Evolution de la complexité du Code Général des Impôts de 1980 à 2022

Analyse du Code Général des Impôts, de ses 4 annexes et du Livre des Procédures Fiscales

Le Code Général des Impôts s'est-il complexifié en 43 ans ?

Méthodes utilisées :

- Dataset obtenue par scraping de Legifrance en Javascript avec axios et cheerio (équivalents requests et bs4) avec choix d'une structure JSON.
- Analyse de données avec pandas et obtention de graphiques.
- Processing d'articles avec des RegExp, création d'un réseau avec NetworkX et visualisation avec pyvis.

Gaëtan Ruet

Evolution du vocabulaire du dataset

```
liste_mots_reduced = pd.read_csv('common_words.txt', header=None, lineterminator=',')
liste_mots_reduced.iloc[5752] = "zones"
df = pd.read_json('CGI_r.json')
content_1980_reduced = df.loc[df.year == 1980].content
content_2022_reduced = df.loc[df.year == 2022].content
results_reduced = []
for mot in liste_mots_reduced[0]:
  word_count_1980_reduced = 0
  word_count_2022_reduced = 0
  for article in content_1980_reduced:
    word_count_1980_reduced += len(re.findall(mot, article, re.IGNORECASE))
  for article in content_2022_reduced:
    word_count_2022_reduced += len(re.findall(mot, article, re.IGNORECASE))
    # I added + 1 because sometimes it was 0 in 1980
  results_reduced.append([mot, word_count_1980_reduced, word_count_2022_reduced,
  (word_count_2022_reduced-word_count_1980_reduced)/(word_count_1980_reduced + 1)])
df_words_reduced = pd.DataFrame(results_reduced)
```

Recherche du nombre d’occurrences de chaque mot d’un dictionnaire des 5000 mots les plus courants dans les versions 1980 et 2022 afin de trouver quels sont les mots dont la popularité a varié.

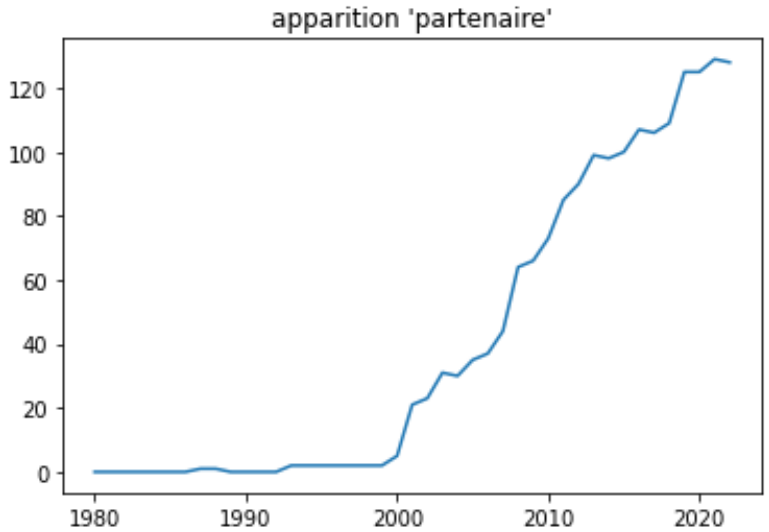
Evolution du vocabulaire du dataset : résultats sur les mots les plus courants (occurrences > 50)

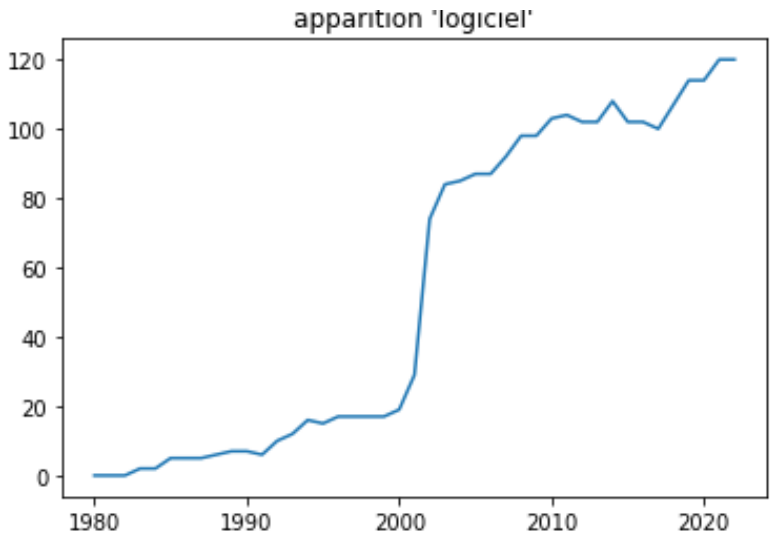
Les plus importants changements sont relatifs à l'émergence de nouvelles technologies, aux changements du tissu économique, à l'émergence de nouvelles pratiques fiscales…

