non-null     object
 8   Purchase_Order_Number  15 non-null     float64
 9   Document_Type          0 non-null      float64
 10  Item_Description       0 non-null      float64
 11  Quantity               9 non-null      float64
```

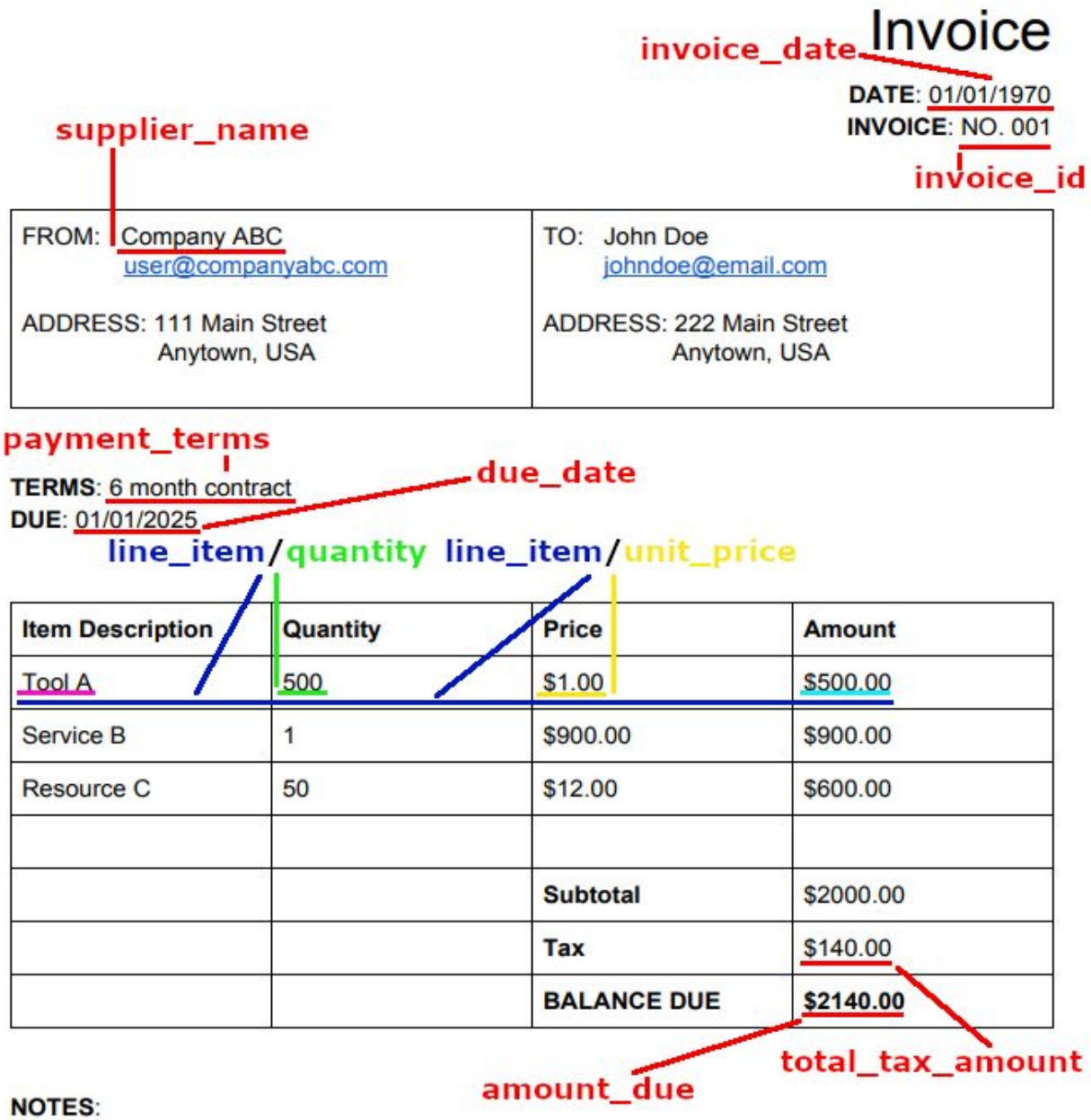
In [27]:
```
## finally download the output into output bucket
final_df.to_csv('merged_output.csv')
```

In [28]:
```
final_df.head(5)
```

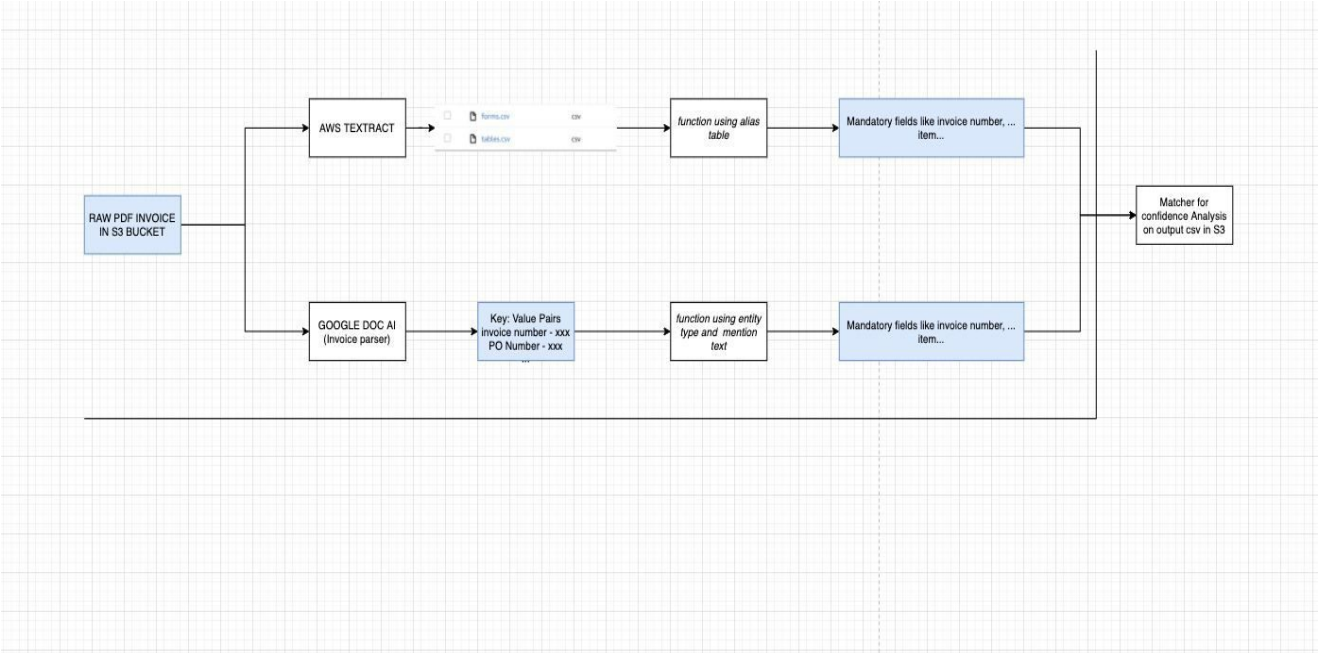
Out[28]:

	DocAi_Extraction_score_bucket	DocAi_Extraction_Score	AWS_Extraction_Score	Invoice_Number_AW
0	2	0.857	0.714	1314
1	2	0.857	0.714	1314
2	2	0.857	0.857	1314
3	2	0.857	0.571	1314
4	2	0.857	0.714	1314

Sample Example Invoice



Pipeline



Description

We use Intelligent automation through Amazon Web Services and Google Document Invoice Processor to parse these invoices.

Step 1: We use a script to load invoice files to AWS S3 bucket. The S3 bucket is a repository for storing these files.

Step 2: The pipeline triggers two processes in parallel. One the AWS Textract and Google Doc AI's Invoice parser.

Step 3: Both these services provide their output file for each input file in a separate folder. The Folder structure can be seen below.

