Collection and indexing process

Extraction -> Documents

Extraction is also called web scraping!

Documents are downloaded from the web site to a local folder: i.e. ~/drugle/<source>/<type>/<file_type>.

<source> represents a short name for the authority, i.e. "emea" and <type> the type of document, such as "spc". <file_type> represents the file extension.

The extraction script should take one argument containing a path to the download folder. If no argument is provided, it must download the documents to a default location.

Example:

```
python emea_pdf_extractor.py
```

will download pdf files to ~/drugle/emea/spc/pdf

The extraction script downloads documents, normally pdf files, to the specified location. In addition the URI for each pdf resource must be saved to <local_path>/uri/<file_name>

<local_path> is the local folder chosen for downloading pdf:s, and <file_name> is the pdf file name minus pdf extension.

Example:

```
martin@martin-laptop:~/drugle/emea/spc$ ls pdf/H-201* -1
pdf/H-201-PI-de.pdf
pdf/H-201-PI-en.pdf
pdf/H-201-PI-es.pdf
pdf/H-201-PI-fr.pdf
pdf/H-201-PI-sv.pdf

martin@martin-laptop:~/drugle/emea/spc$ ls uri/H-201* -1
uri/H-201-PI-de
uri/H-201-PI-en
uri/H-201-PI-es
uri/H-201-PI-fr
uri/H-201-PI-sv
```

Notes, questions & unresolved issues

- Note: Default locations should be under /var/lib/drugle (Paco).

Document processing (Converters -> Splitters -> Index files)

Hash calculating

Example:

```
python emea_pdf_hasher.py
```

Result:

```
martin@martin-laptop:~/drugle/emea/spc$ ls md5/H-201* -1
md5/H-201-PI-de
md5/H-201-PI-en
md5/H-201-PI-es
md5/H-201-PI-fr
md5/H-201-PI-sv
```

ech file containing a serialized dictionary "md5": <md5 hash>. i.e:

```
{'md5': '60581de9e68aec55f496e9133e5e5b61'}
```

Converting pdf to text

Converts pdf files to text files.

Should receive two arguments: source and destination. If arguments ommited, defaults are ~/drugle/<authorithy>/<type>/pdf and /var/drugle/sources/<authority>/<type>/txt

example:

```
sudo python emea_converter.py
```

Generating results:

```
ls /var/drugle/sources/emea/spc/txt/H-201* -1
/var/drugle/sources/emea/spc/txt/H-201-PI-de.txt
/var/drugle/sources/emea/spc/txt/H-201-PI-en.txt
/var/drugle/sources/emea/spc/txt/H-201-PI-es.txt
/var/drugle/sources/emea/spc/txt/H-201-PI-fr.txt
/var/drugle/sources/emea/spc/txt/H-201-PI-sv.txt
```

Please note that if no destination is provided, result is saved under /var directory, as txt files must be in production.

Additionally, pdf files are copied under /var directory

```
ls /var/drugle/sources/emea/spc/pdf/H-201* -1
/var/drugle/sources/emea/spc/pdf/H-201-PI-de.pdf
/var/drugle/sources/emea/spc/pdf/H-201-PI-en.pdf
/var/drugle/sources/emea/spc/pdf/H-201-PI-es.pdf
/var/drugle/sources/emea/spc/pdf/H-201-PI-fr.pdf
/var/drugle/sources/emea/spc/pdf/H-201-PI-sv.pdf
```

Notes, questions & unresolved issues

- Note: We should use /var/lib/drugle as the root directory for Drugle related data files. See FHS /var and FHS /var/lib (Paco).

Data splitting

In this step, index files are first created.

example:

```
sudo python emea_splitter.py
```

Resulting:

```
ls /var/drugle/sources/emea/spc/index
de  en  es  fr  sv
```

One directory for each language

Inside sv directory for example:

```
ls /var/drugle/sources/emea/spc/index/sv/H-201* -1
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.0.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.10.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.11.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.12.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.13.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.14.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.15.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.16.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.17.py
/var/drugle/sources/emea/spc/index/sv/H-201-PI-sv.18.py
..
..
```

Please not that one pdf can contain more than one spc. Each one of them is saved into a different file, distinguished by a point and a number indicating the version.

Indexing

Example:

```
sudo python emea_indexer.py
```

Index file

Each document extracted from the web is transformated with the goal to be indexed.

In first place the document is divided in a number of sections, each of one having a title.

Valid titles are stored in a special database (se aspects). The resulting document in this transform is stored in a file called a index file. Each of these files are uniquely named.

The overall structure of the index file is:

```
{
'id': <globally-unique-id>,
'title': <the-document-title>,
'author': <the-document-author>,
'site': <web site>
'md5': <md5-hash>
'updated': <date-time>,
'link': <url>,
'link_hosted': <file-name>,
'language': <ll_cc>,
'generator': <program>,
'categories': [<cat1>, <cat2>, ...],
'type': <doc-type>,
'atc': <atc-code>
'entries': {
    <id>: {
        'title': <aspect>
        'content': <blob>,
        'div': {
                "name": <result name>,
                "title": <result title>,
                "symbols": {"c": <c>, "t": <t>}, ## section 4.5 and inter_class
                "icon_path": <result icon>,
                "link_to_html_content": <link to html result>,
                "link_to_expand_text": <link to expand result content>,
                "link_to_original": <link to orignal document>,
                "link_to_local": <link to local document>,
                "link_to_text": <link to doucument text version>,
                "link_to_site": <link to host>
               }
        'scores': {<s1>: <v1>, <s2>: <v2>, ...},
        'terms': { <t1>: <ty1>, <t2>: <ty2>, ...}
        },
      ...
    }
}
```

id
: Globally unique id used to identify the file and must be the same as the file name.

title
: The title of the document. Often a descriptive text.

author
: The author of the document. Often an pharmaceutical authority.

site
: The web site where the original document is hosted.

md5
: The md5 hash of the document.

updated
: The timestamp of the latest modification

link
: Url to the original location of the document in the web

link_hosted
: Filename of the document strored

language
: Language and country code of the original document

categories
: Categories used to classify the document, e g: Diabetes, Pediatry, etc

type
: Type of document. E g: SPC, etc

atc
: ATC code of the main medical drug

entries
: Each document may contain one or more entries. Each entry has the identity <id>. Entries are the targets of the search. The result of the search is a list of entries.

entries.<id>.title
: The title of the entry, is unique in the document. Must be one of the valid aspects in the aspect database.

entries.<id>.content
: The content of the entry.

entries.<id>.div
: information to be processed by the web server in order to generate html view of results

entries.<id>.scores
: Score values used to sort the search result list.

entries.<id>.terms
: Typed terms found in this entry.

Indexing text

Because entries in the index file are the targets of the search, it is necessary to specify the text to be indexed.

For each entry <id> the text to be indexed is composed by the following parts separated by a single text space:

1) The whole text in entries.<id>.content

2) The entries.<id>.title

3) All the terms in the field entries.<id>.terms: <t1> + space + <t2> + space + ...