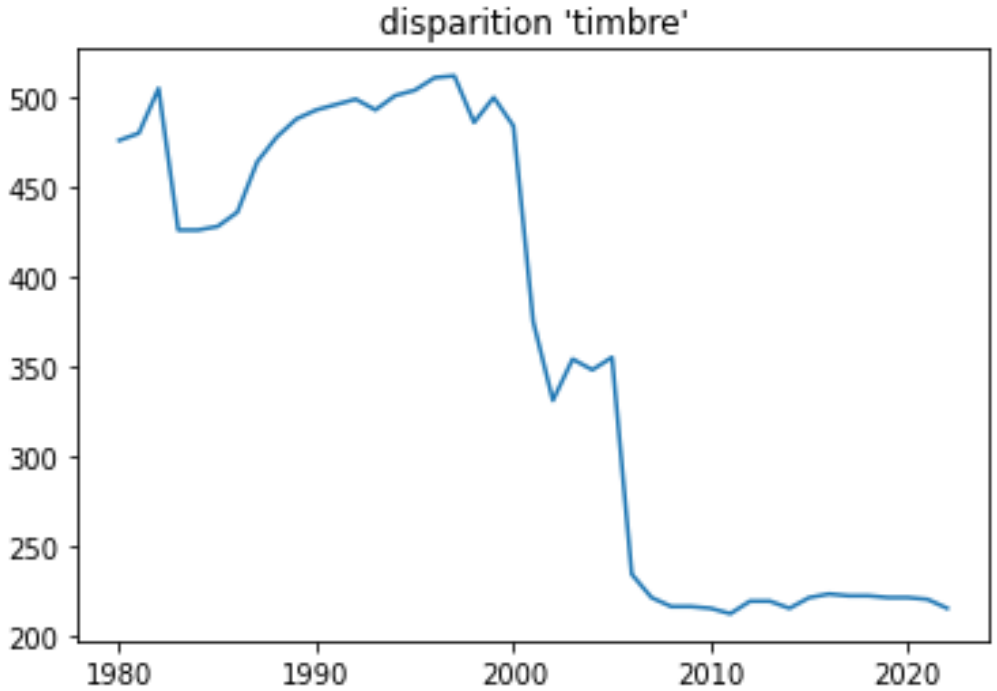
mots	1980	2022	augmentation
audiovisuel	0	181	181.0
accise	0	171	171.0
aides	1	288	143.5
intra	0	136	136.0
partenaire	0	128	128.0
trust	0	123	123.0
logiciel	0	120	120.0
audiovisuelle	0	112	112.0
monétaire	3	437	108.5
réhabilitation	0	107	107.0
informatique	1	214	106.5

mots	1980	2022	augmentation
huile	124	8	-0.928000
raisin	68	6	-0.898551
spiritueux	61	9	-0.838710
marchands	90	17	-0.802198
timbré	52	10	-0.792453
gros	139	35	-0.742857
cote	105	35	-0.660377
voiture	82	28	-0.650602
campagne	94	36	-0.610526
ferment	52	20	-0.603774
vigne	70	28	-0.591549

Evolution du vocabulaire du dataset : résultats sur les mots les plus courants (occurrences > 50)







Evolution de la complexité de la structure : les références à d'autres articles et codes

```python
def processArticle(article):
    """we process the article to use simple functions to match article"""
    # we remove 'dispositions -1' for example, not -0 because many articles use this format
    for w in [*["-1", "-2", "-3"], *traite]:
        article = article.replace(w, "")
    # we remove codes and 40 chars before
    for c in codes:
        article = re.sub(removeBefore(c), "", article)

    # part of the failed attempt to use a less naive technique
    #article = re.sub(myreg, replaceText, article.replace("articles", "article"))

    # we replace 'article 1 à 3' by 'article 1 article 2 article 3'
    article = re.sub(r"articles* (\d+) à (\d+)", rangeReplacement, article)

    # we replace 'article 2, 3' by 'article 2 article 3' recursively
    while True:
        output = re.sub(r"articles* (\d+) *(bis|ter|quater|quinquies|sexies|septies|octies|nonies|
        if output == article:
            break
        article = output
    # we replace 'article 2 et 3' by 'article 2 article 3' recursively
    while True:
        output = re.sub(r"articles* (\d+) *(bis|ter|quater|quinquies|sexies|septies|octies|nonies|
        if output == article:
            break
        article = output
    # we replace 'articles 1 à 3' by 'article 1 article 2 article 3' again to account for
    # situations like 'articles 2 et 5 à 17'
    article = re.sub(r"articles* (\d+) à (\d+)", rangeReplacement, article)
    article = article.replace('articles', 'article').lower().replace(
        ",", " ").replace(".", " ").replace(";", " ")
    return article
```

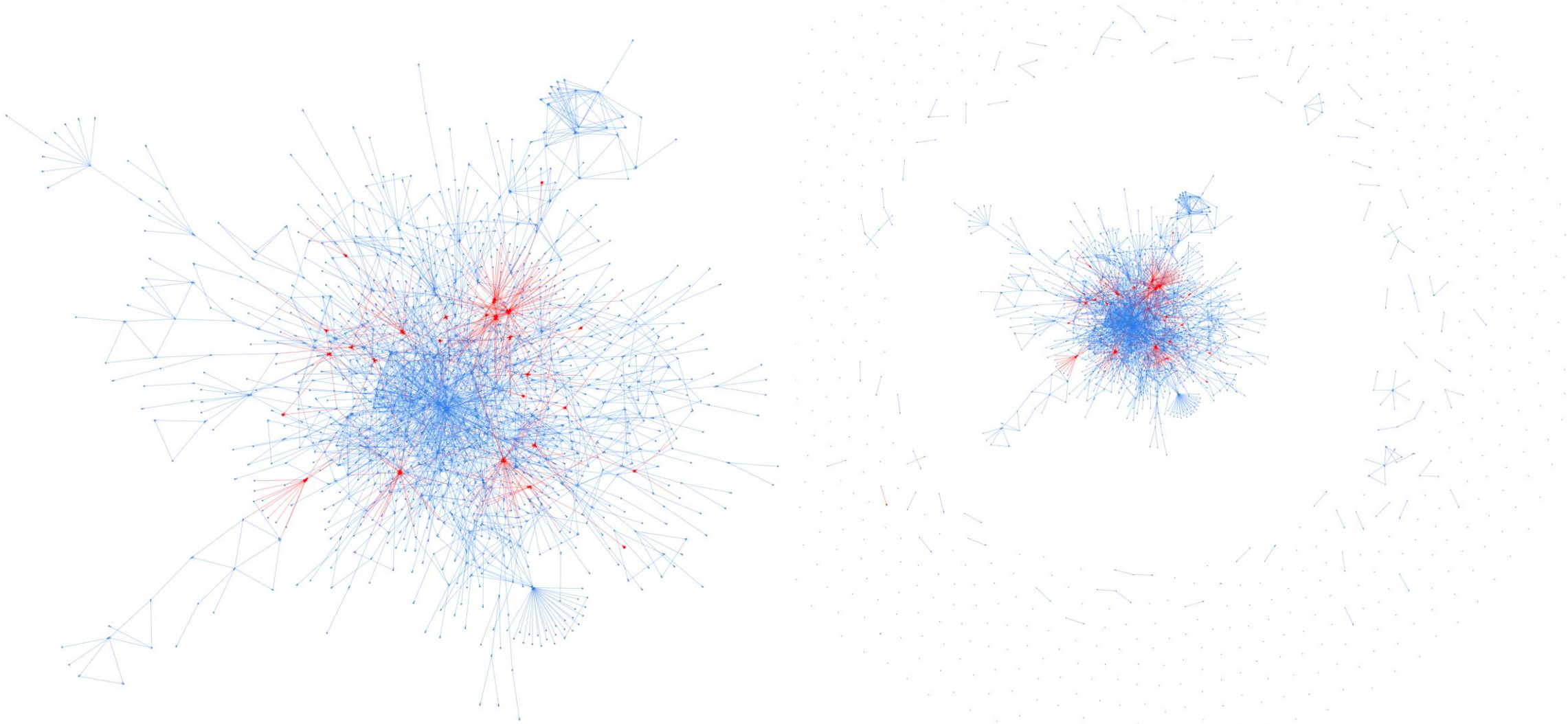
Chaque article est transformé pour trouver les références à d'autres articles. Par exemple : « articles 1, 2 et 10 à 13 » devient « article 1 article 2 article 10 article 11 article 12 article 13 ».

