Projet SD 701 : Détection de Fake News

1. Objectif

L'objectif de ce projet est d'étudier les données extraites d'un jeu de données Kaggle (https://www.kaggle.com/mdepak/fakenewsnet) regroupant des données issues de différentes plateformes d'information (Buzzfeed …) en explorant des méthodes permettant de faciliter la détection ou « prédiction » de Fake news. Après avoir effectué un travail de pré-traitement des données on explorera la piste de la classification en comparant plusieurs classifieurs différent.

2. Extraction des données

Le jeu de données utilisé est disponible en deux parties : un csv pour les données de Real News et un autre pour les Fake news. Ensuite on concatène ces deux jeux donnés en un seul DataFrame pour lequel on peut visualiser les données et leurs caractéristiques.

```
Entrée [10]: News_df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 182 entries, 0 to 181
Data columns (total 13 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   id              182 non-null    object
 1   title           182 non-null    object
 2   text            182 non-null    object
 3   url             174 non-null    object
 4   top_img         172 non-null    object
 5   authors         141 non-null    object
 6   source          174 non-null    object
 7   publish_date    133 non-null    object
 8   movies          25 non-null     object
 9   images          172 non-null    object
 10  canonical_link  170 non-null    object
 11  meta_data       182 non-null    object
 12  news_type       182 non-null    object
dtypes: object(13)
memory usage: 18.6+ KB
```

```
Entrée [11]: News_df.describe()
```

Out[11]:

	id	title	text	url	top_img	authors	source	publish_date	movies
count	182	182	182	174	172	141	174	133	25
unique	182	178	178	171	166	90	28	118	25
top	Fake_44-Webpage	A Hillary Clinton Administration May be Entire...	We're shocked — SHOCKED — to learn that CNN's ...	http://eaglerising.com/36880/a-hillary-clinton...	http://static.politico.com/da/f5/44342c424c68b...	Terresa Monroe-hamilton	http://politi.co	{'$date': 1474588800000}	https://www.youtube.com/embed/z9GptmyPn5A?rel=...	http://static./44342
freq	1	2	2	2	3	8	32	5	1

Après un premier coup d'œil sur les données, on décide de ne pas prendre en compte certaines features car elles présentent des informations redondantes. On garde donc seulement les colonnes qui nous intéressent.

A cette étape on obtient un DataFrame de la sorte :

```
Entrée [13]: New_df_clean
```

Out[13]:

	title	text	source	movies	images	news_type
0	Another Terrorist Attack in NYC...Why Are we STI...	On Saturday, September 17 at 8:30 pm EST, an e...	http://eaglerising.com	NaN	http://constitution.com/wp-content/uploads/201...	Real
1	Donald Trump: Drugs a 'Very, Very Big Factor' ...	Less than a day after protests over the police...	http://abcn.ws	NaN	http://www.googleadservices.com/pagead/convers...	Real
2	Obama To UN: 'Giving Up Liberty, Enhances Secu...	Obama To UN: 'Giving Up Liberty, Enhances Secu...	http://rightwingnews.com	https://www.youtube.com/embed/ji6pl5Vwrvk	http://rightwingnews.com/wp-content/uploads/20...	Real
3	Trump vs. Clinton: A Fundamental Clash over Ho...	Getty Images Wealth Of Nations Trump vs. Clint...	http://politi.co	NaN	https://static.politico.com/dims4/default/8a1c...	Real
4	President Obama Vetoes 9/11 Victims Bill, Sett...	President Obama today vetoed a bill that would...	http://abcn.ws	NaN	http://www.googleadservices.com/pagead/convers...	Real
...	...	...	...	...	...	...
177	Hillary's TOP Donor Country Just Auctioned	Hillary's TOP Donor Country Just Auctioned	http://rightwingnews.com	NaN	http://1.gravatar.com/avatar/d35b77ff6c3900715...	Fake

La colonne ‘title’ contient le titre de chaque article la colonne ‘text’ contient le corps de chaque article, on a également la source, le lien vers une vidéo ou une image s’il y’en a et finalement une nouvelle colonne créée donnant le type de la nouvelle de l’article correspondant, vraie ou fausse.

