IDENTIFIER LE LEADER D'OPINION LE PLUS INFLUENT SUR TWITTER

Introduction

Les entreprises ayant pour ambition de s'ouvrir de plus en plus à la communication digitale, en France comme à l'étranger, ont fini par admettre l'importance des réseaux sociaux dans le développement de leur Business model. Afin de mener au mieux leurs compagnes, il est impératif pour les entreprises d'identifier les leaders de la sphère digitale, alias les influenceurs.

Contexte

« *Le marketing d'influence doit être honnête et authentique. Un influenceur devrait parler de votre produit non pas parce qu'il est payé pour le faire, mais parce qu'il le veut. Idéalement, vous voulez qu'un influenceur vous appuie parce qu'il trouve votre entreprise intéressante* » (Forbes).

« *Pendant la plus grande partie de son existence dans la langue, le terme "influencer" a été utilisé au sens large pour désigner une personne ou une chose ayant le pouvoir de modifier les croyances des individus et, par conséquent, d'influer sur le cours des événements* » (SOLOMON, 2019).

Nous pouvons retrouver les définitions du mot 'influenceur' partout dans la toile aujourd'hui. Pourtant, il existe un vrai débat par rapport à la compréhension de chacun de nous. Qui est l'influenceur? Devrions-nous s'arrêter sur le nombre d'abonnés et likes ?

Ce travail consiste à identifier et mesurer l'impact des influenceurs sur la twittosphère.
Notre mission consiste à analyser une communauté, c'est-à-dire des ensembles de comptes twitter échangeant les uns avec les autres sur un sujet. Grâce à cela nous allons pouvoir identifier le leader d'opinion qui a été le plus influent dans la twittosphère sur le *#IA* sur une période de 1 an (01/01/2020 – 17/01/2021).

Les critères utilisés pour identifier les leaders d’opinion sur Twitter

Nous savons qu'un influenceur X ne peut être considéré comme un influenceur du #IA s'il ne poste pas au moins un contenu/post avec #IA sur la twittosphère. Supposons que Y est un utilisateur twitter. Afin d'établir un lien d'influence de X sur Y par le biais de son post, il faudra analyser les réactions de Y suite au post de X. Les différentes réactions possibles de Y sont au nombre de 5 :

- **Y** a retweeté **X**
- **Y** a mentionné **X** dans un post où il a mis le #IA
- **Y** follow **X**
- **Y** aime le post de **X**
- **Y** a commenté le post de **X**

L'expérience de Stanley Milgram en 1990 a démontré que nous sommes tous séparés par seulement six autres personnes sur cette planète. Ce qui veut dire que pour atteindre n'importe quelle personne sur cette planète, en parcourant les liens sociaux de notre entourage, nous passons par 6 personnes au maximum. Tandis que Facebook trouve que 6 degrés de séparation c'est trop, le fameux réseau social trouve que nous sommes tous séparés uniquement de 3,5 personnes (arrondi à 4 personnes dans la mémoire collective). Donc pour cette analyse, nous retiendrons qu'un contenu est pertinent, et donc potentiellement influent, s'il retient l'attention de au moins 5 personnes. Et de la même manière nous supposons que **Y**, **T**, **U**, **V**, **W** sont des utilisateurs twitter. Afin d'établir un lien d'influence de **X** sur **Y**, **T**, **U**, **V**, **W** par le biais de son post, il faudra analyser les réactions de **Y**, **T**, **U**, **V**, **W** suite au post de **X**. Les différentes réactions possibles de **Y** , **T**, **U**, **V**, **W** sont au nombre de 25:

- **Y** a retweeté **X**
- **Y** a mentionné **X** dans un post où il a mis le #IA
- **Y** follow **X**
- **Y** aime le post de **X**
- **Y** a commenté le post de **X**
- **T** a retweeté **X**
- **T** a mentionné **X** dans un post où il a mis le #IA
- **T** follow **X**
- **T** aime le post de **X** **T** a commenté le post de **X**
- **V** a retweeté **X**
- **V** a mentionné **X** dans un post où il a mis le #IA
- **V** follow **X**
- **V** aime le post de **X**
- **V** a commenté le post de **X**
- **U** a retweeté **X**
- **U** a mentionné **X** dans un post où il a mis le #IA
- **U** follow **X**
- **U** aime le post de **X**
- **U** a commenté le post de **X**
- **W** a retweeté **X**
- **W** a mentionné **X** dans un post où il a mis le #IA
- **W** follow **X**
- **W** aime le post de **X**
- **W** a commenté le post de **X**

Pendant la phase de la collecte, nous avons pu effectuer un scrapping de toutes les données de l'onglet "EXPLORER" sous #IA grâce à l'outil Twint. Les données collectées, ne pouvant servir uniquement à effectuer un graphe avec le lien :Y a mentionné X dans un post où il a mis le #IA. Malheureusement, nous n'avons pas réussi à scrapper les autres liens évoqués ci-dessus car Twitter a modifié sa page HTML depuis fin 2020 et donc le code source que nous avons initialement trouvé pour extraire les followers, par exemple, ne s'exécute plus. Nous avons décidé de nous focaliser, dans un premier temps, sur ce lien.

