Université Constantine2 – Abdelhamid Mehri

Faculté des Nouvelles Technologies de l'Information et de la Communication
Département Informatique Fondamentale et ses Applications

Module : WANLP | M1-SDIA

Solution du TP 02

Solution de l'exercice 1 :

```
import nltk
from nltk import word_tokenize, pos_tag

# Assurez-vous d'avoir téléchargé les ressources nécessaires avec
nltk.download()
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')

# Phrase donnée
phrase = "La JS Kabylie a battu le MC Alger dans un match passionnant au
stade du 1er Novembre."

# Tokenisation de la phrase
tokens = word_tokenize(phrase, language='french')

# Obtention des parties du discours
tags = pos_tag(tokens)

# Affichage des mots avec leurs parties du discours
print(tags)
```

Explication du code :

- Ce script commence par importer les modules nécessaires de NLTK, télécharge les ressources requises pour la tokenisation et l'étiquetage par parties du discours, puis traite la phrase donnée. Il tokenise d'abord la phrase, applique ensuite l'étiquetage par parties du discours, et affiche finalement les mots avec leurs parties du discours correspondantes.
- Notez que l'exemple utilise **language='french'** dans **word_tokenize** pour spécifier la langue de la tokenisation. Cependant, **pos_tag** dans **NLTK** est optimisé pour l'anglais. Pour une analyse précise en français, il est recommandé d'utiliser des outils ou des ressources spécifiques au français, comme ceux disponibles dans la bibliothèque spaCy par exemple.
- Le module **Punkt** est un tokenizeur de phrases qui s'appuie sur un **algorithme non supervisé** pour apprendre les abréviations et les propriétés des points de fin de phrase dans un texte donné. Il est ensuite capable d'identifier les limites des phrases avec une bonne précision dans de nombreux langages.
- Le **'averaged_perceptron_tagger'** fait référence à un modèle pour l'étiquetage des parties du discours (Part-Of-Speech tagging, POS tagging) utilisé par la bibliothèque NLTK. Ce modèle utilise un algorithme de perceptron moyen (Deep Learning) pour prédire les étiquettes grammaticales de chaque mot dans une phrase, telles que nom, verbe, adjectif, etc.

Résultat de l'exécution :

```
[('La', 'NNP'), ('JS', 'NNP'), ('Kabylie', 'NNP'), ('a', 'DT'), ('battu',
'NN'), ('le', 'NN'), ('MC', 'NNP'), ('Alger', 'NNP'), ('dans', 'VBZ'), ('un',
'JJ'), ('match', 'NN'), ('passionnant', 'NN'), ('au', 'NN'), ('stade',
'VBD'), ('du', 'JJ'), ('1er', 'CD'), ('Novembre', 'NNP'), ('.', '.')]
```

Significations des étiquettes :
Les tags les plus courantes utilisées dans l'analyse syntaxique selon la convention du **Penn Treebank**, qui est une référence standard pour **l'anglais** sont :

- CC: Coordinating conjunction (Conjonction de coordination)
- CD: Cardinal number (Nombre cardinal)
- DT: Determiner (Déterminant)
- EX: Existential there (Il existentiel)
- FW: Foreign word (Mot étranger)
- IN: Preposition or subordinating conjunction (Préposition ou conjonction subordonnée)
- JJ: Adjective (Adjectif)
- JJR: Adjective, comparative (Adjectif comparatif)
- JJS: Adjective, superlative (Adjectif superlatif)
- LS: List item marker (Marqueur d'élément de liste)
- MD: Modal (Modal)
- NN: Noun, singular or mass (Nom, singulier ou masse)
- NNS: Noun, plural (Nom, pluriel)
- NNP: Proper noun, singular (Nom propre, singulier)
- NNPS: Proper noun, plural (Nom propre, pluriel)
- PDT: Predeterminer (Prédéterminant)
- POS: Possessive ending ('s)
- PRP: Personal pronoun (Pronom personnel)
- PRP$: Possessive pronoun (Pronom possessif)
- RB: Adverb (Adverbe)
- RBR: Adverb, comparative (Adverbe comparatif)
- RBS: Adverb, superlative (Adverbe superlatif)
- RP: Particle (Particule)
- SYM: Symbol (Symbole)
- TO: to (à)
- UH: Interjection (Interjection)
- VB: Verb, base form (Verbe, forme de base)
- VBD: Verb, past tense (Verbe, passé)
- VBG: Verb, gerund or present participle (Verbe, gérondif ou participe présent)
- VBN: Verb, past participle (Verbe, participe passé)
- VBP: Verb, non-3rd person singular present (Verbe, présent non 3e personne du singulier)
- VBZ: Verb, 3rd person singular present (Verbe, 3e personne du singulier présent)
- WDT: Wh-determiner (Déterminant wh)
- WP: Wh-pronoun (Pronom wh)
- WP$: Possessive wh-pronoun (Pronom possessif wh)
- WRB: Wh-adverb (Adverbe wh)

