Official city desk crawler

Mark Sobolev

28. July 2019

Project description

The aim of the project is to create a crawler, which downloads documents published on official boards of individual cities of the Czech Republic.

Proposed solution

In the first step, the program obtains links to official city pages using the list of Czech cities https://cs.wikipedia.org/wiki/Seznam_m%C4%9Bst_v_%C4%8Cesku. Subsequently, the links to the official boards of individual cities are extracted. In the last step, the program downloads files from individual official boards.

Implementation

The project is programmed in Python3 using the Scrapy framework, which simplifies the documents extraction.

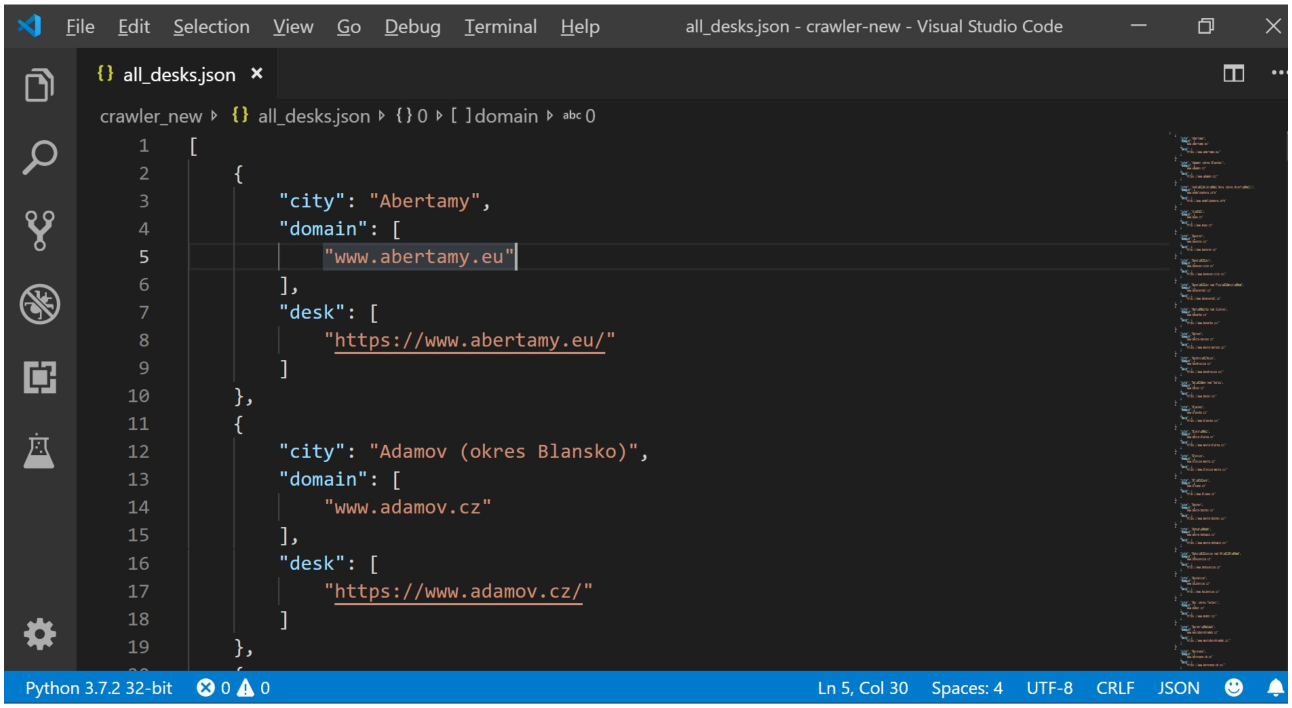
2 scripts are used to download documents:
- *download_desks.py* extracts the municipal websites URLs. Extracted URLs are stored in JSON format. For each web page the file contains the name of the city, its website URL and domain.
- *download_documents.py* downloads documents from individual official websites using extracted URLs. Documents are divided into folders - one for each city.

