

Extracting Data from

PDEs with Python

service station sites that lent themselves to more economic use, the company leased three sites in Montreal and one in Vancouver for non-petroleum redevelopment.

Environmental Protection

It is the company's policy to lobby government regulations for environmental protection and to encourage establishment of environmentally sound and technically achievable standards. In 1973, there was a significant increase in the number of environmental standards proposed or enacted by governments. Imperial assisted in the formulation of many of these standards and devoted increased effort to the planning and execution of programs to meet new regulations.

In 1973, the company's expenditures on environmental projects were \$3.8 million, or 1.4 per cent of total over the last three years to \$84 million. The 1973 expenditures represent 11.4 per cent of

Imperial's total capital spending for the year.

One example is the three-year modernization and environmental control program under way at Ioco refinery in British Columbia involving capital expenditures of about \$12 million. Much of this program was developed by Imperial in research department with assistance from the British Columbia Research Council.

A study of the environmental impact of natural gas production in the Mackenzie Delta, which began in 1972, was carried out through 1973 at a cost for the year of more than \$250,000. In mid-year, Imperial was joined in the work by two

other companies with gas reserves in the area. This field work, together with an analysis of the environmental impact of all the facilities in this area, will be completed in 1974.

Imperial led in the formation of the Delta Environmental Protection Unit. This Mackenzie Delta cooperative unit prepared an oil spill contingency plan and purchased barges, pumps, booms, skimmers, containment boom and other anti-spill equipment at a cost of more than \$225,000, and trained operators in their use.

Government Actions and Implications

The federal and provincial governments took many actions and announced various intentions in 1973 that affect the oil industry. In dealing with the tight supply situation, the federal government placed controls on the exports of crude oil and some petroleum products. It also instituted a voluntary price freeze on crude oil and petroleum products and

placed an export tax on all crude oil exported. Support was announced for pipelines to carry western Canadian crude to Montreal and Arctic gas to southern markets, and the federal government stated its intention to create a national petroleum company. It also passed legislation to allocate petroleum products at the wholesale level and to ration them at the retail level if such actions should become necessary.

As world demand for oil increased, the flow of imported products into Canada was reduced, particularly of heating fuels into Ontario and heavy fuel oils into British Columbia, although other products and areas were also affected. The government's re-

Below: Montreal East refinery supplies the oil to fuel Ontario Hydro's heavy water plant at Douglas Point, Ont. In 1973, these 63-car unit trains made 87 trips and carried 2,350,000 barrels of the fuel

Lubov McKone

lmckone1@jh.edu

Data Services

data bytes

"Bite"-sized Data Talks on Mondays

Learn more and register
at bit.ly/data-bytes



Making Census of the US Census

September 16th, 12 - 1 pm

Extracting Data from PDFs with Python

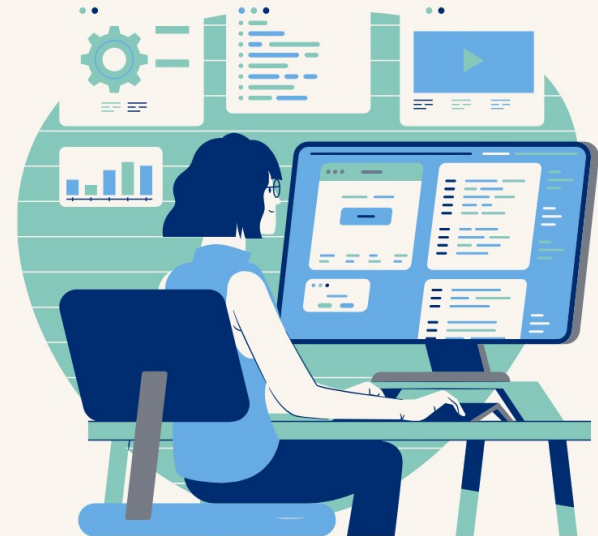
October 7th, 12 - 1 pm

Creating Software Management Plans for Open Science Initiatives

October 28th, 12 - 1 pm

The Hidden Mapping Powers of ArcGIS Arcade

November 18th, 12 - 1 pm



A brief history of PDFs

- Portable Document Format - released by Adobe in 1993
- Developed with the idea that every document should be readable and **printable** on **any device** while preserving the fidelity of the content