Des méthodes moins naïves peuvent être implémentées mais requièrent plus de puissance de calculs ou une meilleure utilisation des RegExp.

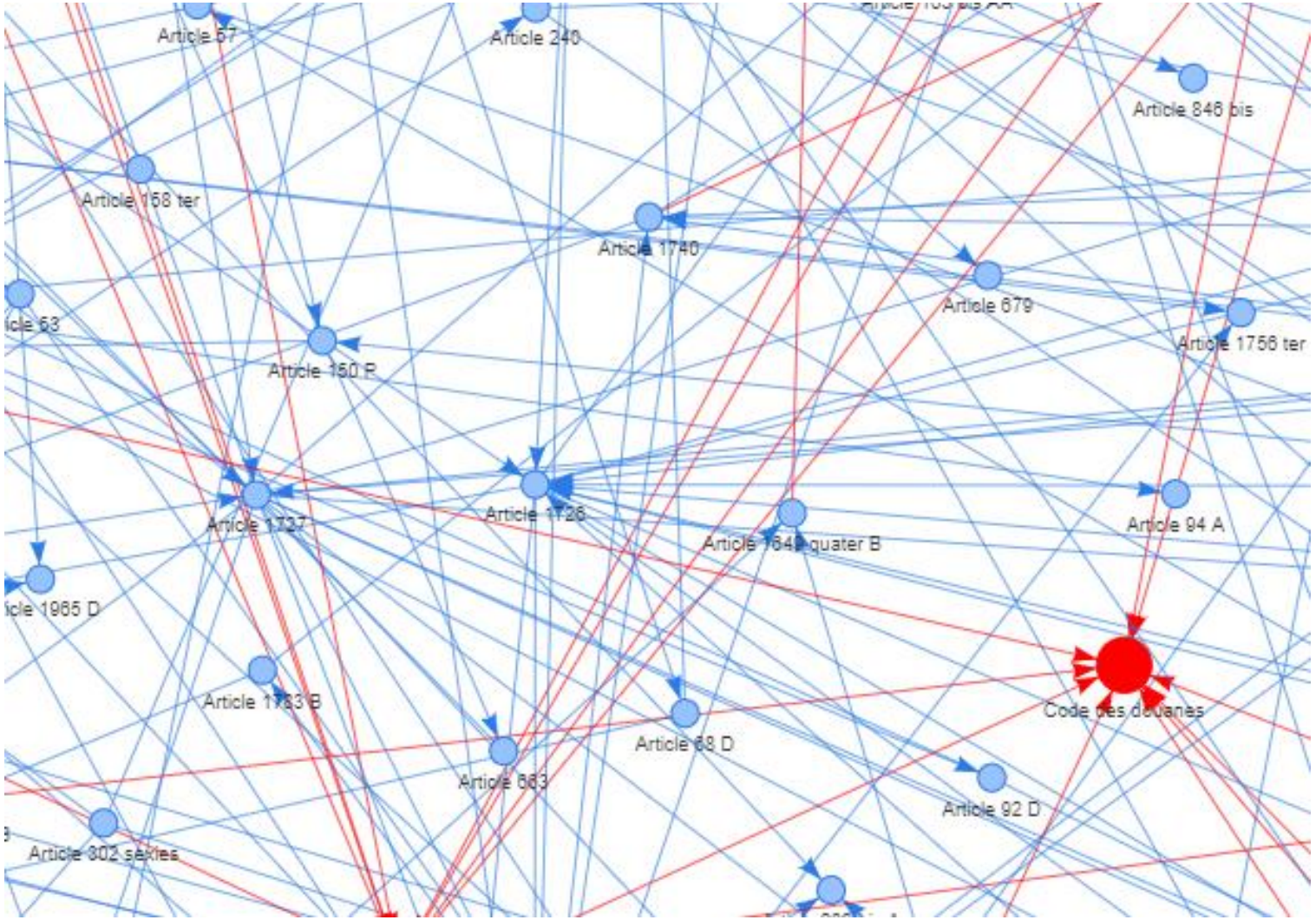
On cherche toutes les références à d'autres articles du Code Général des Impôts dans les articles du Code Général des Impôts.

On crée un réseau à l'aide des données identifiées grâce à la librairie NetworkX puis on utilise une autre librairie pyvis pour visualiser ce réseau.