root folder: have input/ and output/

Name	Type	Last modified	Size	Storage class
inputs/	Folder	-	-	-
outputs/	Folder	-	-	-

output/

Name	Type	Last modified	Size	Storage class
2020-12-04/	Folder	-	-	-
2020-12-05/	Folder	-	-	-
2020-12-08/	Folder	-	-	-
2020-12-09/	Folder	-	-	-

output/2020-12-09/

Name	Type	Last modified	Size	Storage class
aws/	Folder	-	-	-
google/	Folder	-	-	-

output/2020-12-09/aws/

Name	Type	Last modified	Size	Storage class
19934087-GENERAL METALS INC-forms.csv	csv	December 9, 2020, 00:08 (UTC-05:00)	1.1 KB	Standard
19934091-GENERAL METALS INC-forms.csv	csv	December 9, 2020, 00:08 (UTC-05:00)	1.1 KB	Standard
19934093-GENERAL METALS INC-forms.csv	csv	December 9, 2020, 00:07 (UTC-05:00)	1.2 KB	Standard
20107170-ACCU TOOL & DIE INC-forms.csv	csv	December 8, 2020, 23:05 (UTC-05:00)	630.0 B	Standard
20107178-ACCU TOOL & DIE INC-forms.csv	csv	December 8, 2020, 23:05 (UTC-05:00)	607.0 B	Standard
20107181-ACCU TOOL & DIE INC-forms.csv	csv	December 8, 2020, 23:05 (UTC-05:00)	582.0 B	Standard
20107184-ACCU TOOL & DIE INC-forms.csv	csv	December 8, 2020, 23:05 (UTC-05:00)	650.0 B	Standard
20107187-ACCU TOOL & DIE INC-forms.csv	csv	December 8, 2020, 23:05 (UTC-05:00)	653.0 B	Standard
20259782-FUTURE ELECTRONICS CORP-forms.csv	csv	December 9, 2020, 10:31 (UTC-05:00)	1.6 KB	Standard
20259818-FUTURE ELECTRONICS CORP-forms.csv	csv	December 9, 2020, 10:32 (UTC-05:00)	1.5 KB	Standard

output/2020-12-09/google

Name	Type	Last modified	Size	Storage class
forms.csv	csv	December 9, 2020, 12:54 (UTC-05:00)	16.1 KB	Standard
tables.csv	csv	December 9, 2020, 12:54 (UTC-05:00)	23.5 KB	Standard

Sample forms.csv

company_name	file_number	invoice_id	invoice_id-conf	currency	currency-conf	invoice_date	invoice_date-conf	net_amount	net_amount-conf	purchase_order	purchase_order-conf	total_amount	total_amount-conf	total_tax_amount	total_tax_amount-conf
ENDRIES INTERNATIONAL INC	20107450	4951841-00	0.951141417	USD	0.761774957	7/24/19	0.976538241					0	0.920344472	0	0.919695377
ENDRIES INTERNATIONAL INC	20107452	4938807-00	0.948624194	USD	0.74518466	7/24/19	0.992804468			4511209376	0.990905523	381.17	0.912502468		
ENDRIES INTERNATIONAL INC	20107451	4952044-00	0.948948562	USD	0.783095002	7/24/19	0.990732491	156.32	0.162612423	4511206662	0.987678587	156.32	0.920344472		
ENDRIES INTERNATIONAL INC	20107449	4952791-00	0.948888898	USD	0.770995736	7/26/19	0.984686911	204.91	0.17640923	4511298144	0.986244977	204.91	0.917190731		
ENDRIES INTERNATIONAL INC	20107453	4952793-00	0.948397815	USD	0.779284894	7/26/19	0.98731178	69.52	0.191752523	4511298144	0.992853463	69.52	0.915942371		
ACCU FAB	20269527	132628	0.994207144	$	1	9/3/19	0.985894799	$150.90	0.325820684	4511154734	0.99931252	150.9	0.933638692		
