modifier le xrm chaque jour, aller dans requête réseau, recharger la page, xmr et delivery, remplacer le authorization

Extraction

In [ ]:
```
import requests
import json
import os

# ================================================================================
# 1. Paramètres de l'API et configuration générale
# ================================================================================

# URL de l'API Primo Paris Nanterre
url = "https://primo.parisnanterre.fr/primo_library/libweb/webservices/re

# En-têtes pour l'API
headers = {
    "Accept": "application/json, text/plain, */*",
    "Cookie": "JSESSIONID=4A5251AA12BBBD31DD25D2C99D126D9F; TBMCookie_355
    "Referer": "https://primo.parisnanterre.fr/primo-explore/search?insti
    "Accept-Language": "fr-FR,fr;q=0.9",
    "Host": "primo.parisnanterre.fr",
    "User-Agent": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleW
    "Authorization": "Bearer eyJraWQiOiJwcmltb0V4cGxvcmVQcml2YXRlS2V5LVND
    "Accept-Encoding": "gzip, deflate, br",
    "Connection": "keep-alive",
}

# Liste des mots-clés à rechercher (modifiable)
mots_cles = ["EU Emissions Trading System", "EU ETS", "European Carbon Ma
"Carbon pricing", "Carbon allowance trading", "Carbon abatement",
"Low carbon investment", "Green investment", "Technological change",

"Système d’échange de quotas d’émission de l’UE", "Marché du carbone euro
"Tarification du carbone", "Réduction des émissions de CO2",
"Investissement bas carbone", "Transition énergétique"]

# Dictionnaires globaux pour stocker les résultats bruts et les données e
resultats_globaux = {}
resultats_globaux_extraits = {}

# ================================================================================
# 2. Scraping et sauvegarde des résultats bruts
# ================================================================================

for mot_cle in mots_cles:
    print(f"\n🔍 Recherche pour le mot-clé : {mot_cle}")
```

```
    # Paramètres de la requête (le mot-clé est inséré dynamiquement)
    params = {
        "acTriggered": "false",
        "blendFacetsSeparately": "false",
        "citationTrailFilterByAvailability": "true",
        "getMore": "0",
        "inst": "SCD",
        "isCDSearch": "false",
        "lang": "fr_FR",
        "limit": "10",  # Nombre de résultats par page
        "mode": "Basic",
        "newspapersActive": "false",
        "newspapersSearch": "false",
        "otbRanking": "false",
        "pcAvailability": "false",
        "q": f"any,contains,{mot_cle}",
        "scope": "all_blended",
        "sort": "rank",
        "tab": "all_tab",
        "vid": "SCD",
    }

    all_results = []
    offset = 0
    max_pages = 30  # Nombre maximum de pages à scraper
    page_count = 0

    while page_count < max_pages:
        params["offset"] = offset
        response = requests.get(url, headers=headers, params=params)

        if response.status_code == 200:
            data = response.json()
            # La clé "docs" contient les résultats (adapter si la structu
            results = data.get("docs", [])
            if not results:
                break

            all_results.extend(results)
            offset += len(results)
            page_count += 1
        else:
            print(f"  ⚠️ Erreur HTTP {response.status_code} pour '{mot_cl
            print(response.text)
            break

    print(f"  ✅ {len(all_results)} résultats collectés pour '{mot_cle}'.

    # Sauvegarde des résultats bruts dans un fichier
    fichier_brut = f"resultats_{mot_cle}.json"
    with open(fichier_brut, "w", encoding="utf-8") as f:
        json.dump(all_results, f, indent=4, ensure_ascii=False)
    print(f"  📁 Résultats bruts sauvegardés dans '{fichier_brut}'.")
```

```
    resultats_globaux[mot_cle] = all_results

    # ================================================================
    # 3. Extraction des informations pour les enregistrements de type "ar
    # ================================================================

    extracted_data = []

    for record in all_results:
        pnx = record.get("pnx", {})  # Récupérer la section "pnx" si disp
        # Extraction de la valeur "rsrctype" depuis la section "search"
        rsrctype = pnx.get("search", {}).get("rsrctype", [""])[0]
        # On ne conserve que les enregistrements de type "article"
        if rsrctype != "article":
            continue

        extracted_info = {
            "title": pnx.get("display", {}).get("title", [""])[0],
            "authors": pnx.get("display", {}).get("creator", []),
            "description": pnx.get("display", {}).get("description", [""]
            "subjects": pnx.get("search", {}).get("subject", []),
            "creation_date": pnx.get("search", {}).get("creationdate", ["
            "source": pnx.get("display", {}).get("publisher", [""])[0],
            "full_text_link": pnx.get("links", {}).get("linktorsrc", [""]
            "record_id": pnx.get("control", {}).get("recordid", [""])[0],
            "rsrctype": rsrctype
        }
        extracted_data.append(extracted_info)

    # Sauvegarde des données extraites dans un fichier spécifique au mot-
