



**UNIVERSITÉ  
RENNES 2**

PROJET SUR LA FOUILLE DE DONNÉES TEXTUELLES (*TEXT MINING*)

---

## Wenger vs Mourinho : qu'en pense Twitter?

---



PIERRE BUREAU  
GUILLAUME LE FLOCH  
Année 2017-2018

*Encadrante : MME FABIENNE MOREAU*

# Table des matières

<b>1</b>	<b>Introduction et problématique</b>	<b>2</b>
<b>2</b>	<b>La récupération des données</b>	<b>3</b>
<b>3</b>	<b>Analyse préliminaire et nettoyage des données</b>	<b>4</b>
<b>4</b>	<b>Traitement global des tweets</b>	<b>5</b>
4.1	Une analyse simple de la fréquence des mots . . . . .	5
4.2	Une tentative d'amélioration . . . . .	7
4.3	Complément d'analyse : les ngrams . . . . .	9
<b>5</b>	<b>Traitement détaillé des tweets</b>	<b>13</b>
5.1	Analyse de sentiments . . . . .	13
5.2	Analyse des tendances : focus sur les hashtags . . . . .	15
<b>6</b>	<b>La confrontation directe entre Wenger et Mourinho</b>	<b>16</b>
<b>7</b>	<b>Bilan</b>	<b>16</b>
<b>8</b>	<b>Sources</b>	<b>16</b>

# 1 Introduction et problématique

Tout amateur de football, et plus particulièrement du championnat anglais de **Premier League**, est au courant de la rivalité qui existe entre **Arsène Wenger** et **José Mourinho**. En effet, l'histoire entre nos deux protagonistes est riche en péripéties, faisons un bref rappel des faits.

Lorsque José Mourinho est nommé manager du club londonien de **Chelsea** à l'été 2004, Arsène Wenger sort d'une saison incroyable avec le club rival d'**Arsenal**, puisque les *Gunnners* ont remporté le championnat anglais sans avoir perdu le moindre match parmi les 38 journées sur lesquelles se déroule la Premier League. Arsenal termine donc champion avec **90 points**, c'est-à-dire 11 points de plus que le second... Chelsea !

On comprend ainsi qu'Arsène Wenger était à ce moment-là l'homme à abattre, et José Mourinho, qui sortait d'un doublé historique (Championnat du Portugal-Ligue des Champions) avec le **FC Porto** était l' élu pour accomplir cette tâche. Tout est fait pour opposer ces deux-là : l'entraîneur français est connu pour sa classe, pour donner leur chance aux jeunes joueurs, pour faire jouer son équipe avec un style « léché » et offensif, quand son homologue portugais est décrit comme un grand tacticien, un entraîneur vicieux, au pragmatisme légendaire, qui s'appuie plutôt sur une défense de fer et des joueurs expérimentés. Mais ce n'est pas tout, José Mourinho est également un maître dans l'art de la provocation, pour déstabiliser l'adversaire. Il va ainsi lancer sans cesse des piques à Arsène Wenger ou Arsenal dans les médias, en conférence de presse avant ou après les matchs, ce qui va définitivement rendre la relation entre les deux managers tendue et mener à des altercations sur le bord du terrain lors des oppositions entre Arsenal et Chelsea, qui étaient déjà rivaux à la base. Chelsea terminera champion en 2005 et 2006, avant que Mourinho ne soit évincé. Il fera son retour à Chelsea en 2013, ce qui va raviver les tensions dans notre couple favori, remportera à nouveau la Premier League en 2015, puis sera à nouveau limogé 6 mois plus tard après une première partie de saison catastrophique. En 2016, il signe à Manchester United, autre rival historique d'Arsenal. L'histoire se poursuit donc encore aujourd'hui en 2018.

La situation actuelle est la suivante : Arsène Wenger est toujours en poste à Arsenal (depuis 1996) mais n'a plus remporté le championnat depuis cette fameuse saison 2003-2004. José Mourinho a terminé 6<sup>e</sup> du championnat après avoir dépensé une fortune sur le marché des transferts pour bâtir son équipe. Cette saison, Manchester United et Arsenal n'arrivent pas à tenir la cadence infernale du Leader de Premier league, Manchester City, et sont à la traîne en championnat. Cela fait 14 ans qu'Arsenal n'a pas remporté le championnat, l'équipe commet tout le temps les mêmes erreurs, tandis que pour Mourinho son équipe est ennuyée à regarder jouer (à cause de sa tactique du « *park the bus* » qui consiste à défendre très bas en utilisant tous ses joueurs de champ et donc à « garer le bus » devant son but pour empêcher l'adversaire de marquer), il n'a plus autant de succès qu'avant, c'est un « *chequebook-manager* » (dans le sens où il ne bâtit son succès qu'en dépensant des sommes astronomiques pour attirer des grands joueurs, par opposition à Wenger qui développe des jeunes talents n'ayant quasiment rien coûté).

En bref, aujourd'hui ces deux managers sont très décriés, beaucoup de fans d'Arsenal et de Manchester United semblent vouloir le limogeage de leur manager respectif, en tout cas si l'on s'en fie à ce que l'on peut voir dans les journaux anglais ou sur les réseaux sociaux.

Nous avons donc décidé de nous pencher sur le cas de ces deux personnalités du monde du football et de vérifier si ces hypothèses sont vraies, en analysant les tweets dans lesquels ils sont mentionnés. Notre problématique sera la suivante : **Quelle est l'opinion des utilisateurs de Twitter vis-à-vis d'Arsène Wenger et de José Mourinho ?**

Cette problématique est assez large, et pour cette raison, notre étude va s'articuler en plusieurs étapes pour y répondre :

- Une analyse préliminaire des données afin de procéder à une sélection de tweets pertinents
- Une analyse globale des tweets concernant chaque manager, afin de résumer les opinions
- Une analyse plus détaillée pour tenter d'extraire de vraies tendances aux niveaux des **hashtags** et de la **valeur sentimentale** des tweets
- Une confrontation directe entre Wenger et Mourinho en analysant les adjectifs utilisés pour les qualifier ainsi que les tweets dans lesquels les deux managers sont mentionnés

## 2 La récupération des données

Une fois la problématique choisie, l'étape suivante de cette étude a été de construire notre base de données. Comme nous souhaitions analyser l'opinion des utilisateurs de **Twitter**, nous avons procédé au *scraping* des tweets grâce à la librairie **tweepy** en Python, qui possède une API permettant de communiquer directement avec l'interface du célèbre réseau social.

Il est important de noter que l'extraction des tweets ne peut pas s'opérer sur des tweets anciens (c'est la réglementation de Twitter). Ce sont donc tous les tweets qui sont envoyés après le lancement du programme qui sont enregistrés et écrits dans un fichier au format *json*. Ensuite, vous pouvez laisser votre programme tourner indéfiniment du moment que la connexion internet n'est pas coupée. Cependant, il existe des limitations (*cf documentation complète dans les sources*) comme pour toute API. Pour notre cas particulier, les tweets qui ont été extraits contenaient soit le mot « wenger » soit « mourinho » (le code correspondant se trouve dans le script *twitter\_streaming.py*). En ce qui nous concerne, l'extraction s'est faite en 4 fois pour plusieurs raisons :

- Il fallait un volume de données conséquent afin d'avoir suffisamment de tweets à analyser après avoir effectué un nettoyage complet
- Nous ne pouvions pas laisser tourner le programme assez longtemps pour emmagasiner assez de tweets en une seule fois
- L'opinion des fans peut être biaisée à la suite d'un match perdu ou gagné par Arsenal et/ou Manchester United, nous avons donc souhaité capter des tweets à différents moments

Au final, nous nous retrouvons avec **1.60 Go** de données brutes à analyser, après agrégation des tweets extraits entre le **22 décembre 2017** et le **03 janvier 2018** (le code pour agréger les données correspond au script *data\_concat.py*). Ces données sont regroupées dans un fichier texte appelé *twitter\_data.txt*, qui est donc composé de tweets. Ces derniers sont au format *json*, c'est à dire que ce sont des **dictionnaires** en Python. Ils renferment énormément d'informations et sont de la forme suivante :

```
{
  created_at: "Wed Jan 03 08:17:09 +0000 2018",
  id: 948468259682168832,
  id_str: "948468259682168832",
  text: "Stewart Robson is with us now talking Arsenal v Chelsea, and Wenger v Mike Dean.\n\nGet your questions and comments i... https://t.co/uSAc2zEIz7",
  source: "<a href='\"https://about.twitter.com/products/tweetdeck\"' rel='\"nofollow\">TweetDeck</a>",
  truncated: true,
  in_reply_to_status_id: null,
  in_reply_to_status_id_str: null,
  in_reply_to_user_id: null,
  in_reply_to_user_id_str: null,
  in_reply_to_screen_name: null,
  user: {
    id: 401488459,
    id_str: "401488459",
    name: "Off The Ball",
    screen_name: "offtheball",
    location: "📍 - 📺 - 🎧 - 📱",
    url: "http://www.offtheball.com",
    description: "📺 #OTBAM weekdays 7:45-9am 📺 #OffTheBall Mon-Fri from 7pm, Sat-Sun from 1pm. 📧 sport@offtheball.com 📺 offtheballnt",
    translator_type: "none",
    protected: false,
    verified: true,
    followers_count: 82571,
    friends_count: 944,
    listed_count: 427,
    favourites_count: 1263,
    statuses_count: 33882,
    created_at: "Sun Oct 30 16:32:42 +0000 2011",
    utc_offset: 0,
    time_zone: "Casablanca",
    geo_enabled: true,
    lang: "en",
    contributors_enabled: false,
    is_translator: false,
    profile_background_color: "C0DEED",
    profile_background_image_url: "http://abs.twimg.com/images/themes/theme1/bg.png",
    profile_background_image_url_https: "https://abs.twimg.com/images/themes/theme1/bg.png",
    profile_background_tile: false,
    profile_link_color: "1DA1F2",
    profile_sidebar_border_color: "C0DEED",
    profile_sidebar_fill_color: "DDEEFF",
    profile_text_color: "333333",
    profile_use_background_image: true,
    profile_image_url: "http://pbs.twimg.com/profile_images/906172613533339649/6CRlue1x_normal.jpg",
    profile_image_url_https: "https://pbs.twimg.com/profile_images/906172613533339649/6CRlue1x_normal.jpg",
    profile_banner_url: "https://pbs.twimg.com/profile_banners/401488459/1513780128",
    default_profile: true,
    default_profile_image: false,
  }
}
```

Dans notre étude, nous allons nous servir principalement de la clé 'text' qui renferme le contenu des (désormais) 280 caractères maximum à disposition de l'utilisateur pour s'exprimer. La clé 'language' nous servira également dans l'analyse préliminaire pour sélectionner les tweets qui seront dans notre base finale à exploiter.

### 3 Analyse préliminaire et nettoyage des données

Extraire tous les tweets qui contiennent une chaîne de caractère donnée n'est pas compliqué, en revanche on peut se douter à l'avance que tout ne sera pas pertinent et que l'on va rencontrer certains problèmes. C'est pour cette raison qu'il nous fallait une base conséquente au départ, car elle va se retrouver largement affinée par la suite.

Un problème que nous n'avons pas eu, mais que nous aurions pu rencontrer concerne les différents sens du mot cible qui nous sert à effectuer l'extraction des tweets. Par exemple, si nous avons voulu nous pencher sur le cas de l'actuel manager de Chelsea, **Antonio Conte** et que nous avons seulement utilisé le mot clé « conte » pour y faire référence, nous nous serions retrouvés avec des tweets parlant de personnes d'origine africaine portant le même nom, ou bien encore par exemple d'un « conte de Noël ». Et si vous renseignez « antonio conte » (l'API ne tient pas compte de la casse) vous ne pourrez extraire que les tweets contenant la chaîne complète, ce qui réduira donc largement le champs des tweets accessibles.

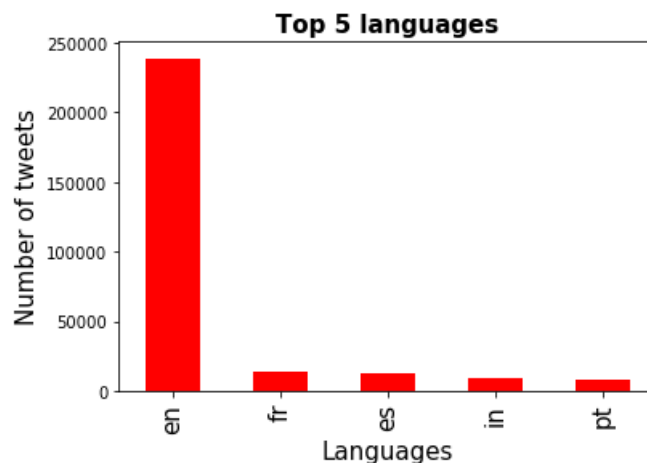
En ce qui nous concerne, il n'existe pas d'ambiguïté sur Mourinho et Wenger, ce sont des noms propres et aucune autre personnalité ne porte le même. En revanche, lorsque les fans ou journalistes parlent d'eux ils utilisent parfois « Mou » ou « José » pour parler de Mourinho et « AW », « the Boss » ou encore « Big Wengz » concernant Arsène Wenger. Nous n'avons donc pas pu capter tout ce qui se disait sur eux.

Néanmoins, nous avons réussi à extraire un total de **304 512** tweets. Un premier nettoyage a consisté à enlever les tweets pour lesquels il n'y avait pas de texte (ce qui correspond à une liste vide associée à la clé 'text' dans le dictionnaire python). En effet, un autre problème lié à la fouille de texte est que l'on ne peut pas exploiter les images, les GIFs ou encore les vidéos qui sont pourtant très porteurs de sens. Après avoir procédé à cette opération, il nous reste tout de même **304 488** tweets.

Un autre problème que l'on peut rencontrer avec les tweets est la présence de **doublons** ou de tweets qui sont quasiment des doublons. Cela concerne les fameux **retweets** : quand une personne va retweeter un tweet, cela va apparaître sous la forme « RT : blabla ». Pour régler ce problème, nous avons donc décidé de supprimer tous les tweets commençant par « RT », on ne garde ainsi que le tweet original.

Le problème des « quasi-doublons » était quant à lui dû aux liens (vers un article de presse par exemple) : on retrouve à chaque fois le même contenu textuel, mais puisque le lien est différent, la méthode `drop_duplicates()` de la librairie **Pandas** en Python ne détecte pas le doublon. La solution que nous avons choisie consiste tout simplement à supprimer au préalable tous les liens dans les tweets à l'aide d'une expression régulière.

Un autre aspect à prendre en compte est la langue dans laquelle est écrit le tweet. En effet, Arsenal et Manchester United sont des clubs de football suivis dans le monde entier, des fans de tous les pays tweetent à propos de ces 2 entités et de leurs managers. Nous avons donc décidé de regarder un *Top 5* des langues dans lesquels nos tweets sont écrits.



Les résultats sont sans appel, une écrasante majorité des tweets sont rédigés en anglais, ce qui n'est pas surprenant puisque la Premier League est beaucoup suivie au Royaume-Uni, en Amérique du Nord, en Inde et au Nigeria. Le français représente la deuxième langue, probablement grâce à Arsène Wenger et à la communauté francophone d'Arsenal. Cependant la quantité de tweets est insuffisante, puisque les doublons n'ont pas encore été enlevés. Nous allons donc conserver uniquement les tweets en anglais.

Après avoir effectué ces différentes opérations, nous allons stocker les tweets concernant José Mourinho dans un dataframe *mourinho.csv*, idem pour Arsène Wenger (*wenger.csv*). Au final, nous avons 26087 tweets pour José Mourinho et 28244 tweets pour Arsène Wenger, à analyser.

## 4 Traitement global des tweets

Lorsqu'on parle de traitement global des tweets (par opposition au traitement détaillé que l'on verra par la suite), nous faisons référence à une première approche assez basique qui consiste à regarder la fréquence des mots dans notre corpus. Pour ce faire, nous avons utilisé les modules de la librairie Python **NLTK** (Natural Language ToolKit). Le code est disponible dans le script *traitement\_nltk.py*. Le but est de regarder les mots les plus fréquents afin d'obtenir un premier résumé des tweets concernant chaque manager, pour créer un **nuage de mots** à l'aide de l'outil [wordle.net](http://wordle.net). Nous trouvons que cette représentation est visuelle et permet d'obtenir une première tendance en ce qui concerne les thèmes liés à nos protagonistes. Pour parvenir à une représentation « propre », il faut effectuer quelques traitements au préalable :

- Enlever les « RT » dans les tweets car ils ne nous sont d'aucune utilité
- Enlever les noms d'utilisateurs (qui sont précédés d'un « @ ») car nous jugeons qu'ils pourraient polluer l'analyse
- Enlever tous les éléments de la ponctuation et les accents à l'aide d'expressions régulières
- Enlever les mots vides et en ajouter certains qui ne figurent pas dans la liste implémentée par le module *nltk*, puis enlever les hashtags puisqu'ils seront traités séparément à la fin de cette étude
- Supprimer des caractères « parasites » propres à l'extraction des tweets et que les expressions régulières n'ont pas pu enlever comme '...' ou encore les emojis : pour cela nous avons implémenté la fonction **remove\_useless** qui ne conserve que les chaînes de caractères contenant seulement des chiffres, des lettres et/ou des symboles monétaires (qui ont leur importance dans notre analyse)
- Regrouper tous les mots en un seul paragraphe (un pour chaque manager) afin de former notre propre corpus
- Enlever les mots « wenger » et « mourinho » qui vont être les plus représentés, puisque leur fréquence sera supérieure ou égale au nombre de tweets, car ils risquent d'écraser les autres mots

Une fois ce travail effectué, on peut se pencher sur l'analyse des corpus « propres » que nous avons formé.

### 4.1 Une analyse simple de la fréquence des mots

Ce qui ressort de cette première analyse s'est retrouvé assez influencé par la période de scraping des tweets. Il est donc logique de trouver dans nos résultats le nom des adversaires d'Arsenal et de Manchester United, ainsi que les noms de certains joueurs ayant fait l'actualité dans ces matchs. A côté de ces aspects, on notera tout de même certains points intéressants que nous allons détailler tout de suite avec les nuages de mots.

En ce qui concerne Arsène Wenger, le résultat visuel est à la page suivante.











La différence n'est pas flagrante, et dans notre analyse elle n'apporte rien de significatif et aurait même tendance à dégrader la qualité des résultats avant de procéder à une normalisation des mots. Nous allons donc nous pencher sur l'analyse des **ngrams** pour tenter d'enrichir l'analyse textuelle.

En ce qui concerne les ngrams, les traitements appliqués seront les mêmes que précédemment, à la différence près que l'on va conserver les hashtags ici (le code se trouve dans le script `n_grams.py`). On regarde la fréquence des ngrams (bigrammes, trigrammes et quadrigrammes) en supprimant à la main les premiers qui ne sont pas pertinents et qui risqueraient d'écraser les autres, voici ce que l'on trouve pour Arsène Wenger.











## 5 Traitement détaillé des tweets

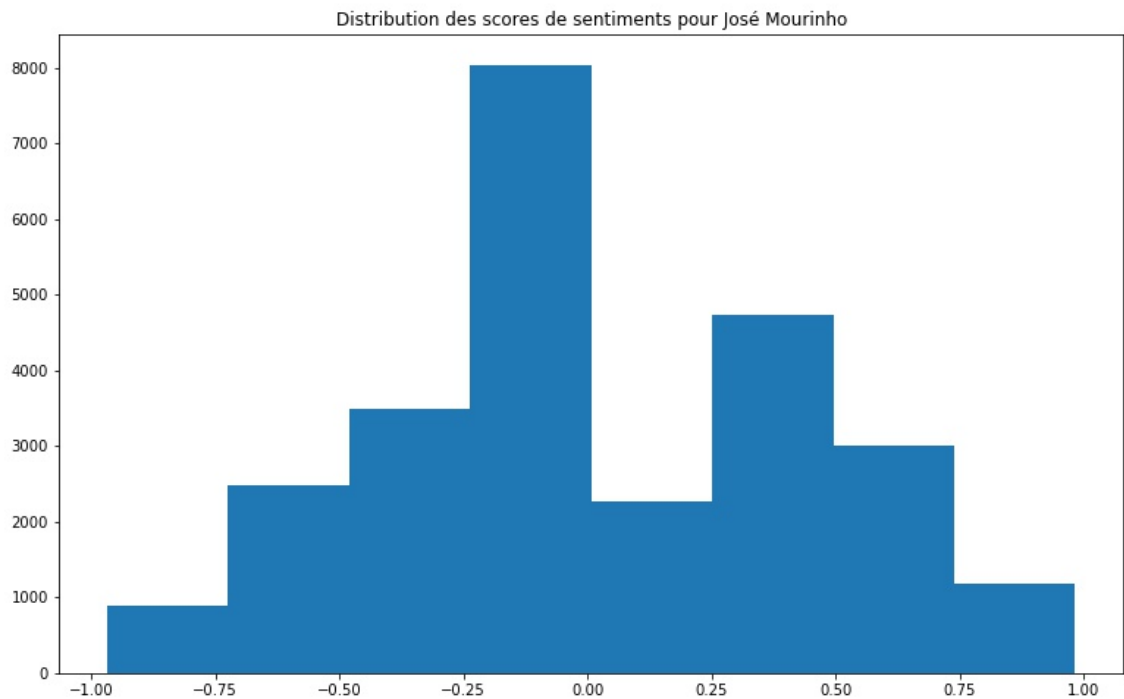
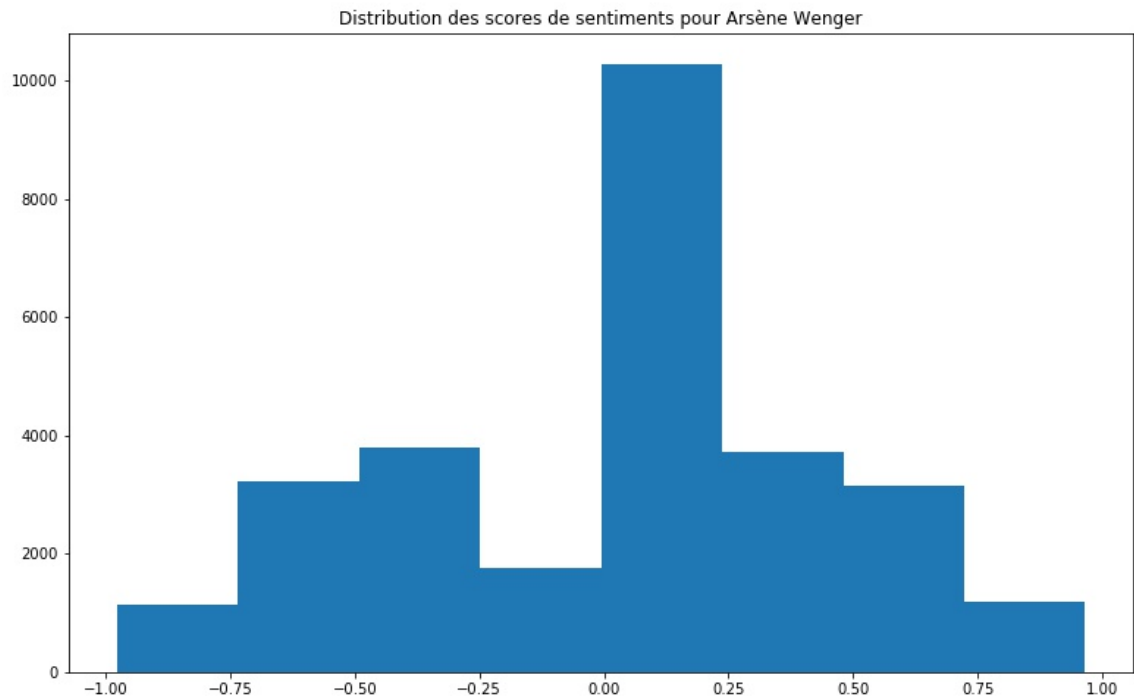
### 5.1 Analyse de sentiments

Dans cette partie, notre but pour comparer les deux managers est de calculer un score de popularité. Nous allons pour cela utiliser un module particulier de la librairie **nltk** nommé **SentimentIntensityAnalyzer**. Quand on utilise ce module, la casse et la ponctuation sont importantes, pour cette raison nous laisserons la ponctuation et ne mettrons pas en minuscule les tweets. Tous les autres traitements effectués précédemment (enlever les « RT », noms d'utilisateurs, emojis, etc) seront appliqués (le code correspondant se trouve dans le script *sentimentAnalysis.py*). Commençons par un exemple visuel pour bien comprendre comment fonctionne ce module.

```
I love this game
compound: 0.6369,
neg: 0.0,
neu: 0.323,
pos: 0.677,
I LOVE THIS GAME !
compound: 0.6696,
neg: 0.0,
neu: 0.308,
pos: 0.692,
Think Wenger was spot on today, congratulations to the team and the manager, credit where it's due !
compound: 0.7777,
neg: 0.0,
neu: 0.688,
pos: 0.312,
The manager is fucking shit #WengerOut !
compound: -0.6352,
neg: 0.456,
neu: 0.544,
pos: 0.0,
He just parks the bus and his teams play unattractive football, just sack Mourinho already ffs I've had enough of this cunt !
compound: -0.8534,
neg: 0.345,
neu: 0.578,
pos: 0.077,
```

Ici on prend tout d'abord un exemple simple pour montrer la différence apportée à une même phrase par la casse et la ponctuation, puis nous prenons des exemples de tweets que nous avons jugés positifs et négatifs. Les résultats renvoyés par le module sont en accord avec ce que nous pensions et reflètent assez bien l'intensité (positive ou négative) des tweets (le score final correspond à « compound »).

