

---

Département : Mathématiques & Informatique

Filière : Cycle d'ingénieur : Génie Informatique (2ème année)

Module : Calcul de Complexité et Méta-Heuristiques

---

# Détection des fausses nouvelles en Python avec Machine Learning

Année Universitaire

2021 - 2022

Responsable :

---

Mr. El KAFHALI Said

Présenté par :

---

AIT HSSAINE Mohamed

AIT ELGHAZI Soufiane

BENDIR Mohammed

## Table des matières

I.	Introduction Générale.....	5
1.	Contexte.....	5
2.	Objectif et contribution .....	5
II.	Conception d'un modèle de détection de fausses informations .....	6
1.	Introduction .....	6
2.	Description et objectif de projet .....	6
3.	L'apprentissage automatique pour la classification du texte.....	6
3.1	L'apprentissage automatique (Machine Learning) .....	7
3.1.1	L'apprentissage supervisé.....	7
3.1.2	L'apprentissage non supervisé .....	7
3.1.3	L'apprentissage par renforcement.....	7
3.2	La classification du texte.....	7
3.2.1	La collection des données.....	7
3.2.2	Le prétraitement de donnés.....	7
3.2.3	L'extraction des caractéristiques .....	8
3.2.4	La classification.....	8
3.2.5	L'évaluation.....	9
4.	Conclusion.....	10
III.	Implémentation et résultats.....	11
1.	Introduction .....	11
2.	L'environnement et les outils de travail .....	11
2.1	Le matériel.....	11
2.2	Le langage de programmation .....	11
2.3	L'éditeur de code.....	11
2.4	Les librairies et les bibliothèques Python.....	11
3.	L'analyse exploratoire de données .....	13
3.1	Informations générales.....	13
3.2	La distribution de données.....	13
3.3	Le prétraitement.....	14
3.4	Les résultats de classification .....	15
3.5	La discussion des résultats de classification .....	16
3.6	La prédiction de fausses informations .....	16
4.	Conclusion.....	17
IV.	Conclusion Générale .....	18

## Liste Figures :

<b>Figure 1</b> : Processus de réalisation de système de détection de fausses informations .....	6
<b>Figure 2</b> : matrice de confusion .....	9
<b>Figure 3</b> : Editeur de code et bibliothèques Python utilisés .....	12
<b>Figure 4</b> : Distribution des types des données « data ». ....	13
<b>Figure 5</b> : Résultats de classification par la régression logistique. ....	15
<b>Figure 6</b> : Résultats de classification par les arbres de décision. ....	15
<b>Figure 7</b> : Résultats de classification par les forêts aléatoires. ....	15
<b>Figure 8</b> : Résultats de classification par Multinomial de Naïve de Bayes .....	15
<b>Figure 9</b> : Résultats de classification par modèles de base.....	16
<b>Figure 10</b> : Exemple de détection d'une fausse information.....	16

## Liste Tableaux :

<b>Tableau 1</b> : Caractéristiques du matériel utilisé .....	11
<b>Tableau 2</b> : Caractéristiques de l'ensemble de données « data » .....	13

# I. Introduction Générale

## 1. Contexte

De nos jours, Internet, le plus vaste réseau mondial de communication, est devenu universel et accessible grâce aux sites web et aux médias sociaux qui ont réussi à rassembler les individus, les entreprises et les organisations côte à côte. Ce nouvel environnement a offert un moyen facile et efficace de partage de publications, d'articles, de nouvelles et des informations par rapport aux médias traditionnels en raison de ses services gratuits qui offrent une large diversité de fonctionnalités. La popularité des médias sociaux est liée à la demande croissante et continuée d'information, devenue plus importante dans nos sociétés modernes et se multipliant à un niveau sans précédent.

Durant cette décennie, le phénomène de fausses nouvelles, *fakes news* en anglais et *infox* en français, a attiré autant d'inquiétudes que d'attention. Les fausses informations portent sur tous les niveaux allant des individus ordinaires jusqu'aux gouvernements et états. Elles ont connu une large propagation en influençant négativement des différents secteurs de vie. De ce fait, l'information est devenue plus précieuse que jamais, et l'existence des solutions pour détecter et bloquer la dissémination des fausses nouvelles qui circulent sur le web est devenue une nécessité.

De nombreuses applications et systèmes ont déjà été développés pour cette tâche. Le processus de détection de la désinformation et la mésinformation vise, automatiquement ou manuellement, à une meilleure détermination d'une fausse information en vue de bloquer sa diffusion ou de minimiser son ampleur.

## 2. Objectif et contribution

Dans ce projet, notre objectif est d'explorer le domaine des fausses informations, en particulier la tâche de détection de fausses nouvelles publiées sur le web et les médias sociaux. Nous avons également exploité les techniques de l'apprentissage automatique pour développer certains modèles de détection de fausses informations tout en respectant la méthodologie de la classification du texte liée au domaine du traitement automatique du langage naturel (NLP).

## II. Conception d'un modèle de détection de fausses informations

### 1. Introduction

Dans le but de mettre techniquement en évidence la tâche de la détection de fausses informations, nous présentons dans cette partie de ce travail la conception d'un cas d'étude concret de classification du texte basé sur l'apprentissage supervisé, et appliqué sur les fausses informations.

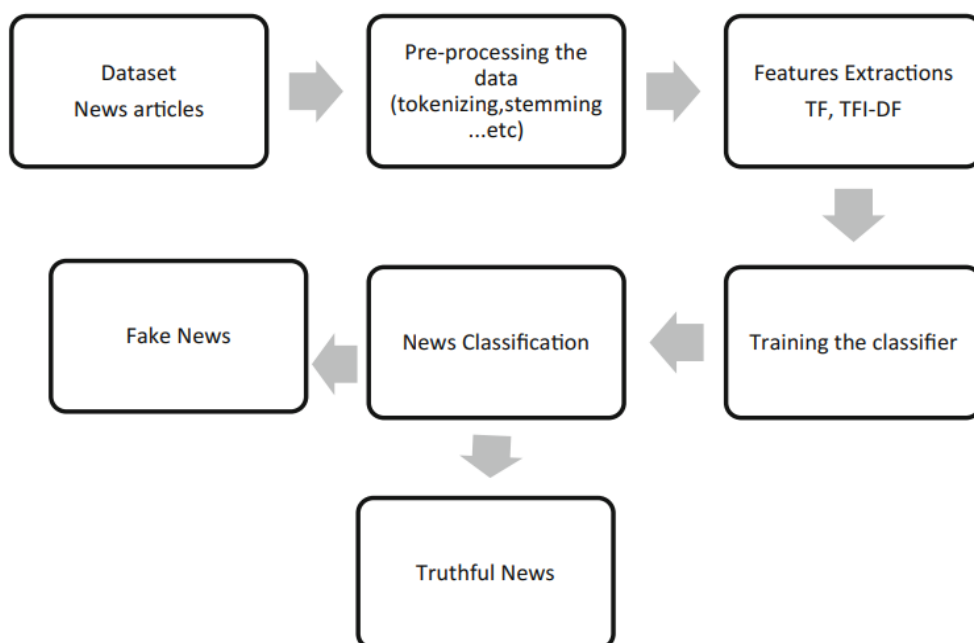
### 2. Description et objectif de projet

Notre projet est une application directe de la détection de fausses informations sur le web et les plateformes en ligne en utilisant l'apprentissage automatique (Machine Learning).

Les informations à examiner et classer concernent une collection des publications partagées sur certains sites web.

### 3. L'apprentissage automatique pour la classification du texte

Parmi les différentes approches de détection des Fakes News, nous avons suivi une approche basée sur La vérification automatique qui s'appuie sur le traitement automatique du langage naturel (NLP).



*Figure 1 : Processus de réalisation de système de détection de fausses informations*

### 3.1 L'apprentissage automatique (Machine Learning)

L'apprentissage automatique (ML : Machine Learning) est un sous-domaine de l'intelligence artificielle, qui donne à un système une capacité de compréhension grâce à ses algorithmes. Il est basé sur l'idée de faire apprendre des algorithmes à partir de données et de faire des prédictions avec ces données et par cela les ordinateurs apprennent à résoudre des tâches spécifiques, sans avoir besoin de les programmer. Il existe trois types ou catégories d'apprentissage :

3.1.1 ***L'apprentissage supervisé*** où, après avoir présenté les données et les résultats souhaités aux ordinateurs, ils auront la capacité de faire des prédictions pour de nouvelles données d'entrée.

3.1.2 ***L'apprentissage non supervisé*** où l'on ne donne à l'ordinateur que les données et où il doit trouver une structure avec un sens par lui-même sans l'intervention d'une supervision extérieure. Elle dépend principalement du clustering.

3.1.3 ***L'apprentissage par renforcement*** où la machine se comporte comme un agent qui apprend de son environnement d'une manière interactive jusqu'à ce qu'il découvre les comportements qui produisent des récompenses.

Et puisque notre jeu de données présente une tâche de classification binaire (faux, vrai), notre conception automatiquement sera suivre le type : apprentissage supervisé.

### 3.2 La classification du texte

La classification des textes est une tâche générique de traitement automatique de la langue naturelle qui consiste à assigner une ou plusieurs catégories, parmi une liste prédéfinie ou non à un document en trouvant une liaison fonctionnelle entre un ensemble de textes et un ensemble de catégories (étiquettes, classes) selon des critères.

Classifier des textes consiste souvent en cinq étapes : collection de données, prétraitement de données, extraction des caractéristiques, classification et évaluation du modèle de classification.

#### 3.2.1 *La collection des données*

Cette étape consiste à collecter les données brutes destinées à la classification depuis des sources crédibles de données. Pour notre projet, nous avons collecté un ensemble de 4009 données publiques fausses et vraies, collectées à partir des articles des journaux partagées sur les sites web.

#### 3.2.2 *Le prétraitement de données*

Le pré-traitement linguistique est un outil puissant pour préparer les données textuelles au traitement automatique. Généralement, les prétraitements comprennent la normalisation textuelle des mots, la normalisation linguistique et la segmentation. La segmentation (en anglais *tokenisation*) consiste à séparer une suite de caractères en éléments sémantiques, ou

mots (en anglais *tokens*). La normalisation est une opération qui consiste à fournir une forme canonique pour chaque mot. Deux types de normalisation à distinguer :

- **La normalisation textuelle** Prenant par exemple : supprimer les valeurs nulles et les mots vides (en anglais stopwords), éliminer la ponctuation, mettre les lettres du texte en minuscules et transformer les chiffres en lettres, développer les formes contractées (exemple : can't devient can not) ...etc.
- **La normalisation linguistique** avec ses deux types. **La racinisation** (en anglais *stemming*) se rapporte au procédé qui cherche à supprimer les flexions et les suffixes des mots. Il est fortement dépendant de la langue utilisée. **La lemmatisation** (en anglais *lemmatization*) fait une analyse linguistique poussée destinée à enlever les variantes flexionnelles des mots afin de les ramener sous leur forme lemmatisée ou encyclopédique.

### 3.2.3 L'extraction des caractéristiques

La vectorisation du texte (en anglais *word embedding*) est un ensemble de techniques de traitement du langage naturel où les mots ou les phrases sont transformées à des vecteurs numériques. La technique la plus utilisée est :

**Term Frequency-Inverse Document Frequency (TF-IDF)** : est une méthode d'évaluation de la pertinence d'un document par rapport à un terme, en tenant compte de deux facteurs : la fréquence de ce mot dans le document (TF) et le nombre de documents contenant ce mot (IDF) dans le corpus étudié.

$$tf - idf_{t,d} = tf_{t,d} \times idf_t = tf_{t,d} \times \log(N/df_{t,d})$$

Tel que :  $tf_{t,d}$  désigne la fréquence d'un terme ou mot  $t$  dans le document  $d$ ,  $df_{t,d}$  le nombre de documents  $d$  contenant le terme  $t$ ,  $N$  est le nombre total de documents dans le corpus.

### 3.2.4 La classification

Il existe différents classificateurs d'apprentissage supervisé qui sont utilisés pour la classification du texte. Les modèles de base sont les suivants :

#### ✚ **Naïve de Bayes (Naive Bayes)**

Cette méthode se base sur le théorème de Bayes permettant de calculer les probabilités conditionnelles. Dans le cas de la classification du texte, la méthode Naïve Bayes est utilisée comme suit : on cherche la classification qui maximise la probabilité d'observer les mots du document. Lors de la phase d'entraînement, le classificateur calcule les probabilités qu'un nouveau document appartient à telle catégorie à partir de la proportion des documents d'entraînement appartenant à cette catégorie. Il calcule aussi la probabilité qu'un mot donné soit présent dans un texte, sachant que ce texte appartient à telle catégorie. Quand un nouveau document doit être classé, on calcule les probabilités qu'il appartienne à chacune des catégories à l'aide de la règle de Bayes La formule :  $P(A/B) = \frac{P(A \cap B)}{P(B)}$

#### ✚ **La régression logistique (Logistic Regression)**



La régression logistique est essentiellement un algorithme de classification supervisé. Dans un problème de classification, la variable cible (ou sortie),  $y$ , ne peut prendre que des valeurs discrètes pour un ensemble donné de caractéristiques (ou d'entrées),  $X$ . Le modèle construit un modèle de régression pour prédire la probabilité qu'une entrée de données donnée appartienne à la catégorie numérotée « 1 ». La régression logistique modélise les données à l'aide de la fonction sigmoïde.

#### ✚ *Les forêts aléatoires (Random Forest)*

C'est une application de graphe en arbres de décision permettant ainsi la modélisation de chaque résultat sur une branche en fonction des choix précédents. On prend ensuite la meilleure décision en fonction des résultats qui suivront.

#### ✚ *Les arbres de décisions (Decision Trees)*

Les arbres de décision cherchent à exploiter une représentation hiérarchique de la structure des données sous forme des séquences des tests en vue de la prédiction d'une classe, tel que chaque donnée entrante est décrite par un ensemble de variables qui sont testées dans les nœuds de l'arbre. Donc, les tests s'effectuent dans les nœuds internes et les décisions sont prise dans les nœuds feuilles.

### 3.2.5 L'évaluation

La performance des modèles de classification est généralement basée sur la façon dont ils prédisent les résultats pour les nouveaux ensembles de donnés. Cette performance est mesurée par rapport à un ensemble de test. Plusieurs métriques déterminent les performances de prédiction d'un modèle, mais nous allons principalement se concentrer sur les métriques suivantes :

✚ **La matrice de confusion** : une matrice de confusion est un résumé des résultats de prédiction sur un problème de classification. Le nombre de prédictions correctes et incorrectes est résumé avec des valeurs de comptage et ventilé par classe sous forme tabulaire comme la montre le tableau suivant :

<i>Ensemble des catégories</i>	<b>Documents appartenant à la catégorie</b>	<b>Documents n'appartenant pas à la catégorie</b>
<i>Documents assignés à la catégorie par le classifieur</i>	True Positif (TP)	False Negative (FN)
<i>Documents rejetés à la catégorie par le classifieur</i>	False Positif (FP)	True Negative (TN)

**Figure 2** : matrice de confusion

On définit à partir des statistiques de cette table les mesures suivantes :

✚ **Le taux de succès ou d'erreur** (en anglais *accuracy*) : désigne le taux des prédictions réussites obtenu par le modèle de classification. C-à-d :

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

✚ **La précision** (en anglais *precision*) : est définie comme le nombre de prédictions faites qui sont réellement correctes ou pertinent parmi toutes les prédictions basées sur la classe positive. Ceci est également connu comme **valeur prédictive** positive et peut être représentée par la formule :

$$\text{Precision} = \frac{TP}{TP + FP}$$

✚ **Le rappel** (en anglais *recall*) : est défini comme le nombre d'instances de la classe positive qui étaient correctement prédit. Ceci est également connu sous le nom de couverture ou de sensibilité et peut être représenté par la formule :

$$\text{Recall} = \frac{TP}{TP + FN}$$

✚ **F1-Score** : est une autre mesure de précision qui est calculée en prenant la moyenne harmonique de la précision et du rappel et peut être représentée comme suit :

$$F1 - \text{Score} = \frac{2 * \text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

## 4. Conclusion

Nous avons proposé une étude suivant l'apprentissage automatique pour la classification et la détection de fausses informations. Nous avons présenté les fonctions les plus importantes du prétraitement des textes et mentionné les différentes techniques d'extraction des caractéristiques. Ensuite, nous avons détaillé la tâche de classification en décrivant les algorithmes de base de Machine Learning. Nous avons également introduit les différentes mesures d'évaluation de performance d'un classificateur. Dans le chapitre suivant les résultats de cette étude.

### III. Implémentation et résultats

#### 1. Introduction

Après avoir décrit notre approche de conception dans le précédent chapitre, le présent chapitre rapporte la mise en œuvre de la tâche de détection de fausses informations en se basant sur l'approche de l'apprentissage automatique. Nous déterminerons l'environnement et les outils de travail d'abord. Puis, après une analyse exploratoire nécessaire, nous listerons les étapes de la détection de fausses informations en les supportant par des illustrations statistiques ou graphiques. Nous clôturons ensuite par une discussion sur les résultats finaux et leurs implications.

#### 2. L'environnement et les outils de travail

##### 2.1 Le matériel

Nous avons réalisé notre projet à l'aide d'un pc portable dont les caractéristiques sont décrites dans le tableau ci-dessous :

*Tableau 1 : Caractéristiques du matériel utilisé*

	Poste de travail
PC	HP
Système d'exploitation	Windows 10 Professionnel
Processeur	Intel(R) Core(TM) i7-3687U CPU @ 2.10GHz 2.60 GHz
RAM	8,00 Go
Type de système	Système d'exploitation 64 bits

##### 2.2 Le langage de programmation

Nous avons choisi Python, version 3.7.4 pour implémenter notre système de détection de fausses informations. Python est un langage de programmation interprété, multiparadigme et multiplateformes. Il est conçu pour optimiser la productivité des programmeurs en offrant des outils de haut niveau et une syntaxe simple à utiliser. Il est doté d'un typage dynamique fort, d'une gestion automatique de la mémoire par ramasse-miettes et d'un système de gestion d'exceptions. Le langage Python est placé sous une licence libre et fonctionne sur la plupart des plates-formes informatiques.

##### 2.3 L'éditeur de code

Pour éditer le code de notre système, nous avons utilisé Jupyter Notebook, version 6.4.5, qui est une application web *open source* qui permet de créer et de partager des documents contenant du code en direct, des équations, des visualisations et du texte narratif. Les utilisations comprennent : le nettoyage et la transformation des données, la modélisation statistique, la visualisation des données, l'apprentissage automatique et bien plus encore.

##### 2.4 Les librairies et les bibliothèques Python



*Figure 3 : Editeur de code et bibliothèques Python utilisés*

- **Pandas** : Pandas est une bibliothèque écrite pour le langage de programmation Python permettant la manipulation et l'analyse des données. Elle propose en particulier des structures de données et des opérations de manipulation de tableaux numériques et de séries temporelles.
- **Nltk** : est une plateforme Python leader pour travailler sur les données du langage naturel. Elle offre des interfaces faciles à manipuler et plus de 50 corpus et ressources lexiques. Nltk offre également des librairies de prétraitement de données, de classification, segmentation, racinisation, et plus d'autres.
- **Scikit-learn** : est une bibliothèque libre Python destinée à l'apprentissage automatique. Elle comprend de nombreuses méthodes pour la classification, la régression, et le clustering. Elle est conçue pour s'harmoniser avec d'autres bibliothèques libres Python, notamment NumPy et SciPy.
- **Matplotlib** : est une bibliothèque de visualisation 2D de données, conçue pour Python. Elle offre des possibilités variées de visualisations statiques, personnalisées et interactifs dans des différents formats.
- **NumPy** : est une bibliothèque destinée à manipuler des matrices ou tableaux multidimensionnels ainsi que des fonctions mathématiques opérant sur ces tableaux.
- **Seaborn** : est une librairie qui vient s'ajouter à Matplotlib, remplace certains réglages par défaut et fonctions, et lui ajoute de nouvelles fonctionnalités.

### 3. L'analyse exploratoire de données

#### 3.1 Informations générales

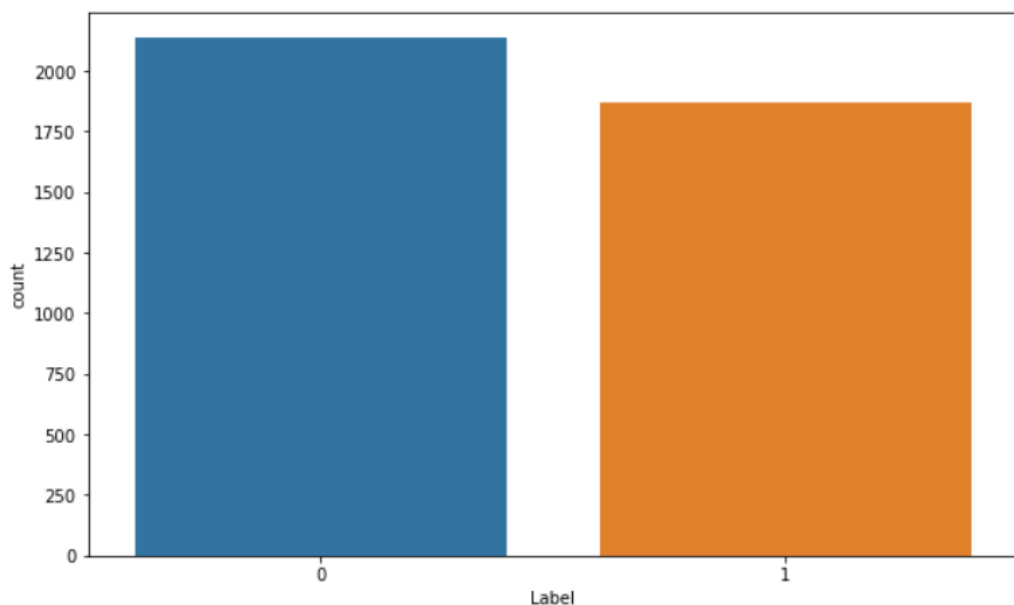
Nous avons nommé notre ensemble de données « data ». Comme nous l'avons déjà décrit dans le chapitre Précédent, cet ensemble est extrait à partir d'un ensemble de données disponibles publiquement. Pour notre objectif, nous avons gardé un corpus avec seulement deux attributs : *Label* (0,1), et *News* (une collection des publications partagées sur certains sites web). Les caractéristiques de l'ensemble de données final sont décrites ci-après :

**Tableau 2 :** Caractéristiques de l'ensemble de données « data »

<i>Nom de l'ensemble de données</i>	<i>Data</i>
<i>Nombre de lignes</i>	<b>4009</b>
<i>Nombre de colonnes</i>	<b>2</b>
<i>Type de données</i>	<b>Textuelles</b>
<i>Noms des colonnes</i>	<b>(News, Label)</b>

#### 3.2 La distribution de données

L'ensemble de données « data » contient 4009 articles et publications postées sur les sites web et les médias sociaux, 2137 sont étiquetées « 0 » c-à-d, sont fausses, et 1872 sont étiquetées « 1 » c-à-d, sont vraies. La figure 3 illustre la distribution de données selon les attributs « Type » et « Label ».



**Figure 4 :** Distribution des types des données « data ».

### 3.3 Le prétraitement

La phase de prétraitement de données textuelle est une étape clé pour un modèle de détection performant. Nous avons appliqué sur le corpus « data » les fonctions de normalisation et prétraitement mentionnées précédemment. L'exemple suivant montre un texte brut avant et après la normalisation superficielle et la normalisation linguistique :

*Texte brut :*

'Four ways Bob Corker skewered Donald TrumpImage copyright Getty Images\nOn Sunday morning, Donald Trump went off on a Twitter tirade against a member of his own party.\nThis, in itself, isn't exactly huge news. It's far from the first time the president has turned his rhetorical cannons on his own ranks.\nThis time, however, his attacks were particularly biting and personal. He essentially called Tennessee Senator Bob Corker, the chair of the powerful Senate Foreign Relations Committee, a coward for not running for re-election.\nHe said Mr Corker "begged" for the president's endorsement, which he refused to give. He wrongly claimed that Mr Corker's support of the Iranian nuclear agreement was his only political accomplishment.\nUnlike some of his colleagues, Mr Corker - free from having to worry about his immediate political future - didn't hold his tongue.\nSkip Twitter post by @SenBobCorker It's a shame the White House has become an adult day care center. Someone obviously missed their shift this morning. - Senator Bob Corker (@SenBobCorker) October 8, 2017 Report\nThat wasn't the end of it, though. He then spoke with the New York Times and really let the president have it. Here are four choice quotes from the Tennessee senator's interview with the Times and why they are particularly damning.\n"I don't know why the president tweets out things that are not true. You know he does it, everyone knows he does it, but he does." "You can't really sugarcoat this one. Mr Corker is flat-out saying the president is a liar - and everyone knows it.\nThe senator, in particular, is challenging Mr Trump's insistence that he unsuccessfully pleaded for his endorsement, but the accusation is much broader.\nMr Corker and the president used to be something akin to allies. The Tennessean was on Mr Trump's short list for vice-president and secretary of state.\nImage copyright Getty Images Image caption Bob Corker at Trump campaign rally in July 2016\nThose days are seemingly very much over now - and it's not like Mr Corker is going anywhere anytime soon. Although he's not running for re-election, he'll be in the Senate, chairing a powerful committee, until January 2019.\nThe president's margin for success in that chamber is razor-thin. If Democrats can continue to stand together in opposition, he can afford to lose only two votes out of 52 Republican senators. That's why healthcare reform collapsed in July - and it could be bad news for tax efforts.\nFrom here on out, Mr Corker isn't going to do the president any favours.\n"Look, except for a few people, the vast majority of our caucus understands what we're dealing with here." "Frustration in Congress has been growing over what Republicans feel has been the president's inability to focus on advancing their agenda. Getting a sharply divided party to come together on plans to repeal Obamacare, reform taxes or boost infrastructure spending is challenging enough. Doing so when the president stirs up unrelated controversies on a seemingly daily basis makes things all the harder.\nOne of the president's gifts has been his ability to shake off negative stories by quickly moving

*Texte final prétraité :*

'four way bob corker skewer donald trumpimag copyright getti imag sunday morn donald trump went twitter tirad member parti exactli huge news far first time presid turn rhetor cannon rank time howev attack particularli bite person essenti call tennesse senat bob corker chair power senat foreign relat committe coward run elect said mr corker beg presid endors refus give wrongli claim mr corker support iranian nuclear agreement polit accomplish unlik colleagu mr corker free worri immedi polit futur hold to ngu skip twitter post senbobcork shame white hous becom adult day care center someon obvious miss shift morn senat bob corker senbobcork octob report end though spoke new york time realli let presid four choic quot tennesse senat interview time particularli damn know presid tweet thing true know everyon know realli sugarcoat one mr corker flat say presid liar everyon know senat particular challeng mr trump insist unsuccess plead endors accus much broader mr corker presid use someth akin alli tennessean mr trump short list vice presid secretari state imag copyright getti imag imag caption bob corker trump campaign ralli juli day seemngli much like mr corker go anywher anytim soon although he'n elect senat chair power committe januari presid margin success chamber razor thin democrat continu stand togeth opposit afford lose two vote republican senat healthcar reform collaps juli could bad news tax effort mr corker go presid favour look except peopl vast major caucu understand deal frustrat congress grow republican feel presid inabl focu advanc agenda get sharpli divid parti come togeth plan repeal obamacar reform tax boost infrastructur spend challeng enough presid stir unrel controversi seemngli daili basi make thing harder one presid gift abil shake neg stori quickli move differ subject work brilliantli presidenti campaign less effect legis slow grind imag copyright getti i imag imag caption corker confirm hear secretari state rex tillerson month republican congress grumbl background among occasion someone like mr mcconnel lament presid understand senat work mr corker state loud clear say almost everyon agre kept silent still hope pass conserv legis presid sign fear mr trump legion back primari challeng next year stay home gener elect calculu ever chance becom riskier stay silent speak mr trump real troubl lot peopl think kind good cop bad cop act underway true time mr trump appear undercut secretari state rex tillerson other administr attempt use soft diplomaci deal rang intern crise war taliban afg hanistan iran complianc multin nuclear agreement ongo disput qatar persian gulf neighbour unrest venezuela recent north korea continue ballist missil test target presid offhand remark twitter invect administr defend said part mr trump strategi updat versi on nixon era madman theori presid forc adversari give way fear unpredict us leader action mr corker buy strategi say possibl chance hope mr trump senior advis abl avoid know fact everi singl day white hous situat tri contain grow collect john kelli face pal photo serv testament chief staff report frustrat deal presid mr trump goe script prais torch bear white nationalist ralli charlottesvil mr kelli captur close eye rub arch nose attempt stave migrain imag copyright reuter imag caption white hous chief staff john kelli look us presid donald trump speak campaign ralli presid call north korean leader crimin speech unit nation mr kelli straight buri face hand white hous commun team often left scrambl tri explain reframe indel presidenti joke remark directli contradict offici administr line even though mr kelli brought disciplin west wing staff presid still march beat drum continu unfetter access phone twitter app bob corker latest person politician journalist sport star celebr feel mercuri presid uncontain irreparable'



