Section 1:

1. https://www.nltk.org/data.html

NLTK 3.4.5 documentation

PREVIOUS | NEXT | MODULES | INDEX

Installing NLTK Data

NLTK comes with many corpora, toy grammars, trained models, etc. A complete list is posted at: http://nltk.org/nltk_data/

To install the data, first install NLTK (see http://nltk.org/install.html), then use NLTK's data downloader as described below.

Apart from individual data packages, you can download the entire collection (using "all"), or just the data required for the examples and exercises in the book (using "book"), or just the corpora and no grammars or trained models (using "all-corpora").

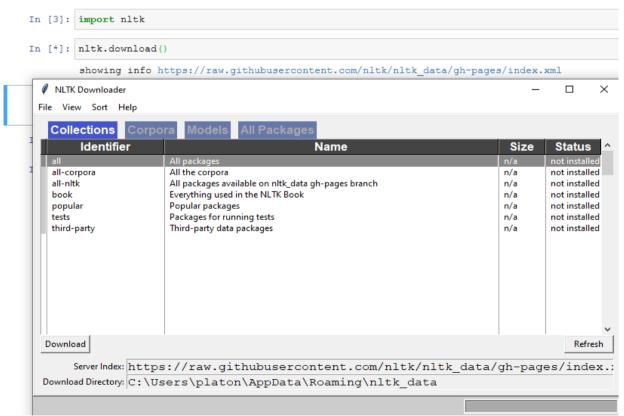
Interactive installer

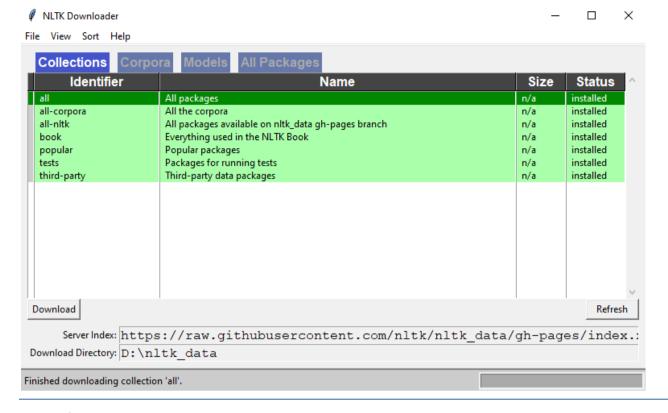
For central installation on a multi-user machine, do the following from an administrator account.

Run the Python interpreter and type the commands:

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

2. Interactive installer





2.1 Kiểm tra dữ liệu sau cài đặt

Accessing in-built corpora



2.2 Sử dụng thư viên nltk để (Sentence Segmentation, Word Segmentation, POS Tagging ..) được giới thiệu trong slide chương 2:

3. Underthesea - Vietnamese NLP Toolkit

Gõ lệnh cài đặt thư viên: pip install underthesea

4. Viết lệnh thể hiện các yêu cầu sau:

- Sentence Segmentation (sent tokenize)
- Word Segmentation (word_tokenize)

- POS Tagging (pos_tag)
- Chunking (*chunk*)
- Dependency Parsing (dependency_parse)
- Named Entity Recognition (ner)
- Text Classification (*classify*)

5. Sử dụng thư viện Pyvi để tách từ tiếng Việt

from pyvi import ViTokenizer, ViPosTagger

```
[18] def clean_document(doc):
         doc = ViTokenizer.tokenize(doc) #Pyvi Vitokenizer library
         doc = doc.lower() #Lower
         tokens = doc.split() #Split in_to words
         tokens = [word for word in tokens if word]
         return tokens
     clean_document('Bác sĩ bây giờ có thể thản nhiên báo tin bệnh nhân bị ung thư?')
     ['bác sĩ',
      'bây_giờ',
      'có_thể',
      'thản_nhiên',
      'báo',
      'tin',
      'bệnh_nhân',
      ˈbi̞ˈ,
      'ung_thư',
      '?'1
```

Section 2:

import một số thư viện, nếu chưa có thì setting thư viện

```
from pyvi import ViTokenizer, ViPosTagger
import gensim
import os
import pickle
import codecs
import re
from tqdm import tqdm
import regex
import pandas as pd
```

6. Xóa các thẻ HTML

```
def remove_html(txt):
    return regex.sub(r'<[^>]*>', '', txt)
```

7. Xóa stopwords

8. Tiền xử lý dữ liệu văn bản.

```
def text_preprocess(document):
   # xóa html code
   document = remove_html(document)
   # chuẩn hóa unicode
   document = convert_unicode(document)
   # tách từ
   document = ViTokenizer.tokenize(document)
   # đưa về lower
   document = document.lower()
   # xóa các ký tự không cần thiết
   # xóa khoảng trắng thừa
   document = regex.sub(r'\s+', ' ', document).strip()
   # Remove các ký tự kéo dài: vd: đẹppppppp
   document = re.sub(r'([A-Z])\1+', lambda m: m.group(1).upper(), document, flags=re.IGNORECASE)
   # remove nốt những ký tự thừa thãi
   document = document.replace(u'"', u' ')
   document = document.replace(u' ', u'')
   return document
```

9. Load dữ liệu

```
idef get_data(folder_path):
    categories = os.listdir(folder_path)
    for path in tqdm(categories):
        file_paths = os.listdir(os.path.join(folder_path, path))
        for file_path in tqdm(file_paths):
            with open(os.path.join(folder_path, path, file_path), 'r', encoding="utf-8") as f:
                lines = f.readlines()
                lines = ' '.join(lines)
                lines = text_preprocess(lines)
                lines = gensim.utils.simple_preprocess(lines)
                lines = ' '.join(lines)
                lines = remove_stopwords(lines)
                if 'neg' in path:
                    label = 0
                else:
                    label = 1
                with open('data.txt', 'a', encoding='utf8') as f:
                    f.write(f"{label} {lines}\n")
```

10. Sử dụng hàm get_data() để load dữ liệu và xem kết quả.

train_path = 'data_train/train'
get_data(train_path)

11. Tìm hiểu, nghiên cứu thêm Chuẩn hóa Unicode tiếng Việt, Chuẩn hóa dấu từ tiếng việt