COMP30810
Intro to Text Analytics

Dr. Binh Thanh Le

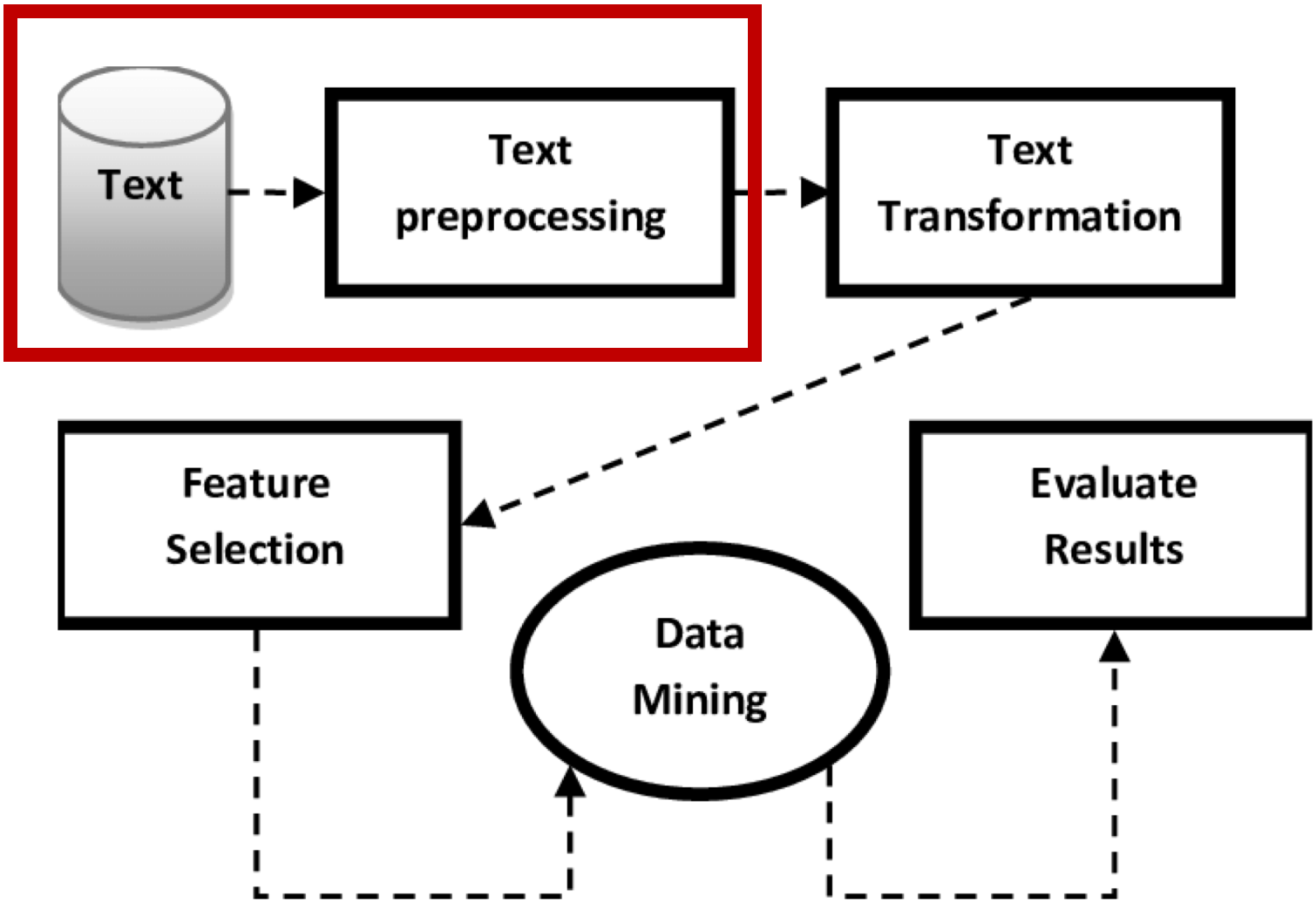
thanhbinh.le@ucd.ie

Insight Centre for Data Analytics

School of Computer Science

University College Dublin

Text Analytics Process



Today Goals:

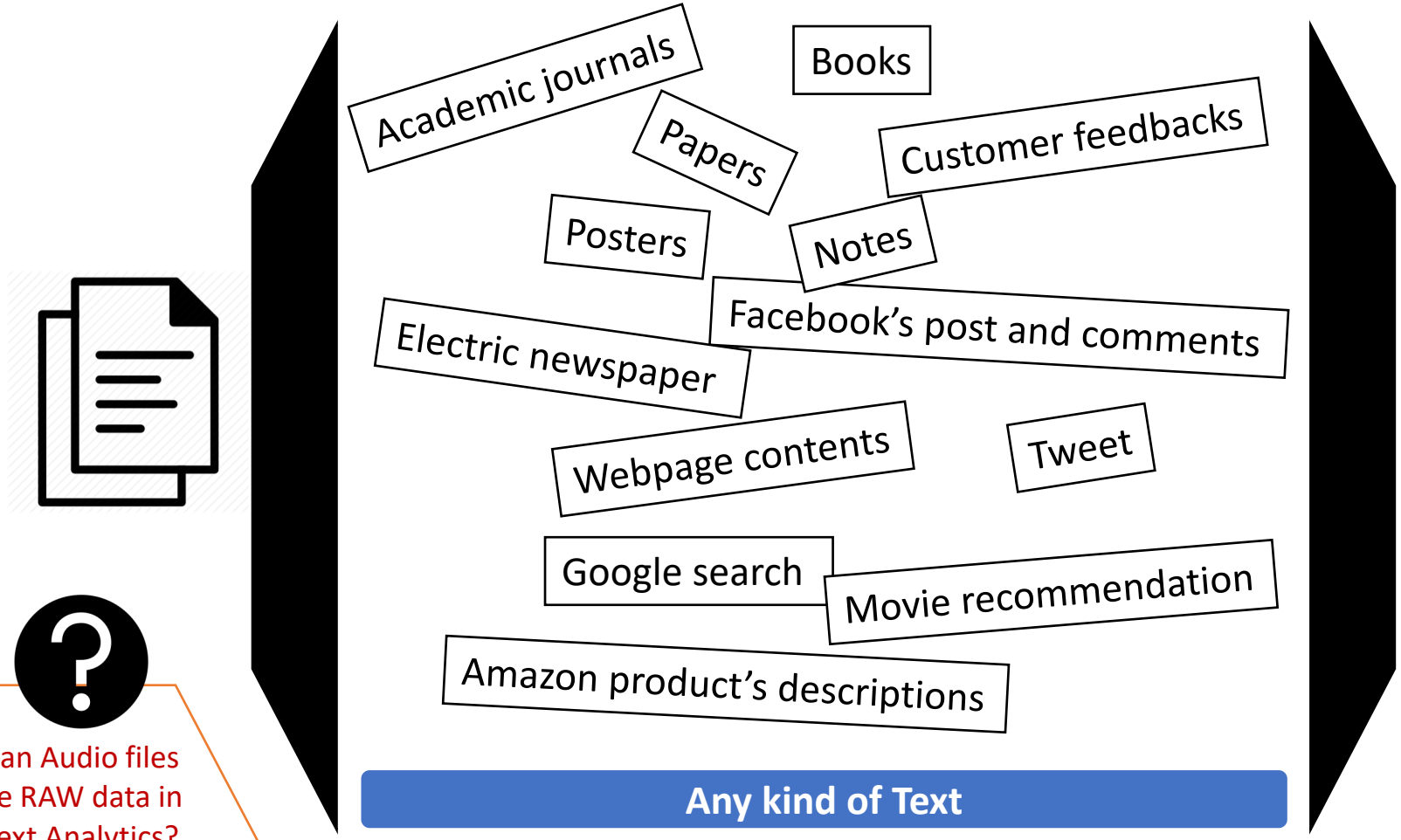
- From RAW Data → Data frame

Why DataFrame?

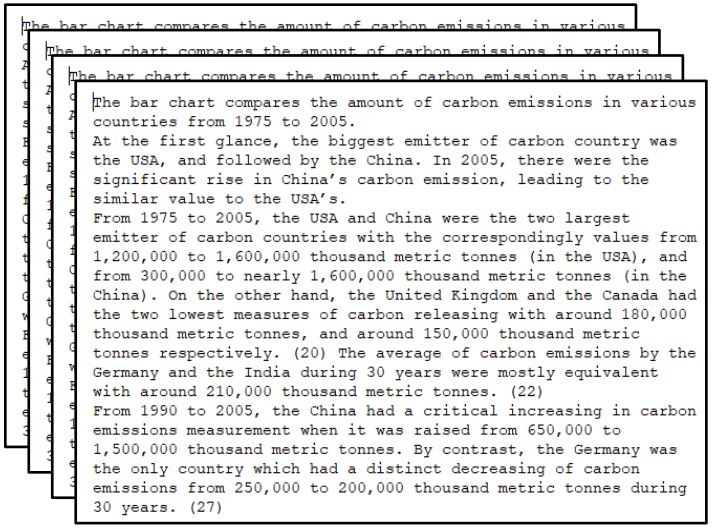
1) Semi-structured data → the features are predefined by user
2) Many support from Python social network
3) Easy to manage the rows and columns
4) Easy to read and write in Python

RAW data

- What are RAW data for Text Analytics?



