

bookshelf_populator

September 15, 2022

```
[ ]: import numpy as np
import wikipedia as wk
import pandas as pd
import matplotlib.pyplot as plt

bkshelf = '/Users/matthewbain/Documents/Notebook/old/bookshelf/up next/film_
↳next.xlsx'

df = pd.read_excel(bkshelf)

for m in range(len(df)):
    film = df["film"][m]
    print(film)

x = wk.WikipediaPage(title = "When Marnie Was There")

content = x.content
y = content.find('L')

content[y:200000]
content.index('hi')
content[content.index('hi'):content.index('hi')+11]

# help(wk.WikipediaPage)

plt.bar(content, 3)
plt.show()

from sklearn import base
```

When Marnie was Here
My Neighbours the Yamadas
The Usual Suspects
Coco
Whisper of the Heart
Weathering With You
Sorry to Bother You
The King

Zodiac
Godfather
The Irishman
Scarface
Children of Men
Intouchables
Dear Ex
A Sun
Blackpink: Light up the Sky
Honey boy
Green Room
Enemy
Moon
Baraka
50/50
It Follows
Proxima
Brokeback Mountain
Little Women
The Master
Sin City
Amores perros
3 idiots
The Invisible Man
2046
Sami Blood
Boy
Risky Business
Taxi Driver
300
The Age of Shadows
Enter the Void
Hard Eight
Gangs of New York
Full Metal Jacket
Once Upon a Time in Hollywood
Grave of the Fireflies
The Tale of Princess Kaguya
Fargo
Is the Man Who Is Tall Happy?
Call me by your Name
Fear and Loathing in Las Vegas
The Favourite
Anomalisa
Moonlight
Phantom Thread
Inherent Vice
Raging Bull

Eyes Wide Shut
Brooklyn
Dear Zachary: A Letter to a Son About His Father
Schindler's List
Roma
Shaun of the Dead
In the Mood for Love
The Lost Boys
Floating Weeds
13 Assassins
Like Father, Like Son
Memories of Murder
Hero
The Grandmaster
Blade Runner
Zombieland
Gravity
The Girl With the Dragon Tattoo
Gone Girl
Prisoners
There Will Be Blood
The Post
Gran Torino
Apocalypse Now
Saving Private Ryan
Beasts of no nation
Hustlers
Molly's Game
The French Connection
Elephant
Free Solo
Once Upon a Time in America
Goodfellas
The King of Comedy
Rounders
Lock, Stock, and Two Smoking Barrels
I Lost my Body
Ocean Waves
Paranoia Agent
Millenium Actress
Wolf Children
Neo Tokyo
Steamboat
Pom Poko
Memories
Liz and the Blue Bird
Only Yesterday
Mirai

Children Who Chase Lost Voices
Metropolis
Jojo Rabbit
Booksmart
Sideways
Life is Beautiful
The Station Agent
Get Real
One Flew Over the Cuckoo's Nest
The Squid and the Whale
Dr Strangelove
Searching for Sugarman
Samsara
Encounters at the End of the World
My Best Fiend
Koyaanisqatsi
Gone with the Wind
Signs
Mala Noche
To Die For
Polytechnique
Raising Arizona
Hail, Ceasar!
Capote
To die for
Selma
Boy
Incendies
The Last Emperor
The Age of Innocence
The game
Empire of the Sun
Climax
Rise the Red Lantern
The Graduate
Dazed and Confused
Short Term 12
Frank
The Color of Pomegranates
Amelie
Shaolin soccer
Masculin Feminin
Pierrot le fou
Fallen Angels
Ashes of Time
10 Cloverfield Lane
WWZ
The Shining

Villains
Carrie
The Evil Dead
Halloween
Mother
The Witch
The Blair Witch Project
Inland Empire
A Tale of Two Sisters
28 Days Later
After the Storm
Dreams
The Handmaiden
Miracle in Cell No. 7
On the beach at night alone
Right now, wrong then
Tale of cinema
In another country
Burning
Lady Vengeance
Thirst
Tokyo!
The Man from Nowhere
Shadow
Hara-kiri: Death of a Samurai
Seven Samurai
Throne of Blood
Ran
Casablanca
Solaris
Close Encounters of the Third Kind
Sunshine
War of the Worlds
The Road
Ad Astra
Space Odyssey
Pi
Split
The Visit
Mean Streets
The Conversation
A Girl Walks Home Alone at Night
Let Me In
Nosferatu
Nosferatu the Vampyre
Bram Stoker's Dracula
Thin Red Line
Merry Christmas Mr Lawrence