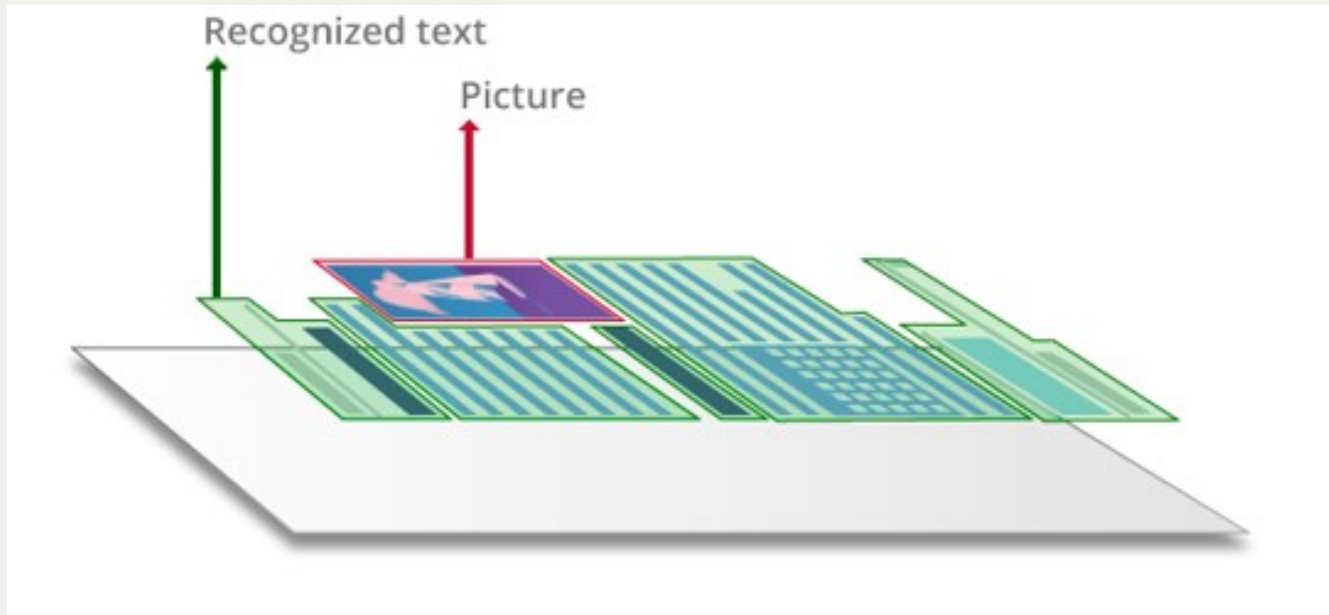
A brief history of PDFs

- Grew in popularity from the 1990s to 2000s, became an **open format** in 2008
- Multiple standards have developed under the PDF format, and not all PDFs are alike when it comes to working with them

3 types of PDFs

“True” or digitally created PDFs

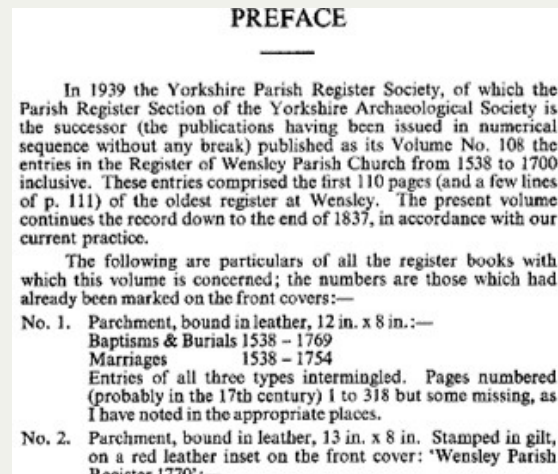
These are PDFs created digitally using software such as Microsoft Word. They consist of searchable text and images.