So, what can we do?



CRISP-DM

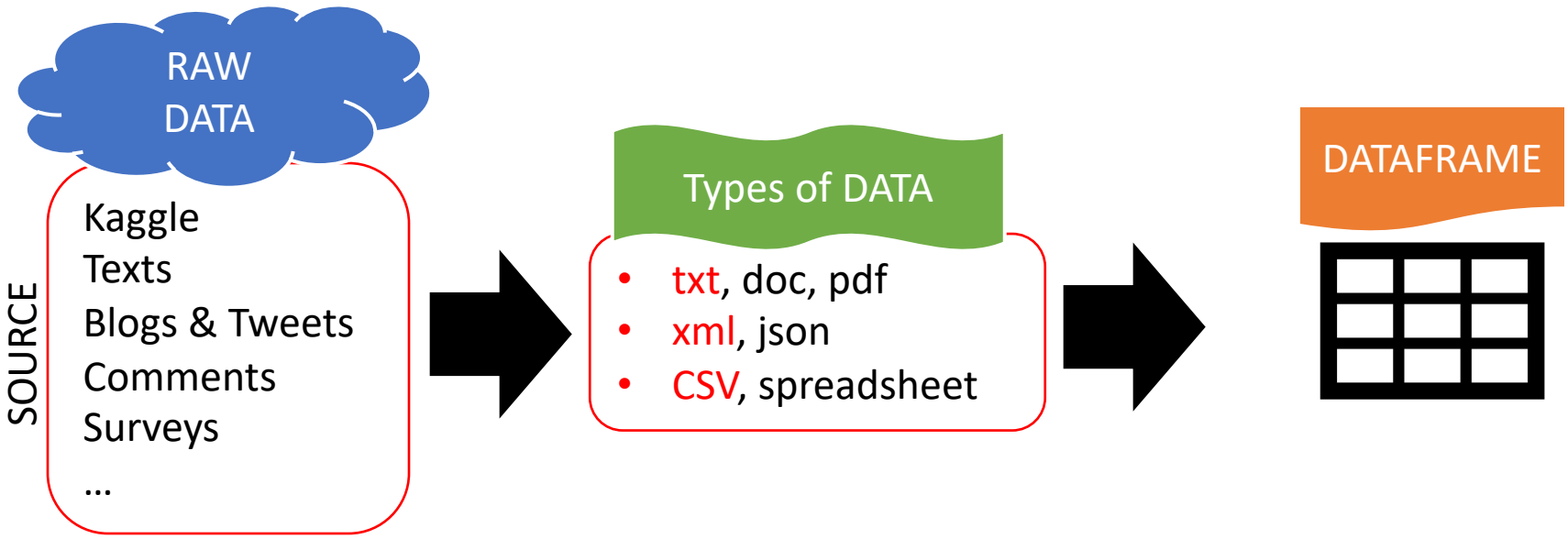
Business Understanding ☑

↓

Data Understanding ☐

- Classify documents to categories
- Group documents into clusters
- Find the most similar document to the input document
- …

Extracting Data from RAW sources



What are Pandas data frames?

DataFrame come with the python Pandas library, and they are defined as a two-dimensional labelled data structure with columns of potentially different types.

Installing

```
>pip install pandas
```

or

```
>conda install pandas
```

Using

```
In [ ]: 1 import pandas as pd
```

```
In [124]: data.head(5)
```

Out[124]:

Columns – features (data.columns)