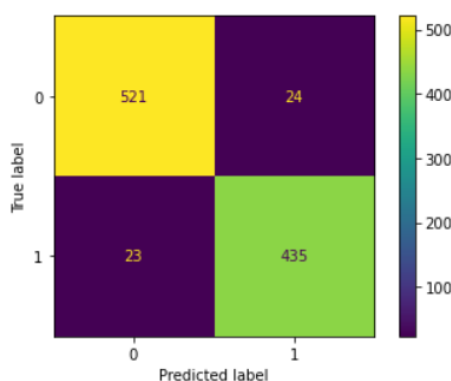
### 3.4 Les résultats de classification

Pour classifier les textes de l'ensemble de données « data », nous avons appliqué 04 modèles de base de classification : Naïve de Bayes, Régression logistique, Forêts aléatoires, et Arbres décisionnels.

Les résultats de classification (sont montrés dans les figures ci-dessous) sont évalués à l'aide des métriques : Précision (*Precision*), Rappel (*Recall*), F1 Score, et Taux de succès (*Accuracy*).

Accuracy of DecisionTree model is: 95.31%

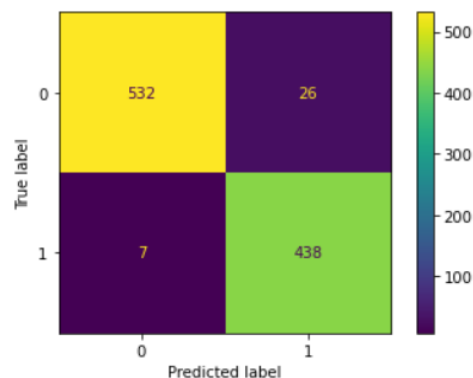
	precision	recall	f1-score	support
0	0.96	0.96	0.96	545
1	0.95	0.95	0.95	458
accuracy			0.95	1003
macro avg	0.95	0.95	0.95	1003
weighted avg	0.95	0.95	0.95	1003



**Figure 6 :** Résultats de classification par les arbres de décision.

Accuracy of LR model is: 96.71%

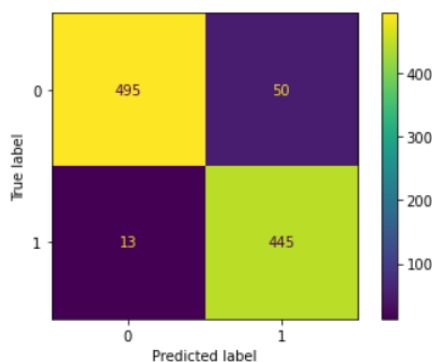
	precision	recall	f1-score	support
0	0.99	0.95	0.97	558
1	0.94	0.98	0.96	445
accuracy			0.97	1003
macro avg	0.97	0.97	0.97	1003
weighted avg	0.97	0.97	0.97	1003



**Figure 5 :** Résultats de classification par la régression logistique.

Accuracy of MultinomialNB model is: 93.72%

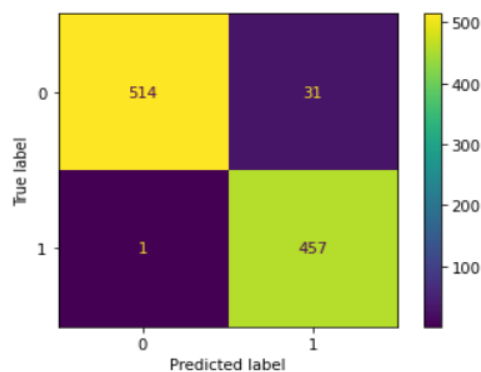
	precision	recall	f1-score	support
0	0.97	0.91	0.94	545
1	0.90	0.97	0.93	458
accuracy			0.94	1003
macro avg	0.94	0.94	0.94	1003
weighted avg	0.94	0.94	0.94	1003



