annif tutorial

TFIDF project

1640
THE NATIONAL
LIBRARY
OF FINLAND

ZBW
Leibniz-Informationszentrum
Wirtschaft
Leibniz Information Centre
for Economics

Annif projects

- A project is used to set a vocabulary, a backend (i.e. algorithm), and other settings.

Annif projects

- A project is used to set a vocabulary, a backend (i.e. algorithm), and other settings.
- Projects are defined in a file usually called `projects.cfg` (or `projects.toml`) located in the current directory where Annif is executed.
    - This default filename/location can be overridden using `ANNIF_PROJECTS` environment variable or `--projects` option after a command.

Annif projects

- A project is used to set a vocabulary, a backend (i.e. algorithm), and other settings.
- Projects are defined in a file usually called `projects.cfg` (or `projects.toml`) located in the current directory where Annif is executed.
  - This default filename/location can be overridden using `ANNIF_PROJECTS` environment variable or `--projects` option after a command.
- A project is identified by a project id, which is typically a short string such as `yso-tfidf-en`.

Annif projects

- A project is used to set a vocabulary, a backend (i.e. algorithm), and other settings.
- Projects are defined in a file usually called `projects.cfg` (or `projects.toml`) located in the current directory where Annif is executed.
    - This default filename/location can be overridden using `ANNIF_PROJECTS` environment variable or `--projects` option after a command.
- A project is identified by a project id, which is typically a short string such as `yso-tfidf-en`.
- `annif list-projects` command shows the configured projects.

Exercise 2: TFIDF project

- The “Hello World” algorithm of automated subject indexing: quick to set up, train and test, but not the final say!

Example projects.cfg file for TFIDF project

```
[yso-tfidf-en]
name=YSO TFIDF project
language=en
backend=tfidf
vocab=yso
analyzer=snowball(english)
```

```
[stw-tfidf-en]
name=STW TFIDF project
language=en
backend=tfidf
vocab=stw
analyzer=snowball(english)
```

Example projects.toml file for TFIDF project

```
[yso-tfidf-en]
name="YSO TFIDF project"
language="en"
backend="tfidf"
vocab="yso"
analyzer="snowball(english"
```

```
[stw-tfidf-en]
name="STW TFIDF project"
language="en"
backend="tfidf"
vocab="stw"
analyzer="snowball(english)"
```

Setting up, training, and testing TFIDF project

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**
2. Load vocabulary: **`annif load-vocab VOCAB_ID SUBJECT_FILE`**

TSV or SKOS/RDF

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**
2. Load vocabulary: **`annif load-vocab VOCAB_ID SUBJECT_FILE`**
3. Train: **`annif train PROJECT_ID TRAINING_DATA`**

TSV or SKOS/RDF

(gzipped) TSV file
or directory

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**
2. Load vocabulary: **`annif load-vocab VOCAB_ID SUBJECT_FILE`**
3. Train: **`annif train PROJECT_ID TRAINING_DATA`**

TSV or SKOS/RDF

(gzipped) TSV file
or directory

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**
2. Load vocabulary: **`annif load-vocab VOCAB_ID SUBJECT_FILE`**
3. Train: **`annif train PROJECT_ID TRAINING_DATA`**
4. Test:
   a. Using one sentence:

```
echo "This is an example." | annif suggest PROJECT_ID
```

TSV or SKOS/RDF

(gzipped) TSV file
or directory

Setting up, training, and testing TFIDF project

1. Add project configuration to `projects.cfg`; verify using **`annif list-projects`**
2. Load vocabulary: **`annif load-vocab VOCAB_ID SUBJECT_FILE`**
3. Train: **`annif train PROJECT_ID TRAINING_DATA`**
4. Test:
   a. Using one sentence:

   ```
   echo "This is an example." | annif suggest PROJECT_ID
   ```

   b. Using a text file:

   ```
   annif suggest PROJECT_ID <FILE.TXT
   ```

TSV or SKOS/RDF

(gzipped) TSV file
or directory

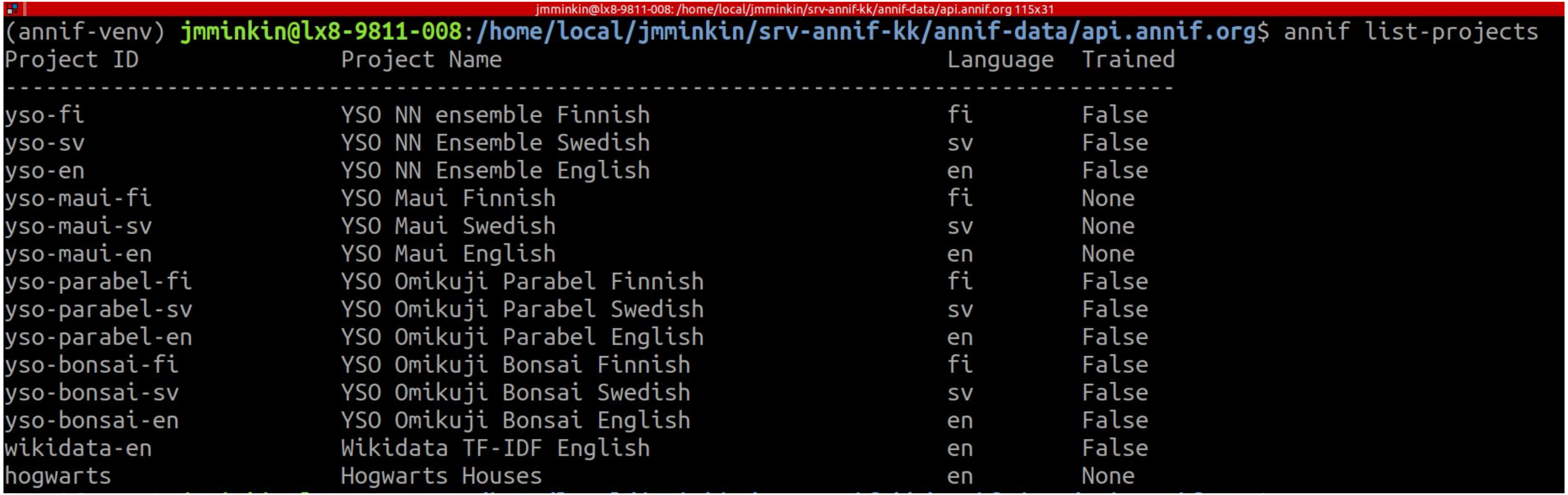
Step 1: Edit the projects.cfg file

```
[yso-tfidf-en]
name=YSO TFIDF project
language=en
backend=tfidf
vocab=yso
analyzer=snowball(english)
```

```
[stw-tfidf-en]
name=STW TFIDF project
language=en
backend=tfidf
vocab=stw
analyzer=snowball(english)
```

Step 2: Check the projects.cfg can be read

```
annif list-projects
```



jmminkin@lx8-9811-008: /home/local/jmminkin/srv-annif-kk/annif-data/api.annif.org 115x31

```
(annif-venv) jmminkin@lx8-9811-008:/home/local/jmminkin/srv-annif-kk/annif-data/api.annif.org$ annif list-projects
Project ID               Project Name                                 Language  Trained
---------------------------------------------------------------------------------------
yso-fi                   YSO NN ensemble Finnish                      fi        False
yso-sv                   YSO NN Ensemble Swedish                      sv        False
yso-en                   YSO NN Ensemble English                      en        False
yso-maui-fi              YSO Maui Finnish                             fi        None
yso-maui-sv              YSO Maui Swedish                             sv        None
yso-maui-en              YSO Maui English                             en        None
yso-parabel-fi           YSO Omikuji Parabel Finnish                  fi        False
yso-parabel-sv           YSO Omikuji Parabel Swedish                  sv        False
yso-parabel-en           YSO Omikuji Parabel English                  en        False
yso-bonsai-fi            YSO Omikuji Bonsai Finnish                   fi        False
yso-bonsai-sv            YSO Omikuji Bonsai Swedish                   sv        False
yso-bonsai-en            YSO Omikuji Bonsai English                   en        False
wikidata-en              Wikidata TF-IDF English                      en        False
hogwarts                 Hogwarts Houses                              en        None
```

Step 3: Load the vocabulary

```
annif load-vocab yso data-sets/yso-nlf/yso-skos.ttl
```

```
annif load-vocab stw data-sets/stw-zbw/stw-skos.ttl
```

You only have to do this once for a particular vocabulary. You can reuse the same vocabulary (by using the same `vocab=` value) in other projects.

Step 4: Train the project using sample data

Use a small training file based on 100,000 records to test the process:

```
annif train yso-tfidf-en data-sets/yso-nlf/yso-finna-small.tsv.gz
```

```
annif train yso-tfidf-en data-sets/yso-nlf/yso-finna-small.tsv.gz
```

Training should take around a minute.

Step 5: Test w/ a sample text using annif suggest

```
echo "Machine learning algorithms build a mathematical model based
on sample data" | annif suggest yso-tfidf-en
```

```
echo "Machine learning algorithms build a mathematical model based
on sample data" | annif suggest stw-tfidf-en
```

Step 6: Train the project using all training data

```
annif train yso-tfidf-en data-sets/yso-nlf/yso-finna.tsv.gz

annif train stw-tfidf-en data-sets/stw-zbw/stw-econbiz.tsv.gz
```

This should take around 5-10 minutes for the stw-zbw data set and around 10-15 minutes for the yso-nlf data set.

Step 7: Test w/ a document using annif suggest

For this step, you need the full text documents of your data set. Fetching them is explained in the data-sets exercise.

Pick any document from the `docs/test/` folder of your chosen data set. In these examples we use the lowest-numbered documents:

```
annif suggest yso-tfidf-en <data-sets/yso-nlf/docs/test/2017-D-52518.txt

annif suggest stw-tfidf-en <data-sets/stw-zbw/docs/test/10008797547.txt
```