3 types of PDFs

“Image-only” or scanned PDFs

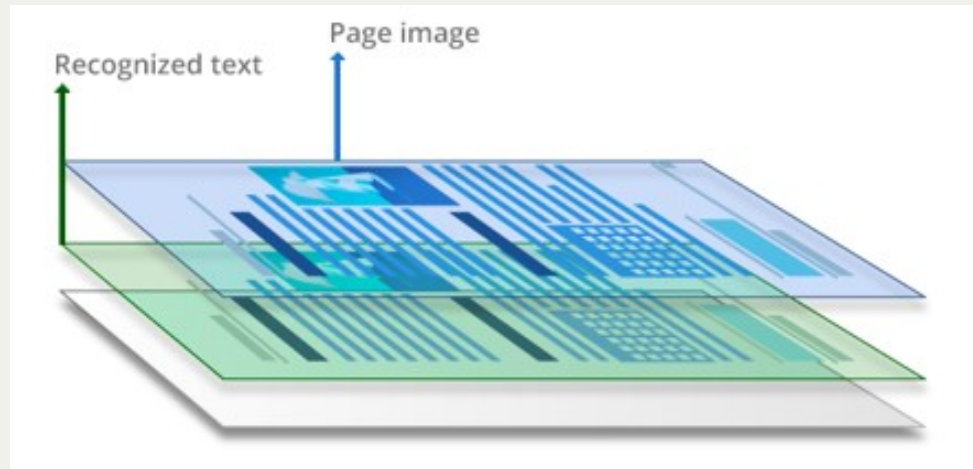
These are PDFs that are created by scanning or photographing hard copy documents. They contain only the scanned/photographed images of pages and are not automatically searchable.



3 types of PDFs

Searchable or OCR'd PDFs

These are the result of applying Optical Character Recognition (OCR) to image-only PDFs. The resulting PDF has two layers – one with the page image, and the other containing the recognized text.



PDFs as data

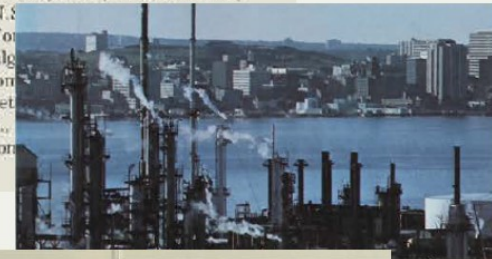
What can be extracted from PDFs?

- Text
- Images
- Structured data (tables)

Imperial Oil Limited was incorporated under the Canada Joint Stock Companies Act, 1877 on September 8, 1880. Its head office is at 111 St. Clair Avenue West, Toronto, Ontario, M5W 1K3.

Imperial Oil Limited shares may be transferred at the following offices: head office of Imperial Oil Limited; principal offices of Montreal Trust Company at St. John's, Nfld., Charlottetown, P.E.I., Halifax, N.S., Montreal, Que., Toronto, Regina, Sask., Calgary, Bankers Trust Company.

The annual meeting held at 11:00 a.m., the Canadian Room, Ontario.



Financial and Operating Highlights				
Financial*		1973	1972	
		millions of dollars		
Earnings	\$	228	157	
Shareholders' dividends		104	77	
Revenue from all sources		2,548	2,104	
Capital and exploration expenditures		333	259	
Taxes charged against income		250	172	
Total taxes generated		553	431	
Earnings per share	\$	1.78	1.22	
Dividends per share		80	60	
		percentages		
Earnings change from previous year		45.4	11.1	
Earnings as a percentage of shareholders' investment		18.5	13.9	
Capital employed		14.5	11.1	
**1972 restated for change to equity accounting				
Operating		1973	1972	
		thousands of barrels per day		
Petroleum product sales		440	417	
Crude oil processed at refineries		441	399	
Crude oil and natural gas liquids				
gross production		345	282	
net production		275	224	
Natural gas sales (millions of cubic feet per day)			480	425
		Gross proved reserves **		
crude oil and natural gas liquids (millions of barrels)		1,338	1,307	
natural gas (trillions of cubic feet)		2,896	3,000	
** Excludes Reservoir Basin discoveries				

A word on libraries

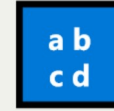
There are several libraries for working with PDFs in Python that do many of the same things. During this training we'll be comparing their performance for different tasks:

- `pdfplumber`
- `pyPDF`
- `PyMuPDF`

PDFs as data

- To Python, every PDF is made up of two components: the document metadata and a set of pages.
- Each page consists of objects that can be classified as:
 - characters
 - lines
 - rectangles
 - curves
 - images
 - and metadata about each of these objects

Extracting text from PDFs



Goal: Extract the text from the 1973 Imperial Oil Report

Some questions:

- Will the text be extracted left to right across the page, or by column?
- How can we combine the text from all of the pages?