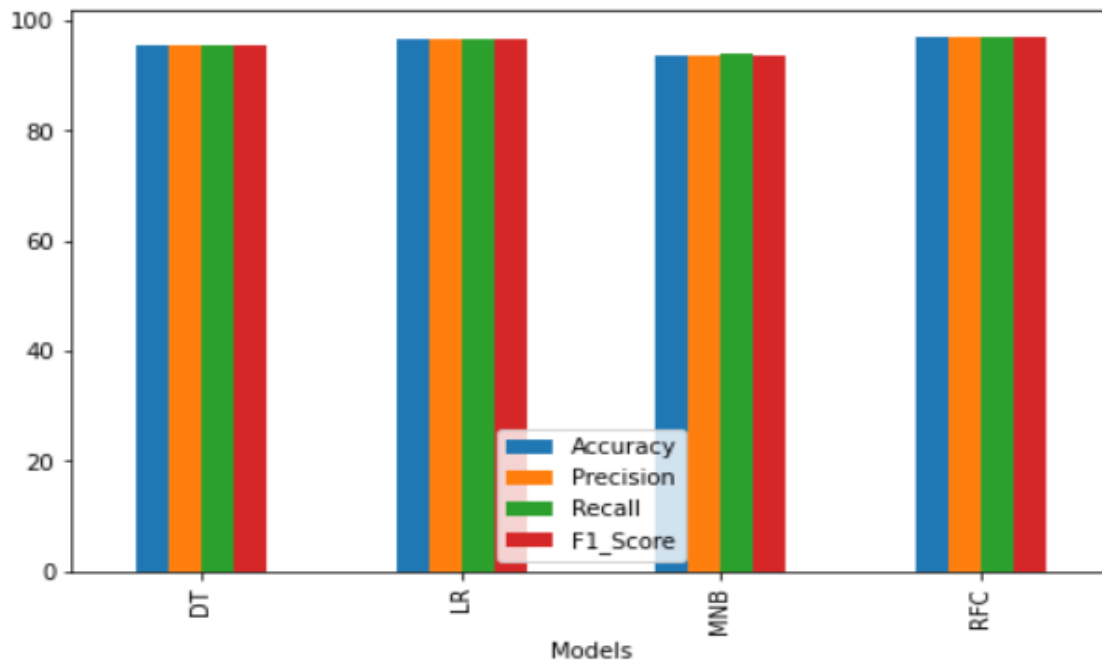
**Figure 8 :** Résultats de classification par Multinomial de Naïve de Bayes

Accuracy of RFC model is: 96.81%

	precision	recall	f1-score	support
0	1.00	0.94	0.97	545
1	0.94	1.00	0.97	458
accuracy			0.97	1003
macro avg	0.97	0.97	0.97	1003
weighted avg	0.97	0.97	0.97	1003



**Figure 7 :** Résultats de classification par les forêts aléatoires.



**Figure 9 :** Résultats de classification par modèles de base

### 3.5 La discussion des résultats de classification

A partir de ce que nous avons comme résultats, nous remarquons que les modèles de classification de base Naïve de Bayes, Régression logistique, Forêts aléatoires, et arbres de décisions, donnent de bonnes performances. D'autre part, nous constatons que le modèle RFC et le modèle de la régression logistique surpassent les autres classificateurs de base et fonctionnent mieux en termes de toutes les mesures. Ceci n'est pas surprenant parce que RFC est connu de sa capacité optimale à supporter le type catégorique de données (comme les données textuelles) par rapport aux autres classificateurs classiques. De même pour la régression logistique qui donne des bons résultats dans le cas où les classes des données peuvent être linéairement séparables.

### 3.6 La prédiction de fausses informations

La dernière étape dans notre projet est la prédiction de fausses informations. Cette étape consiste à prédire la véracité d'un texte en entrée en se basant sur le modèle de classification le plus performant parmi les modèles présentés ci-dessus (dans notre cas, les RFC ou la régression logistique). Le résultat est la classe « Fake » ou « Not a Fake News ».

Comme exemple, nous avons testé le modèle de détection final avec un texte fabriqué aléatoirement. Le résultat est illustré dans la figure 10.

```
Entrée [39]: fake_news_det('U.S. Secretary of State John F. Kerry said Monday that he will stop in Paris later this week|')
```

Not a Fake News

**Figure 10 :** Exemple de détection d'une fausse information



## 4. Conclusion

Nous avons conclu notre travail dans ce Mini Projet par un rapport décrivant la réalisation de la tâche de détection de fausses informations appliquée sur un ensemble de données publiées dans le site Kaggle.com. Nous avons présenté les différentes étapes de la mise en œuvre ainsi que les résultats obtenus par les différents modèles présentés préalablement dans le chapitre de conception précédent. Une interprétation des résultats sur la base des scores des métriques adéquates est offerte à la fin du chapitre afin de bien comprendre les limitations et les performances des modèles de classification appliqués.

## IV. Conclusion Générale

Le concept de détection de fausses informations sur le web est particulièrement nouveau et important. Dans ce contexte, notre travail a visé plus d'un objectif. En premier lieu, il nous a permis d'explorer le domaine d'échange informationnel humain sur le web et les plateformes sociaux et comprendre toutes ses complexités en termes de traitement, de contrôle et d'orientation vers la bonne voie. En outre, nous avons eu l'opportunité de mettre en pratique toutes nos connaissances dans le domaine d'apprentissage automatique sur un thème plus important. Nous avons eu également la chance de travailler sur le domaine de traitement automatique de langage naturel, qui est un domaine très actif et très prometteur et dont la nécessité pour la société moderne s'accroît jour après jour.

Le travail que nous avons réalisé pourrait être complété et poursuivi sous différents aspects. Il serait pertinent d'étendre cette étude avec un *dataset* plus volumineux, et d'évoluer son apprentissage supervisé par un autre en ligne pour une mise à jour continue et intégration automatique des nouvelles fausses informations.