Or dans ce cas, il subsiste des champs vides qui rendent le DataFrame encore impossible à exploiter pour de la détection de fausse nouvelle. On choisit donc de remplacer les colonnes ‘movies’ et ‘images’ par ‘Has_movies’ et ‘Has_image’ qui spécifient si l’article en question contient ou non une vidéo ou une image. Ces features fabriquées sont d’une part exploitables et supposément plus pertinentes pour de l’analyse de données.

Finalement il reste à supprimer les lignes contentant des cases vides et on obtient un DataFrame exploitable.

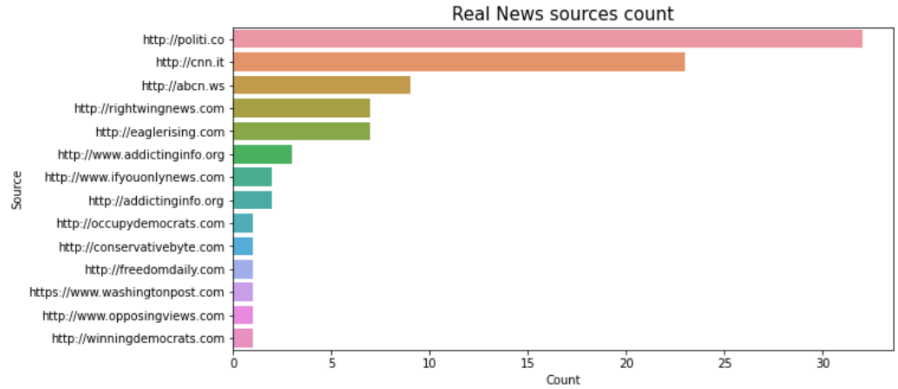
```
Entrée [21]: New_df_clean.info()

<class 'pandas.core.frame.DataFrame'>
Int64Index: 174 entries, 0 to 181
Data columns (total 6 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   title       174 non-null    object
 1   text        174 non-null    object
 2   source      174 non-null    object
 3   news_type   174 non-null    object
 4   Has_movie   174 non-null    int64 
 5   Has_images  174 non-null    int64 
dtypes: int64(2), object(4)
memory usage: 9.5+ KB
```