Bill Burr next

```
-----
JSONDecodeError                                Traceback (most recent call last)
File ~/mambaforge/lib/python3.9/site-packages/requests/models.py:971, in
↳ Response.json(self, **kwargs)
    970 try:
--> 971     return complexjson.loads(self.text, **kwargs)
    972 except JSONDecodeError as e:
    973     # Catch JSON-related errors and raise as requests.JSONDecodeError
    974     # This aliases json.JSONDecodeError and simplejson.JSONDecodeError

File ~/mambaforge/lib/python3.9/json/__init__.py:346, in loads(s, cls,
↳ object_hook, parse_float, parse_int, parse_constant, object_pairs_hook, **kw)
    343 if (cls is None and object_hook is None and
    344         parse_int is None and parse_float is None and
    345         parse_constant is None and object_pairs_hook is None and not kw :
--> 346     return _default_decoder.decode(s)
    347 if cls is None:

File ~/mambaforge/lib/python3.9/json/decoder.py:337, in JSONDecoder.decode(self
↳ s, _w)
    333 """Return the Python representation of ``s`` (a ``str`` instance
    334 containing a JSON document).
    335
    336 """
--> 337 obj, end = self.raw_decode(s, idx=_w(s, 0).end())
    338 end = _w(s, end).end()

File ~/mambaforge/lib/python3.9/json/decoder.py:355, in JSONDecoder.
↳ raw_decode(self, s, idx)
    354 except StopIteration as err:
--> 355     raise JSONDecodeError("Expecting value", s, err.value) from None
    356 return obj, end
```

JSONDecodeError: Expecting value: line 1 column 1 (char 0)

During handling of the above exception, another exception occurred:

```
JSONDecodeError                                Traceback (most recent call last)
/Users/matthewbain/Documents/Programming/projects/bookshelf-populator/code/
↳ bookshelf_populator.ipynb Cell 1 in <cell line: 14>()
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳ projects/bookshelf-populator/code/bookshelf_populator.ipynb#W0sZmlsZQ%3D%3D?
↳ line=10'>11</a>         film = df["film"][m]
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳ projects/bookshelf-populator/code/bookshelf_populator.ipynb#W0sZmlsZQ%3D%3D?
↳ line=11'>12</a>         print(film)
```

```

--> <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W0sZmlsZQ%3D%3D?
↳line=13'>14</a> x = wk.WikipediaPage(title = "When Marnie Was There")
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W0sZmlsZQ%3D%3D?
↳line=15'>16</a> content = x.content
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W0sZmlsZQ%3D%3D?
↳line=16'>17</a> y = content.find('L')

```

```

File ~/mambaforge/lib/python3.9/site-packages/wikipedia/wikipedia.py:299, in
↳WikipediaPage.__init__(self, title, pageid, redirect, preload, original_title
296 else:
297     raise ValueError("Either a title or a pageid must be specified")
--> 299 self.__load(redirect=redirect, preload=preload)
301 if preload:
302     for prop in ('content', 'summary', 'images', 'references', 'links',
↳'sections'):

```

```

File ~/mambaforge/lib/python3.9/site-packages/wikipedia/wikipedia.py:336, in
↳WikipediaPage.__load(self, redirect, preload)
333 else:
334     query_params['pageids'] = self.pageid
--> 336 request = _wiki_request(query_params)
338 query = request['query']
339 pageid = list(query['pages'].keys())[0]

```

```

File ~/mambaforge/lib/python3.9/site-packages/wikipedia/wikipedia.py:742, in
↳_wiki_request(params)
739 if RATE_LIMIT:
740     RATE_LIMIT_LAST_CALL = datetime.now()
--> 742 return r.json()

```

```

File ~/mambaforge/lib/python3.9/site-packages/requests/models.py:975, in
↳Response.json(self, **kwargs)
971     return complexjson.loads(self.text, **kwargs)
972 except JSONDecodeError as e:
973     # Catch JSON-related errors and raise as requests.JSONDecodeError
974     # This aliases json.JSONDecodeError and simplejson.JSONDecodeError
--> 975     raise RequestsJSONDecodeError(e.msg, e.doc, e.pos)

```

JSONDecodeError: Expecting value: line 1 column 1 (char 0)

0.0.1 Hello

\$\$ x

_{ } ^ { }

\$\$

[]:

zsh:fg:1: no job control in this shell.