    fichier_extrait = f"resultats_{mot_cle}_extraits.json"
    with open(fichier_extrait, "w", encoding="utf-8") as f:
        json.dump(extracted_data, f, indent=4, ensure_ascii=False)
    print(f"  ✅ Données extraites sauvegardées dans '{fichier_extrait}'.
    resultats_globaux_extraits[mot_cle] = extracted_data

# ================================================================
# 4. Sauvegarde globale des résultats bruts et extraits
# ================================================================

with open("resultats_globaux.json", "w", encoding="utf-8") as f:
    json.dump(resultats_globaux, f, indent=4, ensure_ascii=False)
with open("resultats_globaux_extraits.json", "w", encoding="utf-8") as f:
    json.dump(resultats_globaux_extraits, f, indent=4, ensure_ascii=False

print("\n✅ Tous les résultats (bruts et extraits) ont été sauvegardés.")
```

```
🔍 Recherche pour le mot-clé : EU Emissions Trading System
  ✅ 300 résultats collectés pour 'EU Emissions Trading System'.
  📁 Résultats bruts sauvegardés dans 'resultats_EU Emissions Trading Sys
tem.json'.
  ✅ Données extraites sauvegardées dans 'resultats_EU Emissions Trading
System_extraits.json'.
```

```
🔍 Recherche pour le mot-clé : EU ETS
  ✅ 300 résultats collectés pour 'EU ETS'.
  📁 Résultats bruts sauvegardés dans 'resultats_EU ETS.json'.
  ✅ Données extraites sauvegardées dans 'resultats_EU ETS_extraits.json'
.

🔍 Recherche pour le mot-clé : European Carbon Market
  ✅ 300 résultats collectés pour 'European Carbon Market'.
  📁 Résultats bruts sauvegardés dans 'resultats_European Carbon Market.j
son'.
  ✅ Données extraites sauvegardées dans 'resultats_European Carbon Marke
t_extraits.json'.

🔍 Recherche pour le mot-clé : Carbon pricing
  ✅ 300 résultats collectés pour 'Carbon pricing'.
  📁 Résultats bruts sauvegardés dans 'resultats_Carbon pricing.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Carbon pricing_extrai
ts.json'.

🔍 Recherche pour le mot-clé : Carbon allowance trading
  ✅ 300 résultats collectés pour 'Carbon allowance trading'.
  📁 Résultats bruts sauvegardés dans 'resultats_Carbon allowance trading
.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Carbon allowance trad
ing_extraits.json'.

🔍 Recherche pour le mot-clé : Carbon abatement
  ✅ 300 résultats collectés pour 'Carbon abatement'.
  📁 Résultats bruts sauvegardés dans 'resultats_Carbon abatement.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Carbon abatement_extr
aits.json'.

🔍 Recherche pour le mot-clé : Low carbon investment
  ✅ 300 résultats collectés pour 'Low carbon investment'.
  📁 Résultats bruts sauvegardés dans 'resultats_Low carbon investment.js
on'.
  ✅ Données extraites sauvegardées dans 'resultats_Low carbon investment
_extraits.json'.

🔍 Recherche pour le mot-clé : Green investment
  ✅ 300 résultats collectés pour 'Green investment'.
  📁 Résultats bruts sauvegardés dans 'resultats_Green investment.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Green investment_extr
aits.json'.

🔍 Recherche pour le mot-clé : Technological change
  ✅ 300 résultats collectés pour 'Technological change'.
  📁 Résultats bruts sauvegardés dans 'resultats_Technological change.jso
n'.
  ✅ Données extraites sauvegardées dans 'resultats_Technological change_
extraits.json'.

🔍 Recherche pour le mot-clé : Système d’échange de quotas d’émission de
```

```
l’UE
  ✅ 39 résultats collectés pour 'Système d’échange de quotas d’émission
de l’UE'.
  📁 Résultats bruts sauvegardés dans 'resultats_Système d’échange de quo
tas d’émission de l’UE.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Système d’échange de
quotas d’émission de l’UE_extraits.json'.

🔍 Recherche pour le mot-clé : Marché du carbone européen
  ✅ 218 résultats collectés pour 'Marché du carbone européen'.
  📁 Résultats bruts sauvegardés dans 'resultats_Marché du carbone europé
en.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Marché du carbone eur
opéen_extraits.json'.

🔍 Recherche pour le mot-clé : Tarification du carbone
  ✅ 154 résultats collectés pour 'Tarification du carbone'.
  📁 Résultats bruts sauvegardés dans 'resultats_Tarification du carbone.
json'.
  ✅ Données extraites sauvegardées dans 'resultats_Tarification du carbo
ne_extraits.json'.

🔍 Recherche pour le mot-clé : Réduction des émissions de CO2
  ✅ 300 résultats collectés pour 'Réduction des émissions de CO2'.
  📁 Résultats bruts sauvegardés dans 'resultats_Réduction des émissions
de CO2.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Réduction des émissio
ns de CO2_extraits.json'.

🔍 Recherche pour le mot-clé : Investissement bas carbone
  ✅ 179 résultats collectés pour 'Investissement bas carbone'.
  📁 Résultats bruts sauvegardés dans 'resultats_Investissement bas carbo
ne.json'.
  ✅ Données extraites sauvegardées dans 'resultats_Investissement bas ca
rbone_extraits.json'.

🔍 Recherche pour le mot-clé : Transition énergétique
  ✅ 300 résultats collectés pour 'Transition énergétique'.
  📁 Résultats bruts sauvegardés dans 'resultats_Transition énergétique.j
son'.
  ✅ Données extraites sauvegardées dans 'resultats_Transition énergétiqu
e_extraits.json'.

✅ Tous les résultats (bruts et extraits) ont été sauvegardés.
```