I] La présentation du jeu de données

Le jeu de données initial se compose de 217633 lignes et 36 colonnes. En-têtes de colonnes, nous retrouvant les étiquettes de colonnes suivantes :

'id', 'conversation_id', 'created_at', 'date', 'time', 'timezone', 'user_id', 'username', 'name', 'place', 'tweet', 'language', 'mentions', 'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count', 'hashtags', 'cashtags', 'link', 'retweet', 'quote_url', 'video', 'thumbnail', 'near', 'geo', 'source', 'user_rt_id', 'user_rt', 'retweet_id', 'reply_to', 'retweet_date', 'translate', 'trans_src', 'trans_dest'.

Afin de mettre en forme notre donnée et pouvoir mieux la manipuler nous avons choisi d'employer la fameuse librairie Pandas. Le Notebook Jupyter en annexes explique étape par étape la phase de nettoyage et de modélisation de données sur un échantillon de 700 lignes. L'échantillon compte 700 lignes représentées respectivement par des tweets. Sur ces 700 tweets, le compte des likes, des replies et des retweets sont de 700 chacun respectivement. En moyenne, beaucoup moins d'une personne répond aux tweets (0,12857), environ 1 seule personne retweet (1,127143) et un peu plus de 2 mettent des likes (2,245714).

La méthode et les étapes pour réaliser l'étude ; les résultats obtenus avec: les cartographies des acteurs les plus influents en fonction des différents critères, la détection des communautés.

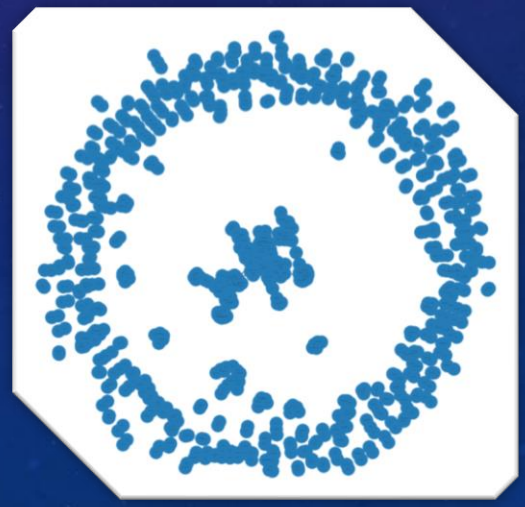
II] Analyse des graphes

L'analyse des réseaux sociaux est une méthode provenant de la sociologie, qui utilise la théorie des réseaux pour étudier les interactions sociales sur les réseaux sociaux. La théorie des réseaux sociaux conçoit les interactions sociales en termes de nœuds et liens. Les nœuds sont les acteurs sociaux dans le réseau mais ils peuvent aussi représenter des utilisateurs, et les liens sont les interactions ou des relations entre ces nœuds.

Il peut exister plusieurs sortes de liens entre les nœuds. Dans sa forme la plus simple, un réseau social se modélise pour former une structure analysable où tous les liens significatifs entre les nœuds sont étudiés. Il en va de même pour les trous structuraux, c'est-à-dire une absence de liens directs entre deux sommets. Il est entre autres possible par cette approche et méthode de déterminer le capital social des acteurs sociaux.

1. La structure des graphes

Le réseau est composé d'une structure plutôt oligopolistique, quelques comptes ont une influence et les autres possèdent une très faible influence, que ce soit par rapport aux liens entrants ou sortants.

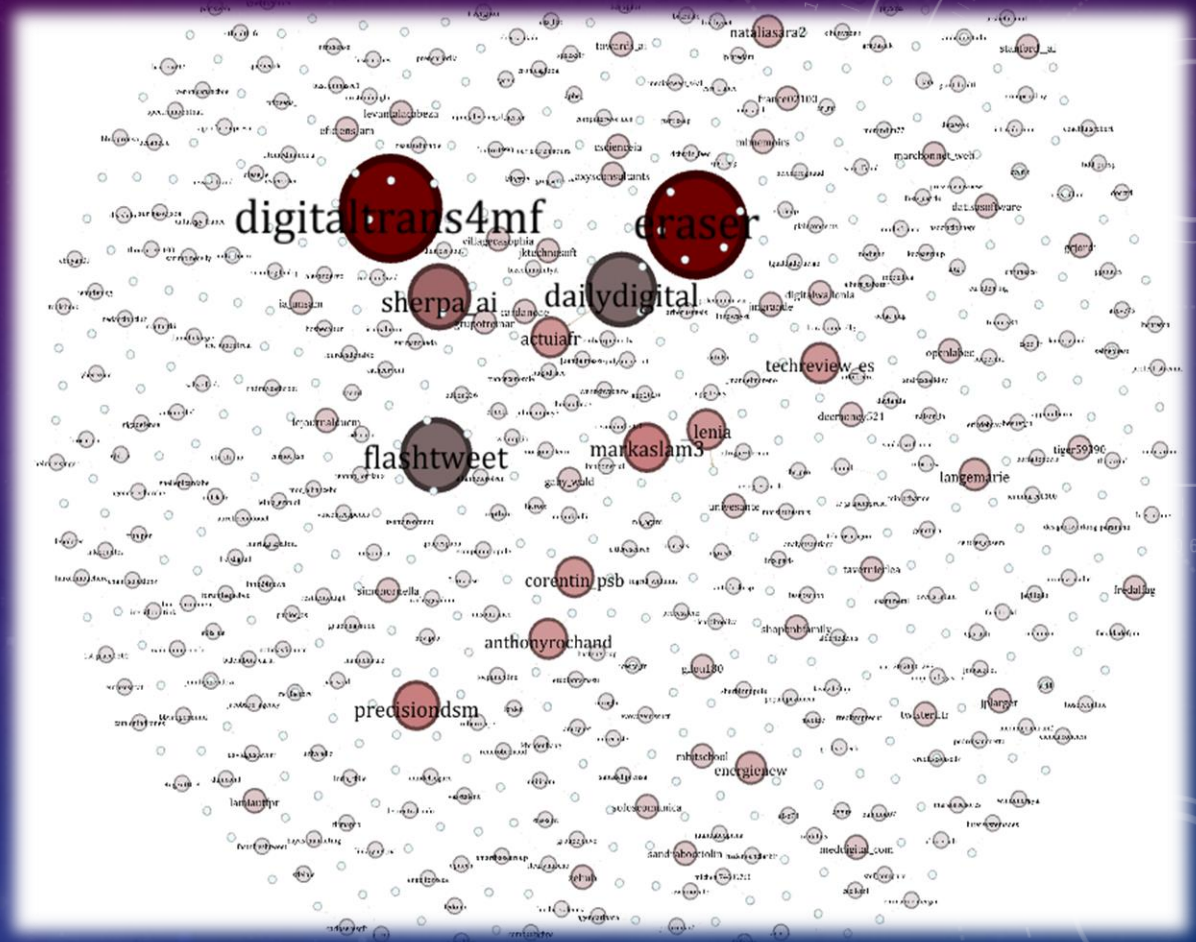


2.1 Les liens sortants

Les personnes ayant le plus de liens sortants représentent les diffuseurs de masse, c'est-à-dire les personnes transmettant le plus d'informations aux personnes suivant le #IA ou les personnes qui sont ses followers.

Nous retrouvons : 'digitaltrans4mf', 'eraser', 'dailydigital', 'flashtweet', 'sherpa_ai' ou encore 'techreview'.

Ces comptes que vous voyez sur le graph sont ceux qui diffusent le plus, malgré leur centralité et leur taille, ils ne représentent pas des leaders d'opinion. Ils sont certes très présents mais ils n'ont pas une grande influence. Cependant ils pourraient le devenir à l'avenir s'ils se feraient mieux connaître et posteraient du contenu qualitatif. »



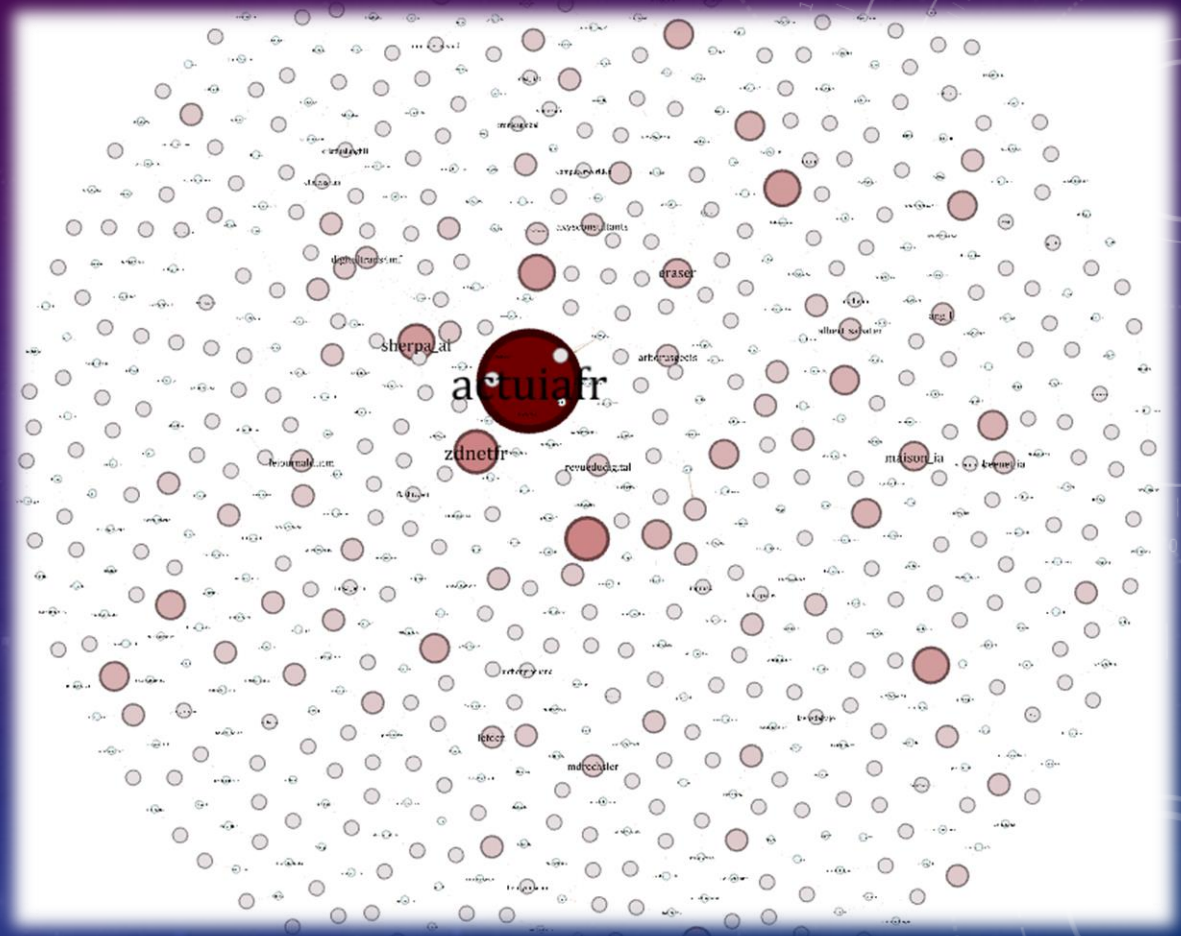
Ce ne sont donc pas ceux qui nous intéressent mais il est intéressant de voir les différentes possibilités d'analyse.

2.2 Les liens entrants

Les personnes ayant le plus de liens entrants représentent les leaders d'opinion. En effet, c'est vers ceux-là que l'attention est la plus élevée. les liens entrants sont symbolisés par des tags de leur @ dans des tweets ou par des retweets.

Nous retrouvons : "Actuiafr", "sherpa_ai", "zdnetfr" ou encore "eraser".

Ces comptes sont en général les plus suivis sur la plateforme, ici, sur le #IA en question. En effet, leur centralité montre en effet leur importance dans ce graphe. Ils sont ceux qui ont le plus de réactions sous leurs tweets et retweets.



Ces comptes sont donc ceux sur lesquels nous allons nous pencher car ils représentent les leaders d'opinion.

2.3 Le profil des plus grands influenceurs du #IA

1.Actuiafr

2.sherpa_ai

3.zdnetfr

4.eraser

Contenu des comptes

1.Actuiafr : Posts sur actualité sur l'IA, améliorations de process par l'IA, toutes nouvelles inventions / découvertes.

Régularité : 2-3 Posts tous les 2 jours

2.sherpa_ai : Posts sur les tendances dans l'IA, L'internet des objets, évènements, concepts novateurs.

Régularité : 3 Posts tous les jours

3.Zdnetfr : Posts sur l'actualité des entreprises en lien avec l'IA, découvertes informatiques.

Régularité : 5 Posts tous les jours

4.Eraser : Posts sur l'éducation des jeunes sur l'informatique, e-learning, conventions TED, blockchain.