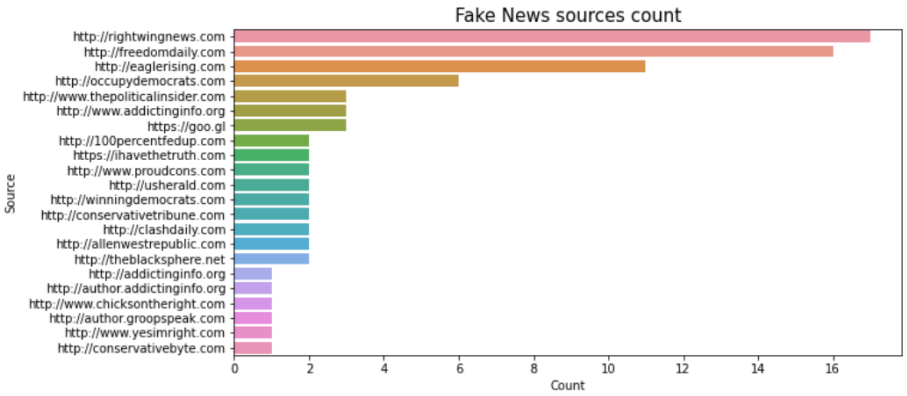
3. Questionnement et analyse de données

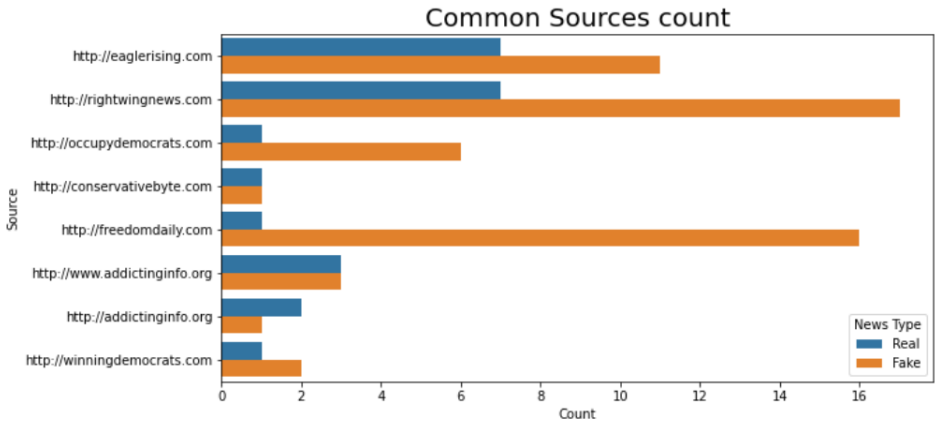
Dans cette partie, on se pose les questions sur l’influence et la pertinence des données sur notre cible de prédiction : La présence ou non de médias dans l’article présente-elle une forte corrélation avec véridicité de ce dernier ? Quels sont les sources les plus fiables ? Les moins fiables ?...

Pour répondre à ces questions nous allons extraire les informations nécessaires du précédent DataFrame et les illustrer sous forme de graphique pour en faciliter l’analyse.

Les graphiques suivants illustrent le nombre total d'articles publié par chaque source en fonction du type de la nouvelle (Fake/Real) ainsi que les sources ayant publié des Fake News et des Real News simultanément.





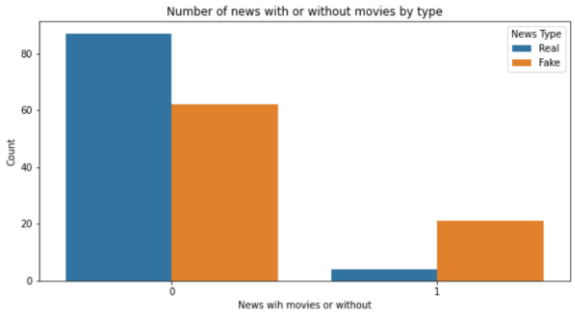


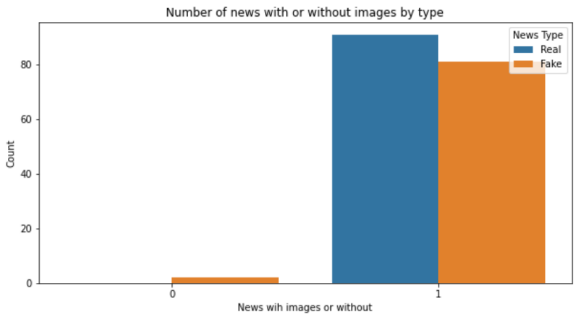
A partir de ces graphiques ont peut porter quelques conclusions sur la fiabilité des sources, en effet les sources ayant publié le plus de « Real News » ne sont pas présentes dans le graphique des sources communes, on peut dire que ces sources sont « fiables ». Au contraire les sources présentant le plus de « Fake News », même si elles ont publié en même temps un certain nombre de « Real News », peuvent être considérés comme non fiable.

De plus on retrouve plus de sources de « Fake News » que de « Real News », ceci, dans un jeu de données plus conséquent pourrait être interprété par la une accessibilité aux « Fake News » plus facile.

Enfin, sur le dernier graphique on remarque une certaine variation entre le nombre de vrais ou faux articles pour chaque source. Par conséquent la source représente une donnée pertinente pour la détection de « Fake News ».

