TP1 - segmentation de textes

Programmation pour le TAL - M1 SDL & SCA

18 février 2013

Pour chacun des exercices, sauvegardez le script Python dans un fichier qui a comme nom *exo <numéro de l'exercice>.py* puis mettre tous les fichiers dans une archive .zip qui a comme nom *TP1_<nom de l'étudiant>_<prénom de l'étudiant >.zip*. A la fin de la séance, envoyez l'archive en fichier attaché à l'adresse *perrier@loria.fr* en mettant comme objet du message *programmation pour le TAL : TP1*.

1 Segmentation d'un texte en phrases

L'analyse d'un texte écrit commence par un découpage de celui-ci en phrases et par un découpage de chaque phrase en mots. On suppose qu'on commence par un découpage en phrases. Le signe de ponctuation qui pose problème est le point car il peut être utilisé à d'autres effets, notamment pour marquer des abréviations (M., etc., cf., fig. . . .) et des acronymes (O.N.U., S.P.A. . . .). Par ailleurs, les points de suspension ne marquent pas toujours la fin d'une phrase.

Exercice 1.1 *Écrivez un script Python prenant en entrée un texte en français sous forme d'une chaîne de caractères et fournissant en sortie une liste des phrases constituant le texte. Chaque phrase est une chaîne de caractères et elle se termine par un signe de ponctuation qui est soit un point, un point-virgule, un point d'interrogation ou un point d'exclamation. Pour ce qui est du traitement des ambiguïtés du point, on supposera que lorsqu'il est utilisé pour terminer une phrase, la phrase suivante débute par un espace ou un passage à la ligne avec ensuite une majuscule (attention au traitement minutieux des points de suspension).*

Utilisez pour écrire le script, un minimum de fonctions prédéfinies. Pour la construction de la liste, utilisez l'opérateur + qui permet de concaténer deux listes. A la fin du script, vous écrirez un texte comportant au moins 5 phrases, choisi soigneusement de façon à mettre en évidence la pertinence du programme.

2 Segmentation d'une phrase en mots

Une fois qu'un texte est découpé en phrases, il faut découper chaque phrase en mots.

Exercice 2.1 *Écrivez un programme qui prenne en entrée une phrase, la découpe en mots et fournisse en sortie une liste des mots. On suppose que les séparateurs de mots*

sont des espaces, des passages à la ligne, des tabulations, des apostrophes ou des signes de ponctuation. Contrairement aux autres séparateurs, les signes de ponctuation apparaissent comme des mots à part entière et les apostrophes sont rattachées au mot qui les précède. Ainsi par exemple, si l'on fournit en entrée la phrase suivante " Jean, dans l'après-midi, vient-t-il jouer ?"*, on doit obtenir en sortie la liste* ["Jean", ",", "dans", "l'", "après-midi", ",", "vient-t-il", "jouer", " ?"].

A la fin du programme, vous choisirez 5 phrases soigneusement pour tester celui-ci.

Dans l'exercice ci-dessus, l'analyse du tiret comme trait d'union est très grossière. On ne prend pas en compte qu'un verbe suivi d'un trait d'union et d'un pronom personnel doit être découpé en deux mots. Par exemple, *prend-il* doit se découper en *prend* et *il*. On ne prend pas en compte non plus qu'un verbe terminé par une voyelle suivi par "t" encadré de deux traits d'union et par un pronom personnel doit être découpé en deux mots avec effacement de "t". Par exemple, *marche-t-il* doit se découper en *marche* et *il*.

Exercice 2.2 *Ecrivez un programme qui prend en entrée les sorties du programme de l'exercice précédent et qui effectue un traitement des tirets de la façon indiquée ci-dessus.*

A la fin du programme, vous choisirez 5 phrases soigneusement pour tester celui-ci.

Une fois qu'une phrase est découpée en mots, il est encore nécessaire de lui faire subir un traitement phonologique pour obtenir des mots qui puissent être retrouvés dans un lexique

Exercice 2.3 *Dans une phrase découpée en mots, voici quelques règles de transformations phonologiques qui peuvent lui être appliquées :*

au	$\longrightarrow$	*à le*
du	$\longrightarrow$	*de le*
cet	$\longrightarrow$	*ce*
qu'	$\longrightarrow$	*que*
m'	$\longrightarrow$	*me*
t'	$\longrightarrow$	*te*
c'	$\longrightarrow$	*ce*
d'	$\longrightarrow$	*de*
s'	$\longrightarrow$	*se*
l'	$\longrightarrow$	*le|la*

Ces règles ne sont pas exhaustives. Ecrivez un programme qui prenne en entrée la sortie du programme de l'exercice précédent et qui transforme un maximum de mots selon des règles semblables à celles exposées ci-dessus afin d'obtenir en sortie une liste de mots qu'on puisse trouver dans un lexique des mots fléchis du français. La sortie du programme doit être une liste de mots.

A la fin du programme, vous choisirez 5 phrases soigneusement pour tester celui-ci.

3 Génération de phrases

En traitement automatique des langues, à l'opposé de l'analyse, on peut générer des textes à partir d'une information non textuelle. Une phase intermédiaire produit en général des phrases sous forme de suites de mots fléchis. Il faut alors transformer ces suites en énoncés phonologiquement acceptables.

Exercice 3.1 *Ecrivez un programme qui prenne en entrée une liste qui est une suite de mots fléchis et qui affiche en sortie une phrase phonologiquement acceptable. Par exemple, si on donne en entre la suite* ["le", "homme", "à", "le", "chapeau", ",", "que", "on", "aperçoit", "à", "le", "aube", "sous", "le", "hêtre", ",", "se", "invitera", "il", "aujourd'hui", " ?"]*, le programme doit afficher* "l'homme au chapeau, qu'on aperçoit à l'aube sous le hêtre, s'invitera-t-il aujourd'hui ?".

A la fin du programme, vous choisirez 5 phrases soigneusement pour tester celui-ci.