Natural Language Research Group

Presentation

Session requirements

Framework

Additional information

Introduction to Human Language Technologies
1. Framework

Natural Language Research Group

UNIVERSITAT POLITÈCNICA DE CATALUNYA
BARCELONATECH
Facultat d'Informàtica de Barcelona

Course 2018/19

Outline

1. Presentation
   - Objectives
   - Evaluation
   - Project
   - Software Installation
2. Session requirements
3. Framework
   - NLTK
   - Corpus Readers
   - Stopwords reader
   - Class Text
4. Additional information
   - Plain Text
   - Web resources

Goal of IHLT lab sessions

- Learn to use basic NLP functions for managing text content
- Solve simple programming exercises

  Programming platform: Jupyter (python)
  [works saved as notebooks - *.ipynb -]
  NLP package for Python: `nltk`

  Similar open-source NLP suites out of this framework:
  Stanford CoreNLP, Freeling, Apache OpenNLP, IXA Pipes

Evaluation of IHLT lab

- groups of 2 people, although individual works will be accepted
- A mandatory project (Semantic Textual Similarity)
- A set of mandatory exercises solved in lab sessions
- A set of optional exercises and projects
- Grade = 0.4 * Project + 0.2 * Exercises
  (this represents the 60% of the final IHLT grade)
- Jupyter notebooks of exercises & projects should be uploaded to `raco.fib.upc.edu`

Topic of the project

How similar two sentences are between them? compare different approaches

Relevance of the topic:
IR, QA, summarization, automatic translation, plagiarism detection, ...

A pair of texts is a paraphrase when both texts describe the same meaning with different words

Example from trial of project data set:

- `The bird is bathing in the sink.`
- `Birdie is washing itself in the water basin.`

Project description

Natural Language Research Group

Presentation
Project
Session requirements
Framework
Additional information

Deadline: 13/12/2018 (oral presentation)

- Implement some approaches to detect paraphrase using sentence similarity metrics.
  - Explore some lexical dimensions.
  - Explore the syntactic dimension alone.
  - Explore the combination of both previous.
- Compare and comment the results achieved by these approaches among them and among the official results.
- Use data set and description of task *Semantic Textual Similarity* in *SemEval 2012*
  `https://www.cs.york.ac.uk/semeval-2012/task6/index.html`
- Jupyter notebook: sts-[Student1]-[Student2].ipynb
- slides: sts-[Student1]-[Student2].pdf
- send files to `raco` in *'IHLT STS Project'* before the oral presentation

Framework installation (Linux)

Framework (python3, jupyter, nltk, numpy, scipy):

> `sudo apt-get install python3`

> `pip3 install -U pip`

> `pip3 install jupyter`

> `sudo pip3 install -U numpy`

> `sudo pip3 install -U nltk`

> `sudo pip3 install -U scipy`

> `jupyter notebook` (select New/Python3)

Stop server with Ctrl-C

Framework installation (Windows)

Framework (python3, jupyter, nltk, numpy, scipy):

- Download python3 from `http://www.python.org`
- Install it checking the option `Add python to the path`
- Start the shell (`cmd`)

```
> pip install jupyter
> pip install -U numpy
> pip install -U nltk
> pip install -U scipy
> jupyter notebook (select New/Python3)
  Stop server with Ctrl-C
```

Execution test

Validate the installation process

- Open a new python3 jupyter notebook
- Change the name of the session to S1-[Student1]-[Student2]
- Import without errors `nltk` library
- Save the session and exit jupyter server.

Outline

1 Presentation
- Objectives
- Evaluation
- Project
- Software Installation

2 Session requirements

3 Framework
- NLTK
- Corpus Readers
- Stopwords reader
- Class Text

4 Additional information
- Plain Text
- Web resources

Session requirements

Gutenberg corpus:

- Both Linux & Windows (via python shell)

```
> import nltk
> nltk.download('gutenberg')
> nltk.download('stopwords')
```

Attached resources:

- `pg35688.txt`
- `projectSTS.zip`

Outline

1 Presentation
- Objectives
- Evaluation
- Project
- Software Installation

2 Session requirements

3 Framework
- NLTK
- Corpus Readers
- Stopwords reader
- Class Text

4 Additional information
- Plain Text
- Web resources

NLTK Resources

Python library:

- List of resources: `http://www.nltk.org/nltk_data/`
- Download non-default resources from nltk

```
import nltk
nltk.download()
```

- **Corpora and lexical resources**: Brown corpus (PoS annotations), sentence_polarity corpus... Lexical resources such as WordNet, SentiWordNet and specialized word lists.
- **Toy grammars**: grammars for English, Spanish, ...
- **Models**: Named Entity recognizer, taggers for English and Russian, ...