Downloaded documents should be prepared for categorization - it is necessary to eliminate duplicates and assign filenames corresponding to the content of the documents.

2 scripts are used to prepare documents:

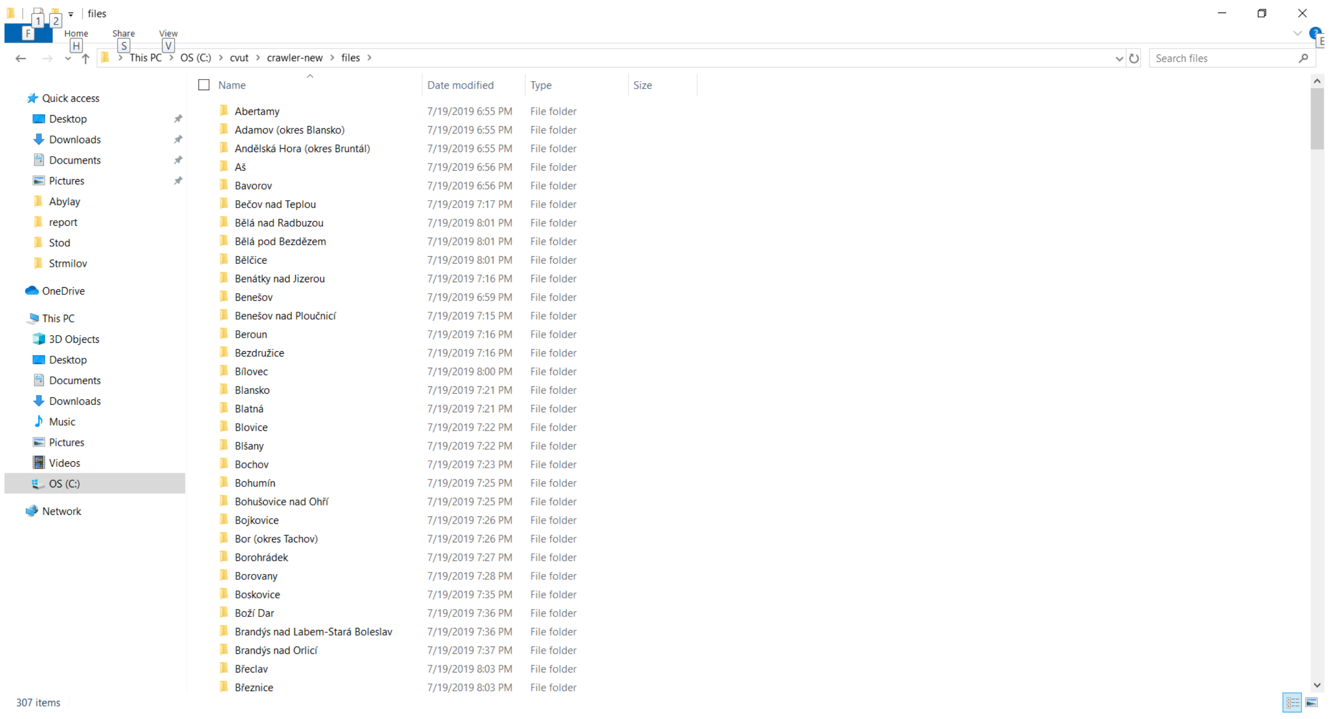
- *remove_duplicates.sh* removes documents which appear in folder more than once (Website may have several links to the same document, so the same document may be downloaded more than once)

- *extract_names.py* extracts the text from the first line of the PDF file and uses it as the new name for the file.



```
[
    {
        "city": "Abertamy",
        "domain": [
            "www.abertamy.eu"
        ],
        "desk": [
            "https://www.abertamy.eu/"
        ]
    },
    {
        "city": "Adamov (okres Blansko)",
        "domain": [
            "www.adamov.cz"
        ],
        "desk": [
            "https://www.adamov.cz/"
        ]
    },
```