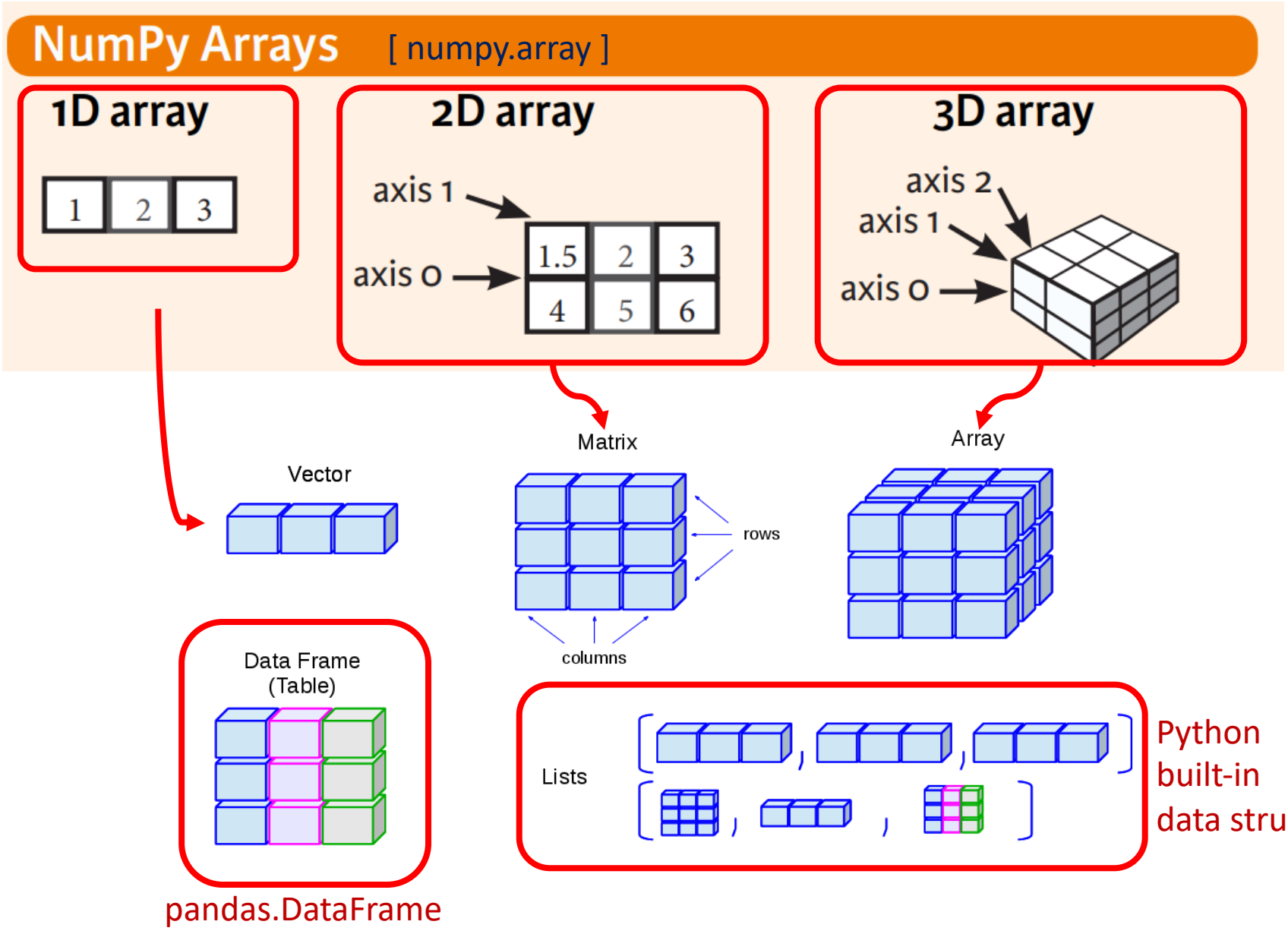
Header

Rows - samples

	first_name	last_name	company_name	address	city	county	postal	phone1	phone2	email	web
0	Aleshia	Tomkiewicz	Alan D Rosenburg Cpa Pc	14 Taylor St	St. Stephens Ward	Kent	CT2 7PP	01835-703597	01944-369967	atomkiewicz@hotmail.com	http://www
1	Evan	Zigomalas	Cap Gemini America	5 Binney St	Abbey Ward	Buckinghamshire	HP11 2AX	01937-864715	01714-737668	evan.zigomalas@gmail.com	http://www
2	France	Andrade	Elliott, John W Esq	8 Moor Place	East Southbourne and Tuckton W	Bournemouth	BH6 3BE	01347-368222	01935-821636	france.andrade@hotmail.com	http://www
3	Ulysses	Mcwalters	Mcmahan, Ben L	505 Exeter Rd	Hawerby cum Beesby	Lincolnshire	DN36 5RP	01912-771311	01302-601380	ulysses@hotmail.com	http://www
4	Tyisha	Veness	Champagne Room	5396 Forth Street	Greets Green and Lyng Ward	West Midlands	B70 9DT	01547-429341	01290-367248	tyisha.veness@hotmail.com	http://www

Index (data.index)