Code for extracting text

```
1 # Extracting text with pdfplumber
2 with pdfplumber.open("../Data/Imperial Oil Annual Report 1973.pdf") as pdfp
3     print(pdfplumber_1973.pages[1].extract_text())
4
5 # Extracting text with pymupdf
6 import pymupdf
7 pymupdf_1973 = pymupdf.open("../Data/Imperial Oil Annual Report 1973.pdf")
8 print(pymupdf_1973[1].get_text())
9
10 # Extracting text with pypdf
11 from pypdf import PdfReader
12 pypdf_1973 = PdfReader("../Data/Imperial Oil Annual Report 1973.pdf")
13 document_text = ''
14 # loop over the pages
15 for page in pypdf_1973.pages:
16     # remove newlines from extracting text
17     page_text = page.extract_text().replace("\n", "")
18     document_text += page_text
```

Extracting text from PDFs - summary

- Libraries for text extraction:
 - `pdfplumber` - has nice filtering features and great documentation, but processes text line-by-line so not great for multi-column spreads.
 - `pyPDF` - great library for high-quality basic text extraction.
 - `pyMuPDF` - similar to `pyPDF` with varying accuracy, so good to check both.

Extracting text from PDFs - tips

- Check your results! Compare outputs across libraries.
- Make sure your output reflects the layout of the text, and doesn't just scan left to right.
- PDFs often contain hidden newlines to preserve the layout of the text, so keep an eye out for strange line breaks and remove `\n` characters if necessary.
- Loop over pages to combine document text.

Extracting images from PDFs

- Because our old document was OCR'd, we were able to extract at least some text from it
- However, OCR doesn't extract images - if you remember, for OCR'd documents, the entire page actually has two layers, the page image and the OCR'd text - so if we tried to extract images, we'd just get the full page images.
- For image detection, we're limited to working with digital PDFs - such as the 2023 Imperial Oil Annual Report

Code for extracting images

```
1 # read in the document with pypdf
2 pypdf_2023 = PdfReader("../Data/Imperial Oil Annual Report 2023.pdf")
3
4 for page in pypdf_2023.pages:
5     # loop over the images
6     for count, image_file_object in enumerate(page.images):
7         # write each image to a .jpg file
8         with open(str(count) + image_file_object.name, "wb") as fp:
9             fp.write(image_file_object.data)
```

Extracting tables from PDFs

Some of these libraries can actually detect and extract tables from PDFs! This feature also performs best on digital PDFs, and doesn't always detect things that we as humans know to be tables.

The best implementaton by far is in [PyMuPDF](#) - let's give it a try on a [Baltimore Police Department Weekly Incident Report](#) from [Open Baltimore](#).

Code for extracting tables

```
1  police_report = pymupdf.open("../Data/police_report_week12.pdf")
2  page = police_report[0]
3  tabs = page.find_tables()
4  df = tabs[0].to_pandas()
5
6  page_text = page.get_text()
7
8  table_dates = page_text[page_text.find("YEAR TO DATE")+len("YEAR TO DATE"):]
9
10 header_list = ["Crime type", table_dates[0] + "-" + table_dates[1],
11 table_dates[2] + "-" + table_dates[3],
12 "7-day percent change", table_dates[4] + "-" + table_dates[5],
13 table_dates[6] + "-" + table_dates[7], "28-day percent change",
14 table_dates[8] + "-" + table_dates[9], table_dates[10] + "-" + table_dates[
15 "YTD percent change"]
16
17 df.columns = header_list
```

Extracting tables - tips 📌

- Check your results, including whether the header is captured
- Don't forget about text extraction!

Summary

- `pypdf` :
 - best OCR interpreter for extracting text from older documents
 - best for extracting images
- `pymupdf` :
 - best for tables
- `pdfplumber` :
 - good package for beginners

Docs and resources

- [pyPDF documentation](#)
- [PyMuPDF documentation](#)
- [pdfplumber documentation](#)
- [Extracting Tables with PyMuPDF](#)

Thanks!

Please give us your feedback on this session at bit.ly/survey-data-bytes and join us at the next Data Bytes!

data bytes

"Bite"-sized Data Talks on Mondays

Learn more and register
at bit.ly/data-bytes



Making Census of the US Census

September 16th, 12 - 1 pm

Extracting Data from PDFs with Python

October 7th, 12 - 1 pm

Creating Software Management Plans for Open Science Initiatives

October 28th, 12 - 1 pm

The Hidden Mapping Powers of ArcGIS Arcade

November 18th, 12 - 1 pm