Corpus reader

- `http://www.nltk.org/howto/corpus.html`
- corpus reader objects & classes
  - `> from nltk.corpus import *resource* [as *variable_name*]`
- Gutenberg corpora
  - `> nltk.download('gutenberg')`
  - `> nltk.corpus.gutenberg.fileids()`
  - `> txt = nltk.corpus.gutenberg.words ('austen-persuasion.txt')`

Corpus reader

Example using the Gutenberg corpus (I):

```
In [1]: import nltk
        # list of files
        nltk.corpus.gutenberg.fileids()

Out[1]: ['austen-emma.txt',
         'austen-persuasion.txt',
         'austen-sense.txt',
         ...]

In [2]: # load a file
        cp = nltk.corpus.gutenberg.words('blake-poems.txt')
        len(cp) # length

Out[2]: 8354

In [3]: cp[100:108] # subset

Out[3]: ['the', 'same', 'again', ',', 'While', 'he', 'wept', 'with']
```

Corpus reader

Example using the Gutenberg corpus (II):

```
In [4]: len(set(cp)) # set

Out[4]: 1820

In [5]: # list of words with more than 2 chars
        lst = [w for w in cp if len(w)>2]
        lst[:30]

Out[5]: ['Poems',
         'William',
         'Blake',
         '1789',
         ...]

In [6]: # tuples with words lowered and length
        tup = [(w.lower(), len(w)) for w in lst]
        sorted(tup)[:5]

Out[6]: [('!"--', 4), ('\',"', 3), (',--', 3), ('1780', 4), ('1789', 4)]
```

Corpus reader

Example using the Gutenberg corpus (III):

```
In [7]: # frequencies of words
        freqs = {w:lst.count(w) for w in set(lst)}
        kmax = max(freqs, key=lambda k: freqs[k])
        kmax, freqs[kmax]

Out[7]: ('the', 351)
```

Stopwords reader

Provide the list of stop words of a specific language. Words that do not have individual meaning (pronouns, determiners, auxiliary verbs, ...)

Natural Language Research Group

Presentation

Session requirements

Framework

Stopwords reader

Additional information

```
In [1]: from nltk.corpus import stopwords

        sw=set(stopwords.words('english'))
        len(sw)

Out[1]: 179

In [2]: 'the' in sw

Out[2]: True

In [3]: sw

Out[3]: {'a', 'about', 'above', 'after', 'again', ...}
```

Mandatory exercise

Natural Language Research Group

Presentation

Session requirements

Framework

Stopwords reader

Additional information

1. Develop a `jupyter notebook` that show the 25 non-stopwords with more number of occurrences in the file `'blake-poems.txt'` of `Gutenberg` corpus.

Upload the jupyter file of the exercise to the `Raco`.

Class Text

Natural Language Research Group

Presentation

Session requirements

Framework

Class Text

Additional information

Consulting occurrences of words:

```
In [1]: from nltk.corpus import gutenberg
        from nltk.text import Text
        crp = Text(gutenberg.words('blake-poems.txt'))

In [2]: # counting words
        crp.count('love')

Out[2]: 15

In [3]: crp.count('Love')

Out[3]: 13

In [4]: # consulting context in a corpus
        crp.concordance('love')

Displaying 25 of 29 matches:
at we may learn to bear the beams of love And these black bodies and this sunb
ing , ' Come out from the grove , my love and care And round my golden tent li
, And be like him , and he will then love me . THE BLOSSOM Merry , merry sparr
 IMAGE To Mercy , Pity , Peace , and Love , All pray in their distress , And t
...
```

Class Text

Consulting contexts:

```
In [5]: # words in the same context
        crp.similar('love')

went youth desires compelled say nuts pointed none by thrush earth
turned ease all see there sight innocence him pitying
```