Let's see …


NumPy Arrays
[numpy.array]
1D array
1 2 3
2D array
axis 1
axis 0
1.5 2 3
4 5 6
3D array
axis 2
axis 1
axis 0
Vector
Matrix
rows
columns
Array
Data Frame
(Table)
Lists
Python
built-in
data structure
pandas.DataFrame

array vs DataFrame

- There are some questions like:

numpy.array or pandas.DataFrame

Matrix

- multiple columns and/or rows of data
- 1 type (numeric or text)

Data Frame

- multiple columns and/or rows of data
- multiple types

Easier to handle:
select, concatenate, merge, apply functions …

1) CSV RAW Source

- Read CSV

```
pandas.read_csv(filepath_or_buffer, sep=', ', delimiter=None, header='infer', names=None, index_col=None,
usecols=None, squeeze=False, prefix=None, mangle_dupe_cols=True, dtype=None, engine=None,
converters=None, true_values=None, false_values=None, skipinitialspace=False, skiprows=None, nrows=None,
na_values=None, keep_default_na=True, na_filter=True, verbose=False, skip_blank_lines=True,
parse_dates=False, infer_datetime_format=False, keep_date_col=False, date_parser=None, dayfirst=False,
iterator=False, chunksize=None, compression='infer', thousands=None, decimal=b'.', lineterminator=None,
quotechar='"', quoting=0, escapechar=None, comment=None, encoding=None, dialect=None, tupleize_cols=None,
error_bad_lines=True, warn_bad_lines=True, skipfooter=0, doublequote=True, delim_whitespace=False,
low_memory=True, memory_map=False, float_precision=None)                                    [source]
```

filepath_or_buffer : *path of file*

sep *: string for separation, default ‘,’*

header *: determine the header of a data frame*

index_col *: determine the index column of the data frame*

na_values *: determine the NaN values*

“example.csv”

	first_name	last_name	age	preTestScore	postTestScore
0	Jason	Miller	42	4	25,000
1	Molly	Jacobson	52	24	94,000
2	Tina	.	36	31	57
3	Jake	Milner	24	.	62
4	Amy	Cooze	73	.	70

In text

```
,first_name,last_name,age,preTestScore,postTestScore
0,Jason,Miller,42,4,"25,000"
1,Molly,Jacobson,52,24,"94,000"
2,Tina,.,36,31,57
3,Jake,Milner,24,.,62
4,Amy,Cooze,73,.,70
```

https://pandas.pydata.org/pandas-docs/stable/generated/pandas.read_csv.html

Load a csv no argument

```
In [5]:
1 df = pd.read_csv('example.csv')
2 df
```

executed in 11ms, finished 11:06:29 2018-07-25

Out[5]:

	Unnamed: 0	first_name	last_name	age	preTestScore	postTestScore
0	0	Jason	Miller	42	4	25,000
1	1	Molly	Jacobson	52	24	94,000
2	2	Tina	.	36	31	57
3	3	Jake	Milner	24	.	62
4	4	Amy	Cooze	73	.	70

Load a csv with no headers

```
In [6]:
1 df = pd.read_csv('example.csv', header=None)
2 df
```

executed in 12ms, finished 11:06:59 2018-07-25

Out[6]:

	0	1	2	3	4	5
0	NaN	first_name	last_name	age	preTestScore	postTestScore
1	0.0	Jason	Miller	42	4	25,000
2	1.0	Molly	Jacobson	52	24	94,000
3	2.0	Tina	.	36	31	57
4	3.0	Jake	Milner	24	.	62
5	4.0	Amy	Cooze	73	.	70

Load a csv with index

```
In [9]:
1 df = pd.read_csv('example.csv', index_col=0)
2 df
```

executed in 11ms, finished 11:13:41 2018-07-25

Out[9]:

	first_name	last_name	age	preTestScore	postTestScore
0	Jason	Miller	42	4	25,000
1	Molly	Jacobson	52	24	94,000
2	Tina	.	36	31	57
3	Jake	Milner	24	.	62
4	Amy	Cooze	73	.	70

https://chrisalbon.com/python/data_wrangling/pandas_dataframe_importing_csv/

Load a csv with index, and NAN values

In [10]:
```
1 df = pd.read_csv('example.csv', index_col=0, na_values='.')
2 df
```

executed in 12ms, finished 11:20:43 2018-07-25

Out[10]:

	first_name	last_name	age	preTestScore	postTestScore
0	Jason	Miller	42	4.0	25,000
1	Molly	Jacobson	52	24.0	94,000
2	Tina	NaN	36	31.0	57
3	Jake	Milner	24	NaN	62
4	Amy	Cooze	73	NaN	70

Example of Text Data in CSV

title	tags	categories	id	fulltitle
Inge Missmahl brings peace to the minds of Afghanistan	['Inge', 'Missmahl', 'TEDTalks', 'TED', 't	['Nonprofits & Activism']	Jc2F3-nawnI	Inge Missmahl brings peace to the minds of Afghanistan
Edith Widder: The weird and wonderful world of biolumin	['Edith', 'Widder', 'TEDTalks', 'TED', 'tal	['Science & Technology']	IDkSDPgrtjs	Edith Widder: The weird and wonderful world of bioluminescence
Jacqueline Novogratz on an escape from poverty	['Jacqueline', 'Novogratz', 'TEDTalks', '	['Nonprofits & Activism']	oD06XPtmLZY	Jacqueline Novogratz on an escape from poverty
Math class needs a makeover | Dan Meyer	['Dan', 'Meyer', 'TEDTalks', 'TED', 'talks	['Science & Technology']	NWUFjb8w9P	Math class needs a makeover | Dan Meyer
Craig Venter unveils "synthetic life"	['TEDTalks', 'TED', 'talks', 'Craig Venter	['Science & Technology']	QHIocNOHd7A	Craig Venter unveils "synthetic life"
Danielle de Niese: A flirtatious aria	['DanielleDeNiese', '2011G', '480p']	['Music']	Nx02V8GBKUs	Danielle de Niese: A flirtatious aria
Philip Howard: Four ways to fix a broken legal system	['Philip', 'Howard', 'TEDTalks', 'TED', 'ta	['Nonprofits & Activism']	IWdzrZdRa38	Philip Howard: Four ways to fix a broken legal system
Why you should define your fears instead of your goals | T	['TEDTalk', 'TEDTalks', 'Choice', 'Fear',	['People & Blogs']	5J6jAC6XxAI	Why you should define your fears instead of your goals | Tim Ferriss
Dennis vanEngelsdorp: Where have the bees gone?	['Dennis', 'vanEngelsdorp', 'TED', 'TED'	['Pets & Animals']	3GXlvP4kLHg	Dennis vanEngelsdorp: Where have the bees gone?
Close-up card magic with a twist | Lennart Green	['Lennart', 'Green', 'TED', 'TEDtalks', 'ta	['Entertainment']	1_oa8m5Oq0C	Close-up card magic with a twist | Lennart Green
Joshua Klein: The intelligence of crows	['Joshua', 'Klein', 'ted', 'Tedtalks', 'ani	['Science & Technology']	bXQAgzfwuNC	Joshua Klein: The intelligence of crows
The psychology of evil | Philip Zimbardo	['TED', 'TEDtalks', 'talks', 'Philip Zimba	['Education']	OsFEV35tWsg	The psychology of evil | Philip Zimbardo
How to make hard choices | Ruth Chang	['TEDTalk', 'TEDTalks', 'TED Talk', 'TED '	['Howto & Style']	8GQZuzIdeQC	How to make hard choices | Ruth Chang
Dan Barasch: A park underneath the hustle and bustle of N	['TEDTalk', 'TEDTalks', 'TED Talk', 'TED '	['Science & Technology']	nLecuOcB3dA	Dan Barasch: A park underneath the hustle and bustle of New York City
Jill Tarter: Why the search for alien intelligence matters	['Jill', 'Tarter', 'TEDTalks', 'TED', 'talks',	['Science & Technology']	EszGIvRdgTE	Jill Tarter: Why the search for alien intelligence matters