ACCU FAB	20269800	132606	0.994934618	$	1	9/3/19	0.982222795	$1,719.60	0.211601779	4511423519	0.999379873	1,719.60	0.919475675		
ACCU FAB	20269804	132611	0.994812429	$	1	9/3/19	0.989617586	$278.48	0.312689275	4511381239	0.998906016	278.48	0.932140946		
ACCU FAB	20269320	132608	0.994437695	$	1	9/3/19	0.982070267	$265.23	0.315760136	4511381239	0.999207079	265.23	0.933692038		
ACCU FAB	20269738	132605	0.99461621	$	1	9/3/19	0.981258333	$1,719.60	0.222934067	4511423520	0.999398112	1,719.60	0.919575095		
ACCU TOOL & DIE INC	20107178	131476	0.991751671	$	1	7/23/19	0.998846948	902	0.295068502	4511232718	0.999150038				
ACCU TOOL & DIE INC	20107187	131419	0.994480968	$	1	7/15/19	0.998796403	2,281.54	0.169761524	4511199078	0.999356925				
ACCU TOOL & DIE INC	20107181	131479	0.991783619	$	1	7/23/19	0.998929203	258	0.291707426	4511213241	0.999319732				
ACCU TOOL & DIE INC	20107170	131472	0.993503988	$	1	7/23/19	0.998940289	240.8	0.298097491	4511232718	0.99931308				
ACCU TOOL & DIE INC	20107184	131461	0.993866801	$	1	7/23/19	0.998861432	540	0.284942299	4510820709	0.998947203				
AK STEEL CORPORATION	20301773	3444347	0.997484028	USD	0.716681838	9/9/19	0.999919653			4511282498	0.998907566	16,856.47	0.920344472		
AK STEEL CORPORATION	20301778	3444239	0.997407317	USD	0.768304527	9/9/19	0.999866962			4511290611	0.998828471	2,862.45	0.920344472		
AK STEEL CORPORATION	20301764	3444815	0.998138845	USD	0.687310994	9/10/19	0.999796152			4511372525	0.999060452	43,938.30	0.920344472		
AK STEEL CORPORATION	20302893	3444350	0.997438252	USD	0.626136124	9/9/19	0.999859631			4511359957	0.99695164	17,096.15	0.920344472		
AK STEEL CORPORATION	20301770	3444346	0.99735564	USD	0.696034431	9/9/19	0.999863684			4511324472	0.999130845	501 00 681 45	0.920344472		
ATLAS METAL PARTS AND FABRICATING	20280627	25562	0.991048992	$	1	9/5/19	0.999320388	$780.00	0.098154403	4511401617	0.999677837			0	0.94610697
ATLAS METAL PARTS AND FABRICATING	20280623	25560	0.991338015	$	1	9/5/19	0.999681234			4511267108	0.999583125	591	0.920344472	0	0.943040848
ATLAS METAL PARTS AND FABRICATING	20280614	25568	0.9917171	$	1	9/5/19	0.999667764	$576.00	0.099769779	4511420985	0.999537885			0	0.945821524
ATLAS METAL PARTS AND FABRICATING	20280619	25567	0.991391897	$	1	9/5/19	0.999669313	$600.00	0.098673038	4511420985	0.999597907			0	0.946705282
ATLAS METAL PARTS AND FABRICATING	20281498	25563	0.989793599	$	1	9/5/19	0.998996735			4511401617	0.999630392	275.2	0.920344472	0	0.944401503
AVEVA SOFTWARE LLC	20304197	4401757	0.994571447	USD	0.633917987	9/10/19	0.996940911			4511452117	0.99296093	0	0.920344472	0	0.935669124
AVEVA SOFTWARE LLC	20302853	4401818	0.994318306	USD	0.6932289	9/10/19	0.996889949			4511403593	0.994071186	4,542.04	0.920344472	0	0.93592149
AVEVA SOFTWARE LLC	20309094	4401865	0.994936824	USD	0.713043869	9/11/19	0.997180879	2,268.00	0.136112243	4509329937	0.994408965	2,268.00	0.920344472	0	0.932030857
