

UNIVERSITE MOULAY IMAIL

FACULTE DES SCIENCES ET TECHNIQUES

**Département :** Informatique

**Filière :** Génie Logiciel

RAPPORT DU PROJET DE FIN D’ETUDES :

**CLASSIFICATION DES MESSAGES SPAM ET**

**NON-SPAM (HAM)**

**Réalisé par : Encadré par :**

* - OUZIZI OTMANE -Dr. AZROUR Mourade
* - OUNACHAD WALID
* - ELAADRAOUI YASSINE

# *Membres Jury :*

* - ……………………….
* -………………………..
* -……………………….. ANNEE UNIVERSITAIRE : 2021/2022

Table des matières

[*Membres Jury :* 1](#_Toc105267283)

[*Remerciements :* 3](#_Toc105267284)

[*Dédicace* 4](#_Toc105267285)

[*Résumé* 5](#_Toc105267286)

[*Liste des abréviations* 6](#_Toc105267287)

[*Liste des tableaux* 7](#_Toc105267288)

[*Liste des figures* 8](#_Toc105267289)

[*Introduction Générale* 10](#_Toc105267290)

[*Chapitre I : Cadre général de projet* 11](#_Toc105267291)

[I.1. Introduction 11](#_Toc105267292)

[I.2. Présentation du projet 11](#_Toc105267293)

[I.3. Spécification des Besoins 11](#_Toc105267294)

[I.4. Problématique 12](#_Toc105267295)

[I.5. Objectifs 12](#_Toc105267296)

[I.6. Conclusion 12](#_Toc105267297)

[Chapitre II : Méthodes d’apprentissage automatique 14](#_Toc105267298)

[II.1. I**ntroduction** 14](#_Toc105267299)

[II.2. Apprentissage supervisé 14](#_Toc105267300)

[II.2.1. Réseau neural network (LSTM) 14](#_Toc105267301)

[II.2.2. Machine à vecteurs de support (SVM) 17](#_Toc105267302)

[II.2.3. Formules mathématiques de maximisation : 19](#_Toc105267303)

[II.3. Apprentissage non supervisé : 19](#_Toc105267304)

[II.3. K-moyennes (K-Means) : 20](#_Toc105267305)

[II.4. Conclusion : 21](#_Toc105267306)

[*Chapitre III : Implémentation et réalisation du projet* 22](#_Toc105267307)

[III.2- Technologies et outils de développement 22](#_Toc105267308)

[*Conclusion générale :* 34](#_Toc105267309)

[*Références* 35](#_Toc105267310)

# *Remerciements :*

***Nous tenons à exprimer toute nos reconnaissances à notre professeur AZROUR Mourade. Nous le remercions de nous avoir encadré, orienté, aidé et conseillé.***

***Nous présentons nos sincères remerciements à tous nos enseignants de la Faculté de Sciences et Techniques Errachidia.***

***Nos profonds remerciements vont aux membres de jury qui ont accepté d’évaluer ce travail.***

***Nous tenons à exprimer également notre gratitude et nos remerciements à nos amis, nos familles et toute personne ayant contribué à la réalisation de ce travail.***

# *Dédicace*

Nous dédions ce modeste travail à ceux qui ont sacrifié leur vie pour la nôtre comme signe d’amour et d’espoir.

A nos très chers parents qui n’ont jamais cessé de prier Dieu pour qu’il nous protège, sans vos efforts et soutiens, ce travail n’aura jamais eu d’existence.

A nos très chers frères et sœurs pour leur soutien.

# *Résumé*

De nos jours, la diffusion de l’information dans les réseaux sociaux prend une grande importance dans la vie de la personne, face à l’augmentation des données, la plupart de celle-ci sont soit des spams soit des messages dupliques, ce qui montre l’accroissement des utilisateurs malveillantes dans le web et dans les réseaux sociaux, le besoin en techniques automatisées, capable d’analyser des données afin de détecter les ressources malveillantes, est devenu nécessaire.

Dans ce projet notre démarche a été, d’abord, de chercher à mieux comprendre le processus de construction d’un classifieur automatique en interprétant premièrement les méthodes d’apprentissages automatique et les algorithmes correspondant à chaque méthode. Puis, on passe à la partie de l’implémentation et la réalisation du projet, en spécifiant tous les technologies et les outils de développement utilisées. Et finalement la présentation de projet qui se représente par l’architecture et le prétraitement des messages afin de la classifier.

# *Liste des abréviations*

*TF : Term frequency.*

*IDF : Inverse Document Frequency.*

*TF-IDF : Terme frequency Invers Document Frequency.*

*SVM : Support Vecteur Machine.*

*LSTM : Long short-term memory*

*NLP : Natural Language Processing*

*RNN : Reccurent Neural Networks*

# *Liste des tableaux*

[Tableau 1 : Rapport comparatif des classifieurs avec pré-traitement 37](#_Toc104969013)

# *Liste des figures*

[Figure 1 : Importation des librairies nécessaires 15](#_Toc104968870)

[Figure 2 : Processus de travail de LSTM 18](#_Toc104968871)

[Figure 3 :Fonctionnement des réseaux de neurones artificiel 19](#_Toc104968872)

[Figure 4 : Fonctionnement des réseaux de neurones récurrent 19](#_Toc104968873)

*Figure 1 : Importation des librairies nécessaires*

*Figure 2 : Processus de travail de LSTM*

*Figure 2-1: Fonctionnement des réseaux de neurones artificiel*

*Figure 2-2: Fonctionnement des réseaux de neurones récurrent*

*Figure 3: le fonctionnement de classifieur SVM*

*Figure 4: Processus de classification selon machine a vecteur de support*

*Figure 5: Principe de l’algorithme K-Means*

*Figure 6: Algorithme de K-means*

*Figure 7-1: Texte avant la normalisation*

*Figure 7-2: Texte après la normalisation*

*Figure 8: Extrait de la partie de la tokenisation*

*Figure 9: Extrait de la partie du stopwords*

*Figure 10: Extrait de la partie de la racinisation*

*Figure 11: Extrait de la partie de la lemmatisation*

*Figure 12: Extrait du rôle de la méthode TF-IDF*

*Figure 13: démarche de la catégorisation de textes*

*Figure 14: Rapport du classifieur de SVM*

*Figure 15: Rapport du classifieur de K-Means*

*Figure 16: Rapport du classifieur de LSTM*

# *Introduction Générale*

Avec le développement de l'internet et de l'internet mobile, les courriers électroniques et les messages instantanés sont devenus les médias plus pratiques pour notre communication quotidienne.

Cependant, les spams, généralement définis comme des courriers électroniques ou des messages non sollicités, sont considérés comme un défi de plus en plus sérieux pour le fondement de l'internet et ont gravement affecté la communication normale des personnes, tant sur le lieu de travail que dans la vie privée. Selon les statistiques de l'UIT (Union internationale des télécommunications), environ 72 à 88 % des courriels actuellement présents sur l'internet sont des spams. Les nombreux spams alertent également la sécurité des systèmes informatiques en réseau lorsqu'ils sont utilisés comme vecteurs de virus et de codes malveillants. La détection et le traitement du spam prennent beaucoup de temps, ce qui réduit considérablement la productivité.

Il existe de nombreux travaux qui traitent le problème de filtrage de spam en utilisant des méthodes d’apprentissage automatique. Le filtrage de spam basé sur le contenu textuel des messages peut être considéré comme un exemple de classification de textes qui consiste en l'attribution de documents textuels à un ensemble de classes prédéfinies. Le but d'un système de classification est de réaliser la tâche de classification et de le faire avec un degré raisonnable d'exactitude. Il existe aujourd'hui une liste plutôt longue de classifieurs développés autour des algorithmes.

On s’est basé dans notre travail sur trois algorithmes d’apprentissage : Machine à Vecteur de support (SVM),Mémoire longue à court terme (LSTM), K-Moyennes (K-Means) pour objectif de sélectionner le meilleur algorithme pour classifier le message ou bien le courriel reçu.

# *Chapitre I : Cadre général de projet*

## Introduction

Les messages de spam sont des messages envoyés à un grand groupe de destinataires sans leur consentement préalable, généralement des publicités pour des biens et services ou des opportunités commerciales. Aujourd’hui, le pourcentage de messages de tricherie parmi les spams a fortement augmenté. Les messages de tricherie incitent généralement les gens à donner de l'argent ou des informations personnelles en proposant une offre plaisante ou fausse.

En d’autre part, la classification du spam est une étape vers la création d’un outil permettant d’identifier les messages tricheurs et de détecter rapidement les messages tricheurs.

## Présentation du projet

A travers notre projet, on a essayé de réaliser toutes les étapes nécessaires pour développer un classifieur intelligent des messages, selon les étapes suivantes :

* 1ère étape : cette étape est consacrée aux collections des documents d'apprentissage
* 2ème étape : c’est une étape pour le prétraitement des données collectés. Elle est considérée comme une étape plus importante dans le Traitement en Langage Naturel (Natural Language Processing (NLP) )
* 3ème étape : pour la construction de modèle d’apprentissage automatique
* 4ème étape : pour l’évaluation, c’est là où on évalue notre model en utilisant la matrice de confusion

## Spécification des Besoins

Les applications concrètes du Machine Learning sont de plus en plus nombreuses. Parmi les plus populaires, on trouve par exemple la classification texte. Alors pour construire un bon modèle de Machine Learning, on peut suivre six étapes clés qui résultent de ces expériences :

* Comprendre et identifier le problème du métier
* Comprendre et identifier les données
* Collecter et préparer les données
* Déterminer les attributs du modèle et entraîner le modèle
* Evaluer les performances du modèle

## Problématique

Le phénomène du spam ne cesse de croître ainsi que les victimes de ce vrai désastre. Pareillement pour les entreprises et pour les particuliers, les dangers du spam sont bien présents et il est important de connaître l’impact de ces derniers :

**Risques de sécurité**: Les spammeurs peuvent amener les utilisateurs à ouvrir des messages ou des pièces jointes contenant des logiciels malveillants, créant ainsi une brèche dans la sécurité de l’appareil.

**Atteinte à la vie privée** : Les messages spam sont capables de faire des captures d’écran, de connaître la localisation géographique de la machine… car le spam en lui-même n’est pas considéré comme un malware.

Et il existe de plusieurs autres risques posés par les messages spams.

## Objectifs

Dans cette recherche, nous évoquons les études les plus récentes dans le domaine de l’apprentissage automatique appliqué pour la classification des messages spams.

L'objectif de notre travail consiste premièrement, à étudier et comparer un ensemble de méthodes d'apprentissage automatique que nous avons découvert pendant notre étude.

Deuxièmement, notre objectif est de préparer les données en faisant l'extraction de nouvelles caractéristiques pour traiter cette problématique.

Finalement, nous réaliserons des expériences en utilisant différentes caractéristiques de spam pour comparer les méthodes d'apprentissage automatique, suivis d'une discussion des résultats obtenus.

## Conclusion

La classification de textes est une tâche de NLP très importante au vu de toutes les applications qui peuvent en découler. Il existe plusieurs approches possibles pour la réalisation de classifieurs. Même si les modèles à base de Machine Learning sont en général plus performants que les moteurs à règles, il peut être intéressant de réaliser un classifieur mixte. Pour finir, il est important de choisir les métriques d’évaluation qui conviennent le mieux en fonction du contexte global dans lequel on se trouve, afin que la mesure de la performance du modèle ait du sens.

# Chapitre II : Méthodes d’apprentissage automatique

1. I**ntroduction**

L’apprentissage automatique est une branche de l’intelligence artificielle (IA) et de l’informatique qui utilise principalement des données et des algorithmes pour affecter la manière dont les êtres humains apprennent, en améliorant progressivement sa précision.

Dans l’apprentissage automatique, on peut distinguer entre trois types d’apprentissage automatiques : Apprentissage supervisé, apprentissage non supervisé et apprentissage de renforcement. Dans notre projet, on va se concentrer sur l’apprentissage supervisé et l’apprentissage non supervisé.

1. Apprentissage supervisé

L’apprentissage supervisé s’intéresse aux données déjà classer, et les algorithmes de ce type d’apprentissage a pour but de prédire la classe d’une donnée étiquetée basant sur la connaissance fournie par le pré-observations et les classes.

L’apprentissage supervisé est classé en deux catégories d’algorithmes :

* Classification : Un problème de classification se pose lorsque la variable de sortie est une catégorie, telle que « Rouge » ou « Bleu », « maladie » ou « pas de maladie ».
* Régression : Un problème de régression se produit lorsque la variable de sortie est une valeur réelle, telle que « dollars » ou « poids ».

Dans notre cas on s’intéresse juste à la classification.

Il existe plusieurs algorithmes et techniques utilisés pour la classification supervisée des spam, on peut citer :

1. Réseau neural network (LSTM)

Avant d’entamer aux réseaux des neurones, on propose le schéma suivant qui explique la hiérarchie où se situe les réseaux de neurones.

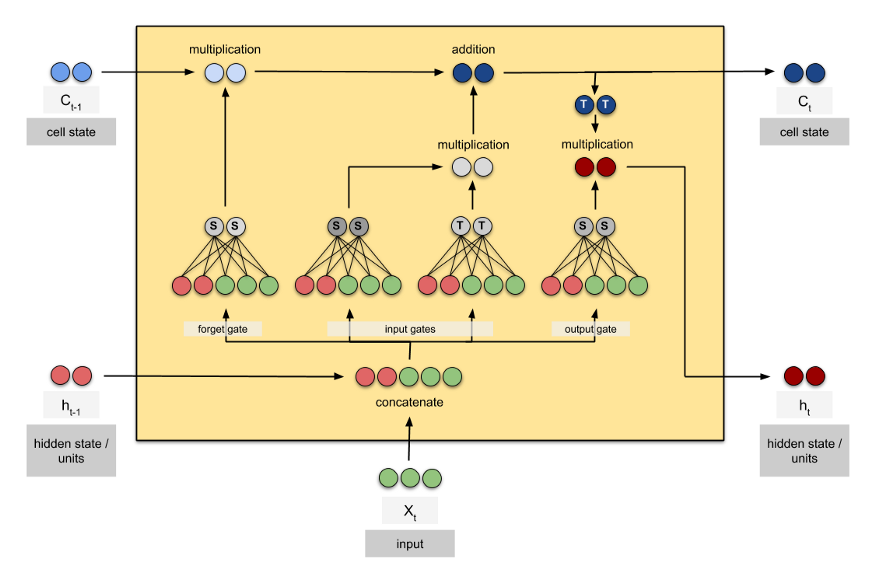


Figure 2 : Processus de travail de LSTM

LSTM est un type des réseaux neurones récurent (RNN), donc pour bien comprendre LSTM, il faut d’abord comprendre les réseaux de neurones et les réseaux de neurones récurrents.

**Un réseau de neurones artificiels** est une séquence des couches de neurones connectés, inspirée des réseaux de neurones biologiques. Ce n’est pas un algorithme mais des combinaisons de divers algorithmes qui nous permettent de faire des opérations complexes sur des données. Parmi les types des réseaux de neurones, on trouve les réseaux de neurones récurrents.

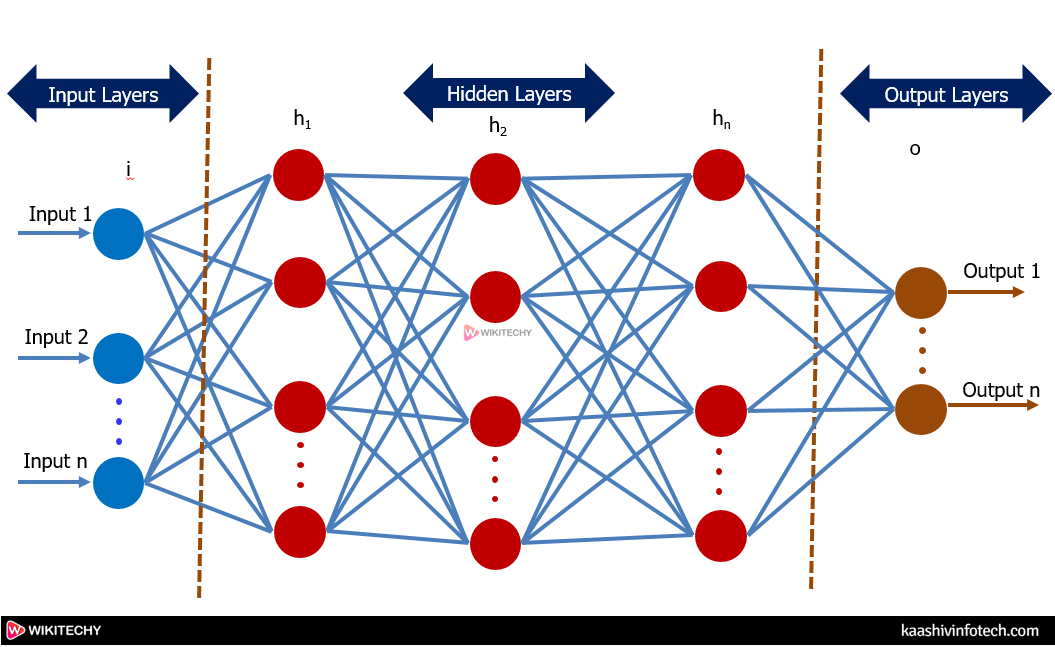


Figure  :Fonctionnement des réseaux de neurones artificiel

**Réseaux de neurones récurrents :** On peut défini ce type des réseaux comme une classe de réseaux neuronaux adaptés pour traiter les données temporelles. Les neurones de RNN ont un état cellulaire / mémoire, et l’entrée est traitée selon cet état interne, qui est obtenu à l’aide de boucles avec dans le réseau neuronal. Il existe des modules récurrents de couches « tanh » dans les RNN qui leur permettent de conserver des informations. Cependant, pas avant longtemps, c’est pourquoi nous avons besoin de modèles LSTM.

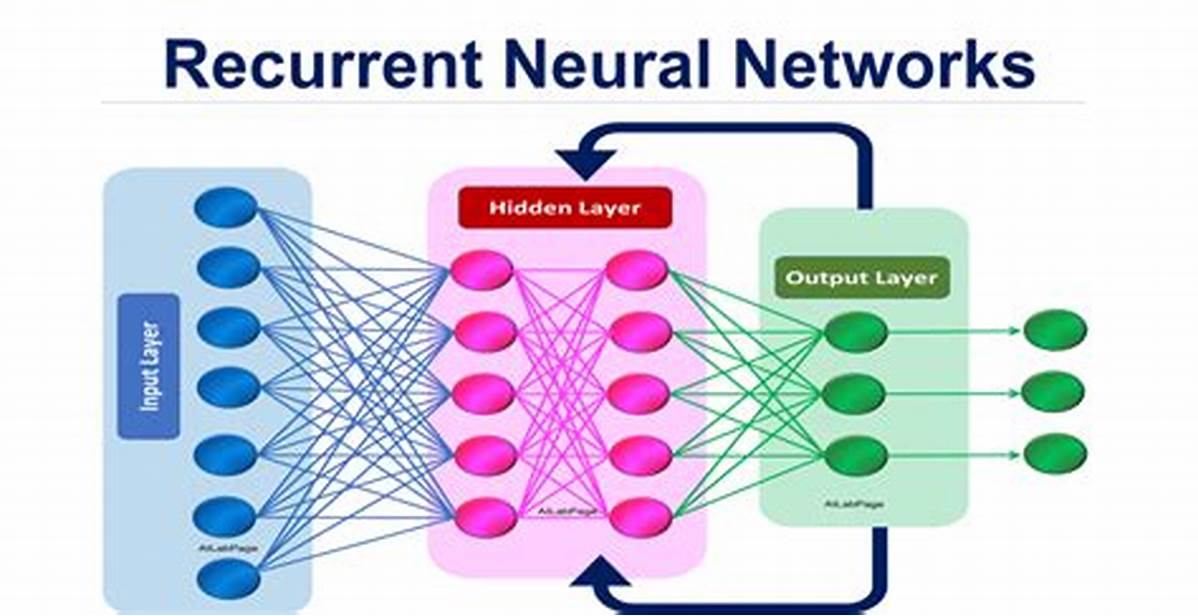


Figure  : Fonctionnement des réseaux de neurones récurrent

**Mémoire longue à court terme (LSTM)**

Actuellement, après qu’on a défini les réseaux de neurones et les réseaux de neurones récurrents, on peut donner une définition au LSTM.

LSTM est un type particulier de réseau neurones récurrent qui est capable d’apprendre les dépendances à long terme dans les données, cette caractéristique qui distingue ce type des réseaux au réseau récurrent.

1. Machine à vecteurs de support (SVM)

Machine a vecteurs de support également nommée séparateurs a vaste marge, elle a été introduite par Vapnik en 1995, ce dernier s’est inspire des travaux sur les hyperplans à marge maximale et sur ceux des fonctions noyaux, et des 1998 que ce classifieur est adaptée à la classification de textes.

Le classifieur SVM a été initialement créé pour répondre à des problèmes de classification binaire, mais reste malgré tout adaptable a la classification multi classes [2]. Avant d’aborder le principe de fonctionnement général des SVM, commençons par quelques termes fondamentaux d’algorithmes :

**Hyperplan** : est un séparateur d’objets des classes. Le SVM consiste à avoir un hyperplan a une caractéristique particulière dont a distance minimale exemples d’apprentissage est maximale, cet hyperplan est appelé l’hyperplan optimal et la distance appelée marge ;

**Vecteurs support** : ce sont les points qui déterminent l’hyperplan tels qu’ils soient les plus proches de ce dernier ;

**Principe de fonctionnement**

Le SVM consiste à trouver une frontière de décision qui sépare l’espace en deux régions, à trouver l’hyperplan qui classifie correctement les données et qui se trouve le plus loin possible de tous les donnes,

On dit qu’on veut maximiser la marge, veut dire la distance di point le plus proches de l’hyperplan

Dans notre contexte de travail, on va considérer les entrées sont les documents textuels et les sorties sont des catégories.

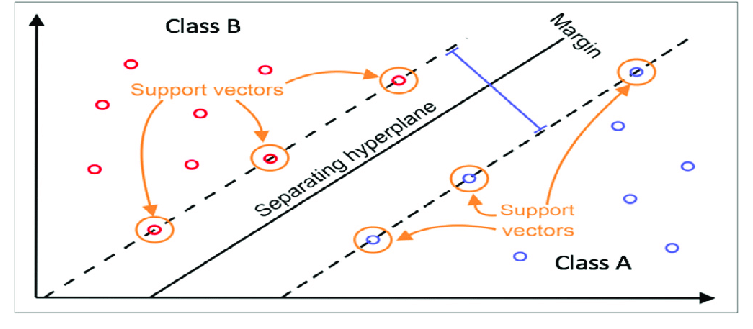
On peut résumer la démarche du fonctionnement de ce classifieur comme suit :

Diagramme suivant :



*Figure 3: le fonctionnement de classifieur SVM*

L’exemple suivant illustre la méthodologie de de travail de cette méthode pour la classification:



*Figure 4: Processus de classification selon machine a vecteur de support*

1. Formules mathématiques de maximisation :

Maximiser la marge revient à minimiser la norme du vecteur de paramètres β .

**Max (2/ ⃦β ⃦)<=>min ⃦β ⃦**

La norme :

**⃦β ⃦=.**

Avec :

**1/ ⃦β ⃦** : C’est la distance entre l’hyperplan et la droite qui passent par les vecteurs de support

1. Apprentissage non supervisé :

Contrairement à l'apprentissage supervisé, le non supervisé traite le cas où on dispose seulement des entrées {X} sans avoir au préalable les sorties. L'apprentissage non supervisé ou le « clustering » vise à construire des groupes (clusters) d'objets similaires à partir d'un ensemble hétérogène d'objets. Chaque cluster issu de ce processus doit vérifier les deux propriétés suivantes :

* La cohésion interne (les objets appartenant à ce cluster sont les plus similaires possible).
* L'isolation externe (les objets appartenant aux autres clusters sont les plus distincts possible).

En effet, le processus de « clustering » repose sur une mesure précise de la similarité des objets que l'on veut regrouper. Cette mesure est appelée distance ou métrique.

On distingue plusieurs algorithmes de « clustering ». Dans notre travail, on s’intéresse seulement à K-means.

1. K-moyennes (K-Means) :

K-Means est un algorithme de partitionnement des données en K nombre de groupes ou clusters. Chaque objet sera associé à un seul cluster. Le nombre K est fixé par l'utilisateur. Cet algorithme vise à regrouper plusieurs données dans un jeu de donné selon les similitudes.



*Figure 5: Principe de l’algorithme K-Means*

**Voici comment fonctionne l’algorithme :**

1. Sélectionnez les numéros d’équipe K.

2. Arrêtez les centroïdes en mélangeant d’abord la configuration des données, puis en sélectionnant K points de données pour les centroïdes sans remplacement.

3. Continuez à répéter qu’il n’y a pas de changement dans les centroïdes. Cela signifie que le traitement des données en groupes ne changera pas.

• Calculer la distance carrée totale entre les points de données et tous les centroïdes.

Affectez chaque point de données au cluster le plus proche (centroïde).

• Calculez les centroïdes des clusters en prenant la moyenne de tous les points de données appartenant à chaque cluster.

SI LES CLASSES

N’EST PAS STABLES

SI LES CLASSES SONT STABLES

REPOSTION DES CENTRES

CONVERGENCE

*Figure 6: Algorithme de K-means*

**Entrée**

Ensemble de n données, notes par x;

Nombre de groupes souhaiter, notée par k;

**Sortie**

Une partition de k groupe {C1, C2…, Ck}

**Début**

1. C1=centre de gravité de l’ensemble des données ;

**Répéter**

1. Initialiser les centres i=1 par le résultat de l’étape précédente :
2. Trouver l’iéme centre :

**Pour chaque donnée x faire**

3-1) Considère x comme étant le iéme centre ;

3-2) Affecter les données aux plus proches centres :

3-3) calculer l’erreur quadratique pour Ci=x

**J=**

**Fin faire**

3-4) Garder le centre Ci=x qui minimise l’erreur quadratique.

4) Appliquer le k-means jusqu’à la convergence

**Jusqu’à** obtenir une partition en k groupes.

1. Conclusion :

Dans ce chapitre, nous avons touché les méthodes d’apprentissage automatique et précisément ses types supervisé et non supervisé, après on a vu quelques exemples d’algorithmes de chaque type.

Le prochain chapitre on va discuter sur l’implémentation et la réalisation de projet.

# *Chapitre III : Implémentation et réalisation du projet*

III.1- Introduction

L'objectif de ce chapitre est de présenter la phase de réalisation en décrivant l’environnement de travail ou on va préciser les technologies et les outils de développement et les jeux de donnée utilisée dans la réalisation de ce projet avec les étapes de prétraitement de corpus pour qu’il soit prêtes et exploitable par le classifieur.

III.2- Technologies et outils de développement

1. **Le langage Python** :

Python est le langage le plus utilisée quand on parle dans le domaine de l’intelligence artificielle. En outre, python est orienté objet et se veut relativement facile d’accès. On a choisi de travailler avec Python car encore est un langage interprété c’est-à-dire que lorsque vous écrivez des instructions, vous pouvez les exécuter directement. En revanche, le langage compilé nécessite d’écrire un programme complet avant de le tester. Le langage interprété facilite la programmation, de manière interactive, pour tester facilement une partie du code. C’est donc le langage idéal pour développer et tester rapidement des prototypes, ou plus précisément, des algorithmes d’analyse de données.

Python est **open source**, il est donc **gratuit** pour tout le monde, y compris les entreprises.

1. **Google Colab** :

Google Colab est un service cloud, offert par google, basé sur *Jupyter Notebook* et destiné à la formation et à la recherche dans l’apprentissage automatique. Cette plateforme permet d’entrainer des modèles de Machine Learning directement dans le cloud, sans donc avoir besoin d’installer quoi que ce soit sur notre ordinateur à l’exception d’u navigateur

*Jupyter Notebook est une application Web Open Source permettant de créer et de partager des documents contenant du code, des équations, des images et du texte.*

En tant que programmeur, vous pouvez effectuer les opérations suivantes :

* Ecrire et Exécuter un code Python.
* Importer/Enregistrer des notes depuis Google Drive.
* Importer des jeux de données externes.

Et d’autres autres opérations…

III.3- Jeux de donnée utilisée (dataset)

1. **Définition** :

Selon Wikipédia : *Un****jeu de données****(en anglais****dataset****ou****data set****) est un ensemble de valeurs « organisées », où chaque valeur est associée à une variable (ou attribut) et à une observation. Une variable décrit l'ensemble des valeurs décrivant le même attribut et une observation contient l'ensemble des valeurs décrivant les attributs d'une*[*unité*](https://fr.wikipedia.org/wiki/Unit%C3%A9_statistique)*.*

1. **Les types d’un corpus** :

Selon *Bowker et Pearson*, ils existent une multitude de types des corpus :

**Les corpus de référence** : un corpus de référence reflète une langue et permet de faire des observations d’ordre général. Ce type de corpus contient des données orales et écrites, c’est un mélange de plusieurs textes de différentes natures (journaux, fictions, reportages, débats …).

**Corpus spécialisé** : un corpus spécialisé est axe sur l’aspect particulier du vocabulaire d’un domaine, sur un type de textes, …

**Corpus oral** : un corpus oral est constitué de transcription de discours oraux

**Corpus écrit** : un corpus écrit contient des textes destines à être lus,

**Corpus monolingue** : un corpus qui contient une seule langue,

**Corpus multilingue** : rassemble des textes dans au moins deux langues. Les corpus multilingues peuvent être divisées en un corpus parallèle et en un corpus comparable

**Un corpus parallèle :** est constitué de textes en langues source accompagnes de leur traduction en langue cible

**Un corpus comparable :** ne contient pas des traductions mais des textes écrits en langue source qui ont tous le même sujet, la même fonction de communication qui sont de même nature (manuel d’instruction, rapport technique, …)

**Corpus synchronique :** est une photo de l’usage d’une langue pendant un temps limite,

**Corpus diachronique** : sert à mesurer l’évolution d’une langue sur une longue période, et beaucoup autre corpus, …

1. **Les qualités d’un corpus** :

Plusieurs caractéristiques sont à prendre en compte pour la création d’un corpus bien formé :

**La taille :** Un bon corpus doit être impérativement de grande taille. Il est impossible d’extraire des informations fiables à partir d’un corpus trop petit.

**La lisibilité :** La lisibilité du corpus incite à la vigilance. Quand des ambiguïtés peuvent être liées à la non prise en compte du sens, il faudra alors craindre une analyse lexicale.

**Le langage du corpus :** Un corpus bien formé doit nécessairement couvrir un seul langage et une seule déclinaison de ce langage.

* Dans notre projet, on a travaillé par un corpus depuis Kaggle.

*Selon Wikipédia : Kaggle est une*[***plateforme web***](https://www.kaggle.com/)*qui accueille la plus grande communauté de Data Science au monde, avec plus de****536 000 membres actifs dans 194 pays****et reçoit près de 150 000 soumissions par mois, et qui lui fournit des outils et des ressources puissants pour aider à atteindre tous les progrès de science des données. Kaggle, de la même manière que*[***Datascientest***](https://datascientest.com/formations-data)*, offre un environnement*[***Jupyter Notebooks***](https://jupyter.org/)*personnalisable et sans configuration. Sont accessibles gratuitement des GPU et une grande quantité de données et de codes publiés par la communauté. À l’intérieur de Kaggle, vous trouverez tout le code et les données dont vous avez besoin pour réaliser vos projets de science des données. Il y a plus de 50 000 jeux de données publics et 400 000 notebooks publics disponibles pour tous.*

1. **Description de corpus** :

Notre jeu de données qu’on a utilisé contient deux colonnes. Le corpus total de 5728 documents. La fonction descriptive se compose d’un texte, la fonctionnalité ciblée se compose de deux classes Ham et Spam, le nom de la première colonne est « Catégorie », et la deuxième colonne « Message ». Les classes sont étiquetées pour chaque document de l’ensemble de données et représentant notre fonctionnalité ciblée avec un alphabet binaire de type chaîne {Ham ; Spam}. Les classes sont ensuite mappées aux entiers 0 (Ham) et 1 (Spam).

* En résumé, le corpus est un outil très important dans la démarche du processus d’apprentissage d’une machine. En effet, il peut jouer comme rôle d’une source des connaissances des classifieurs.
* Dans la partie suivante, on va traiter la procédure générale de traitement des corpus et les méthodes de rendre les corpus bien exploitables avec les classifieurs.

III.4- Notre Modèle

Avant de pouvoir créer et entrainer un quelconque classificateur à partir d'un corpus de textes donnés, nous devions impérativement transformer les documents textes en entrées valides compréhensibles par les différents algorithmes de classification, en respectant le processus du précis. Ces entrées valides sont en fait des vecteurs ou des matrices qui définissent le poids de chaque descripteur (mot ou groupe de mots) dans chaque document texte où ils apparaissent.

1. **Architecture** :

Le processus qui est appliqué au corpus avant de le rendre sous forme d’un document vectoriel, on l’appelle le **prétraitement**, et la transformation entre un document texte (non structurée) et un document vectoriel (les donnée structurée) qu'on appelle **extraction des caractéristiques**.

Les étapes du prétraitement de corpus sont les suivants :

* La normalisation des corpus.
* La tokenisation.
* La suppression des mots vides.
* La racinisation.
* La lemmatisation.

Les méthodes et les techniques d’extraction des caractéristiques sont plusieurs, et on cite :

* La méthode TF\_IDF.
* La méthode TFC.
* La méthode LNU.

**NB :**

Dans notre projet, on va s’intéressé juste sur la méthode TF\_IDF.

1. **Pré-traitement** :
   * + 1. Normalisation des corpus

Normaliser un texte signifie le mettre à la même casse, souvent tout en minuscule.

**Algorithme de normalisation :**

Données : corpus textuels

Sortie : corpus textuelles normalisées

**Début :**

**Pour** tout documents f dans le corpus textuel faire

Ouvrir(f) en mode lecture

Fichier\_normalisée = Normalizer(f)

Corpus textuelles normalisées. ajoutez (Fichier\_normalisee)

**Fin Pour**

**Fin**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Index | catégorie | Message | Longueur | Label |
| 0 | spam | You have an important customer service announcement from premier. | 65 | 1 |
| 1 | spam | Your credits have been topped up for http://www.bubbletext.com your renewal pin is Tgxxrz | 89 | 1 |
| 2 | Non-spam | Science tells that chocolate will melt under the sunlight. please don't walk under the sunlight. bcoz,i don't want to loss a sweet friend. | 138 | 0 |
| 3 | Non-spam | The monthly amount is not that terrible and you will not pay anything till 6months after finishing school. | 106 | 0 |
| 4 | spam | Please call amanda with regard to renewing or upgrading your current T-mobile handset free of charge. offer ends today. tel 0845 021 3680 subject to t's and c's | 160 | 1 |

*Figure 7-1: Texte avant la normalisation*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Catégorie | Message | Longueur | Label |
| 0 | Spam | you have an important customer service announcement from premier. | 65 | 1 |
| 1 | Spam | your credits have been topped up for http://www.bubbletext.com your renewal pin is tgxxrz | 89 | 1 |
| 2 | Non-spam | science tells that chocolate will melt under the sunlight. please don't walk under the sunlight. bcoz,i don't want to loss a sweet friend. | 138 |  |
| 3 | Non-spam | the monthly amount is not that terrible and you will not pay anything till 6months after finishing school. | 106 | 00 |
| 4 | Spam | please call amanda with regard to renewing or upgrading your current t-mobile handset free of charge. offer ends today. tel 0845 021 3680 subject to t's and c's | 160 | 1 |

*Figure 7-2: Texte après la normalisation*

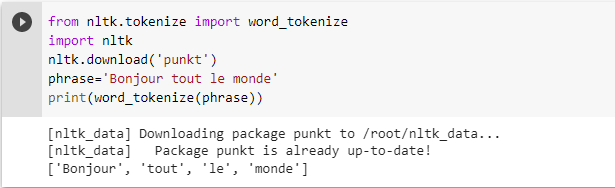
*Figure 7-2: Texte après la normalisation*

La tokenisation

La tokenisation est un processus des décompositions d’un document texte en un ensemble des mots, et un ensemble des phrases. Leur objectif et de bien composer les corpus dans ces formes, pour bien améliorer la précision des classifieurs.

**La décomposition sous forme des mots :**

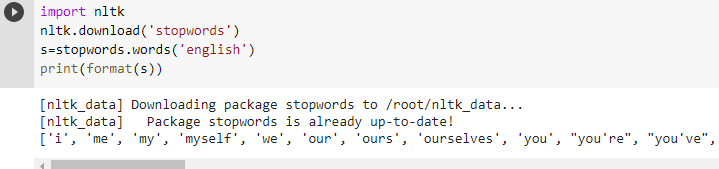
Décomposition sous forme des mots est un processus qui rend un texte d’entrée sous forme un ensemble des mots, incluant les répétitions.



*Figure 8: Extrait de la partie de la tokenisation*

La suppression des mots vides

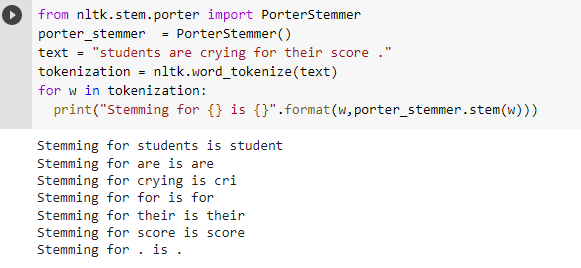
Après la normalisation et la tokenisation, l’étape qui suive dans le processus est la suppression les mots vides ou que l’on nomme ‘’stopwords’’. Ce sont les mots très courants dans la langue étudiée ("et", "à", "le"... en français) qui n'apportent pas de valeur informative pour la compréhension du "sens" d'un document ou bien un corpus.



*Figure 9: Extrait de la partie du stopwords*

La racinisation

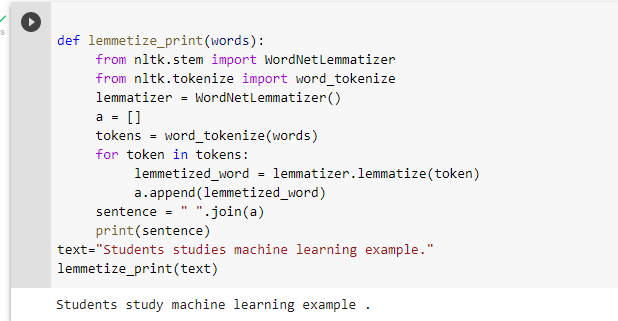
Cette méthode permette de représenter sous un même mot plusieurs dérivées du mot. Dans le cas de la racinisation, nous allons garder le radical du mot.



*Figure 10: Extrait de la partie de la racinisation*

La lemmatisation

La lemmatisation consiste à représenter les mots (ou « lemmes ») sous leur forme canonique. Par exemple pour un verbe, ce sera son infinitif, un nom : son masculin singulier. L'idée étant encore une fois de ne conserver que le sens des mots utilisés dans le corpus.



*Figure 11: Extrait de la partie de la lemmatisation*

Extraction des caractéristiques :

L’extraction des caractéristiques est la dernière, et la plus importante étape qui suivent la phase du pré-traitement des corpus textuelles, cette étape est constituée de plusieurs méthodes d’extraction des caractéristiques dans lesquelles on va étudier juste la méthode TF-IDF.

La méthode TF-IDF :

Le TF-IDF est une méthode de retenue souvent utilisée en recherche d’information et en particulier dans la cavité de textes. Cette mesure statistique permet d'évaluer l'importance d'un terme contenu dans un document, relativement à une collection ou un corpus.

TF-IDF est composé de ces deux termes :

tal14

Avec la fréquence du terme dans un document (dans notre cas un commentaire). Étant le nombre de fois où le terme apparaît dans le document et le nombre de mots du document.

Plus TF est élevé, plus le mot a de l’importance. Il nous reste à calculer IDF :

tal15

Ou n est le nombre de documents et t ∈ d le nombre de documents où le terme est présent.

Nous avons enfin **tƒidƒ(t)=tƒ(t) x idƒ(t)** pour un document en particulier.

C’est un bon exercice de reproduire cet algorithme *from scratch* en python. Comme souvent, il est présent sur plusieurs librairies, notamment scikit-learn. En plus de cela, le tf-idf vectoriel de scikit-learn nous permet directement de spécifier les stop words que l’on souhaite, mais aussi le nombre de N-grams que l’on souhaite prendre en compte, ainsi que plusieurs autres paramètres. Dans notre cas, nous allons ajouter les bi grammes et également prendre en compte les stop words.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | **……** | 191 | 192 | 193 | 194 | 195 | 196 | 197 | 198 | 199 | L1bel |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **……** | 0 | 0 | 0 | 0 | 9381 | 4192 | 5364 | 2988 | 4550 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **……** | 766 | 511 | 9512 | 249 | 7427 | 3947 | 9248 | 5667 | 1997 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **……** | 8443 | 2555 | 4280 | 216 | 4280 | 1171 | 8315 | 8466 | 1637 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | **……** | 4893 | 7850 | 402 | 9867 | 9418 | 8570 | 2433 | 8370 | 1009 | 1 |

*Figure 12: Extrait du rôle de la méthode TF-IDF*

La matrice que nous récupérons comporte quand même 199 colonnes (et le même nombre de lignes qu’au départ, soit 5557) ! Comme vous pouvez le constater, nous ne voyons ici que des zéros, ce qui est normal puisque sur ces 199 tokens, seuls quelques-uns sont présent dans chaque commentaire.

Classification :

La classification de textes est l’une des tâches de traitement du langage naturel les plus courantes. Cependant, l’énorme croissance de la capacité d'informations et en particulier de l’explosif croissance d'Internet a favorisé la recherche technique et scientifique de la classification.

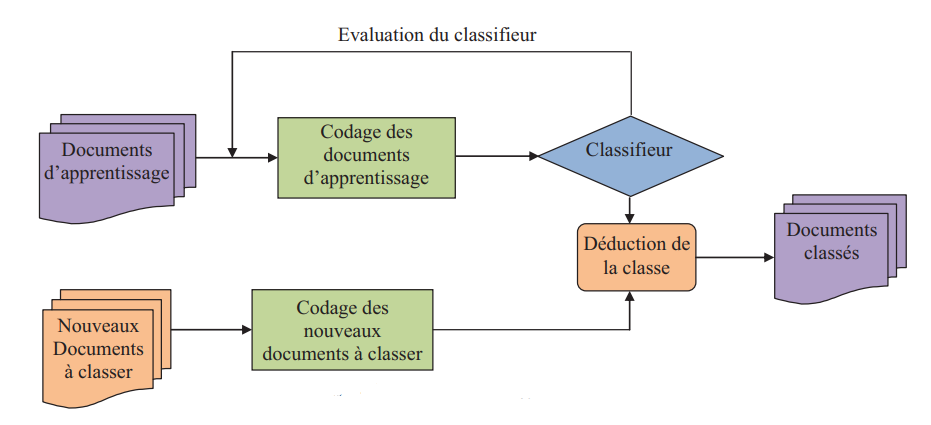
La classification ou bien la catégorisation est utilisée dans des applications très courant dans les systèmes d’informations tell que la recherche numérique des références et ressources pour une meilleur optimisation et l’efficacité de la recherche, la gestion des emails s’il est un spam ou non pour faire au cybercrime, les chat-bots pour bien répond aux plusieurs questions données, ….

La classification de texte a pour fonction première de définir le contenu d’un document en seulement quelques mots. Cet étiquetage va nous permettre notamment de faciliter des recherches par filtre sur une collection de documents textuels.

Dans notre projet, on va étudier l’ensemble des techniques et des instructions pour réaliser un classifieur des documents, en particulier les messages spams et étudier la performance et l’utilité de ces techniques et algorithmes. La démarche d’une approche standard de classification automatique de textes peut être résumée de la manière suivante :

* Eliminer les caractères de séparation, les signes de ponctuations, les mots vides, etc...
* Les termes restants sont tous des attributs.
* Un document devient un vecteur.
* Entrainer le modèle de classification à partir des couples (Message, Classe).
* Évaluer les résultats du classifieur.

La figure suivante illustre la démarche de catégorisation de textes avec ses trois étapes qui peuvent être simplifiées comme suit :

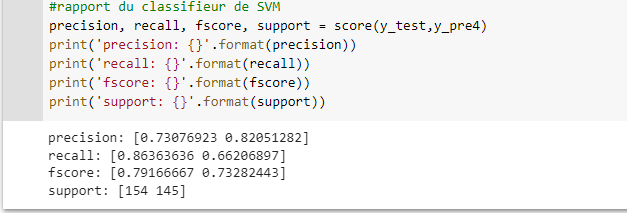


*Figure 13: démarche de la catégorisation de textes*

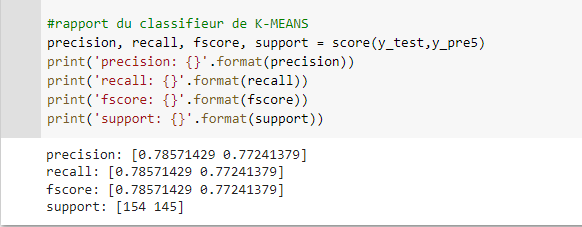
**Présentation sur la phase d’apprentissage du classifieur :**

Après que nous avons rend les corpus prétraités exploitable par les classifieurs grâce à la méthode TF-IDF, on va entamer la phase d’apprentissage et d’évaluation, cette dernière étape on va lui associe un ensemble des calculs sur la précision, rappel, f1-score, support.

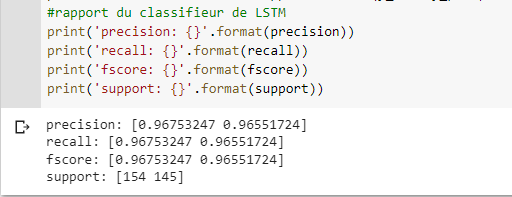
La phase d’apprentissage et grâce à l’objet, résulte les rapports du trois classifieur utilise dans l’apprentissage :



*Figure 14: Rapport du classifieur de SVM*



*Figure 15: Rapport du classifieur de K-Means*



*Figure 16: Rapport du classifieur de LSTM*

**Rapport :**

On peut résumer le rapport d’étude comparative dans le tableau qui suive, on va s’intéressé juste aux valeurs du précision

Tableau 1 : Rapport comparatif des classifieurs avec pré-traitement

|  |  |
| --- | --- |
|  | Avec Pré-traitement |
| SVM | 0.77 |
| K-MEANS | 0.78 |
| LSTM | 0.97 |

**Remarque :**

On peut remarquer que le classifieur LSTM a une haute précision par rapport aux autres méthodes mais on peut proposer comme hypothèse que le classifieur LSTM du NLTk corrige l’erreur.

III.5- Conclusion :

Un modèle avec une précision de 97% a été construit. La précision est plus importante que le rappel, afin d'éviter que les messages de type "Ham" soient classés à tort comme du spam.

Une étape de prétraitement plus personnalisée contribuera à un modèle plus précis.

Le traitement automatique de la langue naturel est une science de recherche très outil pour le développement d’apprentissage automatique, donc il est nécessaire de poser les questions suivantes pour continuer les axes de la recherche et l’amélioration du qualité des classifieurs.

Conclusion générale :

Ce projet a été vraiment très intéressant car cela nous a permis d’approfondir nos connaissances dans le domaine d’apprentissage automatique (machine Learning), cela nous a permis également de travailler en équipe. Il nous a fallu pas mal de temps pour rechercher les documents nous permettant de mener à terme ce projet, surtout en ce qui concerne les méthodes d’apprentissage automatique et aussi les étapes de pré-traitement et l’extraction des caractéristiques du projet.

Ce travail a été divisé en trois grandes parties :

Le premier chapitre, nous avons présenté le cadre général de projet, plus précisément, on a discuté sur les problématiques de projet, les besoins de projet, les objectifs, etc…

Le deuxième chapitre, nous avons vu les méthodes d’apprentissages automatiques qui existent avec leurs types et les algorithmes de chaque type.

Le dernier chapitre, on a présenté les outils avec lesquels on a développé notre classifieur et les étapes nécessaires de pré-traitement et l’extraction des caractéristiques. Finalement, on a fait une comparaison entre les classifieurs.

Références

1. <https://www.datacamp.com/community/tutorials/stemming-lemmatization-python>
2. Khamis, Siti Aqilah, Cik Feresa Mohd Foozy, Mohd Firdaus Ab Aziz, and Nordiana Rahim. "Header based email spam detection framework using Support Vector Machine (SVM) Technique." In International conference on soft computing and data mining, pp. 57-65. Springer, Cham, 2020.
3. Corpus — Wikipédia (wikipedia.org)
4. Text Mining - Introduction à la fouille de textes (univ-lyon2.fr)
5. Python for NLP: Tokenization, Stemming, and Lemmatization with SpaCy Library (stackabuse.com)
6. NLP: Tokenization, Stemming, Lemmatization and Part of Speech Tagging | by Kerem Kargın | MLearning.ai | Medium
7. <https://fr.wikipedia.org/wiki/Kaggle>
8. <https://fr.wikipedia.org/wiki/Jupyter>
9. How to Write to CSV Files in Python (pythontutorial.net)
10. <https://fr.wikipedia.org/wiki/TF-IDF>
11. Feature Extraction Techniques - NLP – GeeksforGeeks
12. https://www.kaggle.com/code/ngawangchoeda/spam-classifier-using-lstm/notebook