faire un filtre qui regroupe les articles selon des sujets à partir de ceux qu'on a extrait

We assess how to slow down climate change through technological advances.•We suggest how to incentivize countries to create innovation clusters.•We use a multi-

country model with emissions permit trade.•We construct a mechanism leading to innovation clusters.•We show how the EU-ETS can be refined with this mechanism. Innovation clusters combining public and private effort to develop breakthrough technologies promise greater technological advances to slow down climate change. We use a multi-country model with an emission trading system to examine whether and how international climate policy can incentivize countries to create such innovation clusters. We find that a minimal carbon price is needed to attract applied research firms, but countries may nevertheless fail to invest in complementary research infrastructure. We construct a mechanism that leads to innovation clusters when emissions targets are set before uncertainty surrounding technological developments is resolved. It is a combination of low permit endowments for the country with the lowest costs to build the needed infrastructure, compensation for this country by profits from permit trade, and maximal possible permit endowments for the remaining countries. We outline how the EU-ETS can be further refined according to this mechanism.

Gersbach and Riekhof (2021)

Permit markets, carbon prices and the creation of innovation clusters

combinaison en français et en anglais

In [ ]:
```
!pip install langdetect
```

```
Collecting langdetect
  Downloading langdetect-1.0.9.tar.gz (981 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 981.5/981.5 kB 7.6 MB/s eta
0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: six in /usr/local/lib/python3.11/dist-packa
ges (from langdetect) (1.17.0)
Building wheels for collected packages: langdetect
  Building wheel for langdetect (setup.py) ... done
  Created wheel for langdetect: filename=langdetect-1.0.9-py3-none-any.whl
size=993222 sha256=d972114c90c66c06874ec6759b6e65ea44c806c88ffc087da9a25a5
1dd02f20a
  Stored in directory: /root/.cache/pip/wheels/0a/f2/b2/e5ca405801e05eb7c8
ed5b3b4bcf1fcabcd6272c167640072e
Successfully built langdetect
Installing collected packages: langdetect
Successfully installed langdetect-1.0.9
```

In [ ]:
```
import json
import pandas as pd
import nltk
import re
from langdetect import detect
```

```
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics.pairwise import cosine_similarity

# Télécharger les stopwords pour les deux langues
nltk.download("stopwords")
nltk.download("punkt")
stop_words_fr = set(stopwords.words("french"))
stop_words_en = set(stopwords.words("english"))
combined_stop_words = list(stop_words_fr.union(stop_words_en))

# Charger les résultats extraits
with open("resultats_globaux_extraits.json", "r", encoding="utf-8") as f:
    articles_data = json.load(f)

# Convertir les articles en un DataFrame pandas
articles_list = []
for mot_cle, articles in articles_data.items():
    for article in articles:
        articles_list.append(article)
df_articles = pd.DataFrame(articles_list)

# Détection de la langue pour chaque article dans la colonne "description
def detecter_langue(text):
    try:
        return detect(text)
    except:
        return "unknown"
df_articles["langue"] = df_articles["description"].astype(str).apply(dete

# Afficher un aperçu des langues détectées avant traitement
print("Répartition des langues détectées :")
print(df_articles["langue"].value_counts())

# Nombre d'articles avant suppression des doublons
nombre_articles_avant = len(df_articles)
print("Nombre d'articles avant suppression des doublons :", nombre_articl

# Supprimer les doublons basés sur le titre et la description
df_articles = df_articles.drop_duplicates(subset=["title", "description"]

#nombre d'article après le nettoyage
nombre_articles_après = len(df_articles)
print("Nombre d'articles après suppression des doublons :", nombre_articl

# Textes de référence en anglais et en français
reference_text_en = """
We assess how to slow down climate change through technological advances.
We suggest how to incentivize countries to create innovation clusters.
We use a multi-country model with emissions permit trade.
We construct a mechanism leading to innovation clusters.
We show how the EU-ETS can be refined with this mechanism.
```

```
Innovation clusters combining public and private effort to develop breakt
We use a multi-country model with an emission trading system to examine w
We find that a minimal carbon price is needed to attract applied research
We construct a mechanism that leads to innovation clusters when emissions
It is a combination of low permit endowments for the country with the low
We outline how the EU-ETS can be further refined according to this mechan