AVEVA SOFTWARE LLC	20310571	4401981	0.994482398	USD	0.691228926	9/11/19	0.998841166			4511458416	0.993703425	1,930.00	0.920344472	0	0.922531366
AVEVA SOFTWARE LLC	20304169	4401817	0.994696558	USD	0.699174464	9/10/19	0.997111499			4511454323	0.994924307	8,875.82	0.920344472	0	0.935817719
BRT EXTRUSIONS INC	20307786	122906	0.996989012			9/10/19	0.999287605	4,330.02	0.181608111	4511393733	0.995854378	4,330.02	0.914950609		
BRT EXTRUSIONS INC	20307784	122903	0.996025205			9/10/19	0.999244452	1,633.38	0.173318803	4511394223	0.99355191	1,633.38	0.914379358		
BRT EXTRUSIONS INC	20307785	122905	0.997092783			9/10/19	0.999198437	4,983.11	0.147195175	4511337479	0.994304717	14,477.02	0.914579749		
BRT EXTRUSIONS INC	20307783	122902	0.997040927			9/10/19	0.999183476	9,017.89	0.177597836	4511371882	0.995661557	9,017.89	0.914988458		

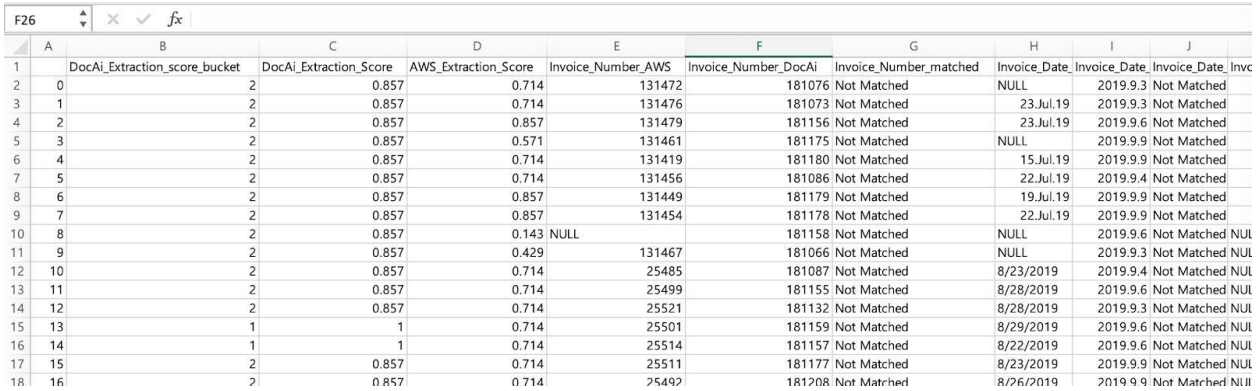
Sample tables.csv

tables

company_name	file_number	line_number	invoice_id	line_item/amount	line_item/amount-conf	line_item/unit_price	line_item/unit_price-conf	line_item/quantity	line_item/quantity-conf	line_item/unit	line_item/unit-conf	line_item/description
ENDRIES INTERNATIONAL INC	20107450	1	4951841-00	0		0				ea		
ENDRIES INTERNATIONAL INC	20107450	2	4951841-00	0		0				EA		
ENDRIES INTERNATIONAL INC	20107450	3	4951841-00									1XBR 5/16-18 GRADE F FLANGE CENTER LOCKNUT ZINC & WAX PO#: SAMPLES/NEW VMI PARTS
ENDRIES INTERNATIONAL INC	20107452	1	4938807-00	44.89		0.44894		100		ea		
ENDRIES INTERNATIONAL INC	20107452	2	4938807-00									EUGW .406x1.25X.064 BRASS FLAT WASHER
ENDRIES INTERNATIONAL INC	20107452	3	4938807-00	7.34		0.0734				EA		100 01 PR 3/8-1681-1/2 GRADE 5 CARRIAGE BOLT PLAIN
ENDRIES INTERNATIONAL INC	20107452	4	4938807-00	105		1.26 OD)		100		EA		
ENDRIES INTERNATIONAL INC	20107452	5	4938807-00	0		0.7231		100		EA		0839288801 2764 1 STEEL LOCKNUT APPLETON BL-100
ENDRIES INTERNATIONAL INC	20107452	6	4938807-00	30		0.3		100		EA		M8X20 A2 STAINLESS DIN6912 LOW SOCKET HEAD CAP S
ENDRIES INTERNATIONAL INC	20107452	7	4938807-00	24.65		0.2465				ea		
ENDRIES INTERNATIONAL INC	20107452	8	4938807-00									100 GRHJ 3/8-16X1-3/4 GRADE 8 CARRIAGE BOLT PLAIN
ENDRIES INTERNATIONAL INC	20107452	9	4938807-00	169.29		0.5643		300		EA		
ENDRIES INTERNATIONAL INC	20107451	1	4952044-00									MRA 2019-07-22-05 TO CREDIT 4920562-00 7/9/19 W/REBILL TO FOLLOW UNDER CORRECT SHIP TO 02006
ENDRIES INTERNATIONAL INC	20107451	2	4952044-00									
ENDRIES INTERNATIONAL INC	20107451	3	4952044-00	156.32		0.3908				EA		
ENDRIES INTERNATIONAL INC	20107451	4	4952044-00									400- 400-X314 3/8-1683-1/2 18-8 STAINLESS STEEL HEX HEAD CAP SCRE
ENDRIES INTERNATIONAL INC	20107449	1	4952791-00	156.46		0.05588		2800		ea		