Régularité : 15 Posts tous les jours

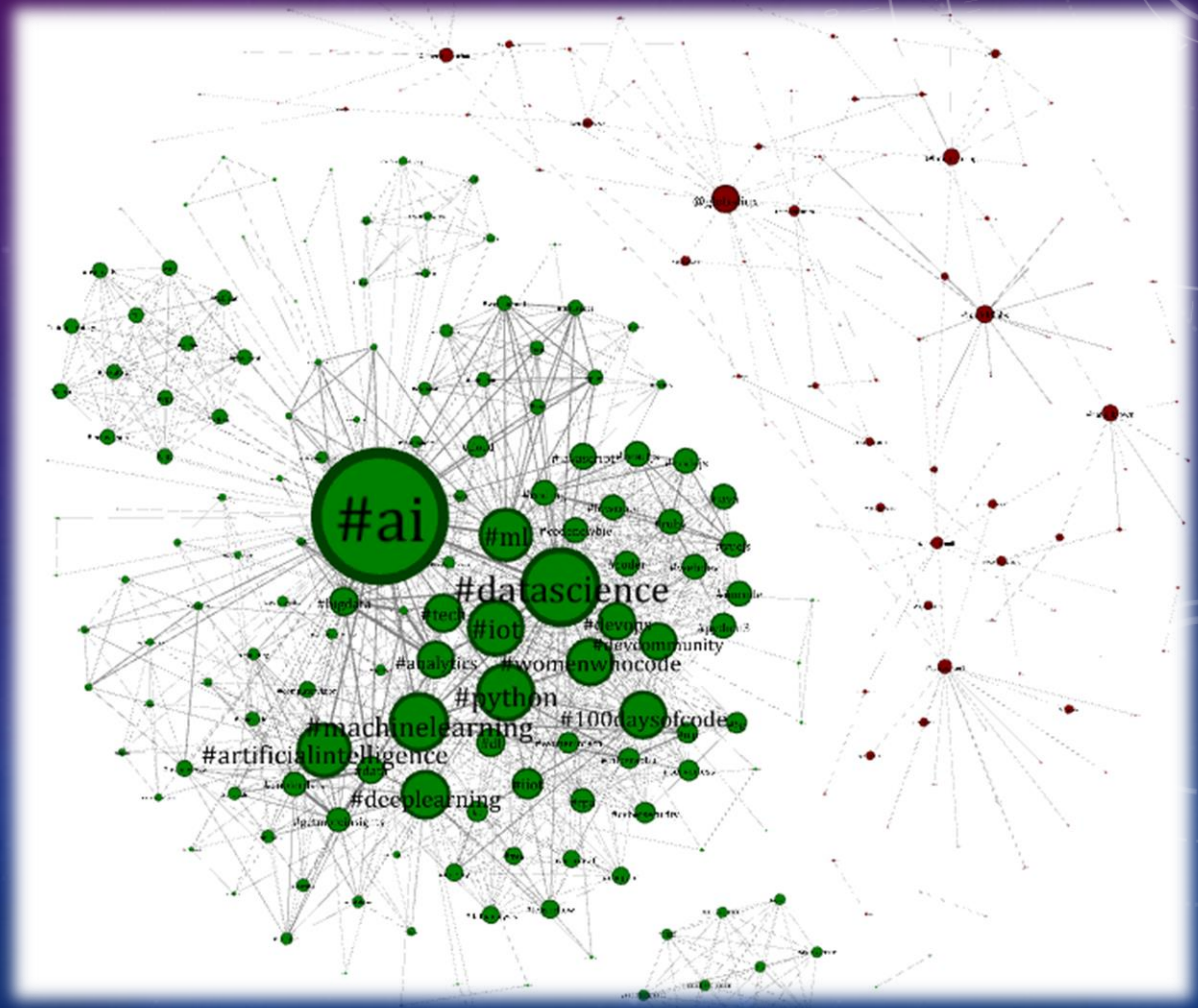
III] Analyse bonus en temps réel

Gephi nous donne accès aux données de Twitter en temps réel, en sélectionnant un '#' ou un '@' nous pouvons avoir toutes les données reliées à ces derniers.
Dans le cadre de cette étude, nous avons sélectionner le #ia et voici les résultats trouvés :

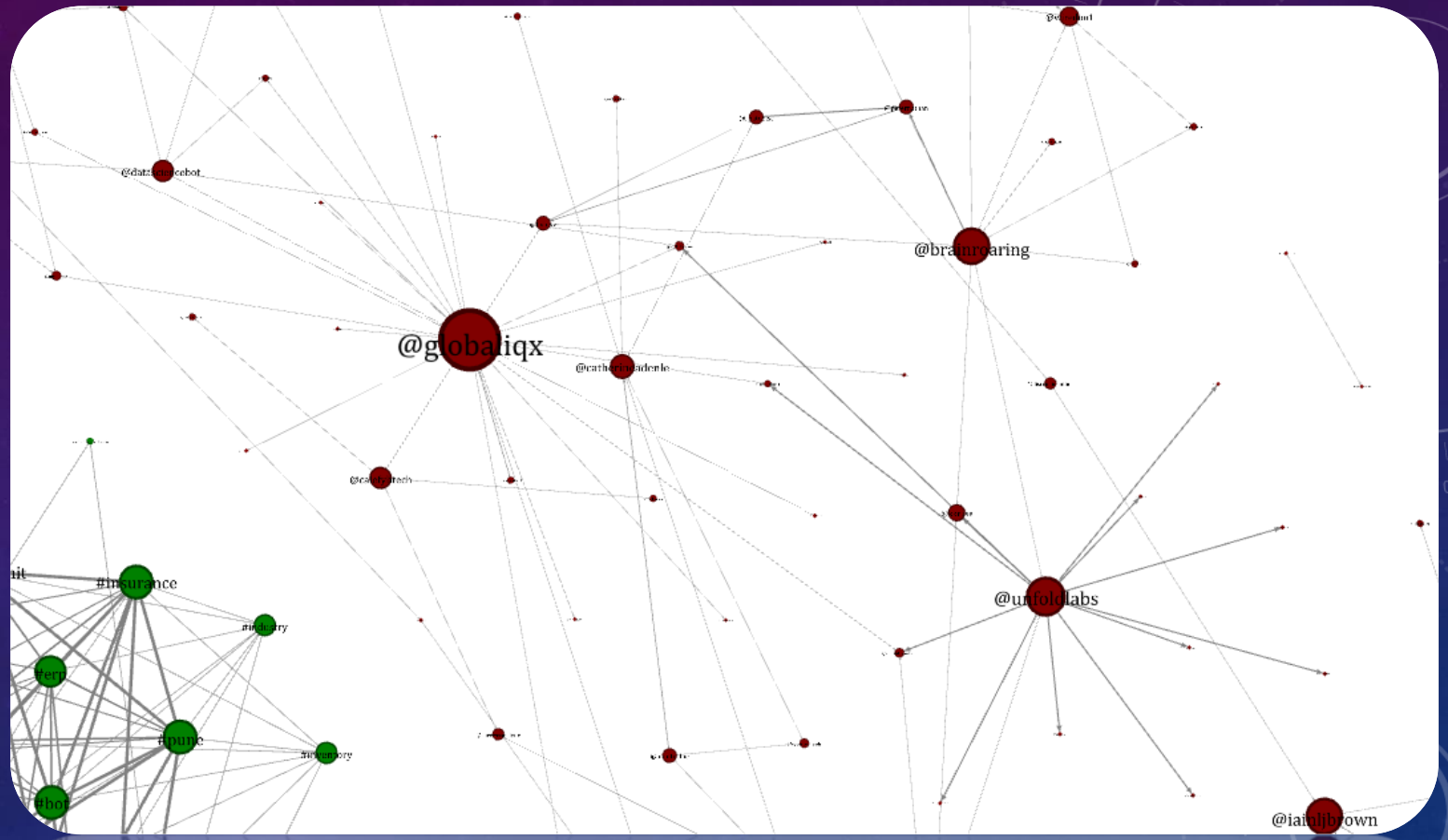
Comme nous pouvons le voir #ia est le centre de tout et nous apercevons les différents sujets selon leur importance.

Par exemple, data science, machine learning et internet of things sont des sujets très discutés.

Nous déduisons donc que ces sujets sont une partie majeure et très importante de l'IA.



Quand nous regardons de plus prêt, nous pouvons aussi voir qui, en temps réel, a le lead d'influence et ici par exemple les personnes les plus influentes sont : « globaliqx » et « unfoldlabs ».
Ces comptes sont actuellement les plus influents du #IA sur Twitter.



1.globaliqx

2.unfoldlabs

Contenu des comptes

1.Globaliqx : Posts sur Machine Learning, enseignement des jeunes, formations, cloud, robots.
Régularité : 5 Posts tous les jours

2.Unfoldlabs : Posts sur la Voice AI, transformation digitale des entreprises, deep learning, CGI.
Régularité : 5 Posts tous les jours

IV] Comment devenir un leader d'opinion sur la twittosphère

La popularité des posts s'articule autour de 3 axes selon moi :

Actualité : Parler de l'actualité est important, elle tient les followers de ce qu'il se passe en ce moment et beaucoup vont suivre un compte parce que le contenu est intéressant.

Régularité : Il faudra poster régulièrement sur des choses qui ont de l'intérêt sans « spammer ». Mettre trop de posts par jours pourrait faire se désabonner des gens car le contenu est trop volumineux, les gens veulent aussi voir les posts des autres personnes qu'ils suivent.

Découverte de nouvelles choses : Les découvertes nouvelles sont des posts très populaires. Savoir créer l'effet « WOW » est super important quand on veut faire découvrir quelque chose à quelqu'un. Par exemple, il peut s 'agir d'un nouveau logiciel révolutionnaire, un objet connecté qui va être utilisé par tout le monde ou encore un robot sachant faire des choses extraordinaires comme Boston Dynamics le montre sur les réseaux.

V] Axes d'amélioration

Le projet étant très ambitieux, nous avons été confrontés à plusieurs difficultés.

Pour commencer, concernant la récolte de données, plusieurs problèmes sont apparus.

Les données provenant de l'API officielle de Twitter sont très limitées au niveau de la récupération de données.

Des contraintes telles que les limites d'appels API est imposées par Twitter. Si l'on dépasse cette limite, nous recevons une pénalité de 15 min avant de pouvoir relancer un appel.

Cependant, nous avons trouvé des outils open source sur internet pour contourner ce problème.

Un outil développé en python se nommant Twint nous à permis de récolter des données sur des années d'historiques.

Twint permet grâce à des commandes qui facilitent son utilisation, de faire du web - scrapping.

En effet, cet outil nous a permis de récolter des tweets qui vont bien au-delà de 2 semaines d'historique.

Choisis arbitrairement, nous avons pu récolter des données de plus d'un an d'historique des tweets faisant référence au #IA.

Ainsi, il a été possible grâce à cette récolte, d'avoir plusieurs types de données tel que la provenance des tweets, la date d'émission, les retweets, le nombre de reply, mentions …

Une fonctionnalité proposée par Twint, et qui nous a semblé pertinente, et de pouvoir récupérer la liste des followers et following d'un utilisateurs.

Avec cette donnée, nous pouvons faire des analyses sur la notoriété d'une personne dans la sphère #IA, grâce à des calculs de centralité, clustering …

Malheureusement, cette fonctionnalité ne fonctionne plus depuis la mise à jour de Twitter le 15 Décembre 2020. Nous avons tout de même essayé de développer notre propre outils de web-scrapping en python afin de récupérer cette précieuse donnée. L'outils que nous avons développé fonctionne à 90 %, mais limité par la contrainte de temps, nous n'avons pas eu le temps nécessaire de la finaliser.

