Γλωσσική Τεχνολογία

Natural Language Toolkit

Natural Language Toolkit

- Πακέτο βιβλιοθηκών και εργαλείων για Natural Language Processing σε Python.
- Δεν εγκαθίσταται με την Python, πρέπει να το εγκαταστήσετε.
- http://www.nltk.org/
  - Download των πακέτων που χρειάζονται
  - Οδηγίες για την εγκατάσταση
  - Διαθέσιμο online το βιβλίο "Natural Language Processing with Python"
- Σε αυτό το φροντιστήριο αναφέρονται κάποια βασικά εργαλεία. Το NLTK περιέχει πολλά περισσότερα!!!

NLTK – Installing Corpora

- Το NLTK δίνει τη δυνατότητα εγκατάστασης corpora.
- Χρησιμοποιούνται για πολλές NLP εργασίες, όπως Normalization, Tagging, Classification etc.

```
>>> import nltk
>>> nltk.download()
NLTK Downloader
---------------------------------------------------------------------------
    d) Download      l) List      c) Config      h) Help      q) Quit
---------------------------------------------------------------------------
Downloader>
```

- Το Brown Corpus και το Wordnet αρκούν.
  - Καλύτερα εγκαταστήστε τα όλα!

Using Corpora

- Μέσω του NLTK είναι δυνατή η προσπέλαση των αρχείων στα corpora:

```
>>> nltk.corpus.brown.fileids()
['ca01', 'ca02', 'ca03', 'ca04', 'ca05', 'ca06', 'ca07', 'ca08', 'ca09', 'ca10', 'ca11',
    'ca12', 'ca13', 'ca14', 'ca15', 'ca16', 'ca17', 'ca18', 'ca19', 'ca20',
    ...
```

- Προσπέλαση του καθαρού κειμένου στα αρχεία:

```
>>> nltk.corpus.brown.raw('ca01')
"\n\n\tThe/at Fulton/np-tl County/nn-tl Grand/jj-tl Jury/nn-tl said/vbd Friday/nr
    an/at investigation/nn of/in Atlanta's/np$ recent/jj primary/nn election/nn
    produced/vbd ``/`` no/at evidence/nn ''/'' that/cs any/dti irregularities/nns
    took/vbd place/nn ./.\n\n\n\tThe/at jury/nn further/rbr said/vbd in/in term-
    end/nn presentments/nns that/cs the/at City/nn-tl Executive/jj-tl
    Committee/nn-tl ,/,.
    ...
```

  - Το κείμενο στο brown corpus είναι tagged!

Corpora Data

- Διάβασμα των κειμένων ανά λέξη

```
>>> from nltk.corpus import brown
>>> brown.words('ca01')
['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', ...]
```

- Διάβασμα των κειμένων ανά πρόταση

```
>>> brown.sents('ca01')
[['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', 'Friday', 'an', 'investigation', 'of',
    "Atlanta's", 'recent', 'primary', 'election', 'produced', '``', 'no', 'evidence', "''",
    'that', 'any', 'irregularities', 'took', 'place', '.'], ['The', 'jury', 'further', 'said', 'in',
    'term-end', 'presentments', 'that', 'the', 'City', 'Executive', 'Committee', ',',
    'which', 'had', 'over-all', 'charge', 'of', 'the', 'election', ',', '``', 'deserves', 'the',
    'praise', 'and', 'thanks', 'of', 'the', 'City', 'of', 'Atlanta', "''", 'for', 'the', 'manner',
    'in', 'which', 'the', 'election', 'was', 'conducted', '.'], ...]
```

  - Λίστα από λίστες λέξεων!

Brown Corpus

- Το Brown Corpus περιέχει κείμενα ταξινομημένα σε κατηγορίες

```
>>> from nltk.corpus import brown
>>> brown.categories()
['adventure', 'belles_lettres', 'editorial', 'fiction', 'government', 'hobbies', 'humor',
    'learned', 'lore', 'mystery', 'news', 'religion', 'reviews', 'romance',
    'science_fiction']
```

- Λέξεις ανά κατηγορία

```
>>> brown.words(categories='science_fiction')
['Now', 'that', 'he', 'knew', 'himself', 'to', 'be', ...]
```

- Προτάσεις ανά κατηγορία

```
>>> brown.sents(categories='science_fiction')
[['Now', 'that', 'he', 'knew', 'himself', 'to', 'be', 'self', 'he', 'was', 'free', 'to', 'grok',
    'ever', 'closer', 'to', 'his', 'brothers', ',', 'merge', 'without', 'let', '.'], ["Self's",
    'integrity', 'was', 'and', 'is', 'and', 'ever', 'had', 'been', '.'], ...]
```

Processing Raw Text

- Στο NLTK περιλαμβάνονται (ανάμεσα στ' άλλα) εργαλεία για:
  - Εξαγωγή κειμένου από ιστοσελίδες
  - Normalization
  - Tokenization
  - Tagging