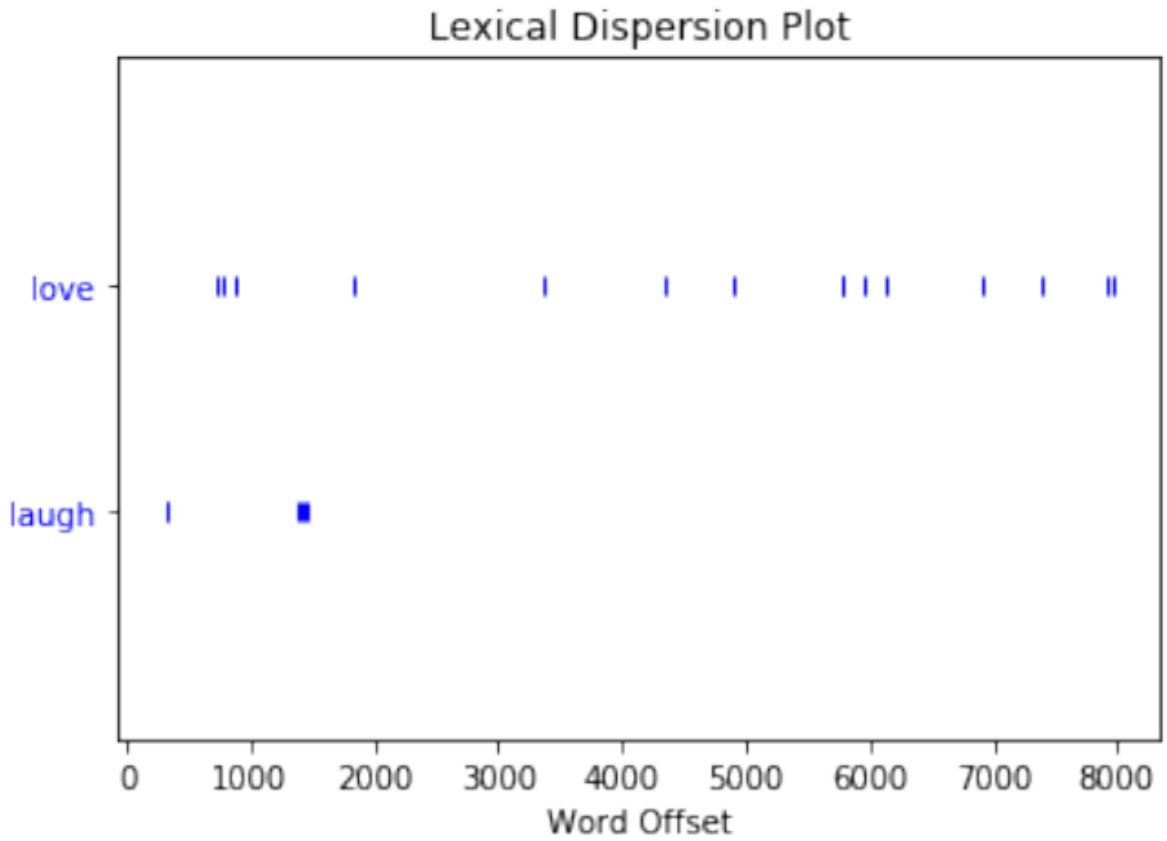
```
In [6]: # words in both context
        crp.common_contexts(['love', 'laugh'])

No common contexts were found
```

Class Text

Dispersion plot:

```
In [8]: # dispersion plot
        crp.dispersion_plot(['love','laugh'])
```



Optional exercise

1. Remake the same steps in the example above but with the file `'austen-sense.txt'` of `Gutenberg` corpus.
2. Compare the results with those in the example.
3. Be sure that the image of the dispersion plot appears in the notebook.

Upload files of the exercise to the `Raco`.

Outline

1 Presentation
- Objectives
- Evaluation
- Project
- Software Installation

2 Session requirements

3 Framework
- NLTK
- Corpus Readers
- Stopwords reader
- Class Text

4 Additional information
- Plain Text
- Web resources

Plain Text Example

Loading corpus from a text file:

```
In [1]: import nltk

        crp = nltk.corpus.PlaintextCorpusReader('../data', '.*\.txt').words()
        len(crp)

Out[1]: 23714

In [2]: crp[:10]

Out[2]: ['The',
         'Project',
         'Gutenberg',
         'EBook',
         'of',
         'Alice',
         'in',
         'Wonderland',
         ',',
         'by']
```

Web Example

Fetching web data as string:

```
In [1]: import urllib.request

        url = 'http://www.gutenberg.org/cache/epub/35688/pg35688.txt'

        with urllib.request.urlopen(url) as response:
           dt = response.read().decode('utf8')

        dt[1:55]

Out[1]: 'The Project Gutenberg EBook of Alice in Wonderland, by'
```