Pour la suite, on étudie la répartition du nombre d'articles contenant des images/vidéos en fonction de leur type.





On constate que la plupart des articles ne contiennent pas de vidéo mais ils présentent des images, d'autre part il n'y a pas beaucoup de variation entre « Fake News » et « Real news ». Donc les données de la présence de médias ne semblent pas être très pertinente pour la détection de Fake News.

Pour la suite on gardera comme features : Le titre, le corps du texte et la source.

4. Preprocessing de données textuelles

L'objectif étant de détecter les fausses nouvelles, on utilisera la méthode de la classification. On possède deux classes (Fake ou Real) et on aimera détecter si une nouvelle est vrai ou fausse à partir de son titre, du texte contenu dans l'article et de la source.

La principale difficulté réside dans le fait que les donné sont entièrement textuelles, il faut donc adapter ces données aux méthodes d'apprentissage que nous allons utiliser.

Pour cela nous allons procéder par étapes. Dans un premier temps nous allons normaliser le texte, c'est-à-dire retirer les majuscules, la ponctuation, les chiffres et les caractères spéciaux. Ensuite nous retirerons les mots de liaison (stopwords) pour ne garder que les mots principaux présents dans le texte. A cette étape pour éviter les variations entre les différentes versions d'un même mot on utilisera la fonction stem pour ne garder que la racine. On obtient finalement une liste de mots adéquate pour appliquer nos différents modèles.

Ensuite on combine les méthodes de vectorisation de texte et TF-IDF pour donner un poids plus important aux termes les moins fréquents, qui sont les plus discriminants et donc les plus pertinents pour la classification. La transformation TF-IDF se fait selon la formule :

$$i\,df_i = \log\left(\frac{|D|}{|\{d_j : t_i \in d_j\}|}\right)$$

Avec $|D|$ nombre total de d'articles et $\left|\{d_j : t_i \epsilon d_j\}\right|$ nombre d'articles où le terme apparaît.

5. Prédiction et tests

Dans cette partie nous allons entrainer trois modèles de classification différents et comparer les résultats obtenus.

Modèle des plus proches voisins :

```
Entrée [217]: print(classification_report(y1_test, predict_KNN))
```

	precision	recall	f1-score	support
Fake	0.83	0.62	0.71	24
Real	0.65	0.85	0.74	20
accuracy			0.73	44
macro avg	0.74	0.74	0.73	44
weighted avg	0.75	0.73	0.73	44

Modèle arbres de décision :

```
Entrée [221]: print(classification_report(y1_test, predict_Dectree))
```

	precision	recall	f1-score	support
Fake	0.59	0.54	0.57	24
Real	0.50	0.55	0.52	20
accuracy			0.55	44
macro avg	0.55	0.55	0.54	44
weighted avg	0.55	0.55	0.55	44

Classification perceptron multicouche :

```
Entrée [225]: print(classification_report(y1_test, predict_MLP))
```

	precision	recall	f1-score	support
Fake	0.86	0.75	0.80	24
Real	0.74	0.85	0.79	20
accuracy			0.80	44
macro avg	0.80	0.80	0.80	44
weighted avg	0.80	0.80	0.80	44

En comparant les différents modèles sur leur précision, on constate que les méthodes de perceptron multicouche et plus proches voisins sont plus précises car plus adéquate aux données textuelles vectorisées. Cependant la précision reste faible, on pourra détecter le type d'une nouvelle avec une précision de 0.75 % avec les plus proches voisins et 80% avec le perceptron multicouche.

6. Conclusion

A travers ce projet, on se rend bien compte de l'importance du traitement des données textuelles en amont de l'apprentissage sur ces données car cela peut influer sur les précisions de certains modèles.

Ensuite, il est possible d'explore la piste de visualisation des données sous formes de graph pour lequel nos nœuds représenteront les nouvelles qui sont liées si les utilisateurs ayant posté ces données sont liés au niveau du réseau social.