hello

```
[ ]: model = torchvision.models.detection.maskrcnn_resnet50_fpn(pretrained=False,
↳progress=False, pretrained_backbone=False)
```

```
[ ]: # %% [hello there \int(4, 5)]
'''
Multinomial Naive Bayes
'''
# %%
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import f1_score, accuracy_score, recall_score,
↳confusion_matrix
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
import pandas as pd
# %%
dataset = pd.read_csv('')
# %%
X = dataset.iloc[:, :-1].values
y = dataset.iloc[:, -1].values
# %%
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.2, random_state=997)
# %%
ss = StandardScaler()
scaled_x_train = ss.fit_transform(X_train)
scaled_x_test = ss.transform(X_test)
# %%
classifier = MultinomialNB().fit(scaled_x_train, y_train)
# %%
y_pred = classifier.predict(scaled_x_test)
# %%
confusion_matrix(y_test, y_pred)
# %%
print(f'F1 Score: {f1_score(y_test, y_pred)}')
print(f'Accuracy Score: {accuracy_score(y_test, y_pred)}')
print(f'Recall Score: {recall_score(y_test, y)}')
```

```
-----
FileNotFoundError                                Traceback (most recent call last)
/Users/matthewbain/Documents/Programming/projects/bookshelf-populator/code/
↳bookshelf_populator.ipynb Cell 3 in <cell line: 12>()
```



```

    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W2sZmlsZQ%3D%3D?
↳line=9'>10</a> import pandas as pd
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W2sZmlsZQ%3D%3D?
↳line=10'>11</a> # %%
---> <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W2sZmlsZQ%3D%3D?
↳line=11'>12</a> dataset = pd.read_csv('')
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W2sZmlsZQ%3D%3D?
↳line=12'>13</a> # %%
    <a href='vscode-notebook-cell:/Users/matthewbain/Documents/Programming/
↳projects/bookshelf-populator/code/bookshelf_populator.ipynb#W2sZmlsZQ%3D%3D?
↳line=13'>14</a> X = dataset.iloc[:, :-1].values

File ~/mambaforge/lib/python3.9/site-packages/pandas/util/_decorators.py:311, in
↳deprecate_nonkeyword_arguments.<locals>.decorate.<locals>.wrapper(*args,
↳**kwargs)
    305 if len(args) > num_allow_args:
    306     warnings.warn(
    307         msg.format(arguments=arguments),
    308         FutureWarning,
    309         stacklevel=stacklevel,
    310     )
--> 311 return func(*args, **kwargs)

File ~/mambaforge/lib/python3.9/site-packages/pandas/io/parsers/readers.py:680,
↳in read_csv(filepath_or_buffer, sep, delimiter, header, names, index_col,
↳usecols, squeeze, prefix, mangle_dupe_cols, dtype, engine, converters,
↳true_values, false_values, skipinitialspace, skiprows, skipfooter, nrows,
↳na_values, keep_default_na, na_filter, verbose, skip_blank_lines, parse_dates,
↳infer_datetime_format, keep_date_col, date_parser, dayfirst, cache_dates,
↳iterator, chunksize, compression, thousands, decimal, lineterminator,
↳quotechar, quoting, doublequote, escapechar, comment, encoding,
↳encoding_errors, dialect, error_bad_lines, warn_bad_lines, on_bad_lines,
↳delim_whitespace, low_memory, memory_map, float_precision, storage_options)
    665 kwds_defaults = _refine_defaults_read(
    666     dialect,
    667     delimiter,
    (...)
    676     defaults={"delimiter": ",",
    677 )
    678 kwds.update(kwds_defaults)
--> 680 return _read(filepath_or_buffer, kwds)

File ~/mambaforge/lib/python3.9/site-packages/pandas/io/parsers/readers.py:575,
↳in _read(filepath_or_buffer, kwds)
    572 _validate_names(kwds.get("names", None))
    574 # Create the parser.
--> 575 parser = TextFileReader(filepath_or_buffer, **kwds)
    577 if chunksize or iterator:

```

```

578         return parser

File ~/mambaforge/lib/python3.9/site-packages/pandas/io/parsers/readers.py:934,
  in TextFileReader.__init__(self, f, engine, **kwargs)
    931     self.options["has_index_names"] = kwargs["has_index_names"]
    933     self.handles: IOHandles | None = None
--> 934     self._engine = self._make_engine(f, self.engine)

File ~/mambaforge/lib/python3.9/site-packages/pandas/io/parsers/readers.py:1218
  in TextFileReader._make_engine(self, f, engine)
    1214         mode = "rb"
    1215     # error: No overload variant of "get_handle" matches argument types
    1216     # "Union[str, PathLike[str], ReadCsvBuffer[bytes], ReadCsvBuffer[str]]"
    1217     # , "str", "bool", "Any", "Any", "Any", "Any", "Any"
-> 1218     self.handles = get_handle( # type: ignore[call-overload]
    1219         f,
    1220         mode,
    1221         encoding=self.options.get("encoding", None),
    1222         compression=self.options.get("compression", None),
    1223         memory_map=self.options.get("memory_map", False),
    1224         is_text=is_text,
    1225         errors=self.options.get("encoding_errors", "strict"),
    1226         storage_options=self.options.get("storage_options", None),
    1227     )
    1228     assert self.handles is not None
    1229     f = self.handles.handle

File ~/mambaforge/lib/python3.9/site-packages/pandas/io/common.py:786, in
  get_handle(path_or_buf, mode, encoding, compression, memory_map, is_text,
  errors, storage_options)
    781     elif isinstance(handle, str):
    782         # Check whether the filename is to be opened in binary mode.
    783         # Binary mode does not support 'encoding' and 'newline'.
    784         if ioargs.encoding and "b" not in ioargs.mode:
    785             # Encoding
--> 786             handle = open(
    787                 handle,
    788                 ioargs.mode,
    789                 encoding=ioargs.encoding,
    790                 errors=errors,
    791                 newline="",
    792             )
    793     else:
    794         # Binary mode
    795         handle = open(handle, ioargs.mode)

FileNotFoundError: [Errno 2] No such file or directory: ''

```

[]: