SCIENCES SORBONNE UNIVERSITÉ

Rapport PLDAC

Analyse de tweets, détection de sentiments et de communautés

Julie Lascar, Matéo Malidin
julie.lascar@etu.sorbonne-universite.fr
mateo.malidin@etu.sorbonne-universite.fr

25 mai 2022

Enseignants encadrants :
Nicolas BASKIOTIS
Vincent GUIGUE

Table des matières

1 Introduction

Les réseaux sociaux en ligne comme Twitter représentent de véritables mines d'informations dès lors que l'on sait en interpréter les interactions. En effet c'est environ 500 millions de messages qui sont publiés sur Twitter chaque jour. Cette source de connaissances est devenue un outil incontournable pour les entreprises, mais aussi pour les personnalités politiques. C'est d'ailleurs ce deuxième aspect qui va nous intéresser dans cette étude. La principale méthode utilisée en politique est le sondage d'opinions (ou détection des sentiments) quant à une cible déterminée. La grande difficulté de la détection de sentiments sur les réseaux sociaux en ligne réside dans l'ambiguïté des messages qui y sont postés. En effet, les utilisateurs de réseaux sociaux sont bien connus pour leur ironie. La détection des sentiments dans un tel contexte représente donc un réel défi [7] et l'unique emploi de méthodes d'analyse des messages textuels semble incomplet à moins d'être capable de comprendre toute l'expressivité de ces derniers. C'est dans cette idée que s'inscrit ce projet. En effet, nous proposons de combiner une analyse topologique du réseau d'interactions entre les utilisateurs, à une analyse des contenus textuels des messages. Nous précisons, que ces travaux sont effectués sur des messages provenant de Twitter, et plus précisément sur un ensemble de tweets collectés au cours de la campagne présidentielle 2017. La section 2 présente les données sur lesquelles nous travaillons.

Notre démarche se divise donc en trois grandes approches : (1) l'étude topologique du réseau d'interactions entre les utilisateurs afin d'y découvrir des mécanismes particuliers d'interactions, (2) l'analyse des contenus textuels des tweets pour la détection des cibles et des sentiments exprimés dans les tweets, et (3) la combinaison de ces deux approches pour améliorer les résultats de la classification.

La section 3 présente l'étude topologique du réseau d'interactions. L'étude topologique du réseau d'interactions entre les utilisateurs consiste à analyser la structure du réseau, à découvrir des structures particulières ou des mécanismes d'interactions. Les réseaux complexes tels que celui sur lequel nous travaillons partagent des spécificités [9] qui jouent un rôle important dans les résultats obtenus. La principale caractéristique qu'il nous faut traiter est la très faible connexité du réseau d'interactions entre les utilisateurs. La principale tâche qui nous intéresse est la détection de communautés topologiques au sein du réseau d'interactions. Une communauté étant un sous-graphe du réseau avec une connexité importante et peu d'interactions avec les autres communautés. De nombreuses approches ont été proposées dans la littérature pour répondre au besoin de regrouper les utilisateurs en communautés [13]. Une des méthodes les plus utilisées est la méthode de Louvain qui fut introduite en 2008 par Vincent D. Blondel et al. [2]. Notre choix s'est porté sur cet algorithme car ce dernier a la particularité de très bien passer à l'échelle sur des graphes très volumineux.

La section 4 présente l'analyse des contenus textuels des tweets. L'analyse des contenus textuels des tweets consiste à construire des classifieurs de cibles et de sentiments en nous appuyant sur une représentation des contenus textuels des tweets. Pour ce faire, nous avons utilisé une représentation des tweets sous forme de sacs de mots (BagOfWords) avec un certain nombre de pré-traitements. La principale difficulté des tâches de classification réside dans la nature non-supervisée du problème. En effet, aucun étiquetage n'est disponible sur cet ensemble de données. Concernant l'identification de la cible des tweets, nous étiquetons les tweets dont le ou les hashtags mentionnent une cible unique (un seul candidat). En ce qui concerne la détection des sentiments exprimés dans les tweets, nous utilisons en premier lieu une méthode de Transfer Learning [16] en entraînant un premier modèle sur un corpus d'avis concernant des films (Allociné), les classificateurs construits à partir des critiques s'avérant être les plus généralisables aux autres flux [10]. Les résultats n'étant pas convaincants, nous avons changé de méthode. Nous avons ensuite généré un étiquetage en agrégeant une partie des hashtags par cible et polarité. A partir de ces données étiquetées, nous avons construit plusieurs classifieurs en faisant varier les pré-traitements et nous avons comparé leurs performances avec les classifieurs de sentiments de l'algorithme BERT [5].

La combinaison des deux approches est décrite dans la section 5. La combinaison de ces deux approches consiste à utiliser l'étiquetage produit en analysant les contenus textuels des tweets pour enrichir le réseau d'interactions. Il s'agit d'analyser la répartition des classes dans les différentes communautés détectées. En utilisant les relations entre tweets, et plus particulièrement entre tweets et retweets, nous propageons les étiquettes de tweet en tweet. Ce procédé permet d'amplifier significativement l'étiquetage issu de l'analyse des contenus textuels des tweets à la condition que les étiquettes initiales soient correctement réparties dans les nombreuses composantes connexes du réseau d'interactions entre les utilisateurs.

2 Présentation des données

2.1 Les spécificités de Twitter

Actuellement, Twitter est l'une des plateformes de microblogage les plus populaires ; elle compte 436 millions d'utilisateurs qui publient environ 500 millions de messages par jour.
Twitter permet aux usagers de partager des messages, des liens vers des sites Web externes, des images ou encore des vidéos. Les messages qui sont publiés sur les microblogs sont courts, contrairement aux blogs traditionnels.

Twitter comporte certaines caractéristiques que nous présentons ci-dessous [7] :

- Un **tweet** est un message unique publié sur Twitter. Le contenu d'un tweet contient au maximum 280 caractères.
- Les **mentions** dans un tweet indiquent que le message mentionne un autre utilisateur. Pour faire cette référence, les utilisateurs utilisent le symbole @ suivi du nom d'utilisateur spécifique auquel ils font référence (@username).
- Les **réponses** dans un tweet sont utilisées pour indiquer que le message est une réponse à un autre tweet. Comme les mentions, elles sont créées en utilisant le symbole @ suivi du nom d'utilisateur auquel elles font référence. Les réponses sont placées à côté du nom d'utilisateur qui crée la réponse.
- Les **followers** font référence aux utilisateurs qui suivent l'activité d'un utilisateur. Suivre d'autres utilisateurs est le principal moyen de se connecter à d'autres utilisateurs sur Twitter. Les utilisateurs de Twitter reçoivent les mises à jour de ceux qu'ils suivent et envoient leurs mises à jour à ceux qui les suivent.
- Un utilisateur peut retweeter un tweet qu'il trouve intéressant afin de le rediffuser. Le **retweet** est considéré comme un outil puissant de diffusion de l'information. Le contenu du retweet est identique au tweet d'origine et il est marqué de l'abréviation RT suivie du nom d'utilisateur de l'auteur (RT @username).
- Les **hashtags** sont utilisés pour indiquer la pertinence d'un tweet par rapport à un certain sujet. Les hashtags, créés à l'aide du caractère # suivi du nom du sujet (#sujet) sont nés du besoin d'étiqueter les informations sur les messages postés. Les hashtags sont générés spontanément par les utilisateurs et la plateforme permet une recherche des tweets contenant un hashtag donné. Les hashtags qui apparaissent dans un nombre élevé de tweets sont caractérisés comme des sujets tendances.

2.2 Modèle de données

Les données étudiées sont extraites d'une base de données de tweets collectés durant la campagne présidentielle 2017. Deux jeux de données ont été extraits, un premier jeu regroupant les tweets publiés entre le 1er et le 15 avril, un deuxième jeu concerne les tweets publiés entre le 15 et le 30 avril. Etant donné le volume des données, nous nous sommes concentrés sur le premier jeu de données.
Le modèle de données est composé de sept tables dont les schémas sont les suivants :

- **hashs_0401_0415** (hash_id, hash),
- **medias_0401_0415** (id, #tweet_id, type, media_url, #source_tweet_id),
- **tweet_hash_0401_0415** (id, #tweet_id, #hash_id),
- **tweet_0401_0415** (tweet_id, text, created_at, geo_lat, geo_long, #user_id, favorite_count, favorited, in_reply_to_status_id, #in_reply_to_user_id, lang, quoted_status_id, #quoted_user_id, retweet_count, retweeted_status_id, #retweeted_user_id, source, place_country, place_bbox_origin_x, place_bbox_origin_y, place_bbox_corner_x, place_bbox_corner_y, place_centroid_x, place_centroid_y, place_type, place_full_name, place_area),
- **user_mentions_0401_0415** (id, #tweet_id, #source_user_id, #target_user_id),
- **users_0401_0415** (user_id, screen_name, name, location, url, description, created_at, followers_count, friends_count, statuses_count, lang, listed_count, verified),
- **users_unknown_0401_0415** (user_id).

Le modèle de données détaillé est fourni sur la plateforme de développement de Twitter [4]. L'annexe 1 décrit les tables et champs utilisés dans notre étude.

2.3 Analyse des données brutes

Le tableau 1 donne un premier aperçu des données contenues dans la base de données.

nombre d'utilisateurs :	662 705
nombre de tweets :	7 094 924
nombre de hashtags :	59 471

Tableau 1 – Aperçu des données contenues dans la base de données

Dans la suite, nous nous proposons de caractériser chaque type de données pour mieux les appréhender.

2.3.1 Caractérisation des utilisateurs

i) Influence des utilisateurs

Si nous ne disposons pas des relations entre les identifiants utilisateurs et les identifiants de leur amis et followers, nous disposons néanmoins du nombre d'amis et de followers pour chaque utilisateur. Ces valeurs montrent l'influence d'un utilisateur et sont importantes pour la suite de notre étude. Dans le tableau 2, nous remarquons que les données sont fortement dispersées, ce qui est attendu dans les bases de données utilisateurs extraites de réseaux sociaux.

statistiques	nombre d'amis	nombre de followers
minimum	0	0
maximum	1 502 622	68 090 337
moyenne	681	4 337
médiane	216	156
écart-type	4 675	216 647

Tableau 2 – Influence des utilisateurs

ii) Activité des utilisateurs

La proportion du nombre total de tweets par rapport au nombre d'utilisateurs de notre base semble montrer une faible activité des utilisateurs. Par activité, nous entendons la publication d'un tweet, qu'il soit un tweet simple, avec ou sans mention, une réponse à un tweet ou un retweet avec ou sans commentaire. La table 3 confirme la faible activité des utilisateurs présents dans notre base avec une médiane égale à 2, ce qui signifie que la moitié des utilisateurs a publié au plus deux tweets. Ceci peut s'expliquer par l'extraction très ciblée des données puisqu'elle a été réalisée à partir des tweets relatifs à la campagne présidentielle de 2017 et sur une période courte de quinze jours. Ces données sont importantes à prendre en compte dans la suite de notre travail puisqu'elles peuvent expliquer en partie la difficulté à regrouper les utilisateurs en communautés.

	nombre de tweets par utilisateur
minimum	1
maximum	10 001
moyenne	11
médiane	2

Tableau 3 – Activité des utilisateurs

2.3.2 Caractérisation des tweets

i) Longueur des tweets

La base de données contient des messages relativement courts, d'environ 19 mots en moyenne par tweet, dont la distribution est détaillée figure 1.

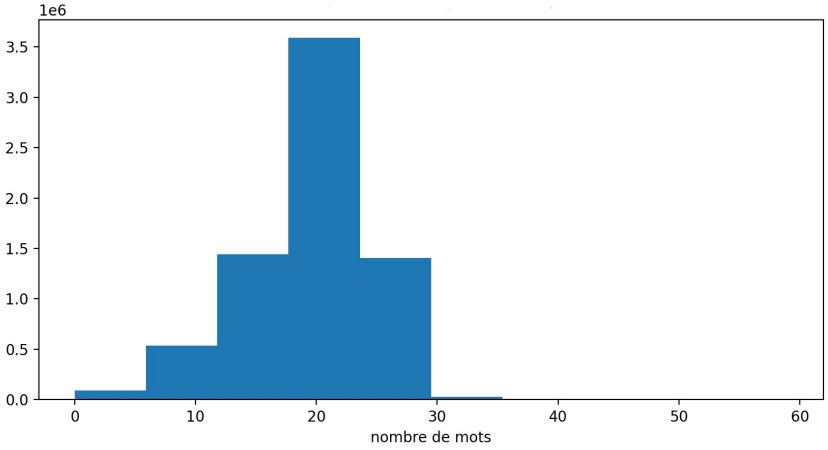


Figure 1 – Distribution du nombre de mots par tweet

ii) Répartition par type de tweet

On peut distinguer différents types de tweets :

- les tweets en réponses à un tweet
- les retweets
- les retweets avec commentaire
- les tweets simples (ni retweet avec ou sans commentaire, ni réponse à un tweet)

Certains tweets appartiennent à plusieurs de ces catégories. En effet ils peuvent être, à la fois, des réponses et des retweets avec ou sans commentaire. La répartition des types de tweets est illustrée sur la figure 2. Comme cela était prévisible les retweets représentent la plus grande proportion de tweets (près de 74% si on cumule les retweets seuls et les retweets combinés aux tweets avec commentaire).

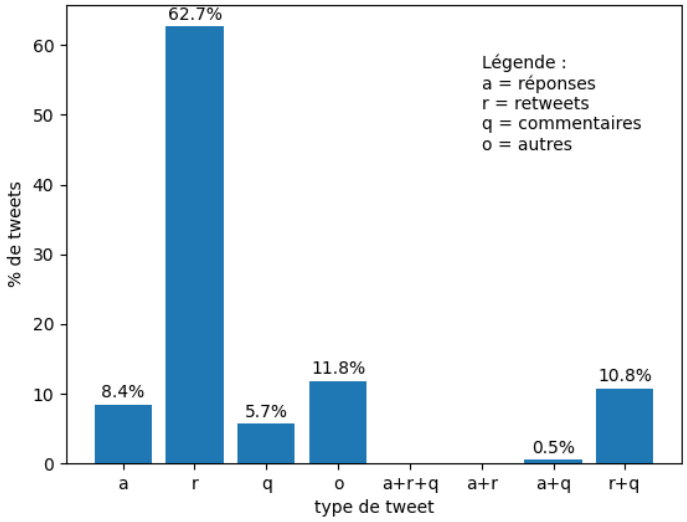
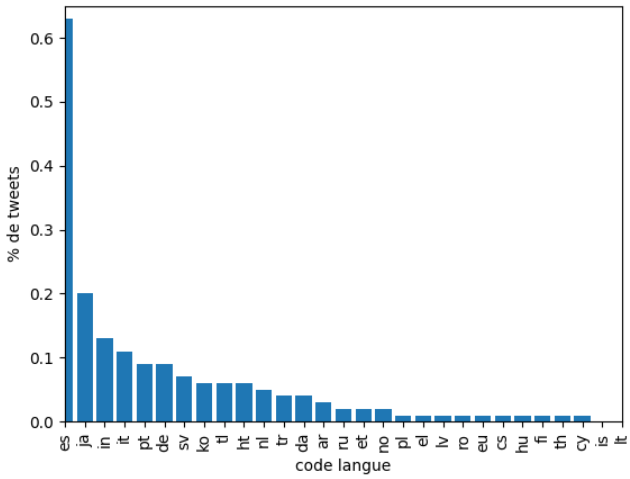


Figure 2 – Distribution par type de tweet

iii) Répartition des tweets par langue

Les langues détectées dans les tweets sont identifiées par leur code dans le champ 'lang'. Près de 88% des tweets sont en français (code : fr), les autres sont en langue étrangère. Parmi ces derniers, environ 8% sont en anglais (code : en) et près de 3% sont en langue indéterminée (code : und) (ce qui peut se produire lorsque plusieurs langues sont détectées dans un même tweet). La figure 3 illustre la distribution des langues détectées autres que français, anglais ou "indéterminé".



es	Espagnol
ja	Japonais
in	Indonésien
it	Italien
pt	Portugais
de	Allemand
sv	Suédois
ko	Coréen
tl	Tagalog (Philippines)
ht	Créole Haïtien

Quelques codes langues

Figure 3 – Répartition des tweets par langue

iv) Répartition temporelle

La base de données étudiée porte sur les tweets écrits entre le 1er et le 15 avril 2017. Les tweets étant horodatés, il est intéressant de regarder leur répartition temporelle sur cette période. Le graphe 4 illustre cette répartition. Nous y remarquons la présence d'un pic le 4 avril qui correspond au jour du débat du premier tour réunissant pour la première fois les onze candidats à l'élection. En regardant de plus près l'évolution du nombre de tweets sur la journée du 4 avril, illustrée sur la figure 5, le pic est bien confirmé pendant le débat (début du pic vers 19h00 UTC soit 21h00 en France).

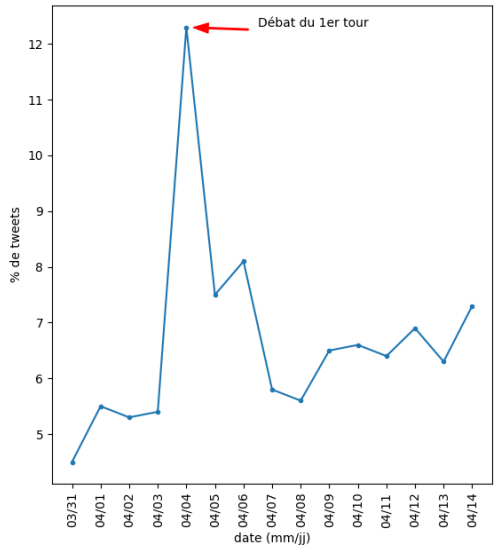


Figure 4 – Répartition temporelle des tweets

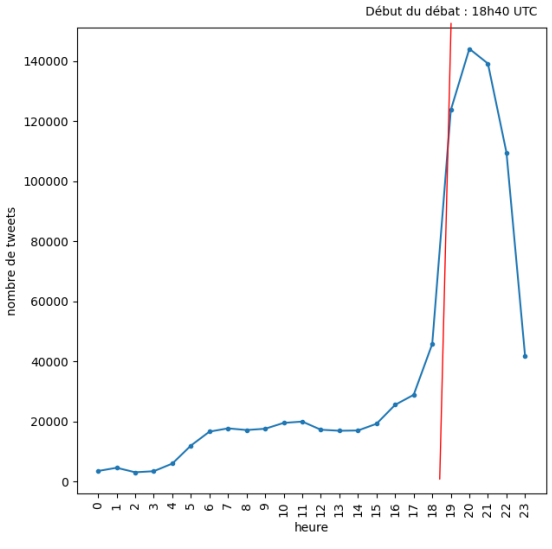


Figure 5 – Evolution sur la journée du 4 avril

2.3.3 Caractérisation des hashtags

Nous nous sommes intéressés à la fréquence d'apparition des hashtags. 59 471 hashtags sont répertoriés dans la base de données, pour 7 094 924 tweets. La figure 6 illustre leur fréquence d'apparition : à gauche, le tableau donne des informations sur les 10 hashtags les plus cités dans les tweets, à droite le nuage de mots réalisé à partir des cent hashtags les plus représentatifs.

hastag	nombre de tweets	% des tweets
macron	714 414	10%
fillon	513 792	7%
LeGrandDebat	417 995	6%
Mélenchon	238 001	3%
JLM2017	229 421	3%
EnMarche	120 973	2%
LePen	119 724	2%
Marine2017	117 529	2%
LEmissionPolitique	105 840	1%
Fillon2017	97 308	1%

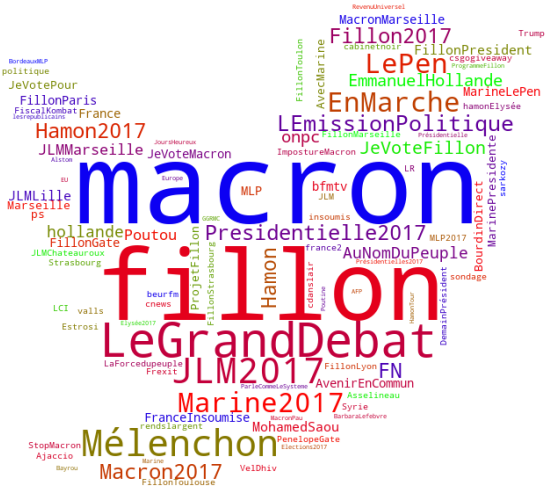


Figure 6 – Fréquence d'apparition des hashtags

Pour étudier plus précisément la distribution des hashtags sur l'ensemble des tweets et afin de mieux percevoir la présence des candidats, nous avons regroupé sous le nom #NomCandidat l'ensemble des hashtags relatifs à ce candidat. Par exemple, #EMacron, #EmmanuelleMacron et #EnMarche ont été remplacés par #Macron. De même, nous avons regroupé sous le nom #médias l'ensemble des hashtags relatifs aux médias. De plus, nous avons observé un grand nombre de hashtags exprimant un sentiment négatif envers les candidats Macron et Fillon (comme #ToutsaufMacron et #FillonGate), que nous avons regroupés sous les noms #antiMacron et #antiFillon. L'histogramme de la figure 7 représente la répartition des 15 hashtags prépondérants sur l'ensemble des tweets ainsi transformés.

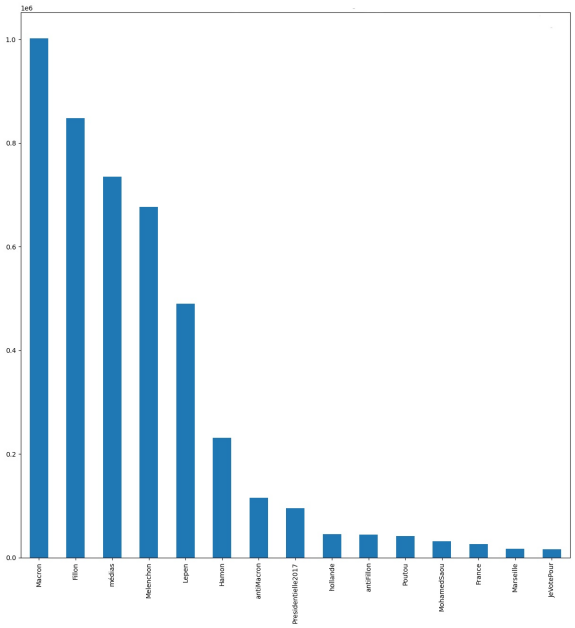


Figure 7 – Les hashtags prépondérants

3 Analyse du réseau d'interactions

3.1 Contexte

Les interactions observées sur les réseaux sociaux en ligne peuvent être modélisées par un réseau multiplexe où les sommets représentent les utilisateurs et les liens modélisent les différentes relations entre eux. Il est donc possible de modéliser ces réseaux par des graphes. Une synthèse des caractéristiques topologiques des graphes obtenus lors de la modélisation de ces activités est présentée dans un état de l'art par Rushed Kanawati [9].

i) Caractéristiques topologiques :

Petit diamètre : La principale caractéristique est l'effet petit-monde qui a été mis en évidence par l'expérience de Milgram sur les six degrés de séparation [15]. Cette caractéristique exprime le fait que les graphes d'interactions possèdent des diamètres très faibles contrairement à leur taille. Le diamètre d'un graphe étant défini par la plus grande longueur des plus courts chemins entre chaque paire de noeuds, il est de l'ordre de $\log(n)$ dans un graphe de taille n.

Distribution hétérogène de degrés : dans ce type de graphe on observe quelques nœuds avec un degré très élevé et beaucoup de nœuds avec des degrés très faibles. La distribution des degrés suit une loi de puissance $p(k) \sim k^{-\gamma}$ avec k le degré et γ compris en pratique entre 2 et 3. Dans ce type de graphe la dispersion des degrés est très importante et le degré moyen n'est donc pas significatif.

Densité faible : La densité du graphe est définie par la proportion d'arêtes existantes dans le graphe par rapport au nombre d'arêtes potentielles. Dans un graphe non orienté, elle est donnée par la formule $d = \frac{2m}{n(n-1)}$ et dans un graphe orienté par $d = \frac{m}{n(n-1)}$ où n est le nombre de sommets et m le nombre d'arêtes (ou arcs). Il a été observé que les réseaux d'interactions ont une densité très faible, de l'ordre de $1/n$.

Fort coefficient de clustering : cette métrique exprime la probabilité que deux nœuds soient connectés sachant qu'ils ont un voisin commun. C'est l'un des paramètres étudiés dans les réseaux sociaux : les amis de mes amis sont-ils mes amis ? Il a été montré que les réseaux d'interactions possèdent un fort coefficient de clustering malgré une densité faible.

Centralité : Il est intéressant de mesurer l'importance de la position d'un nœud dans un réseau social. Pour cela, on dispose du calcul de la centralité du nœud. Il existe plusieurs approches pour évaluer la centralité d'un sommet, de nombreux articles leur sont consacrés [3][8], en synthèse on peut mesurer :

- la centralité de degré (degree centrality) qui mesure la centralité d'un sommet à son nombre de voisins directs
- la centralité d'intermédiarité (betweeness centrality) qui mesure la centralité d'un sommet au nombre de plus courts chemins passant par ce dernier
- la centralité de proximité (closeness centrality) qui mesure la centralité d'un sommet par sa proximité topologique aux autres sommets (inverse de la distance aux autres sommets)

Composante connexe géante : une composante connexe pour un graphe non orienté est un ensemble maximal de sommets tel qu'il existe un chemin entre toutes les paires de sommets de l'ensemble. Pour un graphe orienté, on dit qu'il est de faible connexité si, en ne tenant pas compte de l'orientation des arcs, le graphe est connexe. Certains graphes ne sont pas connexes, il est alors intéressant de mesurer le nombre de ses composantes connexes et leur taille. Dans les réseaux d'interactions on remarque souvent une composante connexe géante.

Structure communautaire : une particularité des réseaux d'interactions est d'être divisible en communautés. Une communauté est un sous-graphe du réseau d'interactions, fortement connecté, et peu connecté avec les autres communautés.

ii) Détection de communautés :

L'existence de zones plus densément connectées dans un graphe montre donc le regroupement de nœuds en communautés en fonction d'une ressemblance, d'un intérêt commun et il est intéressant de pouvoir les détecter. Plusieurs approches et algorithmes permettent la détection et l'extraction de communautés. Une étude de synthèse a été proposée par S. Papadopoulos [13]. On peut citer cinq grandes classes de détection de communauté :

- découverte de structures particulières qui consiste à trouver les structures topologiques particulières telles que les cliques
- clustering : regroupement des sommets basé sur une mesure de similarité
- optimisation d'une mesure de qualité détaillée ci-après
- approche divisive qui consiste à séparer les sommets selon certaines mesures

Dans notre étude nous nous sommes intéressés aux approches d'optimisation. En effet, le problème de détection de communauté peut être ramené à un problème d'optimisation d'une fonction permettant de mesurer la qualité d'une partition, la mesure de qualité la plus utilisée étant la modularité. La modularité d'une partition, introduite en 2004 par M.E.J. Newman [11], est définie par la proportion de liens internes aux communautés moins la valeur de cette même proportion dans un graphe où les arêtes seraient disposées au hasard entre les nœuds du graphe. Pour une partition $P = \{c_1, \ldots, c_k\}$ de k communautés, la qualité d'une communauté c_i est donnée par la formule :

$$\sum_{i,j \in c_i} (A_{ij} - \frac{d_i d_j}{2m})$$

où m désigne le nombre d'arêtes du graphe et A_{ij} les valeurs de la matrice d'adjacence entre les sommets i et j. La qualité de la partition P est la somme des qualités de chacune des communautés, puis on divise par $2m$ pour normaliser les valeurs possibles de Q dans l'intervalle $[-1, 1]$ ce qui donne :

$$Q(P) = \frac{1}{2m} \sum_{c_i \in P} \sum_{i,j \in c_i} (A_{ij} - \frac{d_i d_j}{2m})$$

Le principe d'un bon partitionnement d'un graphe est d'avoir un nombre élevé d'arêtes au sein des communautés mais un nombre réduit d'arêtes entre communautés. On estime qu'un graphe a une structure de communautés significatives quand une partition obtient un score de modularité supérieur à 0.3.

Or un reproche aux approches fondées sur l'optimisation de la modularité est la limite de résolution [6]. Des communautés de taille inférieure à une taille limite peuvent ne pas être distinguées. Pour tenter de corriger ce problème un paramètre de résolution λ est ajouté dans le calcul de la modularité [14] :

$$Q(P) = \frac{1}{2m} \sum_{c_i \in P} \sum_{i,j \in c_i} (A_{ij} - \lambda \frac{d_i d_j}{2m})$$

iii) Algorithme de Louvain :

Pour partitionner un graphe, il existe de nombreux algorithmes. Un des algorithmes reposant sur l'approche d'optimisation de la modularité est l'algorithme de Louvain [2]. Pour maximiser efficacement la modularité, l'algorithme de Louvain a deux phases qui se répètent de façon itérative :

- Etat initial : chaque nœud est affecté à sa propre communauté
- Phase d'affectation des nœuds : pour chaque nœud i, on mesure le gain de modularité si on le déplace dans la communauté de chacun de ses voisins. Le nœud est déplacé dans la communauté qui maximise la modularité. Si aucun gain de modularité n'est possible, i reste dans sa communauté d'origine.
- Phase de compression : on compresse le graphe en créant un nouveau graphe dans lequel chacune des communautés obtenues suite à la première phase devient un nœud. Deux nœuds i et j du nouveau graphe vont être reliés par un lien s'il existait un lien entre un nœud de la communauté représentée par i et un nœud de la communauté représentée par j dans le premier graphe. Le lien sera pondéré de la somme des poids des liens qui existaient entre les nœuds des communautés représentées par i et j.

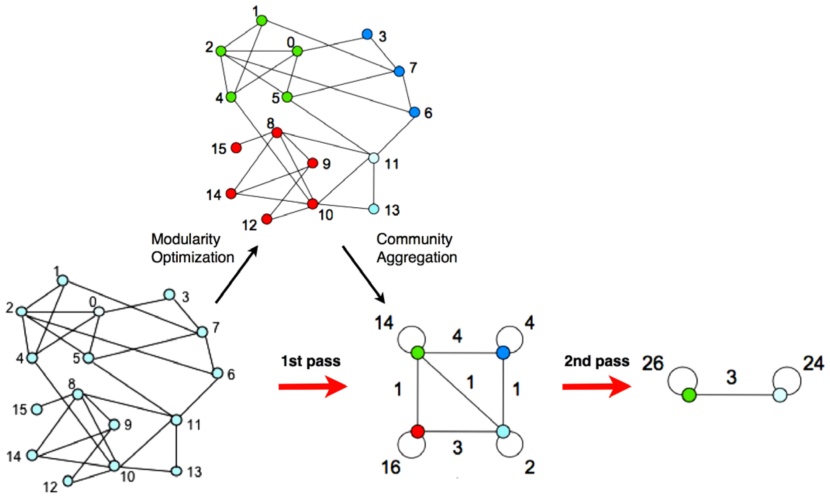


Figure 8 – Illustration de la méthode de louvain[2]

Les deux phases sont répétées jusqu'à ce que les nœuds ne puissent plus être réaffectés ou lorsque la modularité a atteint un maximum. La figure 8 illustre l'exécution de la méthode de Louvain.

La complexité de l'algorithme de Louvain est estimé à $O(n \log n)$, c'est la méthode la plus rapide pour la détection de communautés et est donc bien adaptée pour traiter des volumes de données importants.

3.2 Structure du réseau d'interactions

Les données contenues dans la base de données sur laquelle nous travaillons peuvent être représentées par un graphe multiplex. Plus précisément, sous la forme d'un réseau d'interactions entre les utilisateurs enregistrés, où les sommets du graphe représentent les utilisateurs et où les arcs représentent les tweets qui peuvent être de différentes natures (tweet simple, réponse, retweet, retweet avec commentaire, ou encore tweet mentionnant un autre utilisateur). Nous décrivons dans cette section les caractéristiques topologiques de ce réseau d'interactions.

nombre de sommets :	628 556
nombre d'arcs :	46 309 448

Tableau 4 – caractéristiques du graphe d'interactions entre les utilisateurs

La structure d'un réseau d'interactions peut être analysée à l'échelle globale et à l'échelle locale comme présenté ci-dessous.

3.2.1 Structure du réseau d'interactions à l'échelle globale

Comme nous l'avons décrit en 3.1, la structure topologique globale d'un graphe peut être captée au travers de différentes métriques. Le tableau 5 présente quelques mesures permettant de caractériser la structure globale du graphe d'interactions entre les utilisateurs. Nous indiquons que, pour des soucis d'implémentation, nous devons calculer certaines mesures sur une version non orientée du graphe d'interactions.

Comme le montre le tableau 5, le graphe d'interactions est très peu dense et fortement segmenté. En effet, nous dénombrons un nombre important de composantes connexes et il n'est donc pas possible de calculer le diamètre d'un tel graphe. De plus, nous observons une composante connexe "géante" typique des réseaux d'interactions comme décrit en 3.1. Ces caractéristiques ont leur importance et joueront un rôle prépondérant dans la tâche de détection des communautés.

densité du graphe :	0.0001
nombre de composantes connexes :	3404
plus petite composante connexe :	1 sommet
plus grande composante connexe :	619 434 sommets
taille moyenne des composantes connexes :	185 sommets
taille médiane des composantes connexes :	2 sommets

Tableau 5 – Caractéristiques topologiques du graphe d'interactions à l'échelle globale

3.2.2 Structure du réseau d'interactions à l'échelle locale

La structure d'un graphe peut aussi être analysée à l'échelle locale, comme vu en 3.1. Différentes métriques permettent de caractériser la structure locale d'un graphe d'interactions comme présenté ci-dessous.

i) Degrés des sommets

La première manière d'analyser la structure locale d'un graphe est d'examiner les degrés des sommets qui le composent. Le tableau 6 présente un résumé des degrés des sommets du graphe d'interactions.

degré minimal :	1
degré maximal :	2 080 446
degré médian :	10
degré moyen :	147
dispersion des degrés :	3 648

Tableau 6 – Aperçu des degrés des sommets

La figure 9 décrit la répartition des degrés inférieurs à la moyenne des degrés dans le graphe d'interactions.

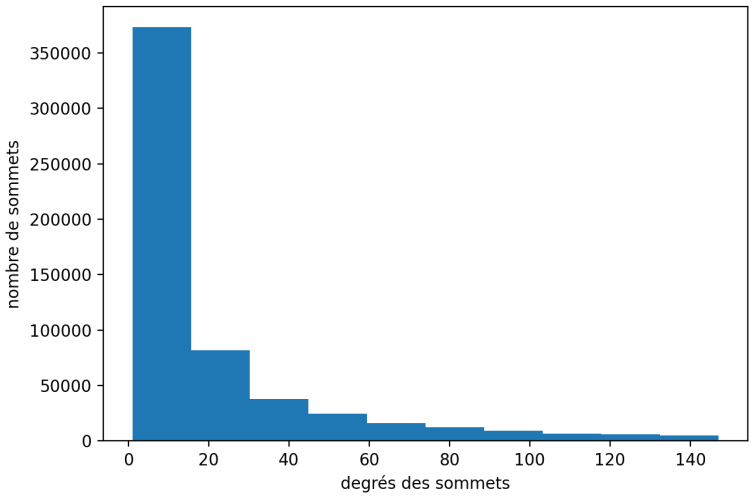


Figure 9 – Répartition des degrés inférieurs à la moyenne des degrés dans le graphe d'interactions

Nous ne présentons pas la répartition des degrés supérieurs à la moyenne des degrés car la forte dispersion des degrés rend le graphique illisible. Comme nous le montre le tableau 6 et la figure 9, et comme annoncé en 3.1, le réseau d'interactions présente une forte hétérogénéité des degrés des sommets qui le composent.

ii) Centralité des sommets

La centralité des sommets dans un réseau d'interactions est un indicateur intéressant sur sa topologie. Plusieurs métriques permettent de mesurer la centralité d'un sommet.

La centralité de degré évalue la centralité d'un sommet en fonction de son degré. Le tableau 7 résume les centralités de degré des sommets du graphe d'interactions.

centralité de degré minimale :	$1.59 \cdot 10^{-6}$
centralité de degré maximale :	3.31
centralité de degré médianne :	$1.59 \cdot 10^{-5}$
centralité de degré moyenne :	$2.34 \cdot 10^{-4}$
dispersion des centralités de degré :	$5.80 \cdot 10^{-3}$

Tableau 7 – Aperçu des centralités de degré des sommets du graphe

La figure 10 décrit la répartition des centralités de degré inférieures à la centralité de degré moyenne.

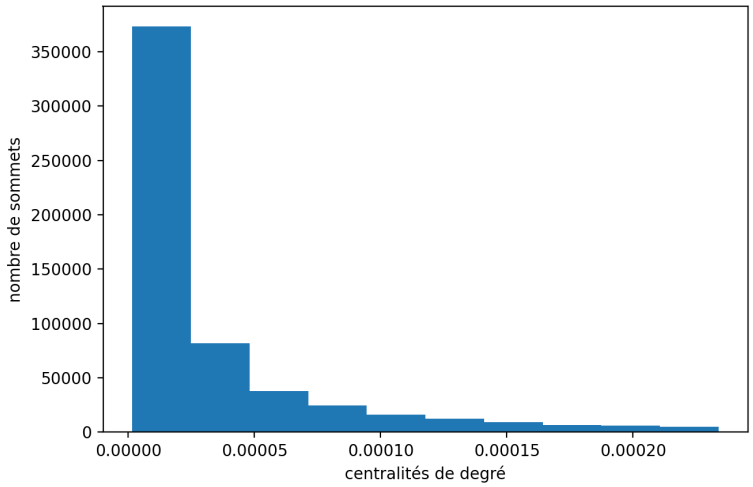


Figure 10 – Répartition des centralités de degré dans le graphe d'interactions

L'étude de la centralité des sommets du réseau d'interactions nous mène au mêmes conclusions que précédemment, une forte hétérogénéité des sommets au sens de la centralité.

L'étude de la topologie locale du graphe nous indique la forte hétérogénéité des sommets tant du point de vue des degrés que par l'étude de leur centralité. Ces analyses correspondent à la forte hétérogénéité des activités des différents utilisateurs décrite en 2.3.1 .

3.3 Analyse des communautés

3.3.1 Algorithme utilisé

Pour cette tâche de détection des communautés dans le réseau d'interactions, nous avons utilisé la méthode de Louvain qui a déjà fait ses preuves sur des graphes à large échelle. Plus précisément, nous avons utilisé l'implémentation réalisée dans la bibliothèque Networkx (networkx/algorithms/community/louvain_communities). L'algorithme de Louvain possède un paramètre nommé résolution qui permet d'ajuster le nombre de communautés retournées. Ce paramètre nous a intéressé afin d'obtenir un nombre réduit de communautés. Cependant, la très faible densité du réseau n'a pas permis de réduire suffisamment le nombre de communautés détectées. Ce paramètre n'a donc eu que peu d'influence sur la partition finale.

3.3.2 Analyse des communautés détectées

i) Partitions obtenues

Comme nous venons de le mentionner, l'algorithme de Louvain possède un paramètre nommé résolution qui permet de réduire le nombre de communautés retournées par l'algorithme. Ainsi, nous avons mené nos analyses sur deux partitions : la partition détectée nativement par l'algorithme (résolution par défaut) et la partition obtenue en diminuant la résolution (environ $1 \cdot 10^{-5}$, plus petite valeur influant sur le nombre de communautés). Nous décrivons ici quelques caractéristiques intéressantes de la partition obtenue.

Le tableau 8 décrit quelques caractéristiques des deux partitions obtenues.

statistiques	résolution par défaut	résolution $= 1 \cdot 10^{-5}$
nombre de communautés	3 688	3 404
modularité de la partition	0.52	$5.61 \cdot 10^{-4}$

Tableau 8 – Caractéristiques des partitions obtenues

Comme nous le montre le tableau 8, la modularité de la partition obtenue en diminuant la résolution est très significativement inférieure à la modularité de la partition obtenue sans modifier la résolution. En effet, l'algorithme de Louvain cherche par défaut à maximiser la modularité ce qui explique cette différence. Comme nous pouvions nous y attendre, la meilleure partition semble bien être celle obtenue sans ajuster la modularité. De plus, la partition obtenue en diminuant la résolution correspond à un partitionnement en composantes connexes, c'est à dire un partitionnement où chaque communauté est une composante connexe. Il est donc clair qu'il n'est pas possible de descendre en dessous de 3404 communautés. Pour ces raisons, nous ne considérons par la suite que la partition optimale. La partition obtenue peut être analysée selon différents critères comme nous le décrivons ci-dessous.

ii) Tailles des communautés

La première manière d'analyser les communautés détectées est d'étudier leur taille, c'est à dire le nombre de sommets (utilisateurs) qu'elles regroupent. Le tableau 9 résume les informations concernant les tailles des communautés détectées.

taille minimale d'une communauté :	1 sommet
taille maximale d'une communauté :	212 195 sommets
taille moyenne d'une communauté :	170 sommets
taille médianne d'une communauté :	2 sommets

Tableau 9 – Tailles des communautés détectées

Le tableau 9 nous indique que la taille d'une communauté varie de manière très importante. Nous observons aussi un grand nombre de communautés de très petite tailles. Ces caractéristiques s'expliquent par la faible densité du réseau d'interactions.

iii) Densités des communautés

La densité d'une communauté est un indicateur intéressant de sa connectivité. Le tableau 10 décrit les densités des communautés obtenues.

densité minimale d'une communauté :	0.0
densité maximale d'une communauté :	1 897,5
densité moyenne d'une communauté :	1.4
densité médianne d'une communauté :	0.5

Tableau 10 – Densités des communautés détectées

De manière similaire à la taille des communautés, nous observons une forte hétérogénéité de la densité des communautés.

iv) Graphe des communautés

A partir de la partition obtenue, il est possible de construire un graphe "induit" (ou graphe des communautés), où chaque sommet du graphe "induit" représente une communauté. Cela permet notamment de visualiser de très grands graphes comme le réseau d'interactions sur lequel nous travaillons. Le tableau 11 décrit les principales caractéristiques du graphe des communautés.

nombre de sommets :	3 688
nombre d'arcs :	838
densité :	0.0001

Tableau 11 – Caracéristiques du graphe des communautés

La figure 11 fournit un aperçu du graphe des communautés. La taille des sommets est proportionnelle à la taille des communautés qu'ils représentent.

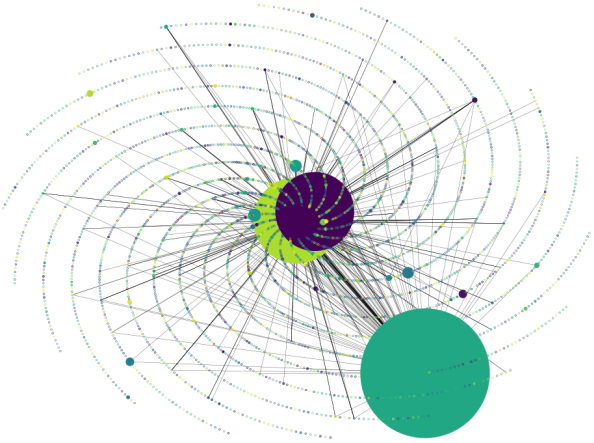


Figure 11 – Graphe des communautés

À noter qu'il s'agit d'un graphe simple et non d'un multigraphe. Une arête apparaît entre deux communautés c_i et c_j si au moins un utilisateur de c_i est connecté avec un utilisateur de c_j.

v) Description interne des communautés

Nous nous proposons d'analyser la topologie interne de quelques communautés. Pour cela nous avons tiré au hasard trois communautés dont voici les caractéristiques.

statistiques	communauté n°23	communauté n°115	communauté n°732
nombre de sommets :	28	167	58
nombre d'arcs :	858	391	107
densité :	1.13	0.01	0.04

Tableau 12 – Principales caractéristiques des communautés tirées au hasard

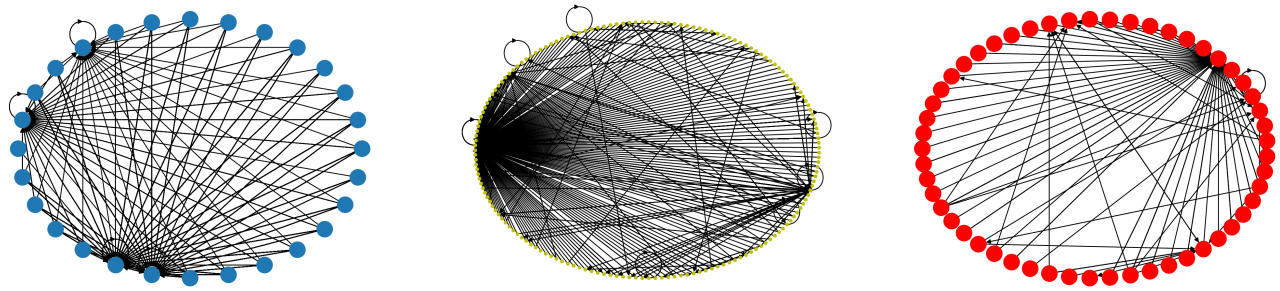


Figure 12 – Visualisation des communautés

Le tableau 12 et la figure 12 nous donnent un aperçu des communautés tirées au hasard. Nous observons des densités significativement supérieures à celle du réseau complet. Nous pouvons aussi observer sur la figure 12 la présence de sommets particulièrement centraux.

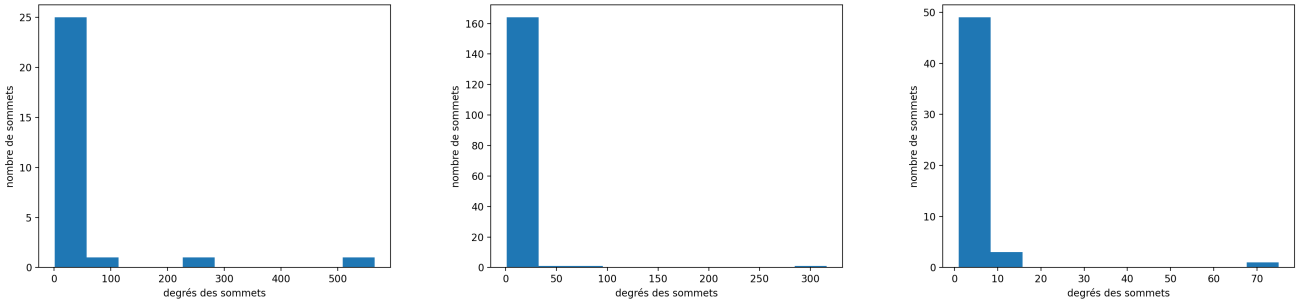


Figure 13 – Répartition des degrés dans les communautés

La figure 13 confirme d'une part l'importante centralité de certains sommets au sein des communautés, et d'autre part la forte hétérogénéité des degrés au sein des communautés.

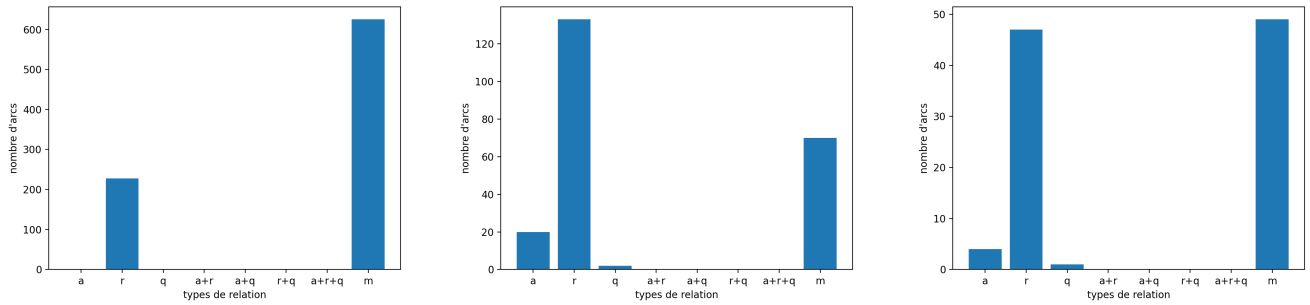


Figure 14 – Répartition des types de relations dans les communautés

Nous observons dans la figure 14 qu'un grand nombre de relations de mention apparaissent au sein d'une communauté ce qui n'est pas très étonnant. Cependant, nous observons aussi que les relations de type "retweets" émergent majoritairement au sein d'une communauté. Ce constat est particulièrement intéressant et nous donne des indices sur le mécanisme de communication au sein d'une communauté. Cette information nous sera très utile pour comprendre les mécanismes de propagation des sentiments dans le réseau d'interactions.

4 Analyse du contenu textuel des tweets

4.1 Identification de la cible des tweets

Nous nommons « cible » d'un tweet le candidat visé par le tweet. Un tweet pouvant avoir plusieurs cibles, nous restreignons notre étude aux tweets ayant une seule cible.
L'un des objectifs de notre étude est de déterminer la cible d'un tweet et d'en prédire la polarité. Pour récupérer les tweets ayant pour cibles des candidats, nous sommes partis de l'hypothèse selon laquelle les noms des candidats visés sont nommés dans les hashtags.

La base de données contient 3 204 023 tweets avec un ou plusieurs hashtags. Parmi l'ensemble de ces tweets, 2 037 322 visent une cible unique c'est-à-dire possèdent des hashtags désignant un seul candidat.
Par la suite, nous chercherons à construire des classifieurs visant à prédire le mieux possible la polarité de l'ensemble de ces tweets (que l'on appellera tweets à cible unique).

4.2 Pré-traitement des données textuelles

Dans le cadre de notre étude, nous utilisons la méthodologie **BagOfWord** (sac de mots en français).
Cette méthode qui était majoritairement utilisée avant l'apparition des modèles basés sur les réseaux de neurones, tient compte du fait que les données brutes ne peuvent pas être transmises directement aux algorithmes de machine Learning car la plupart d'entre eux s'attendent à des vecteurs de caractéristiques numériques de taille fixe plutôt qu'à des documents textuels bruts de longueur variable.
Elle consiste à pré-traiter les données textuelles, puis à les vectoriser, créant ainsi une matrice dont chaque ligne représente un tweet et les colonnes des mots (ou plutôt des tokens) du vocabulaire.

Lors de la phase de pré-traitement, nous avons choisi de :

- supprimer les stop-words : ces termes extrêmement fréquents n'apportent pas de réelles informations et peuvent faire de l'ombre aux termes plus rares et pourtant plus intéressants ;
- supprimer la ponctuation, les accents et les chiffres ;
- utiliser le stemming (ou racinisation) dont le principe consiste à réduire un mot dans sa forme « racine », permettant de regrouper de nombreuses variantes d'un mot, et ainsi de réduire le vocabulaire.
- convertir les textes en minuscules.

Lors de la phase de vectorisation, nous avons fait varier les hyperparamètres suivants :

- max_feature : taille du vocabulaire (nombre de colonnes de la matrice) ;
- min_df : nombre minimum de fois où un mot doit apparaitre dans le corpus pour pourvoir faire partie du vocabulaire ;
- max_df : fréquence maximale d'un mot dans le corpus pour pouvoir faire partie du vocabulaire ;
- n_gram range : choix du nombre de mots consécutifs à considérer pour construire le vocabulaire.

Nous avons de plus choisi la métrique TF-IDF : pour chaque vecteur ligne (chaque document), les coefficients sont d'autant plus grands que le mot est fréquent dans le document et que son nombre d'occurrences dans l'ensemble du corpus est faible.
Plus formellement, le TF-IDF d'un terme « i » dans le document « j » peut être déterminée en multipliant le Term Frequency « i » dans le document « j » par l'Inverse Document Frequency « i » dans l'ensemble des documents.

4.3 Première expérience : Transfer Learning

N'ayant aucune donnée étiquetée dans notre corpus, et le coût d'un étiquetage manuel étant très lourd, une approche d'apprentissage par transfert (transfer Learning en anglais) nous a semblé intéressante dans un premier temps. Cette technique consiste à utiliser des données existantes dans un domaine qui n'est pas identique au domaine d'intérêt pour construire un modèle d'apprentissage. Cette approche s'est considérablement développée dans le traitement automatique du langage ; il existe en effet de nombreuses applications d'apprentissage automatique auxquelles l'apprentissage par transfert a été appliqué avec succès, notamment dans la classification des sentiments [16].

Nous avons utilisé une base de données comprenant un ensemble de critiques de films d'utilisateurs de « allociné », labelisée en sentiments positifs et négatifs. Cette base, bien équilibrée, comprend 1 600 000 données pour l'apprentissage, et 20 000 pour les tests.
Nous avons vectorisé les données en suivant la méthodologie **BagOfWord** décrite en 4.2, avec le paramétrage suivant : max_features=15000, min_df=2, max_df=0.7, ngram_range=(1,2)

Puis, nous avons entrainé deux classifieurs en utilisant les algorithmes SVM et Naive Bayes et avons obtenu des scores sur les données test d'environ 90% pour les deux classifieurs.

Enfin, nous avons testé manuellement les deux classifieurs sur les tweets à cible unique.

Nous présentons ci-dessous les résultats des classifieurs SVM et NB, ainsi que les résultats du classifieur prédisant une polarité au hasard, et les résultats du classifieur prédisant la polarité majoritaire.

nombre de tests	score_SVM	score_NB	score_hasard	score_majoritaire
100	0.54	0.51	0.39	0.6

Ces mauvais scores peuvent être expliqués par le fait que le vocabulaire de la base de données « allociné » est très spécifique et par conséquent non transférable au domaine visé. En effet, en observant les mots ayant un poids significatif, on constate aisément que le vocabulaire utilisé pour prédire les sentiments est bien éloigné de celui utilisé dans notre corpus de tweets. . .

Figure 15 – Les 100 mots ayant le plus grand poids : SVM et Naive Bayes

Il existe des méthodes plus efficaces dans le cadre du transfer Learning comme par exemple l'utilisation de plusieurs sources de données provenant de domaines différents pour construire le classifieur [17]. Yelena Mejova et Padmini Srinivasan [10] montrent en effet que la combinaison de données recouvrant trois sources de médias sociaux (blogs, critiques et twitter), lors de la classification de documents dans une source (la source cible), permet de construire des modèles qui peuvent être aussi bons, voire meilleurs, que ceux formés à partir des données cibles. Ces méthodes, plus sophistiquées, pourraient d'ailleurs faire l'objet d'une autre étude.

Par la suite, nous avons cherché à étiqueter un ensemble de données de notre corpus de manière à construire un classifieur supervisé classique.

4.4 Deuxième expérience : Etiquetage des données et construction de classifieurs

4.4.1 Etiquetage des données

Comme nous l'avons déjà mentionné, certains hashtags attribuent à des candidats un sentiment négatif. Voici la liste de ces hashtags que nous avons relevés :

Candidat	Hashtags négatifs
Macron	'PoissonMacron','ToutSaufMacron','ImpostureMacron','StopMacron','BarbaraLefebvre', 'enmarchearriere', 'MacronPiègeÀCons','macrongate','EmmanuelHollande','LeVraiMacron'
Fillon	'FillonGate','PenelopeGate','PenelopeFillon','penelope','LeVraiFillon'
Lepen	'PasLePen'

De même, nous avons repéré certains hashtags à connotation positive :

Candidat	Hashtags positifs
Macron	'JeVoteMacron','voteMacron','MacronPresident
Fillon	'TousFillon','JeVoteFillon'
Lepen	'JeChoisisMarine','AvecMarine'
Mélenchon	'legoutdubonheur','MélenchonAu2emeTourCestPossible','JoursHeureux'

Une vérification manuelle nous a conduit à classer un tweet comme positif s'il contient un hashtag à connotation positive et s'il ne contient aucun hashtag à connotation négative. En collectant l'ensemble des tweets contenant ces hashtags, nous avons construit une table contenant l'identifiant du tweet, le candidat visé et la polarité. Nous avons ainsi obtenu environ 160 000 données étiquetées « -1 » et 130 000 données étiquetées « +1 ».
Nous avons pu ainsi utiliser ces données étiquetées pour construire des classifieurs.

4.4.2 Choix de l'algorithme de l'apprentissage et paramétrisation commune des classifieurs

Les algorithmes de classification Naive Bayes (NB), maximum entropy classification (ME), and support vector machines (SVM) s'avèrent particulièrement performants dans le domaine de la classification de sentiments. Dans [12], les auteurs montrent que les résultats expérimentaux sur un ensemble de données de critiques de films produits par NB, ME et SVM sont substantiellement meilleurs que les résultats obtenus par les bases de référence générées par l'homme, SVM étant le plus performant sur ce jeu de données.
Par la suite, nous choisirons SVM comme base pour construire nos classifieurs.

Présentation de la démarche :
Nous avons séparé les données étiquetées présentées précédemment en données d'apprentissage et de test. Puis, nous avons entrainé des classifieurs en suivant la démarche classique qui consiste à pré-traiter les données d'apprentissage (de trois manières différentes), les vectoriser puis à entrainer un classifieur SVM. Nous avons ensuite évalué les classifieurs sur les données test.
A la suite de ces tests, nous avons fixé un réglage des hyperparamètres puis nous avons entrainé trois classifieurs (en utilisant cette fois-ci l'intégralité des données étiquetées) dont la construction ne diffère uniquement que dans le choix des pré-traitements.

Nous récapitulons ci-dessous l'ensemble des pré-traitements communs aux trois classifieurs :
- Transformation des liens url en un mot-clé : « lien_url ».
- Remplacement des différents noms des candidats par un unique nom. (description détaillée dans l'annexe page 29)
- Suppression des chiffres, des accents et de la ponctuation (sauf # , @, ! et _).
- Suppression des stop-words.
- Mise du texte en minuscules.
- Stemming.

Vectorisation des données en utilisant la métrique TF-IDF (commun aux trois classifieurs) :
max_features=15000, min_df=2, max_df=0.7, ngram_range=(1,2)

Algorithme de machine learning :
SVM.linear()

4.4.3 Présentation des trois classifieurs

i) Classifieur 1

Pré-traitement : pré-traitement commun
Score en test sur les données étiquetées : 97%
Les prédictions sur les tweets à cible unique sont un peu déséquilibrées :

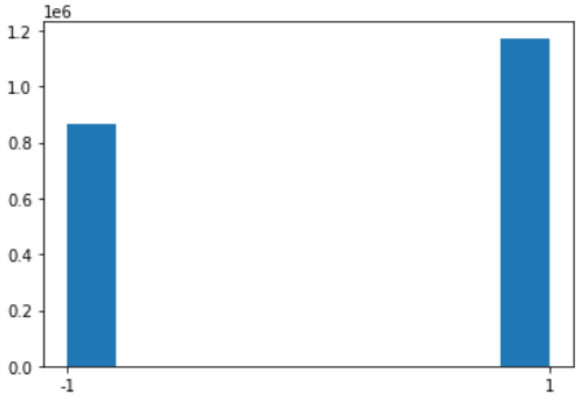


Figure 16 – Répartition des prédictions du classifieur 1

Notons de plus que les mots ayant le plus grand poids sont précisément les hashtags que nous avions utilisés pour étiqueter les données.

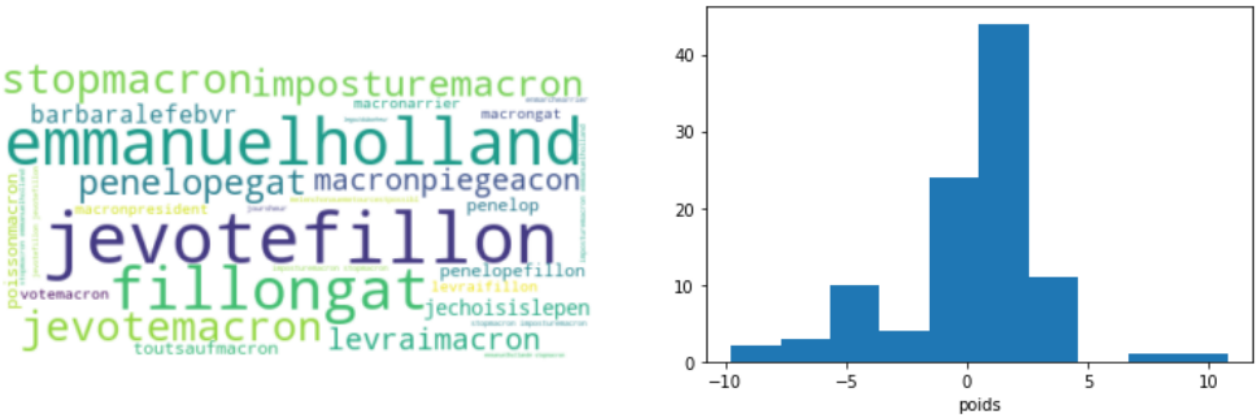


Figure 17 – Les 100 mots ayant les plus grands poids et répartition de ces poids

ii) Classifieur 2

Pré-traitement : pré-traitement commun + suppression de tous les hashtags
Score en test sur les données étiquetées : 90,7%
Les prédictions sur les tweets à cible unique sont assez équilibrées :

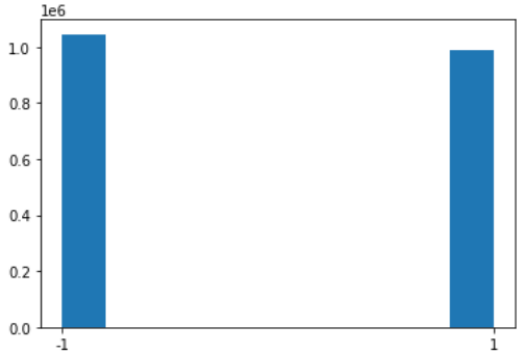


Figure 18 – Répartition des prédictions du classifieur 2

Nous remarquons que certains bi-grammes du type « RT @username » ont un grand poids dans ce classifieur, et pourraient par ailleurs être au centre de leurs communautés.

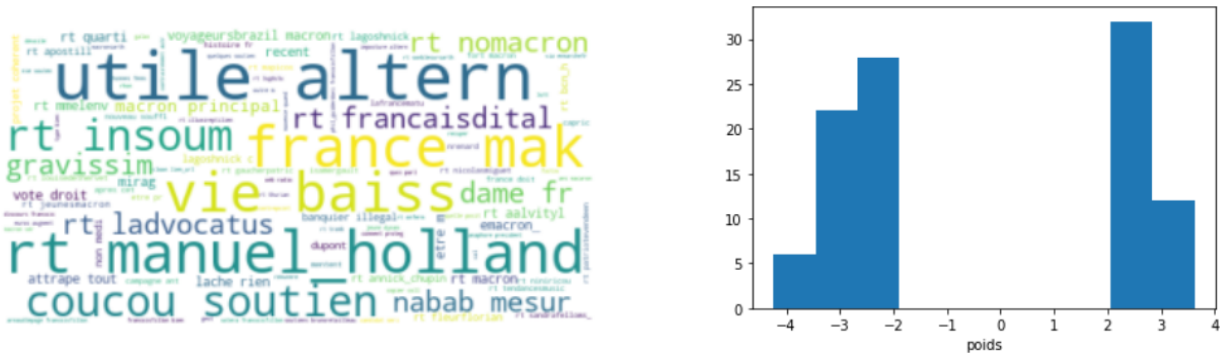


Figure 19 – Les 100 mots ayant les plus grands poids et répartition de ces poids

iii) Classifieur 3

Pré-traitement : Pré-traitement commun + suppression de tous les hashtags + suppression de « RT @username »
Score en test sur les données étiquetées : 88,8%
Les prédictions sur les tweets à cible unique sont assez équilibrées :

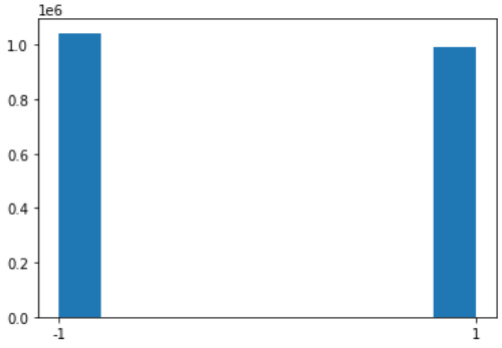


Figure 20 – Répartition des prédictions du classifieur 3

Le nuage des mots ayant les plus grands poids est encore bien différent des deux précédents classifieurs.

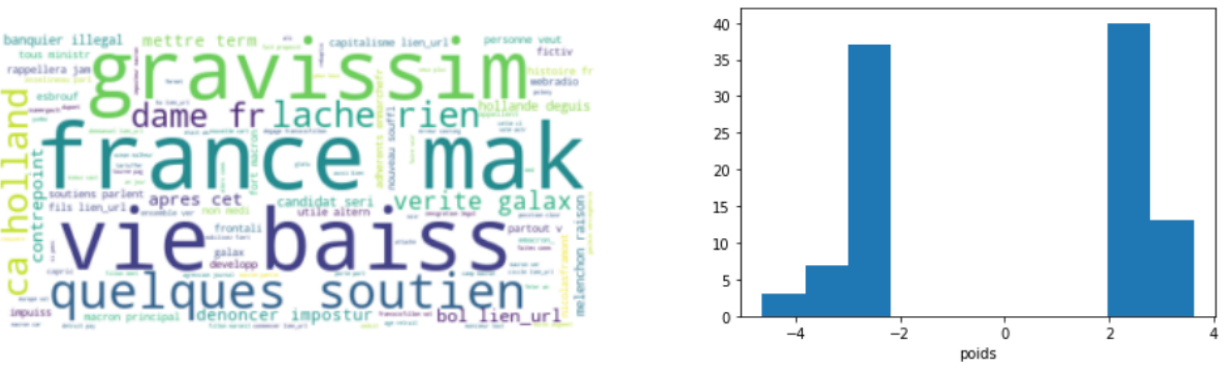


Figure 21 – Les 100 mots ayant les plus grands poids et répartition de ces poids

4.4.4 Evaluation des classifieurs

Nous avons établi les prédictions des 2 037 322 tweets à cible unique pour chaque classifieur. Puis, nous avons testé manuellement (avec deux évaluateurs différents) ces classifieurs sur deux séries de 100 tweets tirés aléatoirement.

nombre de tests	score_classifieur1	score_classifieur2	score_classifieur3	score_hasard	score_majoritaire
100	0.69	0.72	0.76	0.58	0.61
100	0.51	0.73	0.68	0.40	0.55

Malgré son très bon score en test, le classifieur 1 ne semble pas performant. Les mots ayant les plus grands poids sont les hashtags qui avaient servi à étiqueter les données, ce qui vraisemblablement crée un biais. Lorsque l'on supprime l'ensemble des hashtags lors du pré-traitement des données, nous obtenons des résultats bien meilleurs qui, bien que variables, nous semblent encourageants.

4.4.5 Comparaison des classifieurs avec BERT

BERT (Bidirectional Encoder Representation for Transformer) est un modèle développé par Google Research en 2018, qui utilise l'architecture d'encodeur Transformer pour traiter chaque token du texte d'entrée dans le contexte des tokens situés avant et après [5].
Les modèles BERT, qui connaîssent un grand succès sur une variété de tâches en TAL sont généralement pré-entraînés sur un grand corpus de texte, puis affinés pour des tâches spécifiques.

Nous avons retenu les deux modèles suivants :

- « bert-base-multilingual-uncased-sentiment » qui est un modèle affiné pour l'analyse de sentiments sur des critiques de produits en six langues (anglais, néerlandais, allemand, français, espagnol et italien). Il prédit le sentiment de la critique sous la forme d'un nombre d'étoiles (entre 1 et 5). Pour pouvoir le comparer à nos classifieurs, nous avons associé le label « -1 » aux notes 1 et 2, le label « +1 » aux notes et 4 et 5, et le label « 0 » à la note 3. Par la suite, ce modèle sera associé au classifieur que nous nommerons « Bert1 ».
- « twitter-XLM-roBERTa-base for Sentiment Analysis » qui est un modèle entraîné sur environ 198 000 tweets et affiné pour l'analyse des sentiments [1]. Il prédit le sentiment du tweet sous la forme « -1 », « 1 », et « 0 ». Ce modèle sera associé au classifieur que nous nommerons « Bert2 ».

Les classifieurs 1,2 et 3 prédisant uniquement des valeurs égales à « -1 » et « 1 », nous avons dans un premier temps considéré les tweets dont les prédictions par Bert1 et Bert2 sont différentes de 0.

Voici les résultats obtenus sur une série de 100 tweets :

nb_tests	sc_classif1	sc_classif2	sc_classif3	sc_Bert1	sc_Bert2	sc_hasard	sc_maj
100	0.46	0.76	0.69	0.67	0.72	0.45	0.66

Les classifieurs 2 et Bert2 obtiennent les meilleures performances. Néanmoins, un nombre important de tweets ne sont pas considérés (les tweets classés neutre par Bert), ce qui semble augmenter la proportion de tweets à connotation négative et pourrait biaiser les résultats.
Pour atteindre l'objectif de classifier tous les tweets de la base tweets à cible unique, et de comparer les trois classifieurs avec les modèles de Bert, nous avons construit 4 classifieurs Bert 1_rand, Bert 2_rand, Bert 1_1, Bert 2_1 qui « prolongent » les classifieurs définis précédemment.
Bert 1_rand, Bert 2_rand : si le tweet est étiqueté par « 0 », prédit une valeur au hasard parmi « -1 » et « 1 ».
Bert 1_1, Bert 2_1 : si le tweet est étiqueté par « 0 », prédit la valeur « 1 » .

Nous avons obtenu les scores suivants :

nb_tests	classif1	classif2	classif3	Bert1_rand	Bert1_1	Bert2_rand	Bert2_2	hasard	maj
100	0.58	0.77	0.68	0.55	0.57	0.66	0.69	0.43	0.61

A travers l'ensemble de ces tests, nous constatons que le classifieur 2 a la meilleure performance. Dans ce classifieur, certains termes du type « RT @username » ont un poids important, et pourraient être des connecteurs reliant des utilisateurs d'une même communauté.

5 Combinaison des deux approches

Notre but final est d'analyser de manière précise et performante les sentiments exprimés dans les tweets. Comme décrit en 4, nous avons construit des classifieurs fournissant des performances intéressantes. Dans l'objectif d'améliorer et d'affiner la détection des sentiments, nous avons combiné la représentation des données sous la forme d'un réseau d'interactions avec l'étiquetage réalisé. L'étiquetage n'étant réalisé que sur une petite proportion des tweets, cette approche pourrait nous permettre d'étendre l'étiquetage et de découvrir des règles de propagation des sentiments en nous basant sur la topologie du réseau.

5.1 Fusion des approches

Afin de combiner nos deux approches d'analyse des données, nous avons étiqueté les arcs du réseau (représentant les tweets) avec l'étiquetage réalisé en 4. Cela nous permet d'analyser la répartition des classes (polarité et cible) dans le réseau d'interactions et d'en déduire des mécanismes de propagation dans le l'objectif de généraliser l'étiquetage.

5.2 Analyse du graphe étiqueté

5.2.1 Analyse globale

Les étiquettes générées son réparties dans 23 communautés, la figure 22 décrit la répartition des étiquettes dans les communautés.

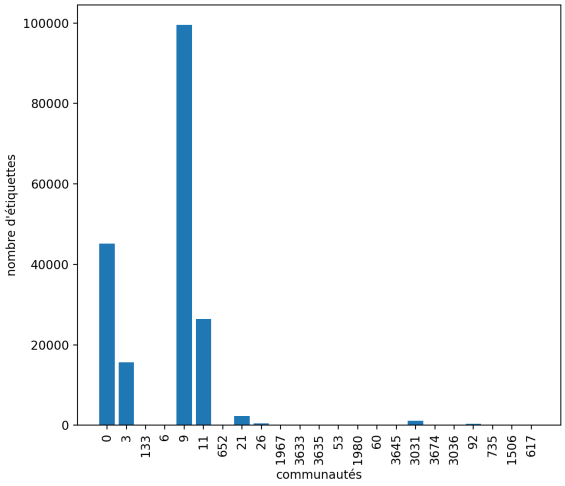


Figure 22 – Répartition des étiquettes dans les 23 communautés

Nous observons dans la figure 22 que les étiquettes sont réparties de manière hétérogène dans les 23 communautés. En effet, seules 8 communautés regroupent la majorité des étiquettes.

5.2.2 Analyse d'une communauté

Nous nous proposons d'analyser la communauté (communauté n°9) qui regroupe la majorité des étiquettes générées lors de l'analyse du contenu textuel des tweets. Les caractéristiques principales de cette communauté sont décrites ci-après.

nombre de sommets :	30 032
nombre d'arcs :	5 668 356
densité :	0.006

Tableau 13 – Principales caractéristiques de la communauté n°9

La figure 23 décrit la répartition des étiquettes (cible et polarité) au sein de la communauté n°9.

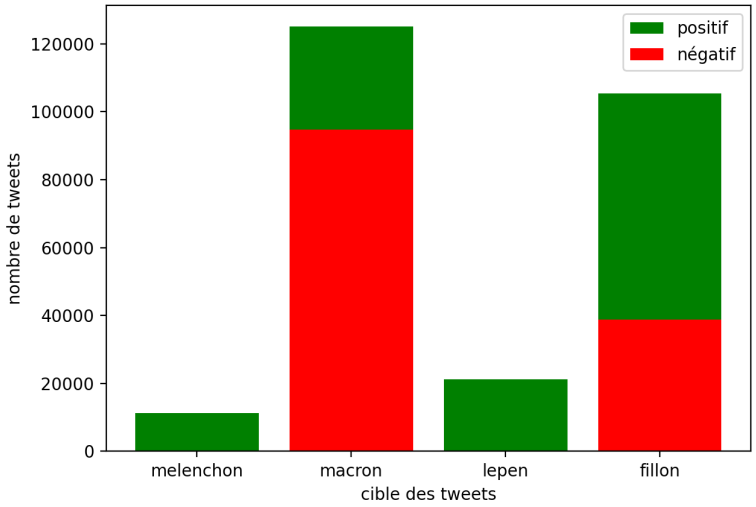


Figure 23 – Répartition des étiquettes (cible et plarité) au sein de la communauté n°9

Comme nous le montre la figure 23, nous observons peu de tweets ciblant M. Mélenchon et Mme. Lepen, mais cependant plutôt favorables. Nous observons bien plus de tweets ciblant M. Macron et M. Fillon, mais plus partagés quant à leur polarité. Cependant il faut bien garder à l'esprit que seule une petite partie des tweets est annotée et que, donc, ces observations ne représentent peut-être pas le comportement de la communauté.

5.3 Propagation de l'étiquetage dans le réseau d'interactions

L'intérêt de combiner l'étude du réseau d'interactions avec l'analyse des contenus textuels des tweets est de s'appuyer sur les interactions utilisateurs pour enrichir l'étiquetage généré par la classification textuelle des tweets. L'étiquetage réalisé est incomplet et ne couvre qu'une petite partie des interactions entre utilisateurs. Nous espérons pouvoir propager l'étiquetage le long des différentes relations qui lient les utilisateurs.

5.3.1 Propagation locale des étiquettes

Nous nous proposons ici de retravailler sur la communauté n°9 car elle regroupe la majorité des étiquettes. La figure 24 décrit la répartition des types de relation entre les utilisateurs de la communauté n°9.

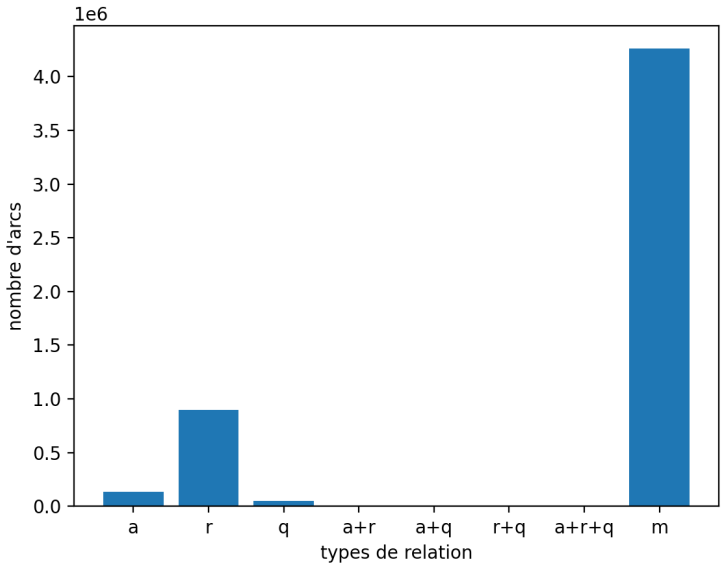


Figure 24 – Répartition des types de relation au sein de la communauté n°9

La figure 24 confirme les observations précédemment réalisées en sections 2 et 3. Les relations de type "retweets" sont les plus fréquentes dans le graphe d'interactions.

Analysons les mécanismes de propagations des étiquettes pour les différents types de relation :

- **réponse à un tweet :** Un tweet écrit en réponse à un autre tweet peut, soit confirmer l'opinion exprimée dans le tweet original, soit la contredire. Nous ne pouvons prédire avec certitude l'étiquette d'un tweet réponse à partir du tweet auquel il répond.
- **retweet :** un retweet est une copie à l'indentique du tweet d'origine. Nous pouvons donc raisonnablement affirmer qu'un retweet hérite de l'étiquette du tweet d'origine.
- **retweet avec commentaire :** un retweet avec commentaire est plus délicat à traiter dans la mesure où le commentaire peut confirmer ou contredire l'opinion exprimée dans le retweet.
- **mention :** Il n'y a pas de raison de propager une étiquette le long d'une relation "mention".

Ainsi, le seul type de relation exploitable pour propager les étiquettes de tweet en tweet est la relation "retweet". Bien heureusement, la majorité des interactions sont de type "retweet", nous devrions donc étiqueter une importante partie des tweets en propageant leurs étiquettes.

Le tableau 14 présente un extrait des tweets nouvellement étiquetés grâce au procédé décrit précédemment. Sur cet extrait, les tweets semblent correctement étiquetés. C'est ce que nous attendions et nous pouvons être confiants dans cette méthode.

contenu du tweet	cible du tweet	polarité (-1 ou 1)
RT @RaphaelChombart : .@FrancoisFillon, le candidat des classes moyennes ! FillonLyon Fillon2017 JeVoteFillon ? ? ? ? https ://t.co/HJjACOvyj4	fillon	1
RT @Charlescohenboy : "Il me reste 17 jours pour convaincre les Français !" @FrancoisFillon sur @franceinter Fillon2017 JeVoteFillon Inte...	fillon	1
RT @CharlesdASTORG : "C'est du vide, du vent" participants à un meeting de Macron. EmmanuelHollande @Manuel_Hollande https ://t.co/wjAXMiU...	macron	-1
RT @Fillon_78 : .@Samuel_Lafont : Qd Macron achète des fans sur Facebook pour feindre un soutien des Français... EmmanuelHollande https ://t...	macron	-1
RT @Charlescohenboy : "C'est de liberté dont notre pays a besoin !" @FrancoisFillon LeGrandDebat JeVoteFillon Fillon2017 https ://t.co/1R1...	fillon	1

Tableau 14 – Extrait des tweets étiquetés par propagation

5.3.2 Propagation globale des étiquettes

Nous venons de montrer que l'étiquetage pouvait être propagé de tweet en tweet le long des relations de retweets au sein d'une communauté. De manière analogue, le même procédé peut être appliqué sur la totalité du réseau. Les résultats après propagation des étiquettes dans l'ensemble du réseau sont présentés ci-dessous.

nombre initial d'étiquettes :	262 974
nombre d'étiquettes après propagation :	264 806

Tableau 15 – Nombre d'étiquettes avant et après propagation

Lorsque nous propageons les étiquettes à un instant t, il est possible de propager les nouvelles étiquettes créées au temps $t+1$. Nous proposons un algorithme itératif permettant de maximiser le nombre d'étiquettes propagées à partir de l'étiquetage initial :

1. Initialisation avec l'étiquetage généré en section 4
2. Propagation des étiquettes de tweet en retweet
3. Enrichissement l'étiquetage avec les nouvelles étiquettes
4. Tant qu'il reste des étiquettes à propager : retour à l'étape 2

Le tableau 16 décrit le résultat de la propagation itérative des étiquettes.

nombre initial d'étiquettes :	262 974
nombre d'étiquettes après propagation :	278 152

Tableau 16 – Nombre d'étiquettes avant et après propagation

Ce procédé permet d'augmenter le nombre d'étiquettes de manière significative. Cependant, nous sommes loin d'avoir couvert l'ensemble des retweets. Ce phénomène est dû à la faible connexité du réseau qui ne permet pas de propager les étiquettes dans la totalité du réseau. De plus, nous avons observé que les étiquettes produites dans la section 4 sont regroupées dans un petit nombre de communautés. Un étiquetage plus dispersé (avec au moins une étiquette par composante connexe) aurait certainement permis de propager plus efficacement les étiquettes dans le réseau d'interactions.

6 Conclusion

Pour conclure sur cette étude, nous avons combiné une étude de la topologie du réseau d'interactions entre les utilisateurs à une analyse des contenus textuels des tweets.

Pour revenir sur l'étude de la topologie du réseau d'interactions, nous avons cherché à détecter les communautés au sein du réseau. Nous avons obtenu une partition de qualité critiquable en raison des très nombreuses micro-communautés détectées. Ce phénomène étant intrinsèquement lié aux très faibles densités et connexités du réseau, rendant un regroupement plus "macroscopique" impossible.

Concernant l'analyse des contenus textuels des tweets, nous avons identifié les cibles en nous appuyant sur les ensembles de hashtags mentionnant une unique cible. Nous avons cherché à mettre en place une méthodologie de Transfer Learning afin de passer d'un problème non-supervisé à un problème supervisé. Pour cela nous avons utilisé un corpus d'avis concernant des films (Allociné). Cependant, la trop grande dissimilarité entre le vocabulaire employé dans le corpus Allociné et celui utilisé dans les tweets n'a pas permis d'obtenir de bonnes performances. Nous avons ensuite construit plusieurs classifieurs supervisés par un étiquetage résultant de l'agrégation des hashtags sur les cibles et les polarités. Les résultats avec ces classifieurs sont plutôt concluants dans la mesure où il concurrencent les performances de BERT sur cette tâche.

Nous avons ensuite combiné ces deux approches en étiquetant les arcs (tweets) du réseau d'interactions avec les étiquettes produites. L'étude du réseau nous a permis de définir un mécanisme de propagation des étiquettes de tweet en retweet permettant ainsi d'amplifier l'étiquetage tout en garantissant la justesse des nouvelles étiquettes. Cependant l'amplification de l'étiquetage n'a pas été aussi important que nous le pensions car les étiquettes étaient réparties dans un nombre réduit de composantes connexes, limitant grandement la propagation dans le réseau.

7 Perspectives

Cette étude a permis de faire émerger un certain nombre de perspectives et d'améliorations.

Concernant l'analyse topologique du réseau, la seule réserve que nous émettons réside dans la modélisation du réseau. Nous avons choisi de le représenter par un multigraphe orienté où les sommets sont les utilisateurs et les arcs, les tweets qui peuvent être de différente nature. Une autre représentation aurait peut-être été plus adaptée. Nous pensons notamment à un réseau multicouches avec, une couche où les sommets représentent les utilisateurs et une autre couche où les sommets représentent les tweets, et les arcs représentent les différentes relations. Une autre modélisation permettrait d'augmenter le coefficient de clustering et d'améliorer la détection des communautés.

Concernant l'analyse des contenus textuels des tweets, nous pensons qu'une approche plus complexe de Transfer Learning, en variant les sources par exemple, permettrait d'obtenir des résultats intéressants. Une approche de bootstrap serait aussi interéressante pour palier le problème d'apprentissage supervisé.

En ce qui concerne la combinaison des deux approches, les résultats sont encourageants dans la mesure où nous avons montré qu'une propagation efficace des étiquettes est réalisable. La faible amplification de l'étiquetage est causée par la faible connexité du réseau et par le regroupement des étiquettes initiales dans un nombre réduit de composantes connexes. Cependant, nous pensons qu'un étiquetage équilibré (au moins une étiquette par composante connexe) permettrait d'améliorer fortement les résultats.

Références

[1] Francesco Barbieri, Luis Espinosa Anke et Jose Camacho-Collados. *XLM-T : Multilingual Language Models in Twitter for Sentiment Analysis and Beyond*. arXiv :2104.12250. type : article. arXiv, 11 mai 2022. arXiv : 2104.12250[cs]. URL : http://arxiv.org/abs/2104.12250.

[2] Vincent D. Blondel et al. "Fast unfolding of communities in large networks". In : *Journal of Statistical Mechanics : Theory and Experiment* 2008.10 (oct. 2008). Publisher : IOP Publishing, p. 12. ISSN : 1742-5468. DOI : 10.1088/1742-5468/2008/10/P10008. URL : https://doi.org/10.1088/1742-5468/2008/10/p10008.

[3] Stephen P. Borgatti et Martin G. Everett. "A Graph-theoretic perspective on centrality". In : *Social networks* 28.4 (2006). Place : Amsterdam Publisher : Elsevier B.V, p. 466-484. ISSN : 0378-8733. DOI : 10.1016/j.socnet.2005.11.005.

[4] *Data dictionary : Standard v1.1*. Twitter Developer Platform. URL : https://developer.twitter.com/en/docs/twitter-api/v1/data-dictionary/object-model/tweet.

[5] Jacob Devlin et al. "BERT : Pre-training of Deep Bidirectional Transformers for Language Understanding". In : *arXiv :1810.04805 [cs]* (2018). arXiv : 1810.04805. URL : http://arxiv.org/abs/1810.04805.

[6] Santo Fortunato et Marc Barthélemy. "Resolution limit in community detection". In : *Proceedings of the National Academy of Sciences of the United States of America* 104.1 (2 jan. 2007), p. 36-41. ISSN : 0027-8424. DOI : 10.1073/pnas.0605965104. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1765466/.

[7] Anastasia Giachanou et Fabio Crestani. "Like It or Not : A Survey of Twitter Sentiment Analysis Methods". In : *ACM Computing Surveys* 49.2 (juin 2017), 28 :1-28 :41. ISSN : 0360-0300. DOI : 10.1145/2938640. URL : https://doi.org/10.1145/2938640.

[8] Per Hage et Frank Harary. "Eccentricity and centrality in networks". In : *Social Networks* 17.1 (1er jan. 1995), p. 57-63. ISSN : 0378-8733. DOI : 10.1016/0378-8733(94)00248-9. URL : https://www.sciencedirect.com/science/article/pii/0378873394002489.

[9] Rushed Kanawati. *Détection de communautés dans les grands graphes d'interactions (multiplexes) : état de l'art*. Nov. 2013. URL : https://hal.archives-ouvertes.fr/hal-00881668.

[10] Yelena Mejova et Padmini Srinivasan. "Crossing Media Streams with Sentiment : Domain Adaptation in Blogs, Reviews and Twitter". In : *Proceedings of the International AAAI Conference on Web and Social Media* 6.1 (3 août 2021). Number : 1, p. 234-241. ISSN : 2334-0770. URL : https://ojs.aaai.org/index.php/ICWSM/article/view/14242.

[11] M. E. J. Newman. "Modularity and community structure in networks". In : *Proceedings of the National Academy of Sciences of the United States of America* 103.23 (6 juin 2006), p. 8577-8582. ISSN : 0027-8424. DOI : 10.1073/pnas.0601602103. URL : https://www.ncbi.nlm.nih.gov/pmc/articles/PMC1482622/.

[12] Bo Pang, Lillian Lee et Shivakumar Vaithyanathan. "Thumbs up ? : sentiment classification using machine learning techniques". In : *Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02*. the ACL-02 conference. T. 10. Not Known : Association for Computational Linguistics, 2002, p. 79-86. DOI : 10.3115/1118693.1118704. URL : http://portal.acm.org/citation.cfm?doid=1118693.1118704.

[13] Symeon Papadopoulos et al. "Community detection in Social Media". In : *Data Mining and Knowledge Discovery* 24.3 (1er mai 2012), p. 515-554. ISSN : 1573-756X. DOI : 10.1007/s10618-011-0224-z. URL : https://doi.org/10.1007/s10618-011-0224-z.

[14] Joerg Reichardt et Stefan Bornholdt. "Statistical Mechanics of Community Detection". In : *Physical Review E* 74.1 (18 juill. 2006), p. 016110. ISSN : 1539-3755, 1550-2376. DOI : 10.1103/PhysRevE.74.016110. arXiv : cond-mat/0603718. URL : http://arxiv.org/abs/cond-mat/0603718.

[15] Jeffrey Travers et Stanley Milgram. "An Experimental Study of the Small World Problem". In : *Sociometry* 32.4 (1969). Publisher : [American Sociological Association, Sage Publications, Inc.], p. 425-443. ISSN : 0038-0431. DOI : 10.2307/2786545. URL : http://www.jstor.org/stable/2786545.

[16] Quang Hong Vuong et Atsuhiro Takasu. "Transfer Learning for Emotional Polarity Classification". In : *2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT)*. 2014 IEEE/WIC/ACM International Joint Conferences on Web Intelligence (WI) and Intelligent Agent Technologies (IAT). T. 2. Août 2014, p. 94-101. DOI : 10.1109/WI-IAT.2014.85.

[17] Karl Weiss, Taghi M. Khoshgoftaar et DingDing Wang. "A survey of transfer learning". In : *Journal of Big Data* 3.1 (déc. 2016), p. 9. ISSN : 2196-1115. DOI : 10.1186/s40537-016-0043-6. URL : http://journalofbigdata.springeropen.com/articles/10.1186/s40537-016-0043-6.

Annexe 1 - Modèle de données
Description des tables et champs utilisés

- La table **hashs_0401_0415** décrit les hashtags présents dans les tweets.
  - hash_id (int) : identifiant du hashtag
  - hash (varchar) : texte du hashtag
- La table **tweet_hash_0401_0415** décrit la relation entre les hashtags et les tweets dans lesquels ils sont présents.
  - #tweet_id (int) : identifiant du tweet dans lequel le hashtag apparait
  - #hash_id (varchar) : identifiant du hashtag
- La table **tweet_0401_0415** décrit les tweets.
  - tweet_id (int) : identifiant du tweet
  - text (varchar) : texte du tweet
  - created_at (datetime) : date du tweet (time UTC)
  - #user_id (int) : identifiant de l'auteur du tweet
  - #in_reply_to_status_id (int) : si le tweet est une réponse, identifiant du tweet original
  - #in_reply_to_user_id (int) : si le tweet est une réponse, identifiant de l'auteur du tweet original
  - #quoted_status_id (int) : si le tweet est un commentaire, identifiant du tweet original
  - #quoted_user_id (int) : si le tweet est un commentaire, identifiant de l'auteur du tweet original
  - #retweeted_status_id (int) : si le tweet est un retweet, identifiant du tweet original
  - #retweeted_user_id (int) : si le tweet est un retweet, identifiant de l'auteur du tweet original
  - lang (varchar) : identifiant de la langue BCP 47 correspondant à la langue détectée ou 'und' si aucune langue n'a pu être détectée.
- La table **users_0401_0415** décrit les utilisateurs.
  - _user_id (int) : identifiant de l'utilisateur
  - screen_name (varchar) : pseudo de l'utilisateur
  - name (varchar) : nom de l'utilisateur
  - followers_count (int) : nombre de followers
  - friends_count (int) : nombre d'amis
- La table **user_mentions_0401_0415** décrit les tweets qui mentionnent un utilisateur.
  - #tweet_id (int) : identifiant du tweet
  - #source_user_id (int) : identifiant de l'auteur du tweet
  - #target_user_id (int) : identifiant de l'utilisateur mentionné

Annexe 2 - Table des différentes dénominations des candidats

Candidat	**dénominations du candidat**
Macron	'EmmanuelMacron', 'Emmanuel Macron','EnMarche','Macron2017' ,'MacronMarseille', 'MacronPau','TeamMacron','emmanuel macron', 'macron'
Fillon	'françois fillon', 'françoisfillon', 'fillon', Fillon2017', 'FillonPresident','FillonParis', ProjetFillon','FillonStrasbourg','FillonToulon','LR','FillonMarseille', 'FillonToulouse', 'FillonLyon','lesrepublicains', 'ProgrammeFillon','FrançoisFillon', 'FillonMontpellier', 'FillonClermont','FillonProvins'
Lepen	'LePen2017', 'MLP2017','Marine LePen','LePen','LePen','Marine2017','FN','Marine', 'AuNomDuPeuple','MLP','MarineLePen','AvecMarine','MarinePresidente','BordeauxMLP', 'JeChoisisMarine','LaFranceVoteMarine','MLPAjaccio','AjaccioMLP','world4marine'
Hamon	'BHRennes','Hamon2017','Hamon','ps','hamonElysée','HamonTour','RevenuUniversel', 'FuturDésirable'
Melenchon	"JLmelenchon','jean luc melenchon','JLMelenchon','Mélenchon','legoutdubonheur', 'JLMMarseille', 'JLMLille', 'JLMLeHavre', 'FranceInsoumise', 'AvenirEnCommun', 'JLM2017', 'JLMChateauroux', 'LaForcedupeuple','JLM','insoumis','JoursHeureux','JLMFrance2', '6eRépublique','LaFranceInsoumise','Melenchon2017','MélenchonAu2emeTourCestPossible'
Dupont Aignan	'NDA2017','DupontAignan','NDA'
médias	'RTLMatin', 'TF1','LeGrandDebat','LEmissionPolitique','onpc','bfmtv', 'LCI', 'BourdinDirect', 'cdanslair', 'cnews', 'beurfm', 'france2', 'DemainPrésident', 'granddebat', '19hruthelkrief','rediff','cavous','DebatBfm','BFM','LeGrandJury','BFMPolitique', 'TDinfos', 'Bourdin','Emissionpolitique','Les4Vérités','le79inter','Replay'

Annexe 3 - Code du projet

Lien vers le GitHub : https ://github.com/JulieLascar/PLDAC