Fig 1. List of official municipal webpages in JSON format



Name	Date modified	Type	Size
Abertamy	7/19/2019 6:55 PM	File folder	
Adamov (okres Blansko)	7/19/2019 6:55 PM	File folder	
Andělská Hora (okres Bruntál)	7/19/2019 6:55 PM	File folder	
Aš	7/19/2019 6:56 PM	File folder	
Bavorov	7/19/2019 6:56 PM	File folder	
Bečov nad Teplou	7/19/2019 7:17 PM	File folder	
Bělá nad Radbuzou	7/19/2019 8:01 PM	File folder	
Bělá pod Bezdězem	7/19/2019 8:01 PM	File folder	
Bělčice	7/19/2019 8:01 PM	File folder	
Benátky nad Jizerou	7/19/2019 7:16 PM	File folder	
Benešov	7/19/2019 6:59 PM	File folder	
Benešov nad Ploučnicí	7/19/2019 7:15 PM	File folder	
Beroun	7/19/2019 7:16 PM	File folder	
Bezdružice	7/19/2019 7:16 PM	File folder	
Bílovec	7/19/2019 8:00 PM	File folder	
Blansko	7/19/2019 7:21 PM	File folder	
Blatná	7/19/2019 7:21 PM	File folder	
Blovice	7/19/2019 7:22 PM	File folder	
Blšany	7/19/2019 7:22 PM	File folder	
Bochov	7/19/2019 7:23 PM	File folder	
Bohumín	7/19/2019 7:25 PM	File folder	
Bohušovice nad Ohří	7/19/2019 7:25 PM	File folder	
Bojkovice	7/19/2019 7:26 PM	File folder	
Bor (okres Tachov)	7/19/2019 7:26 PM	File folder	
Borohrádek	7/19/2019 7:27 PM	File folder	
Borovany	7/19/2019 7:28 PM	File folder	
Boskovice	7/19/2019 7:35 PM	File folder	
Boží Dar	7/19/2019 7:36 PM	File folder	
Brandýs nad Labem-Stará Boleslav	7/19/2019 7:36 PM	File folder	
Brandýs nad Orlicí	7/19/2019 7:37 PM	File folder	
Břeclav	7/19/2019 8:03 PM	File folder	
Březnice	7/19/2019 8:03 PM	File folder	

Fig 2. Folders with downloaded documents

How to use scripts

To work with scripts for downloading documents, one needs to install the Scrapy tool, which can be downloaded using the *pip install scrapy* command.

First run *python download_desks.py* to extract the municipal websites URLs. Then run *python download_documents.py* with appropriate arguments to download documents from individual official websites using extracted URLs.

Examples of valid argument combinations:

python download_documents.py --file all_desks.json --cities adamov bavorov downloads documents for Adamov and Bavorov

python download_documents.py --file all_desks.json --begin adamov --end bavorov downloads documents for cities starting with Adamov and ending with Bavorov. Cities in the list should be sorted lexicographically.

python download_documents.py --file all_desks.json --begin adamov --count 5

downloads documents for 5 cities starting with Adamov.

python download_documents.py --file all_desks.json --begin adamov downloads documents for all cities in the list starting with Adamov.

python download_documents.py --file all_desks.json downloads documents for all cities in the list.

Arguments for *download_documents.py*			
Argument	Description	Example	Note
--file	Name of the file with the URLs which is used for downloading documents.	*--file all_desks.json*	Required. Should be the first argument.
--cities	Names of the cities to download documents for.	*--cities adamov bavorov*	Optional. Can not be combined with *--begin*. Case insensitive.