https://chrisalbon.com/python/data_wrangling/pandas_dataframe_importing_csv/

2) XML RAW Source

Sample of XML

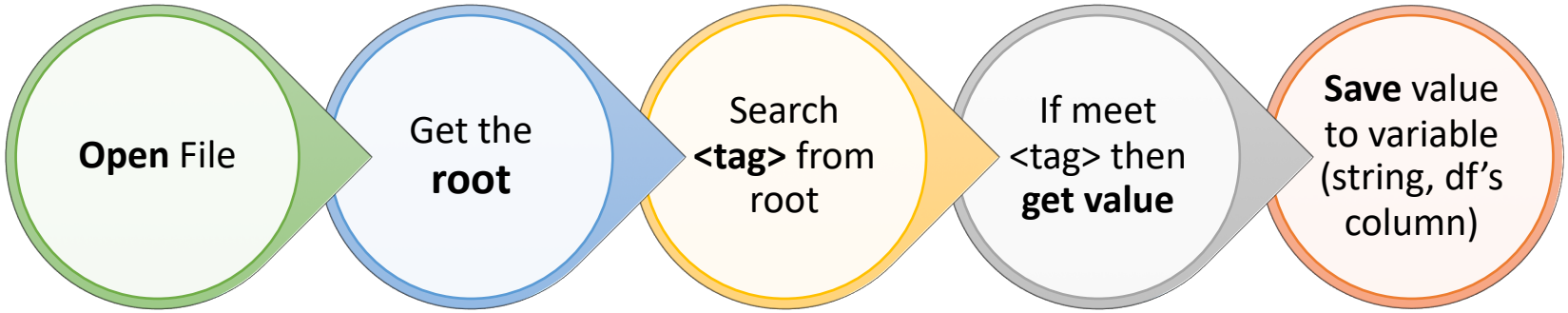
```xml
<?xml version="1.0"?>
<data>
    <country name="Liechtenstein">
        <rank>1</rank>
        <year>2008</year>
        <gdppc>141100</gdppc>
        <neighbor name="Austria" direction="E"/>
        <neighbor name="Switzerland" direction="W"/>
    </country>
    <country name="Singapore">
        <rank>4</rank>
        <year>2011</year>
        <gdppc>59900</gdppc>
        <neighbor name="Malaysia" direction="N"/>
    </country>
    <country name="Panama">
        <rank>68</rank>
        <year>2011</year>
        <gdppc>13600</gdppc>
        <neighbor name="Costa Rica" direction="W"/>
        <neighbor name="Colombia" direction="E"/>
    </country>
</data>
```

```xml
<xml_root>

    <object>
        <id>1</id>
        <name>First</name>
    </object>

    <object>
        <id>2</id>
        <name>Second</name>
    </object>

    <object>
        <id>3</id>
        <name>Third</name>
    </object>

    <object>
        <id>4</id>
        <name>Fourth</name>
    </object>

</xml_root>
```

Main point: parsing the content under the specific <tagname>

Tools in python: **lxml** , **etree** , **BeautifulSoup**

2) XML RAW Source




```
1  import xml.etree.ElementTree as ET
2  tree = ET.parse('../DATA/country_data.xml')
3  root = tree.getroot()
```

executed in 7ms, finished 12:29:58 2018-08-06

```
1  root.tag
```

executed in 5ms, finished

```
'data'
```

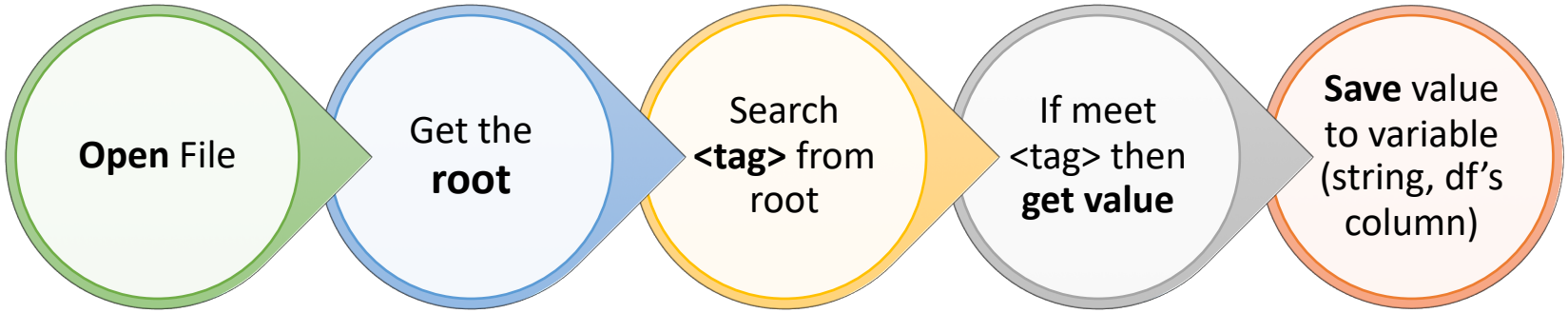
```
1  for country in root.findall('country'):
2      rank = country.find('rank').text
3      name = country.get('name')
4      print(name, rank)
```

executed in 12ms, finished 12:46:49 2018-08-06

```
Liechtenstein 1
Singapore 4
Panama 68
```

```
<?xml version="1.0"?>
<data>
    <country name="Liechtenstein">
        <rank>1</rank>
        <year>2008</year>
        <gdppc>141100</gdppc>
        <neighbor name="Austria" direction="E"/>
        <neighbor name="Switzerland" direction="W"/>
    </country>
    <country name="Singapore">
        <rank>4</rank>
        <year>2011</year>
        <gdppc>59900</gdppc>
        <neighbor name="Malaysia" direction="N"/>
    </country>
    <country name="Panama">
        <rank>68</rank>
        <year>2011</year>
        <gdppc>13600</gdppc>
        <neighbor name="Costa Rica" direction="W"/>
        <neighbor name="Colombia" direction="E"/>
    </country>
</data>
```

XML RAW Source



Note:

1) Should overview the structure of XML before parsing

2) Save different variables for different <tags> → increase memory, but safer for parsing

3) Should handle the type of values before saving

4) Be careful with NAN values → should be replaced by some default value

3) TXT RAW Source

We will think that inside one text file, we need:

1. Determine the separation, e.g. comma, enter, or page break … ?
2. Determine the end of paragraph, file ?
3. Ignored the blank sentences ?
4. Ignored the special characters?

