Traitement automatique du langage naturel(NLP)
Série d'exercices

27 février 2022

Objet du TD : TALN : expressions régulières, analyse lexicale, analyse syntaxique, analyse sémantique. Analyse de sentiments.

---

1 Expressions régulières

1. Définir une expression régulière décrivant les noms d'utilisateurs du réseau social Twitter. Ces derniers se composent de 4 à 15 caractères (uniquement des lettres, des chiffres ou des traits de soulignement) précédés par le caractère @. Écrire un code Python permettant d'extraire tous les noms d'utilisateurs des tweets du module *nltk*.
2. Définir une expression régulière décrivant les hashtags de Twitter en supposant dans un premier temps que ces hashtags ne comportent que des des lettres, des chiffres ou des traits de soulignement, puis en supposant qu'ils peuvent contenir tous les caractères sauf les signes de ponctuations et les espaces et les caractères réservés.É crire un code Python permettant de :
   (a) extraire et comparer des tweets de *nltk* les hashtags décrits par les deux expressions régulières,
   (b) compter les nombre d'occurences de chaque hashtag et afficher les 10 hashtags les plus populaires.
3. Définir une expression régulière décrivant les URL (voir annexe A) et appliquez la aux données du fichier *url4regs.csv*.

2 Analyse lexicale

Ecrire un code python permettant de :

1. tokeniser et "nettoyer" un texte,
2. associer à chaque mot son tag POS, puis lister les mots par tag.
3. Afficher la description d'un tag ou d'une famille de tags (aide Python),
4. lemmatiser les mots du texte.

On appliquera ce traitement, par exemple, au texte du fichier *theguardian.txt*.

3 Une grammaire hors-contexte (CFG) pour le langage naturel

On considère la grammaire définie par les règles de production suivantes :

S	$\longrightarrow$	si S alors S.
S	$\longrightarrow$	soit S soit S.
S	$\longrightarrow$	NP VP.
NP	$\longrightarrow$	Det N.
NP	$\longrightarrow$	PN.
NP	$\longrightarrow$	NP Conj NP.
Det	$\longrightarrow$	un une le la.
VP	$\longrightarrow$	V NP.
VP	$\longrightarrow$	V PP.
PP	$\longrightarrow$	Prep NP.
N	$\longrightarrow$	garçon|fille|chien|chat|nourriture|restaurant|maison|glace|cantine.
V	$\longrightarrow$	mange|aime|monte|va.
PN	$\longrightarrow$	il|elle.
Prep	$\longrightarrow$	à| au.
Conj	$\longrightarrow$	et|ou.

1. Vérifiez que cette grammaire est hors contexte.
2. Recensez les symbôles terminaux et les symbôles non terminaux de cette grammaire.
3. Vérifiez que les phrases suivantes sont reconnues par la grammaire et donnez leurs arbre d'analyse syntaxique.
   (a) Si la fille a faim alors elle va au restaurant.
   (b) Soit il mange à la maison soit il va au restaurant.
   (c) Si le chat a faim alors soit il mange à la maison soit il va au restaurant.

4 Ambiguité(s) du langage naturel

On considère la phrase "Time flies like an arrow' dont Chomski affirmait qu'elle avait 4 sens possibles (d'après le référence [1]).

1. Quel(s) type(s) d'ambiguité rencontrons-nous dans cette phrase ?
2. Construisez les arbres syntaxiques correspondants aux 4 sens évoqués par Chomsky.

5 Un classifieur bayésien pour l'analyse des sentiments

Dans cet exercice, notre objectif est de créer un classifieur bayésien capable de classer des tweets en deux classes : Positif et Négatif. Pour ce faire, nous avons besoin de :

1. Récupérer les tweets inclus dans le module NLTK. On remarquera que ces tweets sont déjà répartis en deux ensembles (tweets positifs et tweets négatifs).
2. Tokenizer ces tweets.
3. Nettoyer les tweets.
4. Créer et tester le classifieur.

Références

[1] I. Tellier. Introduction au TALN et à l'ingénierie linguistique. Université Lille 3. Année non précisée, 104 pages.

A Description d'une URL

La description des URL que nous donnons ici est une description simplifiée[1]. Une URL ($url1 = https : //partage.cyu.fr/?loginOp = logout$ et $url2 = https : //mail.google.com/mail/u/0/\#inbox$) se compose des éléments suivants :

1. un **schéma** (http ou https) suivi de : et de //.

1. On peut trouver une description plus complète des URL dans les RFC 3986 et 1738 les décrivant, ainsi que dans les articles *Wikipedia* qui leur sont consacrés.

2. un **nom de domaine** qui se présente sous la forme *www.dom1.dom2.....dom* (exemple : *www.boursier.com*) ou *dom1.dom2.....dom* (exemple *journal.lemonde.fr*). On supposera que chacune des chaînes *domi* comporte des caractères alphanumériques et le caractère et que la chaîne *dom* ne comporte que des lettres [2],
3. un **chemin** se présentant sous la forme d'une suite éventuellement vide de chaînes de caractères précédées par le caractère '/' (exemple : $/mail/u/0$). Ces chaînes peuvent contenir tous les caractères à l'exception des cacartères réservés ou interdits.
4. une **requête** optionnelle qui commence par le caractère ? éventuellement précédé par le caractère / (exemple : $?loginOp = logout$) et pouvant comporter tous les caractères à l'exception de ceux réservés ou interdits.
5. un **fragment** optionnel qui commence par le caractère # éventuellement précédé par le caractère / (exemple : $\#inbox$) et pouvant comporter tous les caractères à l'exception de ceux réservés ou interdits.

2. Cette composante d'une URL comporte d'autres éléments servant notamment à l'authentification que nous ignorons ici.