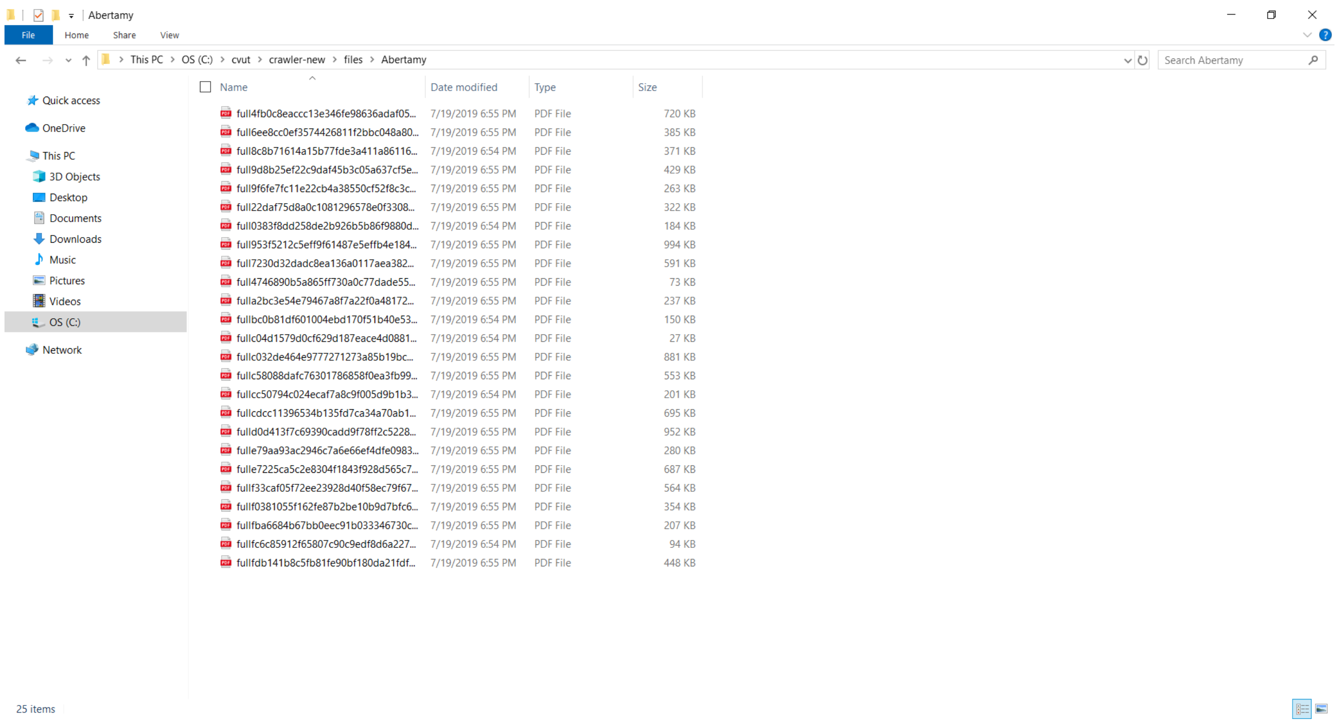
--begin	Name of the city to start with.	*--begin adamov*	Optional. Can not be combined with *--cities*. Case insensitive. If *--count and --end* are not used, documents will be downloaded for all cities in the list
--count	Number of cities to download documents for.	*--count 5*	Optional. Must be preceded by *--begin*. Case insensitive.
--end	Name of the last city to download documents for.e	*--end bavorov*	Optional. Must be preceded by *--begin*. Case insensitive.

Then you can run *remove_duplicates.sh* and *extract_names.py* to prepare documents for categorization. Both scripts should be placed in one folder with folders containing downloaded documents. *remove_duplicates.sh* removes duplicates from all folders in the same directory and *extract_names.py* extracts names from all folders in the same directory.

extract_names.py requires *Pillow* (Python Imaging Library), *pdf2* (wrapper around the pdftoppm and pdftocairo command line tools to convert PDF to a PIL Image list) and *pytesseract* (python wrapper for Google's Tesseract-OCR).

Results

During testing, the task was to download documents for 307 cities. As a result, documents for 291 cities were downloaded (95% success rate). Downloaded documents were than renamed using *extract_names.py*. As seen on Figure 4 not all the names extracted correspond to the actual titles of the documents.

