Annif Use and Explanation

Link to original doc

What is Annif? Why did we experiment with it?

Annif (http://annif.org/) is a tool built using natural language processing and machine learning techniques for recommending subjects for a document after being fed a particular controlled vocabulary. For the SMASH! Phase of our work, we experimented with retrieving related entities given a user query. We wanted to use ANNIF to use the Library of Congress Subject Headings and the subject headings used within the library catalog metadata to recommend subject headings given a user query (instead of an entire document).

Annif resources, data and algorithms used for this project

We followed the documentation at https://github.com/NatLibFi/Annif and https://github.com/NatLibFi/Annif-tutorial to understand how to set up the system and what are the data and metadata requirements. The steps documented at https://github.com/NatLibFi/Annif-tutorial/tree/master/exercises were invaluable!

Algorithm: We configured Annif to use its built-in TFIDF (http://www.tfidf.com/) algorithm for this phase, although ANNIF allows for using a combination of algorithms. If given enough additional time, it would be useful to try out the combination of algorithms to suggest subject headings.

Data:

- Vocabulary: We queried Dave's backup Fuseki to retrieve a TSV of LCSH heading titles and URIs. Relying on this data set means that we don't have the most recent LCSH headings but we do have content which is reasonable to use in an experiment (i.e. based on a data dump from 2019?). (Trying the RDF representation from the Library of Congress resulted in some errors. We are unclear why the error occurred but we would explore if blank nodes within the RDF download were problematic.) The TSV file has [X] subject to URI mapping lines. One of the lines from the file is below and shows the subject heading URI followed by the preferred label fo the subject:

<http://id.loc.gov/authorities/subjects/sh00000231> Antique and classic aircraft

- We retrieved 300,000 Solr documents from our production catalog, requesting the title (fulltitle_display) and subject heading (subject_display) fields. We wrote a script to iterate through the subject heading strings to query LCSH to retrieve URIs. The script then generated a TSV file that listed each title that had subject headings for which the script was able to find a URI in the following way: [resource title] [subject heading uri 1] [subject heading uri 2]. This file has 194,555 lines and thus that many titles mapped to subject headings.
    - An excerpt from the Solr results shows an example of the information used to generate the TSV


```
{
    "fulltitle_display":"Using R for item response theory model applications",
    "subject_display":["Item response theory",
      "R (Computer program language)"]},
```


    - One of the lines from the file is included below to show how the full title of the library resource is followed by LCSH URIs that correspond:
    - Using R for item response theory model applications http://id.loc.gov/authorities/subjects/sh85069055 http://id.loc.gov/authorities/subjects/sh2002004407

Installation, setup, and deployment on dev vm

- The version of Python running on the development VM is 2.7.5 whereas Annif requires 3.5+ . In order to maintain the default version while being able to use the version required by Annif, I used SCL (See https://phoenixnap.com/kb/how-to-install-python-3-centos-7 for an example of instructions for how to install SCL). "annif-venv" is the name of the virtual environment used in this project.
- I used the installation instructions at https://github.com/NatLibFi/Annif for the Basic Install option. I had to change permissions for directories to allow processes run under my username to run without requiring sudo.
- The project configuration file listed the algorithms. The first one, "tfidf-en" (TF-IDF English), is the one we used.
- Vocabulary data was saved in a TSV file (lcsh.tsv) with URIs and preferred labels for LCSH from Dave's backup Fuseki server and cached LCSH triples.
- Training data was also saved in a TSV file (subjecttrainres.tsv).
- The steps are documented below. If running the steps independently, we always need to start with the "running python" step described below.
    - Running python
        - scl enable rh-python36 bash
    - Setting up virtual environment. (This step only occurs once.)
        - Python -m venv annif-venv
    - Enable virtual environment
        - source annif-venv/bin/activate
    - Checking project list (This is included here just for reference and can be run anytime after the virtual environment is enabled)
        - annif list-projects
    - Load vocabulary (The vocabulary only needs to be loaded once).
        - annif loadvoc tfidf-en [path to file]/lcsh.tsv
    - Load training data (The training data also only needs to be loaded once)
        - annif train tfidf-en [path to file]/subjecttrainres.tsv
    - Try out suggestion for query (e.g. "myocardial infarction")

        - echo "myocardial infarction" | annif suggest tfidf-en
    - Run server for REST API calls
        - annif run --host 0.0.0.0
- Request for suggestions based on particular query
    - POST request to “[base URL]/annif/v1/projects/tfidf-en/suggest”
        - Data posted {text: “[query]”, limit: 10} if one wants only 10 results to be returned. A threshold can also be specified in case only results that are above this threshold for being a potential match should be returned.
        - CURL example
            - curl -X POST --header 'Content-Type: application/x-www-form-urlencoded' --header 'Accept: text/html' -d 'text=vienna%20architecture&limit=10' 'http://[URL]/v1/projects/tfidf-en/suggest'
- Explore SWAGGER UI
    - [Base URL]/v1/ui/ opens up SWAGGER documentation and enables exploration of sample requests

Suggestions example

Results example: Searching for “vienna architecture” with a limit of 10 results yields


```
{
  "results": [
    {
      "label": "International style (Architecture)",
      "score": 0.47264978289604187,
      "uri": "http://id.loc.gov/authorities/subjects/sh85067451"
    },
    {
      "label": "Architectural practice, International",
      "score": 0.47264978289604187,
      "uri": "http://id.loc.gov/authorities/subjects/sh85006606"
    },
    {
      "label": "Architecture--Philosophy",
      "score": 0.4689684808254242,
      "uri": "http://id.loc.gov/authorities/subjects/sh2007101285"
    },
    {
      "label": "Quality (Aesthetics)",
      "score": 0.4567379951477051,
      "uri": "http://id.loc.gov/authorities/subjects/sh94009536"
    },
    {
      "label": "Four elements (Philosophy)",
      "score": 0.447782963514328,
      "uri": "http://id.loc.gov/authorities/subjects/sh85051080"
    },
    {
      "label": "Black in interior decoration",
      "score": 0.4383489787578583,
      "uri": "http://id.loc.gov/authorities/subjects/sh2011002117"
    },
```

```
    {
      "label": "Architecture and literature",
      "score": 0.41448304057121277,
      "uri": "http://id.loc.gov/authorities/subjects/sh85006888"
    },
    {
      "label": "Architects--Professional ethics",
      "score": 0.41201311349868774,
      "uri": "http://id.loc.gov/authorities/subjects/sh85006572"
    },
    {
      "label": "Architecture and philosophy",
      "score": 0.40943264961242676,
      "uri": "http://id.loc.gov/authorities/subjects/sh95008375"
    },
    {
      "label": "Architecture and technology",
      "score": 0.40000787377357483,
      "uri": "http://id.loc.gov/authorities/subjects/sh97005373"
    }
  ]
}
```