Notre base de tweets étant tout de même conséquente, on ne va pas pouvoir s'amuser à regarder le score de chaque tweet un à un. Cependant, pour tenter de résumer l'information nous avons décidé de calculer le score moyen de popularité (avec l'écart type associé) pour chaque manager et de représenter la distribution des scores à l'aide d'un histogramme. On rappelle que les scores sont distribués entre -1 et 1. Pour Wenger, le score moyen est de -0.0017 (écart type de 0.4196) et pour Mourinho on obtient 0.0482 (écart type de 0.4232). Infime avantage pour José Mourinho selon ces indicateurs, même si on voit que l'écart type est très élevé. En revanche, en ce qui concerne la distribution des scores, la donne est différente. On a plutôt l'impression que les sentiments à l'égard de Wenger sont positifs globalement et que c'est le contraire pour José Mourinho. Les représentations graphiques sont les suivantes.



Un des problèmes est que beaucoup de tweets sont écrits par des journalistes et vont être « neutres », et vont finalement polluer cette analyse. On va tout de même utiliser ces scores pour classer des tweets comme étant positifs si le score est supérieur à 0.25 et négatif s'il est inférieur à -0.25. Cela nous permet d'éliminer les tweets dits « neutres », et nous allons faire une analyse plus poussée de ces tweets dans la dernière partie sur la confrontation entre les deux managers.





La première image concerne Arsène Wenger et outre les hashtags d'encouragement pour l'équipe (#COYG pour « Come On You Gunners ») et autres hashtags classiques, on remarque la prédominance du #WengerOut (196 en tout) ce qui semble montrer la tendance quant au désir d'une partie des supporters d'Arsenal. Pour Mourinho c'est aussi similaire, on retrouve les mêmes types de hashtags et un #MourinhoOut mais moins marqué (98 occurrences soit moitié moins que Wenger).

Si l'on s'en fie uniquement aux hashtags, il semblerait donc que le manager français d'Arsenal soit soumis à une plus grosse pression venant de ses supporters que son homologue portugais de Manchester United. Le bilan n'est pas pour autant positif pour José Mourinho...

Notons tout de même qu'en général les gens auront plus tendance à s'exprimer pour critiquer quand les choses ne vont pas bien, que pour féliciter le manager après une victoire de l'équipe qu'il vont juger comme étant normale, presque comme étant une formalité. Cela peut donc biaiser les résultats.

Autre problème potentiel qu'il ne nous est pas possible de détecter : l'ironie ! Sur Twitter, les gens aiment plaisanter et par exemple on peut trouver des tweets du style « Another defeat for Arsenal... #WengerIn » ou au contraire, des fervents supporters d'Arsène Wenger qui tweetent « A brilliant win today by Arsenal, Mesut Özil was superb, but of course these idiots still want #WengerOut ». Il est donc compliqué d'extraire le véritable sens d'un tweet de façon automatique, et il y aura donc quelques erreurs. Néanmoins, les résultats jusqu'ici tendent tout de même à confirmer le sentiment général concernant chaque manager. Nous allons maintenant nous pencher sur la confrontation entre Arsène Wenger et José Mourinho, en regardant de quelle façon ils sont décrits sur Twitter, puis en analysant dans un second temps les tweets dans lesquels les mots « wenger » et « mourinho » figurent en même temps.

## 6 La confrontation directe entre Wenger et Mourinho

## 7 Bilan

## 8 Sources

- Réglementation de Twitter, <https://developer.twitter.com/en/developer-terms/agreement-and-policy>
- Scraping de tweets, <https://marcobonzanini.com/2015/03/02/mining-twitter-data-with-python-part-1/>
- Analyse de sentiment, <http://www.nltk.org/howto/sentiment.html>
- Elements des cours et des TP de Text Mining vus pendant le semestre