Solution de l'exercice 2 :

```
import nltk
from nltk import word_tokenize, pos_tag, ne_chunk

# Téléchargement des ressources nécessaires, si ce n'est déjà fait
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
nltk.download('maxent ne chunker')
nltk.download('words')

# Phrase à analyser
phrase = "Le mois du Ramadan en 2024, débutant en mars, verra les habitants
de Constantine participer à des veillées nocturnes, des séances de prière à
la Grande Mosquée Emir Abdelkader, et à des actions de bienfaisance envers
les plus démunis."

# Tokenisation et étiquetage par parties du discours
tokens = word_tokenize(phrase)
tags = pos_tag(tokens)

# Reconnaissance d'entités nommées
entites = ne_chunk(tags)

# Affichage des entités reconnues
print(entites)
```

Explication du code :

- Ce script effectuera les opérations suivantes :
  1. Tokeniser la phrase en mots individuels.
  2. Étiqueter chaque token avec sa partie du discours (POS tagging).
  3. Identifier les entités nommées dans la phrase et les catégoriser (par exemple, comme personnes, organisations, ou lieux).
- En raison du fait que NLTK est principalement conçu pour l'anglais, les résultats exacts de la reconnaissance d'entités nommées pour une phrase en français ne peuvent pas être garantis ici. Pour des analyses en français, considérez l'utilisation de bibliothèques comme spaCy qui offrent un support multilingue, y compris pour la reconnaissance d'entités nommées en français.
- Le modèle **'maxent_ne_chunker'** utilisé par NLTK pour la reconnaissance d'entités nommées. Il est basé sur un **classificateur d'entropie maximale** (Maximum Entropy), qui est une **approche statistique** pour **prédire l'étiquette** ou la catégorie d'un élément en se basant sur les caractéristiques d'entrée.
- **'words'** est un ensemble de données de NLTK qui contient une liste de mots. Cette ressource est souvent utilisée en conjonction avec des outils du NLP, comme le (**maxent_ne_chunker**) ou les tokeniseurs, pour aider à identifier correctement les mots dans un texte et faciliter des tâches telles que la reconnaissance d'entités nommées, la tokenisation, et l'étiquetage par parties du discours (POS tagging).

Résultat de l'exécution :

- Date/temps : "2024", "mars"
- Location : "Constantine", "Grande Mosquée Emir Abdelkader"
- Événement : "Ramadan", "veillées nocturnes", "séances de prière", "actions de bienfaisance"

Signification des types d'entités nommées :

Les types d'entités nommées les plus couramment identifiées par NLTK lors de l'utilisation de sa fonction de reconnaissance d'entités nommées (Named Entity Recognition, NER) :

- ORGANIZATION : Entités représentant des organisations, des institutions, des entreprises, des agences gouvernementales, etc.
- PERSON : Noms de personnes.
- LOCATION : Noms de lieux géographiques comme les villes, les pays, les rivières, les montagnes, etc.
- DATE : Dates, périodes, ères.
- TIME : Heures de la journée.
- MONEY : Quantités monétaires.
- PERCENT : Valeurs en pourcentage.
- FACILITY : Infrastructures telles que les aéroports, les stades, les ponts, les routes.
- GPE (Geo-Political Entity) : Entités géopolitiques telles que les noms de pays, villes, États.