ENDRIES INTERNATIONAL INC	20107449	2	4952791-00	15.09		0.0503				EA		1/2 TYPE B NARROW 18-8 STAINLESS STEEL FLAT WASHE PO#: 4511298144 Line #: 160 Bin Location: 0093 .531X1.00x.063
ENDRIES INTERNATIONAL INC	20107449	3	4952791-00									300 300 X104 1/4-20X1 18-8 STAINLESS STEEL HEX HEAD CAP SCREW
ENDRIES INTERNATIONAL INC	20107449	4	4952791-00	33.36		0.33362				EA		
ENDRIES INTERNATIONAL INC	20107449	5	4952791-00									100 100 X502 1/2-13x1/2 18-8 STAINLESS STEEL HEX HEAD CAP SCREW
ENDRIES INTERNATIONAL INC	20107453	1	4952793-00	42.64		0.2132		200		EA		
ENDRIES INTERNATIONAL INC	20107453	2	4952793-00									X503 1/2-13X3/4 18-8 STAINLESS STEEL HEX HEAD CAP SCREW
ENDRIES INTERNATIONAL INC	20107453	3	4952793-00	26.88		0.26882				EA		
ACCU FAB	20269527	1	132628	150.9		$15.09		10		EA		REAR PANEL OULET BOX
ACCU FAB	20269800	1	132606	1,719.60		$28.66		17948		EA		BATT TRAY, IBC-L WELD

Step 4: Each field extracted has a confidence level associated with it. Our matching function does an analysis of this data, to suggest which output to use. In our case it was Google Doc AI (Invoice Processor). Entity Extraction Score (EES) : Scale between 0 to 1, it indicates how many mandatory fields are extracted out of 7 mandatory fields by each cloud service.

Sample matched.csv



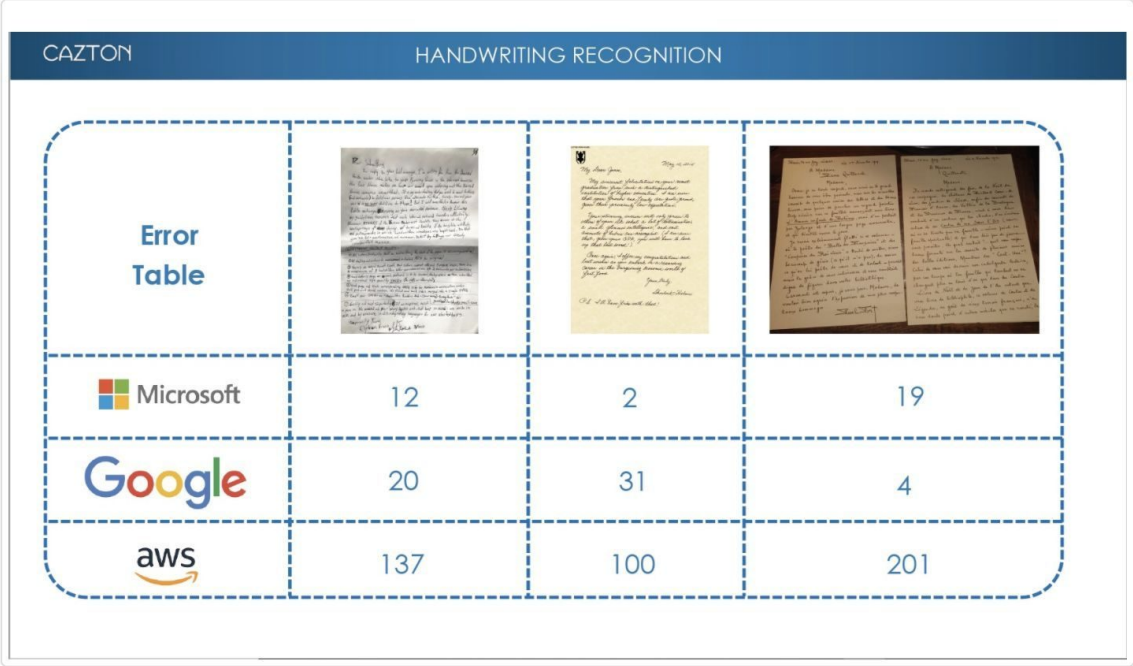
F26

	A	B	C	D	E	F	G	H	I	J	
1		DocAi_Extraction_score_bucket	DocAi_Extraction_Score	AWS_Extraction_Score	Invoice_Number_AWS	Invoice_Number_DocAi	Invoice_Number_matched	Invoice_Date_	Invoice_Date_	Invoice_Date_	Invc
2	0	2	0.857	0.714	131472	181076	Not Matched	NULL	2019.9.3	Not Matched	
3	1	2	0.857	0.714	131476	181073	Not Matched	23.Jul.19	2019.9.3	Not Matched	
4	2	2	0.857	0.857	131479	181156	Not Matched	23.Jul.19	2019.9.6	Not Matched	
5	3	2	0.857	0.571	131461	181175	Not Matched	NULL	2019.9.9	Not Matched	
6	4	2	0.857	0.714	131419	181180	Not Matched	15.Jul.19	2019.9.9	Not Matched	
7	5	2	0.857	0.714	131456	181086	Not Matched	22.Jul.19	2019.9.4	Not Matched	
8	6	2	0.857	0.857	131449	181179	Not Matched	19.Jul.19	2019.9.9	Not Matched	
9	7	2	0.857	0.857	131454	181178	Not Matched	22.Jul.19	2019.9.9	Not Matched	
10	8	2	0.857	0.143	NULL	181158	Not Matched	NULL	2019.9.6	Not Matched	NUL
11	9	2	0.857	0.429	131467	181066	Not Matched	NULL	2019.9.3	Not Matched	NUL
12	10	2	0.857	0.714	25485	181087	Not Matched	8/23/2019	2019.9.4	Not Matched	NUL
13	11	2	0.857	0.714	25499	181155	Not Matched	8/28/2019	2019.9.6	Not Matched	NUL
14	12	2	0.857	0.714	25521	181132	Not Matched	8/28/2019	2019.9.3	Not Matched	NUL
15	13	1	1	0.714	25501	181159	Not Matched	8/29/2019	2019.9.6	Not Matched	NUL
16	14	1	1	0.714	25514	181157	Not Matched	8/22/2019	2019.9.6	Not Matched	NUL
17	15	2	0.857	0.714	25511	181177	Not Matched	8/23/2019	2019.9.9	Not Matched	NUL
18	16	2	0.857	0.714	25492	181208	Not Matched	8/26/2019	2019.9.9	Not Matched	NUL