Evolution de la complexité de la structure : les références à d'autres articles et codes : *1980*



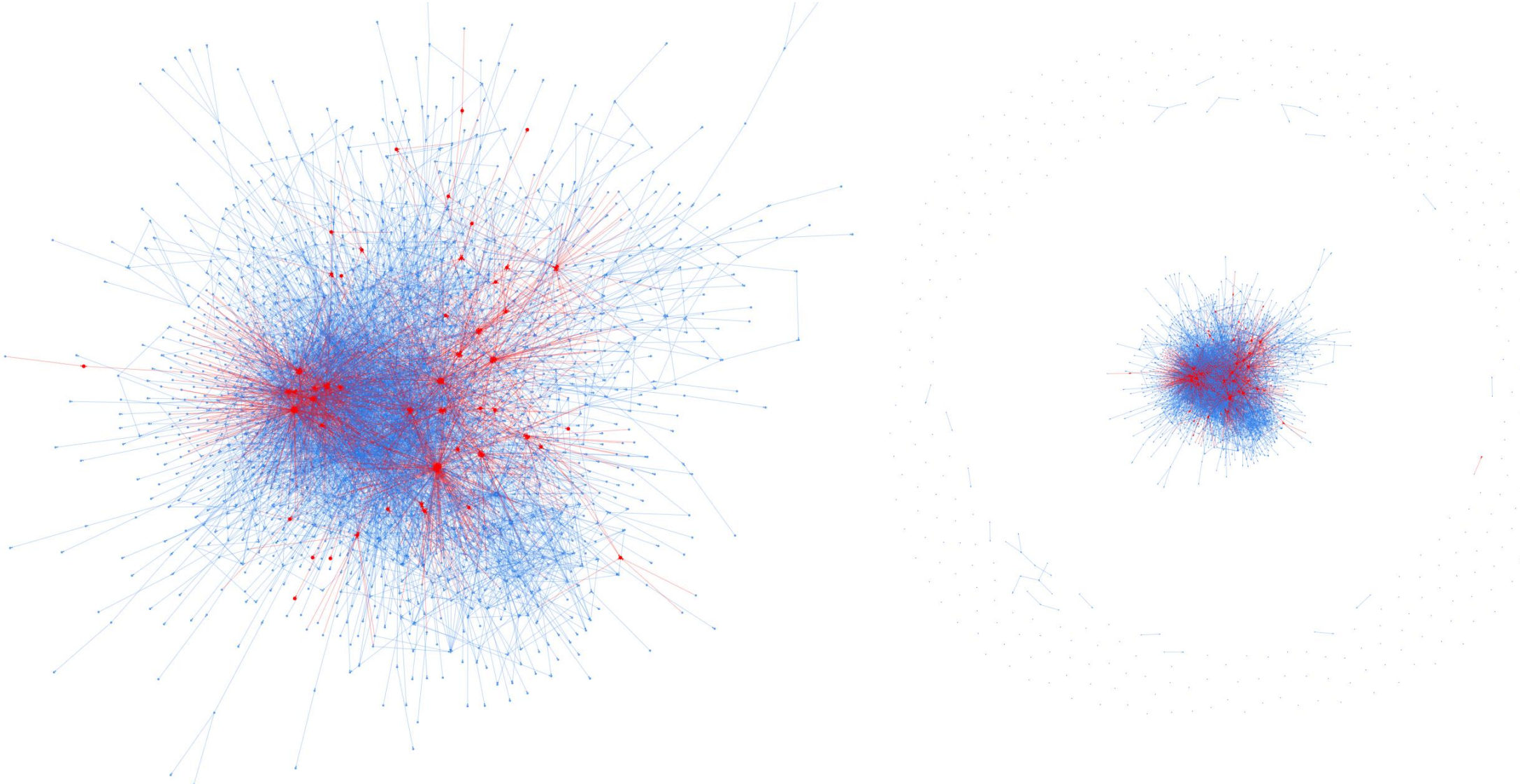
Evolution de la complexité de la structure : les références à d'autres articles et codes : *1980*