Conclusion

Nous nous apercevons à travers cette étude qu'il s'avère utile d'analyser les réseaux sociaux afin de comprendre et expliquer des phénomènes réels.
Cette étude nous a permis de trouver les personnes ayant de l'influence au sein d'une communauté juste à partir des liens que ces derniers ont.

Avec un peu de recul nous nous apercevons que ce qu'il se passe sur les réseaux sociaux est quelque chose d'applicable dans la vraie vie. En effet, certaines personnes représentent les centres d'intérêt, attire du monde et d'autres les suit et commence à imiter certains traits de personnalité.

Par exemple, dans le sport en général, il y a un certain nombre d'influenceurs / idoles et ces personnes ont un impact sur d'autres personnes qui les suivent. Ils vont soudainement avoir un rythme de vie plus sain, commencer à faire du sport…

Les réseaux sociaux renforcent la puissance d'influence des idoles, ce qui leur permet d'avoir encore plus d'impact sur les personnes qui les suit.

Le marketing d'influence semble donc une bonne idée afin d'étendre le rayon d'action d'une entreprise.

Annexes

Annexe 1 : Data Transformation P1

Annexe 2 : Data Transformation P2

Annexe 3 : Data Transformation P3

Annexe 4 : Data Transformation P4

Annexe 5 : Data Transformation P5

Annexe 6 : Data Transformation P6

Annexe 7 : Data Transformation P7

Annexe 8 : raw data (pièce jointe)

Annexe 9 : Notebook Data Transformation (pièce jointe)

Annexe 10 : Graph Structure (pièce jointe)

Annexe 11 : Graph Gephi Influenceurs (pièce jointe)

Annexe 12 : Graph Gephi #IA (pièce jointe)

Annexe 1 : Data Transformation P1

In [1]:
```
# Importation des modules
import pandas as pd
import datetime
import json
import numpy as np
```

In [2]:
```
# Ouverture du fichier CSV de la dataset source avec Pandas
df = pd.read_csv("file.csv",sep='\t',header=0)
```

In [3]:
```
# Vérification de la taille de la dataframe
df.shape
```

Out[3]:
```
(700, 36)
```

In [4]:
```
# Récupération des étiquettes de colonnes de la data frame
df.columns
```

Out[4]:
```
Index(['id', 'conversation_id', 'created_at', 'date', 'time', 'timezone',
       'user_id', 'username', 'name', 'place', 'tweet', 'language', 'mentions',
       'urls', 'photos', 'replies_count', 'retweets_count', 'likes_count',
       'hashtags', 'cashtags', 'link', 'retweet', 'quote_url', 'video',
       'thumbnail', 'near', 'geo', 'source', 'user_rt_id', 'user_rt',
       'retweet_id', 'reply_to', 'retweet_date', 'translate', 'trans_src',
       'trans_dest'],
      dtype='object')
```

In [5]:
```
# Supression des colonnes non pertinentes
df = df.drop(['name','place','language','urls','photos','cashtags','quote_url',
              'video','thumbnail','near','geo','source','user_rt_id','user_rt',
              'retweet_id','retweet_date','translate','trans_src','trans_dest'],
               axis=1)
```

Annexe 2 : Data Transformation P2

In [6]:
```
# Aperçu des 50 premières lignes de la dataframe
df.head(50)
```

Out[6]:

	id	conversation_id	created_at	date	time	timezone	user_id	username	tweet	
0	1383296665826697217	1383296665826697217	2021-04-17 07:50:00 CEST	2021-04-17	07:50:00	200	3122211	eraser	How China Is Using Artificial Intelligence in ...	[{'scre 'wsj', 'n
1	1383296246366932993	1383296246366932993	2021-04-17 07:48:20 CEST	2021-04-17	07:48:20	200	808671822761730049	sarimomojelly	完成しました(*ค'ค)💞✨ #IA #VOCALOID https://t.co/...	
2	1383295466226388995	1383295466226388995	2021-04-17 07:45:14 CEST	2021-04-17	07:45:14	200	613074939	sherpa_ai	Written by @ReidBlackman for @HarvardBiz: If Y...	[{'scre 'reidb 'nan
3	1383294235944701958	1383294235944701958	2021-04-17 07:40:20 CEST	2021-04-17	07:40:20	200	47255864	jmgrande	#IA de #Facebook mantiene #sesgo de #género ...	
			2021-04-							

Annexe 3 : Data Transformation P3

```
In [7]: # Renseignements sur la dataframe en général
        df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 700 entries, 0 to 699
Data columns (total 17 columns):
 #   Column           Non-Null Count  Dtype
---  ------           --------------  -----
 0   id               700 non-null    int64
 1   conversation_id  700 non-null    int64
 2   created_at       700 non-null    object
 3   date             700 non-null    object
 4   time             700 non-null    object
 5   timezone         700 non-null    int64
 6   user_id          700 non-null    int64
 7   username         700 non-null    object
 8   tweet            700 non-null    object
 9   mentions         700 non-null    object
 10  replies_count    700 non-null    int64
 11  retweets_count   700 non-null    int64
 12  likes_count      700 non-null    int64
 13  hashtags         700 non-null    object
 14  link             700 non-null    object
 15  retweet          700 non-null    bool
 16  reply_to         700 non-null    object
dtypes: bool(1), int64(7), object(9)
memory usage: 88.3+ KB
```