Step 5: Our Dataset was 1000 files, we ran a test on 400 files and found a 99% accuracy in terms of extraction of fields by Google Doc AI (Invoice processor).

Learnings

Extracting data from unstructured pdf is not an easy task. Coming up with a single solution that solves all problems is tough, and would not lead to great levels of accuracy. Google's Invoice processor is a great service that is customized for invoices and involves Named-entity-relation that generates Key-value pairs which can be consumed by the CRM tool.



[Reference: https://cazton.com/blogs/executive/form-recognition-azure-aws-gcp]

- This was done using google doc ai not the latest invoice processor that we are using.

Future Scope

Some of the following areas can be improved/added to further improve Invoice parsing:

- Various other language support, Google Doc AI works well on various languages, so the current implementation can be further extended
- Man in the middle approach, for invoices that fail to meet our expectations (in terms of fields extracted or accuracy) we can flag those to be manually handled.
- Continuous learning, Google Doc Invoice Parser phase 2 will have a feedback loop that takes failed Invoices to learn. This ensures that the system improves continuously.
- Integrating other approaches, adding other approaches to have assembled technologies, to improve the accuracy.

Conclusion

Overall, by leveraging cognitive services from these various cloud providers, we extend beyond the capabilities of Robotic Process Automation(RPA) and orient the solution towards the use of Intelligent Automation. The solution can process such varying formatted files and yet through alias matching, the model can retrieve the required fields and their corresponding values.
This solution benefits the client, Eaton, in terms of reducing the manual indexing work that would be monotonous and time-consuming. It would be a highly repetitive task given the large volume of invoices that Eaton receives. With the solution coming in place, Eaton can reduce the number of employees working on this process and use them for more creative and complex tasks.

Like every other automation, there will be some sort of manual intervention required to validate the output of these cognitive services. The manual workforce would also be required to check the exceptions and mismatched fields. While the aim is to reduce manual work as much as possible, the model can be improved with newer features being developed by these various providers, to attain results that require no human involvement at any stage. Moving towards automation would also help Eaton with faster processing of these invoices and reduce the error rate that would otherwise be more probable by manual indexing.