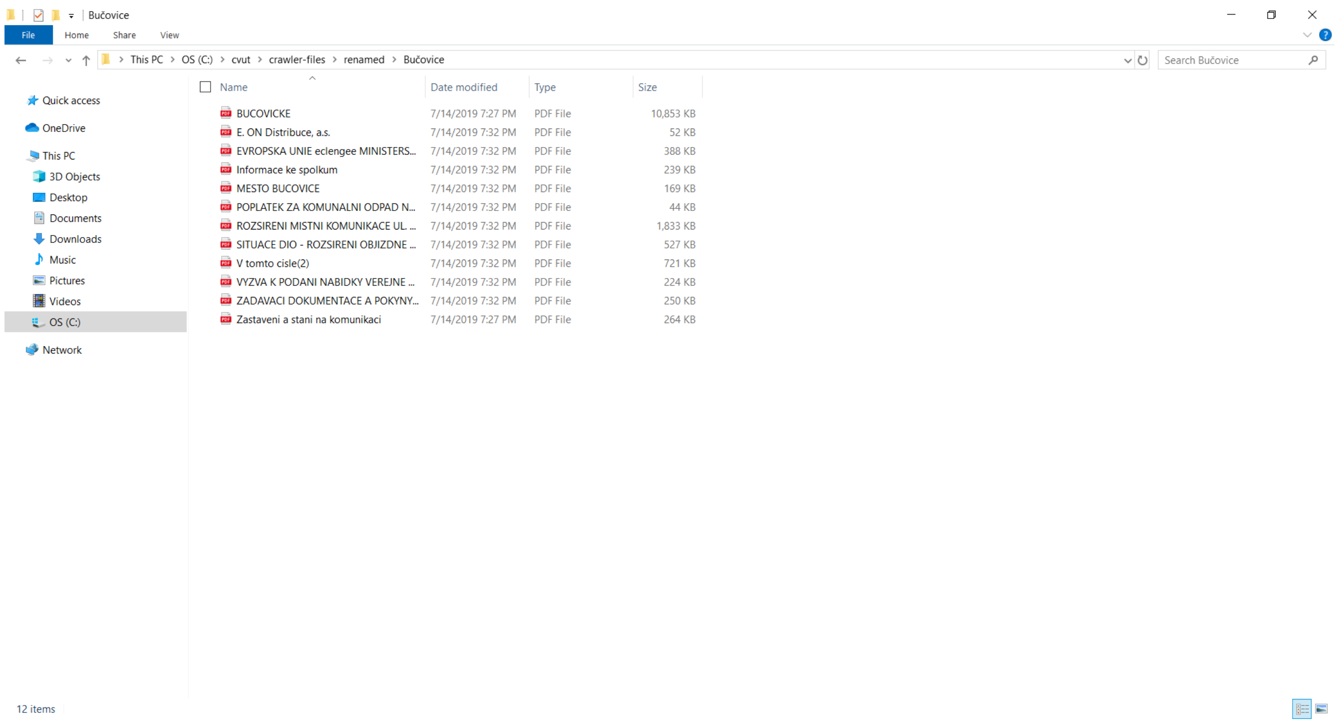

Abertamy

This PC > OS (C:) > cvut > crawler-new > files > Abertamy

Name	Date modified	Type	Size
full4fb0c8eaccc13e346fe98636adaf05...	7/19/2019 6:55 PM	PDF File	720 KB
full6ee8cc0ef3574426811f2bbc048a80...	7/19/2019 6:55 PM	PDF File	385 KB
full8c8b71614a15b77fde3a411a86116...	7/19/2019 6:54 PM	PDF File	371 KB
full9d8b25ef22c9daf45b3c05a637cf5e...	7/19/2019 6:55 PM	PDF File	429 KB
full9f6fe7fc11e22cb4a38550cf52f8c3c...	7/19/2019 6:55 PM	PDF File	263 KB
full22daf75d8a0c1081296578e0f3308...	7/19/2019 6:55 PM	PDF File	322 KB
full0383f8dd258de2b926b5b86f9880d...	7/19/2019 6:54 PM	PDF File	184 KB
full953f5212c5eff9f61487e5effb4e184...	7/19/2019 6:55 PM	PDF File	994 KB
full7230d32dadc8ea136a0117aea382...	7/19/2019 6:55 PM	PDF File	591 KB
full4746890b5a865ff730a0c77dade55...	7/19/2019 6:55 PM	PDF File	73 KB
fulla2bc3e54e79467a8f7a22f0a48172...	7/19/2019 6:55 PM	PDF File	237 KB
fullbc0b81df601004ebd170f51b40e53...	7/19/2019 6:54 PM	PDF File	150 KB
fullc04d1579d0cf629d187eace4d0881...	7/19/2019 6:54 PM	PDF File	27 KB
fullc032de464e977271273a85b19bc...	7/19/2019 6:55 PM	PDF File	881 KB
fullc58088dafc76301786858f0ea3fb99...	7/19/2019 6:55 PM	PDF File	553 KB
fullcc50794c024ecaf7a8c9f005d9b1b3...	7/19/2019 6:54 PM	PDF File	201 KB
fullcdcc11396534b135fd7ca34a70ab1...	7/19/2019 6:55 PM	PDF File	695 KB
fulld0d413f7c69390cadd9f78ff2c5228...	7/19/2019 6:55 PM	PDF File	952 KB
fulle79aa93ac2946c7a6e66ef4dfe0983...	7/19/2019 6:55 PM	PDF File	280 KB
fulle7225ca5c2e8304f1843f928d565c7...	7/19/2019 6:55 PM	PDF File	687 KB
fullf33caf05f72ee23928d40f58ec79f67...	7/19/2019 6:55 PM	PDF File	564 KB
