Premiers pas avec Python et la boite à outils NLTK

Cours d'initiation au TAL - M1 SCA & SDL

3 septembre 2013

Après avoir ouvert une session Windows, lancez l'interface interactive *Idle* de *Python*. Parallèlement, à l'aide d'un navigateur, affichez le tutoriel NLTK en ouvrant la page web à l'URL : *nltk.org/book*.

1 Traiter des textes et des mots

Suivez attentivement le tutoriel en ouvrant le chapitre 1 *Language Processing and Python* et exécutant tous les exemples qui accompagnent les explications dans la fenêtre *Idle* ouverte en parallèle.

La section 1.1 est une initiation à Python et au traitement des textes et des mots qu'ils contiennent à l'aide des outils Python fournis par la bibliothèque NLTK.

Après une brève introduction à l'utilisation de l'interprète de Python, on montre comment télécharger certains corpus qui vont être traités par la suite. La commande *nltk.download()* ne fonctionne pas du fait que le téléchargement passe par un serveur proxy. Ce n'est pas gênant car les corpus sont déjà téléchargés et on peut passer la commande.

Ensuite, sont présentés différents outils qui permettent d'extraire des propriétés des mots contenus dans un corpus : contextes d'un mot, mots similaires, contextes communs à une liste de mots, dispersion d'un mot dans un texte, comptage des mots, des mots différents.

2 Les listes, les variables et les chaînes de caractères

La section 1.2 présente certaines formes de données manipulées par Python et très utiles en TAL. Elle le fait justement à travers des exemples issus du TAL. Il s'agit tout d'abord de la notion de *liste* et la façon d'accéder à un élément d'une liste ou à une sous-liste. Ensuite, est abordée la notion de *variable* qui permet de stocker une valeur pour pouvoir la réutiliser par la suite. A la fin de la section, on trouve une brève présentation des *chaînes de caractères*.

3 Calcul statistique sur un corpus

La section 1.3 aborde le calcul statistique sur un corpus.Cette section peut être passée sans que cela ne nuise à la compréhension de la suite.

Parmi les calculs simples qui sont décrits dans la section 1.3, il y a celui de de la distribution de fréquence des mots dans un corpus. On peut aussi chercher les mots suffisamment fréquents et longs (les mots courts sont en général des mots grammaticaux qui ne nous renseignent pas sur le contenu informatif d'un texte). Les *collocations* sont

des ensembles de mots qui sont présents simultanément dans un texte. Ici, on se contente de chercher les collocations qui sont des suites de deux mots.

4 Instructions de choix et de répétition de Python

La section 1.4 commence par une présentation des *instructions conditionnelles* qui permettent d'effectuer des choix dans les programmes en fonction de conditions. Les *boucles*, qui sont abordées ensuite, permettent d'itérer une instruction.

5 Programmes Python réutilisables

Passer la section 1.5 qui donne une idée de différentes applications du TAL et ouvrir le chapitre 2 *Accessing Text Corpora and Lexical Resources*. Aller directement à la section 2.3 qui explique tout d'abord qu'il est possible d'utiliser Python autrement qu'en mode interactif. A l'aide de l'éditeur de *Idle*, on peut écrire un programme Python de plusieurs lignes, le sauvegarder dans un fichier avec l'extension *.py* pour ensuite l'exécuter d'un seul coup. Un tel fichier porte le nom de *module*. On peut l'utiliser à l'aide de l'instruction *import*.

Python utilise la variable *path* du module *sys* pour stocker une liste de chemins possibles où seront recherchés les modules à importer à l'aide de l'instruction *import*. Pour ajouter un nouveau chemin dans la liste , il faut exécuter les instructions suivantes :

import sys
sys.path += [<nouveau chemin>]

Dans un module, on peut définir des *fonctions* qu'on peut ensuite appeler. La définition d'une fonction commence par le mot clé *def* suivi du nom de la fonction puis de la liste de ses paramètres entre parenthèses. Dans le corps de la fonction, une instruction *return* suivie d'une expression indique quelle valeur sera retournée par la fonction. Une fois qu'elle est définie une fonction peut être utilisée. il suffit de l'appeler avec son nom suivi entre parenthèses de la valeur de ses paramètres ou arguments.

6 Exercices

Mettre les réponses à chacun des exercices ci-dessous dans un fichier désigné ainsi : *TP1<numéro de l'exercice>.py*. Les réponses qui sont des explications et non des programmes Python doivent être mises sous forme de commentaires. En Python un commentaire est une ligne précédée par le caractère #. Pour que l'interprète Python puisse lire les lettres accentuées de votre fichier, mettez en entête de ce fichier le commentaire suivant *#-*- coding : utf-8 -*-*, qui indique que le codage des caractères utilisé est UTF8.

Exercice 6.1 *Dans le texte* "Sense and Sensibility by Jane Austen 1811" *qui fait partie des textes présentées dans la sous-section 1.1, chercher tous les contextes du mot* sensibility.

Exercice 6.2 *Faire les exercices 4, 6, 13, 15 de la section 1.8. Pour l'exercice 13,* sent1 *est la variable utilisée plus haut dans le chapitre.*

Exercice 6.3 *Faire les exercices 24 et 25 de la section 1.8.*

Exercice 6.4 *En s'aidant de la section 2.3, expliquer comment stocker la fonction* plural*, définie dans la même section, dans un module* morphology *qui est placé dans le répertoire* /home/nltk *et comment faire ensuite pour l'utiliser.*

Exercice 6.5 *(plus difficile) Ecrire un programme qui prenne en entrée un texte sous forme d'une liste de mots et qui retourne en sortie la liste des mots lexicaux de ce texte (par opposition aux mots grammaticaux). Par exemple, si on lui donne en entrée le texte* ["Je", "la", "connais", "très", "bien", ".", "Je", "crois", "que", "le", "directeur", "de", "l'", "entreprise", "la", "rencontrera", "."]*, le programme retournera* [""connais", "bien", "crois", "directeur", "entreprise", "rencontrera"]

Réunir les cinq fichiers contenant les réponses aux exercices dans une archive nommée *<nom-étudiant>.<prénom-étudiant>.TP1.zip* et envoyez-le par mail en pièce jointe à l'adresse *perrier@loria.fr* en mettant dans l'objet du message *initiation au TAL : TP1.*