Raw Text Extraction From HTML

- Κατέβασμα του περιεχομένου ενός url

```
>>> from urllib import urlopen
>>> url="http://en.wikipedia.org/wiki/Natural_Language_Toolkit"
>>> raw=urlopen(url).read()
>>> raw
'<!DOCTYPE html>\n<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
    lang="en">\n<head>\n<title>Wikimedia Error</title>\n<meta http-equiv="Content-
    Type" content="text/html; charset=UTF-8"/>\n<meta name="author" content="Mark
    Ryan"/>\n<meta name="copyright" content="(c) 2005-2007 Mark Ryan and others.
    Text licensed under the GNU Free Documentation License.
    http://www.gnu.org/licenses/fdl.txt“
...
```

- Εξαγωγή κειμένου

```
>>> pure=nltk.clean_html(raw)
>>> pure
'Wikimedia Error \n \n \n \n\n\n\n \n\n \n Wikimedia Foundation \n\n\n Error \n\n\n\n
    \n\n \n Our servers are currently experiencing a technical problem. This is probably
    temporary and should be fixed soon. Please try again in a few minutes. \n You may be
    able to get further information in the #wikipedia channel on the Freenode IRC
    network . \n The Wikimedia Foundation is
...
```

  - Δεν επιτρέπεται παντού το crawling ☺

Tokenization

- Μετατροπή ενός κειμένου σε λίστα από tokens
- Simple split

```
>>> text="When it's over, I want to go. It's 15:30!"
>>> tokens=text.split(" ")
>>> tokens
['When', "it's", 'over,', 'I', 'want', 'to', 'go.', "It's", '15:30!']
```

- Using Regular Expressions

```
>>> import re
>>> tokens=re.split(r'\W+',text)
>>> tokens
['When', 'it', 's', 'over', 'I', 'want', 'to', 'go', 'It', 's', '15', '30', '']
```

- NLTK

```
>>> tokens=nltk.word_tokenize(text)
>>> tokens
['When', 'it', "'s", 'over', ',', 'I', 'want', 'to', 'go.', 'It', "'s", '15', ':', '30', '!']
```

Text Normalization

- Κανονικοποίηση λέξεων: μετατροπή σε τύπους που μπορούν να ομαδοποιηθούν.
- Stemming (αποκατάληξη)

```
>>> porter=nltk.PorterStemmer()
>>> tokens=['baby','babies','child','children']
>>> stemms=[porter.stem(t) for t in tokens]
>>> stemms
['babi', 'babi', 'child', 'children']
```

- Lemmatization (αναγωγή στον πρώτο κλιτικό τύπο)

```
>>> wnl=nltk.WordNetLemmatizer()
>>> tokens=['baby','babies','child','children']
>>> lemmas=[wnl.lemmatize(t) for t in tokens]
>>> lemmas
['baby', 'baby', 'child', 'child']
```

  - Αλλά

```
>>> wnl=nltk.WordNetLemmatizer()
>>> tokens=['go','goes','went']
>>> lemmas=[wnl.lemmatize(t) for t in tokens]
>>> lemmas
['go', 'go', 'went']
```

Tagging

- Αναγνώριση του Part of Speech
- Using NLTK

```
>>> text="Natural Language Processing is a growing field."
>>> tokens=nltk.word_tokenize(text)
>>> nltk.pos_tag(tokens)
[('Natural', 'NNP'), ('Language', 'NNP'), ('Processing', 'NNP'), ('is', 'VBZ'), ('a', 'DT'),
    ('growing', 'VBG'), ('field', 'NN'), ('.', '.')]
```

- Χρήση tagged κειμένων

```
>>> from nltk.corpus import brown
>>> tagged_text=brown.raw('ca01')
>>> tagged_text
"\n\n\tThe/at Fulton/np-tl County/nn-tl Grand/jj-tl Jury/nn-tl said/vbd Friday/nr an/at
    investigation/nn of/in Atlanta's/np$ recent/jj primary/nn election/nn produced/vbd ``/``
    no/at evidence/nn ''/'' that/cs any/dti irregularities/nns took/vbd place/nn ./.
>>> tagged_tokens=[nltk.tag.str2tuple(t) for t in tagged_text.split()]
>>> tagged_tokens
[('The', 'AT'), ('Fulton', 'NP-TL'), ('County', 'NN-TL'), ('Grand', 'JJ-TL'), ('Jury', 'NN-TL'), ('said',
    'VBD'), ('Friday', 'NR'), ('an', 'AT'), ('investigation', 'NN'), ('of', 'IN'), ("Atlanta's", 'NP$'),
    ('recent', 'JJ'), ('primary', 'NN'), ('election', 'NN'), ('produced', 'VBD'), ('``', '``'), ('no', 'AT'),
    ('evidence', 'NN'), ("''", "''"), ('that', 'CS'), ('any', 'DTI'), ('irregularities', 'NNS'), ('took', 'VBD'),
    ('place', 'NN'), ('.', '.'),
```

Για το project

- Η καλύτερη λύση είναι:
  - Tokenization με το NLTK
  - Normalization με lemmatization και όχι με stemming
  - Normalization & tagging με χρήση των εξωτερικών taggers που δίνονται στη σελίδα του εργαστηρίου (κάνουν και τα δύο).
  - Γιατί: το lemmatization του wordnet σε άλλα μέρη του λόγου εκτός των ουσιαστικών δεν είναι καλό!