Annexe 4 : Data Transformation P4

```
In [8]: # Renseignements sur les types de données par colonne
        df.dtypes

Out[8]: id                int64
        conversation_id   int64
        created_at       object
        date             object
        time             object
        timezone          int64
        user_id           int64
        username         object
        tweet            object
        mentions         object
        replies_count     int64
        retweets_count    int64
        likes_count       int64
        hashtags         object
        link             object
        retweet            bool
        reply_to         object
        dtype: object

In [9]: # Focus sur la colonne que nous allons utiliser
        print(df.iloc[1,9])

        []

In [10]: # Extraire liste de toute personne mentionnée par id et et la liste de leur index dans la df
         l_index_mentions = []
         l_mentions = []
         for i in df.index:
             mentions = df['mentions'][i]
             if mentions != "[]":
                 mentions = mentions.replace('"',"'").replace("{'", '{"').replace("': '",'": "').replace("', '", '", "').repl
                 json_mentions = json.loads(mentions)
                 for mention in json_mentions:
                     mention_id = mention['id']

                     l_index_mentions.append(i)
                     l_mentions.append(mention_id)
```

Annexe 5 : Data Transformation P5

```
In [13]: # conversion des listes en numpy arrays
         np_index_mentions = np.array(l_index_mentions,int)
         np_mentions = np.array(l_mentions)

In [14]: ''' Créaton d'une dataframe avec index de l'id de la mention
         et l'id du mentionn'''

         df_mentions = pd.DataFrame({'index_joint_mentions':np_index_mentions,
                                     'id_users_mentions':np_mentions})

In [15]: # Aperçu des 5 premières lignes de la dataframe
         df_mentions.head()
```

Out[15]:

	index_joint_mentions	id_users_mentions
0	0	3108351
1	2	1161699135587782656
2	2	14800270
3	4	113055050
4	5	1591237452

Annexe 6 : Data Transformation P6

```
In [16]: # Ajout d'une colonne d'indexation pour la jointure
         df['index_joint'] = range(len(df.index))
         df.head()
```

Out[16]:

	id	conversation_id	created_at	date	time	timezone	user_id	username	tweet	mentions	rep
0	1383296665826697217	1383296665826697217	2021-04-17 07:50:00 CEST	2021-04-17	07:50:00	200	3122211	eraser	How China Is Using Artificial Intelligence in ...	[{'screen_name': 'wsj', 'name': 'the wall stre...	
1	1383296246366932993	1383296246366932993	2021-04-17 07:48:20 CEST	2021-04-17	07:48:20	200	808671822761730049	sarimomojelly	完成しました ("ŕ‿ŕ")💞✨ #IA #VOCALOID https://t.co/...	[]	
2	1383295466226388995	1383295466226388995	2021-04-17 07:45:14 CEST	2021-04-17	07:45:14	200	613074939	sherpa_ai	Written by @ReidBlackman for @HarvardBiz: If Y...	[{'screen_name': 'reidblackman', 'name': 'reid...	
3	1383294235944701958	1383294235944701958	2021-04-17 07:40:20 CEST	2021-04-17	07:40:20	200	47255864	jmgrande	#IA de #Facebook mantiene #sesgo de #género ...	[]	
4	1383294064582217734	1383137732659142657	2021-04-17 07:39:40 CEST	2021-04-17	07:39:40	200	95177547	adsuara	"Una aproximación basada en #riesgo. Los siste...	[{'screen_name': 'vrbenjamins', 'name': 'richa...	

```
In [17]: # fusion des deux df
         df_merge =  df_mentions.merge(df,left_index=True,right_index=True,suffixes=("index_joint","index_joint_mentions"))
```

Annexe 7 : Data Transformation P7

In [18]:
```
#Aperçu des 5 premières lignes de la dataframe
df_merge.head()
```

Out[18]:

	index_joint_mentions	id_users_mentions	id	conversation_id	created_at	date	time	timezone	user_id	us
0	0	3108351	1383296665826697217	1383296665826697217	2021-04-17 07:50:00 CEST	2021-04-17	07:50:00	200	3122211	
1	2	1161699135587782656	1383296246366932993	1383296246366932993	2021-04-17 07:48:20 CEST	2021-04-17	07:48:20	200	808671822761730049	sarim
2	2	14800270	1383295466226388995	1383295466226388995	2021-04-17 07:45:14 CEST	2021-04-17	07:45:14	200	613074939	sl
3	4	113055050	1383294235944701958	1383294235944701958	2021-04-17 07:40:20 CEST	2021-04-17	07:40:20	200	47255864	jn
4	5	1591237452	1383294064582217734	1383137732659142657	2021-04-17 07:39:40 CEST	2021-04-17	07:39:40	200	95177547	

In [20]:
```
# Export de la Df en CSV
df_merge.to_csv('mentions.csv')
```

Merci de votre attention !