DO NOT THINK ABOUT THEM SO EARLY

This is the most difficulty parsing!

1. One sentence is one column ??? → No, it generates a large feature space
2. Title of text file has its own meaning, also the path → Yes, have to check those before parsing.

3) TXT RAW Source

Read whole TXT from python:

```
In [31]:
1 file1 = open("adele.txt","r")
2 print(file1.read())

executed in 7ms, finished 15:20:06 2018-07-25

Looking for some education
Made my way into the night
```

Read line-to-line TXT from python:

```
In [32]:
1 lines = [line.rstrip('\n') for line in open('adele.txt')]
2 lines

executed in 16ms, finished 15:22:27 2018-07-25

Out[32]: ['Looking for some education',
          'Made my way into the night',
```

File Handle

1.Read Only ('r') : Open text file for reading. The handle is positioned at the beginning of the file. If the file does not exists, raises I/O error. This is also the default mode in which file is opened.

2.Read and Write ('r+') : Open the file for reading and writing. The handle is positioned at the beginning of the file. Raises I/O error if the file does not exists.

3.Write Only ('w') : Open the file for writing. For existing file, the data is truncated and over-written. The handle is positioned at the beginning of the file. Creates the file if the file does not exists.

4.Write and Read ('w+') : Open the file for reading and writing. For existing file, data is truncated and over-written. The handle is positioned at the beginning of the file.

5.Append Only ('a') : Open the file for writing. The file is created if it does not exist. The handle is positioned at the end of the file. The data being written will be inserted at the end, after the existing data.

6.Append and Read ('a+') : Open the file for reading and writing. The file is created if it does not exist. The handle is positioned at the end of the file. The data being written will be inserted at the end, after the existing data.

https://docs.python.org/2/tutorial/inputoutput.html

Example:

```
1  import os
2  import pandas as pd
3
4  path = '../Comp30810/DATA/Kaggle/poetry'
5  df = pd.DataFrame([],columns=['filename','content','path'])
6
7  for root, directories, files in os.walk(path):
8      for filename in files:
9          _name = os.path.splitext(filename)[0]
10
11         filepath = os.path.join(root, filename)
12         f = open(filepath,"r",encoding="utf8")
13         content = f.read()
14         pieces = {'filename': _name, 'content': content, 'path': filepath}
15         df = df.append(pieces,ignore_index=True)
16 df.head(5)
```

executed in 112ms, finished 14:43:36 2018-08-06

	filename	content	path
0	adele	Looking for some education\nMade my way into t...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\adele.txt
1	al-green	Let's stay together I, I'm I'm so in love with...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\al-gre...
2	alicia-keys	Ooh..... New York x2 Grew up in a town that ...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\alicia...
3	amy-winehouse	Build your dreams to the stars above\nBut when...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\amy-wi...
4	beatles	Yesterday, all my troubles seemed so far away\...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\beatle...

We will manage the content in next lecture!

Write to CSV file

```
1  import pandas as pd
2
3  # Write dataframe to CSV
4  df.to_csv('myfile.csv', sep=',',index=0)
5
6  # Read again CSV to check
7  df_check = pd.read_csv('myfile.csv',sep=',')
8  df_check.head(5)
```

executed in 198ms, finished 14:46:08 2018-08-06

	filename	content	path
0	adele	Looking for some education\r\nMade my way into...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\adele.txt
1	al-green	Let's stay together I, I'm I'm so in love with...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\al-gre...
2	alicia-keys	Ooh..... New York x2 Grew up in a town that ...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\alicia...
3	amy-winehouse	Build your dreams to the stars above\r\nBut wh...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\amy-wi...
4	beatles	Yesterday, all my troubles seemed so far away\...	D:/git/TA_/Comp30810/DATA/Kaggle/poetry\beatle...

Summary

- What RAW text data is.
- How to extract data from RAW source:
  - CSV
  - XML
  - Text file (.txt)
- How to save raw data as CSV dataframe.