"""

reference_text_fr = """
Nous évaluons comment ralentir le changement climatique grâce aux avancée
Nous suggérons comment inciter les pays à créer des pôles d'innovation.
Nous utilisons un modèle multi-pays avec des échanges de permis d'émissio
Nous construisons un mécanisme conduisant à la création de pôles d'innova
Nous montrons comment l'EU-ETS peut être affiné avec ce mécanisme.
Les pôles d'innovation combinant les efforts publics et privés pour dével
Nous utilisons un modèle multi-pays avec un système d'échange de quotas d
Nous constatons qu'un prix minimal du carbone est nécessaire pour attirer
Nous construisons un mécanisme qui conduit à des pôles d'innovation lorsq
Il s'agit d'une combinaison de faibles dotations en permis pour le pays d
Nous expliquons comment le système européen d'échange de quotas d'émissio
"""

# Combiner les textes de référence
reference_text = reference_text_en + "\n" + reference_text_fr

# Déterminer la langue de référence en utilisant le texte anglais
reference_lang = detect(reference_text)



# Vectorisation des descriptions avec TF-IDF en utilisant les stopwords c
vectorizer = TfidfVectorizer(stop_words=combined_stop_words)
tfidf_matrix = vectorizer.fit_transform([reference_text] + df_articles["d

# Calcul de la similarité cosinus entre le texte de référence et chaque d
similarites = cosine_similarity(tfidf_matrix[0:1], tfidf_matrix[1:]).flat

# Ajouter les scores de similarité au DataFrame
df_articles["similarite_reference"] = similarites

# Filtrer les articles avec un seuil de similarité > 0.1
df_articles_filtres = df_articles[df_articles["similarite_reference"] > 0
    by="similarite_reference", ascending=False
)

# Exporter les articles filtrés en CSV
df_articles_filtres.to_csv("articles_pertinents_sans_doublons.csv", index

# Afficher les 10 premiers résultats des articles filtrés
print("\nPremiers articles filtrés :")
print(df_articles_filtres.head(5))

# Afficher le nombre total d'articles filtrés
```

```
nombre_articles_filtrés = df_articles_filtres.shape[0]
print("\nNombre total d'articles filtrés :", nombre_articles_filtrés)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
Répartition des langues détectées :
langue
en         2269
fr          453
unknown     169
es            2
lv            1
pl            1
pt            1
Name: count, dtype: int64
Nombre d'articles avant suppression des doublons : 2896
Nombre d'articles après suppression des doublons : 2706

Premiers articles filtrés :
                                                  title  \
2290  Measuring regional innovation: A critical insp...
2221  Functions of innovation systems: A new approac...
1780  Why low-carbon technological innovation hardly...
1932  Executive green investment vision, stakeholder...
2459  Carbon constraint in the Mediterranean: Differ...

                                                authors  \
2290  [Hauser, Christoph ; Siller, Matthias ; Schatz...
2221  [Hekkert, M.P. ; Suurs, R.A.A. ; Negro, S.O. ;...
1780  [Li, Wenchao ; Xu, Jian ; Ostic, Dragana ; Yan...
1932  [Wan, Xiaole ; Wang, Yuxuan ; Qiu, Lulian ; Zh...
2459                 [Boisgibault, Louis ; Mozas, M A]

                                           description  \
2290  The disparities in regional innovation are oft...
2221  The central idea of this paper is that innovat...
1780  •China's energy efficiency is heterogeneous in...
1932  During the 14th Five Year Plan period, the gre...
2459  European Union's energy goals for 2020, inclus...

                                             subjects creation_date  \
2290  [Community Innovation Survey, Indexes, Indicat...          2018
2221  [Determinants, Dynamical systems, Dynamics, Em...          2007
1780  [Agglomeration, Energy efficiency, FCM, Hetero...          2021
1932  [business ecosystem, enterprise green innovati...          2022
2459  [Economics and Finance, Geography, Humanities ...          2012

                     source  \
2290  New York: Elsevier Inc
2221            Elsevier Inc
1780            Elsevier Ltd
1932     Frontiers Media S.A
```

```
2459

                                      full_text_link  \
2290
2221
1780
1932
2459  $$Uhttps://shs.hal.science/halshs-00942495$$EV...

                                           record_id rsrctype langue  \
2290              TN_cdi_proquest_journals_2084459774  article     en
2221            TN_cdi_proquest_miscellaneous_743050087  article     en
1780  TN_cdi_crossref_primary_10_1016_j_cie_2021_107566  article     en
1932  TN_cdi_doaj_primary_oai_doaj_org_article_fb954...  article     en
2459         TN_cdi_hal_primary_oai_HAL_halshs_00942495v1  article     fr

      similarite_reference
2290              0.205408
2221              0.172222
1780              0.159091
1932              0.159025
2459              0.146363

Nombre total d'articles filtrés : 46
```

In [ ]:
```
df_articles_filtres["langue"].value_counts()
```

Out[ ]:

langue	count
en	29
fr	17

dtype: int64

sur ce code, il est possible de choisir des mots clés pour les différents nommer les différents groupes

Regrouper les articles selon des méthodes quatitatives

In [ ]:
```
import re
import nltk
import pandas as pd
```

```
from sklearn.cluster import KMeans  # Vous pouvez aussi utiliser MiniBatc
from sklearn.feature_extraction.text import TfidfVectorizer
from nltk.corpus import stopwords

# Téléchargement des stopwords si nécessaire
nltk.download("stopwords")

# Définir les stopwords pour le français et l'anglais
stop_words_fr = set(stopwords.words("french"))
stop_words_en = set(stopwords.words("english"))
combined_stop_words = list(stop_words_fr.union(stop_words_en))

# Fonction de nettoyage du texte
def clean_text(text):
    text = str(text)  # S'assurer que c'est bien une chaîne de caractères
    text = text.lower()  # Mise en minuscule
    text = re.sub(r'\s+', ' ', text)  # Normaliser les espaces
    text = re.sub(r'[^\w\s]', '', text)  # Supprimer la ponctuation
    return text.strip()

# ------------------ Extraction des mots-clés de méthodes quantitatives -
def extract_quantitative_keywords(text):
    text_lower = clean_text(text)
    # Liste des mots-clés (ajoutez ou modifiez les termes ici)
    keywords = [
        # Termes en français
        "maximum de vraisemblance",
        "méthode des moments",
        "tobit",
        "logit",
        "probit",
        "heckman",
        "mco",
        "2sls",
        "iv",
        "effets fixes",
        "effets aléatoires",
        "données de panel",
        "régression poisson",
        "régression binomiale négative",
        "régression quantile",
        "gmm",
        "panel dynamique",
        "var",
        "autorégression",
        "séries temporelles",
        "arima",
        "arch",
        "garch",
        "test de racine unitaire",
        "cointégration",
        "modèle de correction d'erreur",
        "modèle de cox",
```

```
        "modèle multinomial",
        "modèle loglinéaire",
        "bootstrap",
        "inférence bayésienne",
        "mcmc",
        "test de durbinwatson",
        "test de granger",
        "économétrie spatiale",
        "modèles hiérarchiques",
        "modèles multiniveaux",
        "régression non linéaire",
        "erreurs standards robustes",
        "monte carlo",
        # Termes en anglais
        "maximum likelihood",
        "method of moments",
        "tobit",
        "logit",
        "probit",
        "heckman",
        "ols",
        "2sls",
        "instrumental variable",
        "fixed effects",
        "random effects",
        "panel data",
        "poisson regression",
        "negative binomial regression",
        "quantile regression",
        "gmm",
        "dynamic panel",
        "var",
        "vector autoregression",
        "autoregression",
        "time series",
        "arima",
        "arch",
        "garch",
        "unit root",
        "cointegration",
        "error correction model",
        "cox regression",
        "multinomial model",
        "log-linear",
        "bootstrap",
        "bayesian inference",
        "mcmc",
        "durbin-watson",
        "granger causality",
        "spatial econometrics",
        "hierarchical models",
        "multilevel models",
        "mixed effects",
```

```
        "nonlinear regression",
        "robust standard errors",
        "markov chain monte carlo",
        # Mots-clés supplémentaires
        "binomial",
        "log-binomial"
    ]

    found_keywords = []
    for keyword in keywords:
        if keyword in text_lower:
            found_keywords.append(keyword)

    if found_keywords:
        # Conserver l'ordre d'apparition et supprimer les doublons
        found_unique = sorted(set(found_keywords), key=lambda x: found_ke
        return ", ".join(found_unique)
    else:
        return None

# Création d'une nouvelle colonne qui extrait les mots-clés détectés dans
df_articles_filtres["quantitative_keywords_extracted"] = df_articles_filt

# ------------------ Attribution des phases selon l'année ---------------
def assign_phase(year):
    if pd.isnull(year):
        return "Autre"
    if 2005 <= year <= 2007:
        return "Phase 1"
    elif 2008 <= year <= 2012:
        return "Phase 2"
    elif 2013 <= year <= 2030:
        return "Phase 3"
    else:
        return "Autre"

if "creation_date" in df_articles_filtres.columns:
    df_articles_filtres["creation_date"] = pd.to_datetime(df_articles_fil
    df_articles_filtres["year"] = df_articles_filtres["creation_date"].dt
    df_articles_filtres["phase"] = df_articles_filtres["year"].apply(assi
else:
    print("La colonne 'creation_date' n'a pas été trouvée.")

# ------------------ Clustering par TF-IDF ------------------
descriptions_clean = df_articles_filtres["description"].apply(clean_text)
vectorizer_cat = TfidfVectorizer(stop_words=combined_stop_words)
tfidf_matrix_cat = vectorizer_cat.fit_transform(descriptions_clean)

n_clusters = 5
try:
    kmeans = KMeans(n_clusters=n_clusters, random_state=42)
    cluster_labels = kmeans.fit_predict(tfidf_matrix_cat)
except ValueError as e:
```

```
    print("Conversion en matrice dense pour KMeans en raison de :", e)
    tfidf_dense = tfidf_matrix_cat.toarray()
    cluster_labels = kmeans.fit_predict(tfidf_dense)

df_articles_filtres["categorie"] = cluster_labels

print("Répartition des catégories dans df_articles_filtres :")
print(df_articles_filtres["categorie"].value_counts())

order_centroids = kmeans.cluster_centers_.argsort()[:, ::-1]
terms = vectorizer_cat.get_feature_names_out()
for i in range(n_clusters):
    top_terms = [terms[ind] for ind in order_centroids[i, :10]]  # 10 ter
    print("Catégorie {}: {}".format(i, ", ".join(top_terms)))

# Affichage d'un aperçu intégrant l'extraction des mots-clés, l'attributi
print("\nExemple d'analyse des articles :")
print(df_articles_filtres[["description", "quantitative_keywords_extracte
```

```
Répartition des catégories dans df_articles_filtres :
categorie
2    11
3    11
1    10
4     7
0     7
Name: count, dtype: int64
Catégorie 0: pays, co2, quotas, directive, carbone, modèle, prix, éligible
s, entreprises, temps
Catégorie 1: green, innovation, environmental, investment, institutional,
development, enterprises, rd, corporate, sustainability
Catégorie 2: innovation, lowcarbon, energy, ets, policy, firms, carbon, de
regulation, technology, efficiency
Catégorie 3: pays, carbone, carbon, countries, plus, leurs, instruments, t
echnologies, france, deux
Catégorie 4: innovation, change, technological, systems, policy, system, i
mportant, functions, framework, design

Exemple d'analyse des articles :
                                            description  \
2290  The disparities in regional innovation are oft...
2221  The central idea of this paper is that innovat...
1780  •China's energy efficiency is heterogeneous in...
1932  During the 14th Five Year Plan period, the gre...
2459  European Union's energy goals for 2020, inclus...

     quantitative_keywords_extracted    phase
2290                  iv, var, arch  Phase 3
2221                           None  Phase 1
1780                             iv  Phase 3
1932                       iv, arch  Phase 3
2459                       iv, arch  Phase 2
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [ ]:
```
df_articles_filtres["quantitative_keywords_extracted"].value_counts(dropn
```

Out[ ]:

quantitative_keywords_extracted	count
iv, arch	13
iv	12
iv, var, arch	5
arch	4
None	3
iv, var	3
iv, arch, panel data, dynamic panel	1
iv, var, arch, panel data	1
iv, arch, panel data, binomial	1
iv, panel data, dynamic panel	1
iv, negative binomial regression, binomial	1
iv, arch, panel data	1

dtype: int64

au dessus indique spécifiquement le mot extrait

In [ ]:
```
import re
import nltk
import pandas as pd
from sklearn.cluster import KMeans  # Vous pouvez aussi utiliser MiniBatc
from sklearn.feature_extraction.text import TfidfVectorizer
from nltk.corpus import stopwords

# Téléchargement des stopwords si nécessaire
nltk.download("stopwords")

# Définir les stopwords pour le français et l'anglais
stop_words_fr = set(stopwords.words("french"))
stop_words_en = set(stopwords.words("english"))
combined_stop_words = list(stop_words_fr.union(stop_words_en))

# Fonction de nettoyage du texte
def clean_text(text):
    text = str(text)  # s'assurer que c'est bien une chaîne de caractères
    text = text.lower()  # mise en minuscule
    # Optionnel : normalisation des accents (si besoin, avec par exemple
```

