**Cours :** Traitement Automatique du Langage Naturel

**Promo :** Master 2 Polytech

**Auteur :** Amir HAZEM

**Date :** **7** Novembre 2018

# TP2 : Classifieurs

Dans ce second TP nous nous intéresserons aux classifieurs type SVM, Naïve Bays, etc. Le but étant de pouvoir caractériser les tweets par des traits (features) afin de les classer selon un ensemble d'étiquettes (positif, négatif, neutre, figuratif, etc.).

## Exercice 1 (Prise en main des classifieurs) :

Récupérez le fichier *run_classifiers_sample.py (*http://www.amirhazem.ovh/teaching.html).

Ce fichier contient des exemples de classifieurs.

Récupérez aussi, les fichiers *train-tmp.csv* et *dev-tmp.csv* afin de pouvoir tester les différents classifieurs.

Exécutez le script run_classifiers_sample.py et testez les différents classifieurs pour constater le score d'exactitude (Accuracy) renvoyé par chaque classifieur sur le fichier dev-tmp.csv.

## Exercice 2 (Application à Deft 2017) :

Dans cet exercice vous devez dans un premier temps réfléchir aux traits qui permettront de discriminer la nature des tweets pour ensuite entraîner les classifieurs tels que : Naïve Bays, SVM RandomForest, Maximum Entropy, MLP, etc. Appuyez-vous sur les traits extraits lors du premier Tp. Les traits peuvent être par exemple la longueur des tweets, la présence ou non d'émoticônes, la présence d'un vocabulaire positif, la présence d'un vocabulaire négatif, etc.

### 2-1 Extraction des traits

À l'aide des fonctions implémentées lors du premier Tp, extraire pour chacune des trois tâches les traits suivants :

- la longueur du tweet en nombre de mots
- la longueur du tweet en nombre de caractères
- la présence de hashtags

- la présence d’arobases
- la présence d’émoticônes positives :) :-) :D =) :’) :o) :P >:) :”> >:| <3 ;> ;)  ;-)  ;>,(:  (;
- la présence d’émoticônes négatives  :( : ( :-( :’( :/ :< ;(
- ...

**2-2 Préparation des données**

Une fois les traits choisis, représentez-les dans un fichier .csv.
Chaque colonne correspondra à l’ensemble des valeurs pour chaque trait.  La dernière valeur sera la classe ou l’étiquette de la tâche traitée.

Exemple :
1,103,30,38,83,43.3,0.183,33,0
1,115,70,30,96,34.6,0.529,32,1
3,126,88,41,235,39.3,0.704,27,0
8,99,84,0,0,35.4,0.388,50,0
les 8 premières valeurs correspondent à vos traits et la dernière valeur correspond à la classe 0 ou 1 (0 : positive / 1 : négative)

Faire varier les traits et comparer les classifieurs entre eux. Vous avez à disposition un corpus de développement pour tester les performances de vos classifieurs sur les trois tâches ici http://www.amirhazem.ovh/teaching.html.