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

Enoncés du TP 03

```
import nltk
from nltk.tokenize import word_tokenize
from nltk.corpus import stopwords
import re

# Téléchargement des ressources nécessaires de NLTK
nltk.download('punkt')
nltk.download('stopwords')
nltk.download('averaged_perceptron_tagger')
nltk.download('maxent_ne_chunker')
nltk.download('words')

# Texte original pour l'exercice
text = ("En cette belle journée ensoleillée du 5 juillet, la fête de
l'Indépendance de l'Algérie a été célébrée avec éclat sur la Place
des Martyrs : c'est-à-dire concerts par l'Orchestre Symphonique
National, feux d'artifice au-dessus de la Baie d'Alger, et discours
patriotiques par le Président Abdelmadjid Tebboune. Détails sur notre
page Facebook 'Festivités-DZ'! Pour plus d'informations, visitez-nous
sur: www.festivités-dz.com. #FêteNationale #1erNovembre")

# Étape 1: Tokenisation
tokens = word_tokenize(text, language='french')
print("Étape 1 - Tokenisation:", tokens)

# Étape 2: Normalisation
# Conversion en minuscules et suppression de la ponctuation
normalized_tokens = [token.lower() for token in tokens if
token.isalpha()]
print("Étape 2 - Normalisation:", normalized_tokens)

# Étape 3: Suppression du Bruit
# Suppression des stop words (et normalement des URLs et hashtags,
nécessite des expressions régulières)
filtered_tokens = [word for word in normalized_tokens if word not in
stopwords.words('french')]
print("Étape 3 - Suppression du bruit:", filtered_tokens)

# Note: Pour le traitement des mots concaténés, des guillemets et des
contractions comme "c'est-à-dire", une approche plus avancée serait
nécessaire,
# impliquant potentiellement des expressions régulières et une
logique personnalisée.
```

```
# Étape 4: Analyse des Parties du Discours (POS)
# Cette étape requiert une conversion des tokens pour le français ou l'utilisation d'un outil adapté au français, car NLTK est optimisé pour l'anglais.
pos_tags = nltk.pos_tag(filtered_tokens)
print("Étape 4 - Analyse POS:", pos_tags)

# Étape 5: Reconnaissance d'Entités Nommées (NER)
# Comme pour l'analyse POS, la reconnaissance d'entités nommées avec NLTK est optimisée pour l'anglais.
ner_result = nltk.ne_chunk(pos_tags)
print("Étape 5 - NER:", ner_result)

# Remarque: Les étapes 4 et 5 nécessitent des adaptations ou des ressources supplémentaires pour être pleinement fonctionnelles en français.
```