```
    # from unidecode import unidecode
    # text = unidecode(text)
    text = re.sub(r'\s+', ' ', text)  # normaliser les espaces
    text = re.sub(r'[^\w\s]', '', text)  # supprimer la ponctuation
    return text.strip()

# ------------------ Détection de méthodes quantitatives ----------------
def detect_method_quantitative(text):
    text_lower = clean_text(text)
    keywords = [
        # Termes techniques en français
        "maximum de vraisemblance",
        "méthode des moments",
        "tobit",
        "logit",
        "probit",
        "heckman",
        "mco",
        "2sls",
        "iv",
        "effets fixes",
        "effets aléatoires",
        "données de panel",
        "régression poisson",
        "régression binomiale négative",
        "régression quantile",
        "gmm",
        "panel dynamique",
        "var",
        "autorégression",
        "séries temporelles",
        "arima",
        "arch",
        "garch",
        "test de racine unitaire",
        "cointégration",
        "modèle de correction d'erreur",
        "modèle de cox",
        "modèle multinomial",
        "modèle loglinéaire",
        "bootstrap",
        "inférence bayésienne",
        "mcmc",
        "test de durbinwatson",
        "test de granger",
        "économétrie spatiale",
        "modèles hiérarchiques",
        "modèles multiniveaux",
        "régression non linéaire",
        "erreurs standards robustes",
        "monte carlo",
        # Termes techniques en anglais
        "maximum likelihood",
```

```
        "method of moments",
        "tobit",
        "logit",
        "probit",
        "heckman",
        "ols",
        "2sls",
        "instrumental variable",
        "fixed effects",
        "random effects",
        "panel data",
        "poisson regression",
        "negative binomial regression",
        "quantile regression",
        "gmm",
        "dynamic panel",
        "var",
        "vector autoregression",
        "autoregression",
        "time series",
        "arima",
        "arch",
        "garch",
        "unit root",
        "cointegration",
        "error correction model",
        "cox regression",
        "multinomial model",
        "log-linear",
        "bootstrap",
        "bayesian inference",
        "mcmc",
        "durbin-watson",
        "granger causality",
        "spatial econometrics",
        "hierarchical models",
        "multilevel models",
        "mixed effects",
        "nonlinear regression",
        "robust standard errors",
        "markov chain monte carlo"
    ]
    for keyword in keywords:
        if keyword in text_lower:
            return True
    return False

# Création de la colonne indiquant la présence d'une méthode quantitative
df_articles_filtres["methode_quantitative"] = df_articles_filtres["descri

# ------------------ Attribution des groupes de méthodes ----------------
def detect_method_groups(text):
    groups_found = set()
```

```
    method_groups = {
        "Estimation": [
            "maximum de vraisemblance", "method of moments", "bootstrap",
            "inférence bayésienne", "bayesian inference", "mcmc", "markov
        ],
        "Limited Dependent Variable Models": [
            "tobit", "logit", "probit", "heckman"
        ],
        "Regression Models": [
            "mco", "ols", "2sls", "iv",
            "poisson regression", "régression poisson",
            "negative binomial regression", "régression binomiale négativ
            "quantile regression", "régression quantile", "nonlinear regr
        ],
        "Panel Data": [
            "effets fixes", "effets aléatoires", "données de panel", "pan
            "fixed effects", "random effects", "panel data", "dynamic pan
        ],
        "Time Series": [
            "var", "autorégression", "séries temporelles", "vector autore
            "autoregression", "time series", "arima", "arch", "garch"
        ],
        "Econometric Tests": [
            "test de racine unitaire", "cointégration", "cointegration",
            "modèle de correction d'erreur", "error correction model",
            "durbin-watson", "test de durbinwatson", "granger causality",
        ],
        "Advanced Models": [
            "économétrie spatiale", "spatial econometrics",
            "modèles hiérarchiques", "hierarchical models",
            "modèles multiniveaux", "multilevel models", "mixed effects"
        ]
    }
    text_lower = clean_text(text)
    for group, keywords in method_groups.items():
        for keyword in keywords:
            if keyword in text_lower:
                groups_found.add(group)
                break  # Dès qu'un mot-clé est trouvé dans un groupe, on
    if groups_found:
        return ", ".join(sorted(groups_found))
    else:
        return None

# Création de la colonne indiquant les groupes de méthodes détectés
df_articles_filtres["groupe_methodes"] = df_articles_filtres["description

# ------------------ Attribution des phases selon l'année ---------------
def assign_phase(year):
    if pd.isnull(year):
        return "Autre"
    if 2005 <= year <= 2007:
        return "Phase 1"
```

```
    elif 2008 <= year <= 2012:
        return "Phase 2"
    elif 2013 <= year <= 2030:
        return "Phase 3"
    else:
        return "Autre"

if "creation_date" in df_articles_filtres.columns:
    df_articles_filtres["creation_date"] = pd.to_datetime(df_articles_fil
    df_articles_filtres["year"] = df_articles_filtres["creation_date"].dt
    df_articles_filtres["phase"] = df_articles_filtres["year"].apply(assi
else:
    print("La colonne 'creation_date' n'a pas été trouvée.")

# ------------------ Clustering par TF-IDF ------------------
# On applique le nettoyage à la volée sur la colonne "description"
descriptions_clean = df_articles_filtres["description"].apply(clean_text)

vectorizer_cat = TfidfVectorizer(stop_words=combined_stop_words)
tfidf_matrix_cat = vectorizer_cat.fit_transform(descriptions_clean)

n_clusters = 5
try:
    # Tenter de lancer KMeans directement sur la matrice creuse
    kmeans = KMeans(n_clusters=n_clusters, random_state=42)
    cluster_labels = kmeans.fit_predict(tfidf_matrix_cat)
except ValueError as e:
    # Si une erreur survient (par exemple liée au format sparse), convert
    print("Conversion en matrice dense pour KMeans en raison de :", e)
    tfidf_dense = tfidf_matrix_cat.toarray()
    cluster_labels = kmeans.fit_predict(tfidf_dense)

df_articles_filtres["categorie"] = cluster_labels

print("Répartition des catégories dans df_articles_filtres :")
print(df_articles_filtres["categorie"].value_counts())

# Afficher les termes les plus représentatifs de chaque catégorie
order_centroids = kmeans.cluster_centers_.argsort()[:, ::-1]
terms = vectorizer_cat.get_feature_names_out()

for i in range(n_clusters):
    top_terms = [terms[ind] for ind in order_centroids[i, :10]]  # 10 ter
    print("Catégorie {}: {}".format(i, ", ".join(top_terms)))

# Affichage d'un aperçu intégrant la détection des méthodes quantitatives
print("\nExemple d'analyse des articles :")
print(df_articles_filtres[["description", "methode_quantitative", "groupe
```

```
Répartition des catégories dans df_articles_filtres :
categorie
2    11
3    11
1    10
4     7
0     7
Name: count, dtype: int64
Catégorie 0: pays, co2, quotas, directive, carbone, modèle, prix, éligible
s, entreprises, temps
Catégorie 1: green, innovation, environmental, investment, institutional,
development, enterprises, rd, corporate, sustainability
Catégorie 2: innovation, lowcarbon, energy, ets, policy, firms, carbon, de
regulation, technology, efficiency
Catégorie 3: pays, carbone, carbon, countries, plus, leurs, instruments, t
echnologies, france, deux
Catégorie 4: innovation, change, technological, systems, policy, system, i
mportant, functions, framework, design

Exemple d'analyse des articles :
                                              description  methode_quantitat
ive  \
2290  The disparities in regional innovation are oft...                  T
rue
2221  The central idea of this paper is that innovat...                 Fa
lse
1780  •China's energy efficiency is heterogeneous in...                  T
rue
1932  During the 14th Five Year Plan period, the gre...                  T
rue
2459  European Union's energy goals for 2020, inclus...                  T
rue

                     groupe_methodes    phase
2290  Regression Models, Time Series  Phase 3
2221                            None  Phase 1
1780              Regression Models  Phase 3
1932  Regression Models, Time Series  Phase 3
2459  Regression Models, Time Series  Phase 2
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

In [ ]:
```
df_articles_filtres["source"].value_counts(dropna=False)
```

Out[ ]:

source	count
Elsevier Ltd	6
	4
Elsevier B.V	4
Frontiers Media S.A	3

Dalloz	3
New York: Elsevier Inc	2
Kidlington: Elsevier Ltd	2
Association d'Economie Financière	2
San Francisco: Public Library of Science	1
PERSÉE : Université de Lyon, CNRS & ENS de Lyon	1
New York: Hindawi	1
Guelph: University of Toronto Press	1
Pretoria: African Online Scientific Information Systems (Pty) Ltd t/a AOSIS	1
Kidlington: Elsevier B.V	1
De Boeck Supérieur	1
London: Routledge	1
Cambridge: Routledge	1
Piscataway: IEEE	1
London: Nature Publishing Group UK	1
Elsevier Inc	1
La Documentation française	1
Association d'économie financière	1
Oxford: Elsevier Ltd	1
Minefi - Direction de la prévision	1
Paris: Dalloz	1
Routledge	1
Ministère de l'Économie	1
American Economic Association	1

dtype: int64

In [ ]:
```
df_articles_filtres["methode_quantitative"].value_counts()
```

Out[ ]:

methode_quantitative	count
True	43
False	3

dtype: int64

trier les articles avec rééditions , éviter les doublons

In [ ]:
```
import pandas as pd

# 1. Extraire les 6 premiers mots du titre, en minuscules
df_articles_filtres["first_6_words"] = (
    df_articles_filtres["title"]
    .astype(str)                # conversion en chaîne de caractères si
    .str.lower()                # mise en minuscule
    .str.split()                # découpage en mots
    .apply(lambda x: " ".join(x[:6]))  # on prend les 6 premiers mots
)

# 2. Détecter les doublons sur la base de ces 6 premiers mots
# keep=False signale que toutes les lignes faisant partie d'un doublon se
df_articles_filtres["doublon"] = df_articles_filtres.duplicated(subset="f

# 3. (Optionnel) Supprimer la colonne temporaire si vous ne souhaitez pas
# df_articles_filtres.drop(columns=["first_6_words"], inplace=True)

# Vérification rapide
print(df_articles_filtres[["title", "first_6_words", "doublon"]].head(10)
```

```
                                                  title  \
2290  Measuring regional innovation: A critical insp...
2221  Functions of innovation systems: A new approac...
1780  Why low-carbon technological innovation hardly...
1932  Executive green investment vision, stakeholder...
2459  Carbon constraint in the Mediterranean: Differ...
2429  Premières simulations de la directive européen...
1713  Perceived uncertainty, low-carbon policy, and ...
2545  La Stern Review : le parti pris de l'action fa...
2430  Premières simulations de la directive européen...
2163  Sustainability as a driver of green innovation...

                                        first_6_words  doublon
2290  measuring regional innovation: a critical insp...    False
2221            functions of innovation systems: a new    False
1780  why low-carbon technological innovation hardly...    False
1932  executive green investment vision, stakeholder...    False
2459  carbon constraint in the mediterranean: differ...    False
2429   premières simulations de la directive européenne     True
1713  perceived uncertainty, low-carbon policy, and ...    False
2545                         la stern review : le parti    False
2430   premières simulations de la directive européenne     True
2163              sustainability as a driver of green    False
```