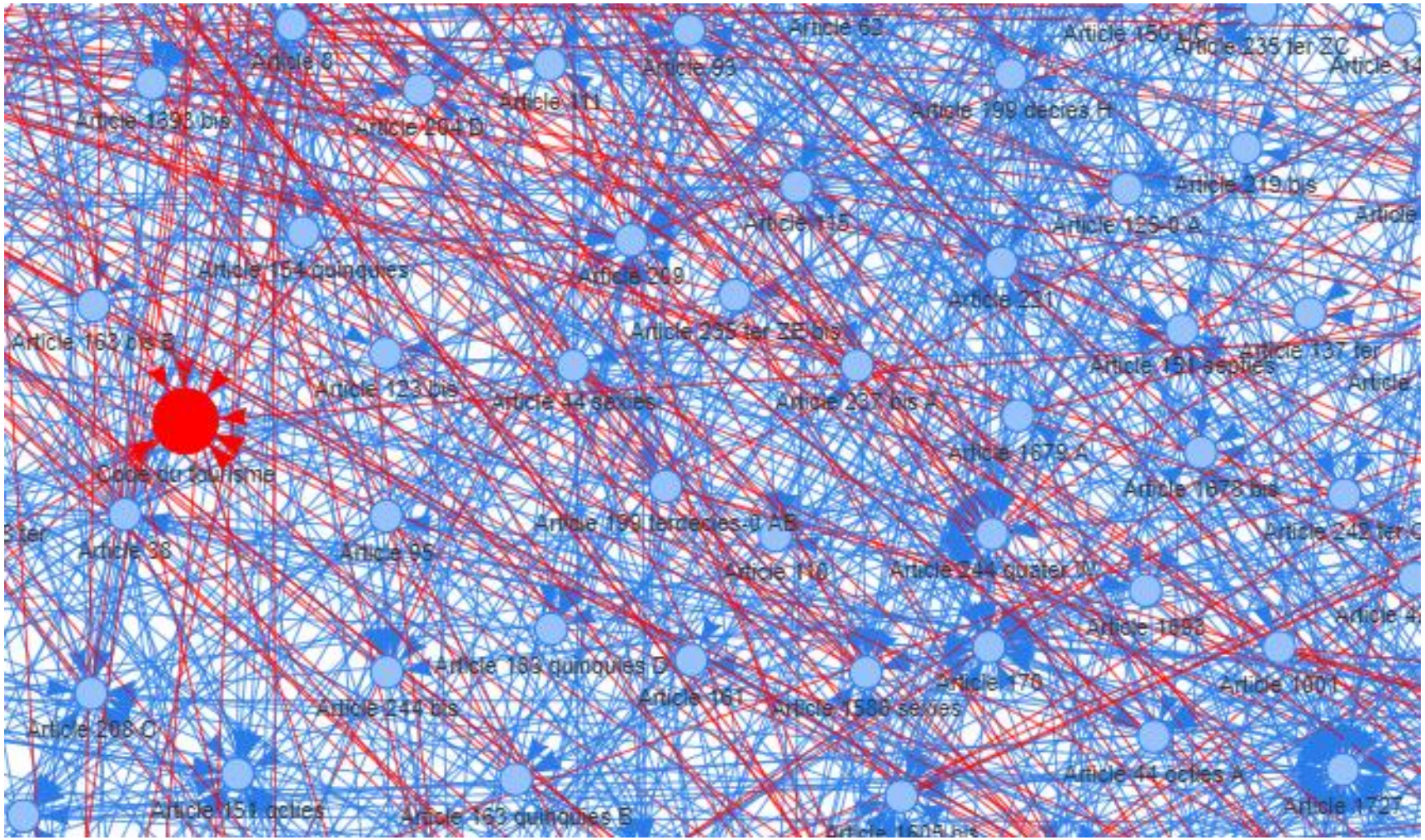
Les articles les plus importants au sein de l'algorithme pagerank sont :

Article 231
Article 1020
Article 8
Article 1649 quinquies A
Article 196
Article 6
Article 1726
Article 39
Article 1727
Article 1679

Evolution de la complexité de la structure : les références à d'autres articles et codes : *2022*



Evolution de la complexité de la structure : les références à d'autres articles et codes : *2022*



Les articles les plus importants au sein de l'algorithme pagerank sont :

Article 8
Article 39
Article 206
Article 1639 A bis
Article 4 B
Article 1020
Article 6
Article 1594 D
Article 170
Article 39 duodecies

Conclusion

La complexité du CGI a fortement augmenté en 43 ans malgré les diverses promesses et tentatives de simplification. Non seulement, il y a plus d'articles, plus de caractères mais les articles sont aussi de plus en plus liés les uns les autres.

Mais l'augmentation de la complexité du CGI n'est pas qu'endogène mais aussi exogène, elle est le reflet de l'augmentation de la complexité de la société.

Questions ?