fullf0381055f162fe87b2be10b9d7bfc6...	7/19/2019 6:55 PM	PDF File	354 KB
fullfba6684b67bb0eec91b033346730c...	7/19/2019 6:55 PM	PDF File	207 KB
fullfc6c85912f65807c90c9edf8d6a227...	7/19/2019 6:54 PM	PDF File	94 KB
fullfdb141b8c5fb81fe90bf180da21fdf...	7/19/2019 6:55 PM	PDF File	448 KB

25 items

Fig 3. Downloaded documents


Bučovice

This PC > OS (C:) > cvut > crawler-files > renamed > Bučovice

Name	Date modified	Type	Size
BUCOVICKE	7/14/2019 7:27 PM	PDF File	10,853 KB
E. ON Distribuce, a.s.	7/14/2019 7:32 PM	PDF File	52 KB
EVROPSKA UNIE eclengee MINISTERS...	7/14/2019 7:32 PM	PDF File	388 KB
Informace ke spolkum	7/14/2019 7:32 PM	PDF File	239 KB
MESTO BUCOVICE	7/14/2019 7:32 PM	PDF File	169 KB
POPLATEK ZA KOMUNALNI ODPAD N...	7/14/2019 7:32 PM	PDF File	44 KB
ROZSIRENI MISTNI KOMUNIKACE UL. ...	7/14/2019 7:32 PM	PDF File	1,833 KB
SITUACE DIO - ROZSIRENI OBJIZDNE ...	7/14/2019 7:32 PM	PDF File	527 KB
V tomto cisle(2)	7/14/2019 7:32 PM	PDF File	721 KB
VYZVA K PODANI NABIDKY VEREJNE ...	7/14/2019 7:32 PM	PDF File	224 KB
ZADAVACI DOKUMENTACE A POKYNY...	7/14/2019 7:32 PM	PDF File	250 KB
Zastaveni a stani na komunikaci	7/14/2019 7:27 PM	PDF File	264 KB

12 items

Fig 4. Renamed documents

Conclusion

The proposed solution has high success rate for downloading documents, but is insufficient to extract document names for further categorization of documents. It is assumed that the reason is too primitive method of determining the titles, based on the fact that the title text should be the first line of the text of the document. It is proposed to use PDF metadata extraction or pattern recognition to get better success rate for title extraction.