Design and use of linguistic tools II
Building linguistic resources with NLP tools

Pablo Gamallo

CiTIUS
Universidade de Santiago de Compostela

Master EmLex

Table of Contents

Table of Contents

Objectives

- To use and apply NLP tools on text corpora:
  - tokenization and lemmatization
  - PoS tagging
  - syntactic analysis
  - multi-word extraction
  - named entity recognition
  - sentiment analysis
  - authorship attribution

Tools for Natural Language Processing (NLP)

Analysis

- tokenization
- lemmatization
- morpho-syntactic analysis (PoS-taggers)
- sintactic analysis (dependency parsers)

Extraction

- terms (multi-words)
- entities
- semantic relations
- concepts
- opinions, polarity

Applications

- summarization
- spelling/grammar checking
- (translation)

Tools for Natural Language Processing (NLP)

Analysis

- tokenization
- lemmatization
- morpho-syntactic analysis (PoS-taggers)
- sintactic analysis (dependency parsers)

Extraction

- terms (multi-words)
- entities
- semantic relations
- concepts
- opinions, polarity

Applications

- summarization
- spelling/grammar checking
- (translation)

Tools for Natural Language Processing (NLP)

Analysis

- tokenization
- lemmatization
- morpho-syntactic analysis (PoS-taggers)
- sintactic analysis (dependency parsers)

Extraction

- terms (multi-words)
- entities
- semantic relations
- concepts
- opinions, polarity

Applications

- summarization
- spelling/grammar checking
- (translation)

LinguaKit

- Web demo: `https://linguakit.com`
- Open source code: `https://github.com/citiususc/Linguakit`

Table of Contents

Tokenization

```
cat text.txt | PATH/Linguakit-master/linguakit tok es
```

Counting and sorting

```
cat text.txt | PATH/Linguakit-master/linguakit tok es | wc

cat text.txt | PATH/Linguakit-master/linguakit tok es -sort

cat text.txt | PATH/Linguakit-master/linguakit tok es |
sort | uniq -c | sort -nr
```

PoS tagging and Lemmatization

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es
```

Counting PoS tags and lemmas

- Count common nouns:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es |
cut -d `` `` -f 3 | grep ``NC" | wc
```

- Count lemma “comer”:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es |
cut -d `` `` -f 2 | grep ``^comer$'' | wc
```

Sorting PoS tags and lemmas

- Sorting lemmas by frequency:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es |
cut -d `` `` -f 2 | sort | uniq -c | sort -nr
```

- Sorting PoS tags by frequency:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es |
cut -d `` `` -f 3 | cut -c1-2 | sort | uniq -c | sort -nr
```

Dependency Parsing

```
cat text.txt | PATH/Linguakit-master/linguakit dep es
```

Dependency Parsing: Argument identification

Select the direct objects of the verb “comer”

```
cat text.txt | PATH/Linguakit-master/linguakit dep es |grep Dobj |
grep "comer\_VERB" |awk -F ";" '{print \$3}' |awk -F "\_" '{print \$1}
```

Named Entity Recognition-Classification (NER-NEC)

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es -ner
```

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es -nec
```

NERC: Selecting Locations and Organizations

Select locations:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es -nec |
grep NP00G | cut -d " " -f 1 | sort | uniq -c | sort -nr
```

Select organizations:

```
cat text.txt | PATH/Linguakit-master/linguakit tagger es -nec |
grep NP00O | cut -d " " -f 1 | sort | uniq -c | sort -nr
```

Table of Contents

Multi-Word Extraction

```
cat text.txt | PATH/Linguakit-master/linguakit mwe es
```

Multi-Word Extraction: Class Practice

Look for texts on a specific field (e.g. medicine, archeology,...) and use the multi-word extractor to build a terminology.

You can use a PDF to TXT conversor:

```
cat text.pdf | pdftotext > text.txt
```

Opinion Mining / Sentiment Analysis

```
cat text.txt | PATH/Linguakit-master/linguakit sent es
```

Opinion Mining: Class Practice

Open the polarity lexicon and introduce new terms

You can edit the Spanish lexicon as follows:

```
gedit PATH/Linguakit-master/sentiment/es/lex_es
```

Semantic Relation Extraction

```
cat text.txt | PATH/Linguakit-master/linguakit rel es
```

Open Information Extraction approach, described in:

Gamallo, P. and Marcos Garcia (2015). Multilingual Open Information Extraction, Lecture Notes in Computer Science, 9273, Berlin: Springer-Verlag: 711-722. ISNN: 0302-9743.

Table of Contents

Summarization

```
cat text.txt | PATH/Linguakit-master/linguakit sum es -p 5
```

Grammar Checking: Avalíngua

```
echo Vou a aportar a documentasción |
PATH/Linguakit-master/linguakit aval gl -xml
```

Online demos for Spanish:

`http://fegalaz.usc.es/nlpapi`

`http://fegalaz.usc.es/avalingua`

Authorship Attribution

Source code in:

`https://github.com/gamallo/Autoria`

Requirements:

```
cpan Math::KullbackLeibler::Discrete
```

Authorship Attribution: Class Practice

1. Select one book to be identified, for instance, “Fortunata y Jacinta”, de Galdós.
2. Select three other works by Galdós.
3. Select three works by other two authors, for instance, Borges and Unamuno.
4. Create four files in folder `./corpus/all`:
   - `FortunataYJacinta.txt` (to be compared against the rest of files)
   - `Galdos.txt` (merging the other 3 works by Galdós)
   - `Borges.txt` (merging the selected 3 works by Borges)
   - `Unamuno.txt` (merging the selected 3 works by Unamuno)
5. Run the script:

```
sh run.sh FortunataYJacita.txt
```