In [ ]:
```
df_articles_filtres["doublon"].value_counts()
```

Out[ ]:

doublon	count
False	39
True	7

dtype: int64

In [ ]:
```
df_articles_filtres_nodoublons = df_articles_filtres[df_articles_filtres[
```

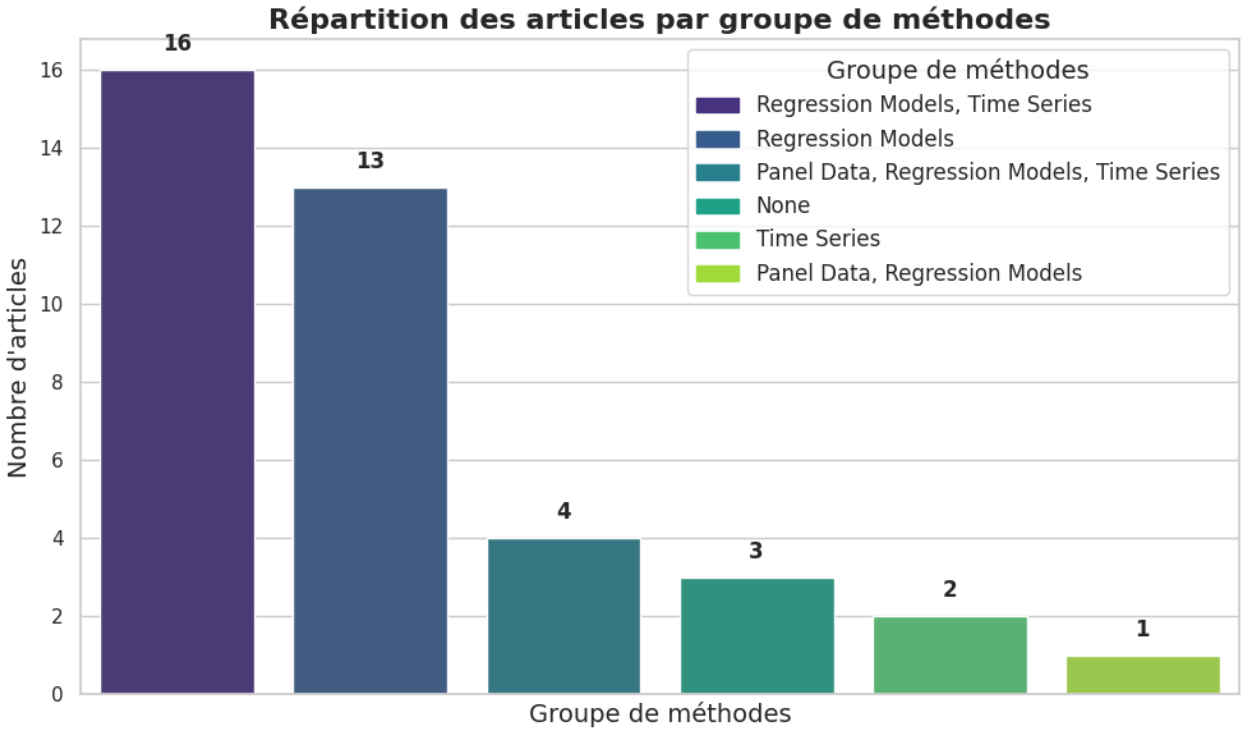
In [ ]:
```
print(df_articles_filtres_nodoublons["groupe_methodes"].value_counts(drop
```

```
groupe_methodes
Regression Models, Time Series                16
Regression Models                             13
Panel Data, Regression Models, Time Series     4
None                                           3
Time Series                                    2
Panel Data, Regression Models                  1
Name: count, dtype: int64
```

In [ ]:
```
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.patches as mpatches
```

```
# Calculer le nombre d'articles par groupe (en incluant les valeurs manqu
groupe_counts = df_articles_filtres_nodoublons["groupe_methodes"].value_c

# Configurer le style Seaborn pour un rendu propre
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))

# Créer l'histogramme avec la palette "viridis"
bar_plot = sns.barplot(x=groupe_counts.index.astype(str), y=groupe_counts

# Ajouter le titre et les labels aux axes
plt.title("Répartition des articles par groupe de méthodes", fontsize=16,
plt.xlabel("Groupe de méthodes", fontsize=14)
plt.ylabel("Nombre d'articles", fontsize=14)

# Afficher le nombre d'articles au-dessus de chaque barre
for i, count in enumerate(groupe_counts.values):
    plt.text(i, count + 0.5, str(count), ha='center', fontsize=12, fontwe

# Masquer les étiquettes de l'axe des x pour ne conserver que la légende
plt.xticks([])

# Créer une légende personnalisée à partir des couleurs utilisées
colors = sns.color_palette("viridis", len(groupe_counts))
patches = [mpatches.Patch(color=colors[i], label=str(groupe_counts.index[
plt.legend(handles=patches, title="Groupe de méthodes", loc="upper right"

plt.tight_layout()
plt.show()
```

```
<ipython-input-30-4c9fc3fa4d98>:13: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be remove
d in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for
the same effect.

  bar_plot = sns.barplot(x=groupe_counts.index.astype(str), y=groupe_count
s.values, palette="viridis")
```



In [ ]:
```
df_articles_filtres_nodoublons["phase"].value_counts()
```

Out[ ]:

phase	count
Phase 3	27
Phase 2	9
Phase 1	3

dtype: int64

Ce truc permet de répondre que Oui, selon notre code. Ca a un impact positif.

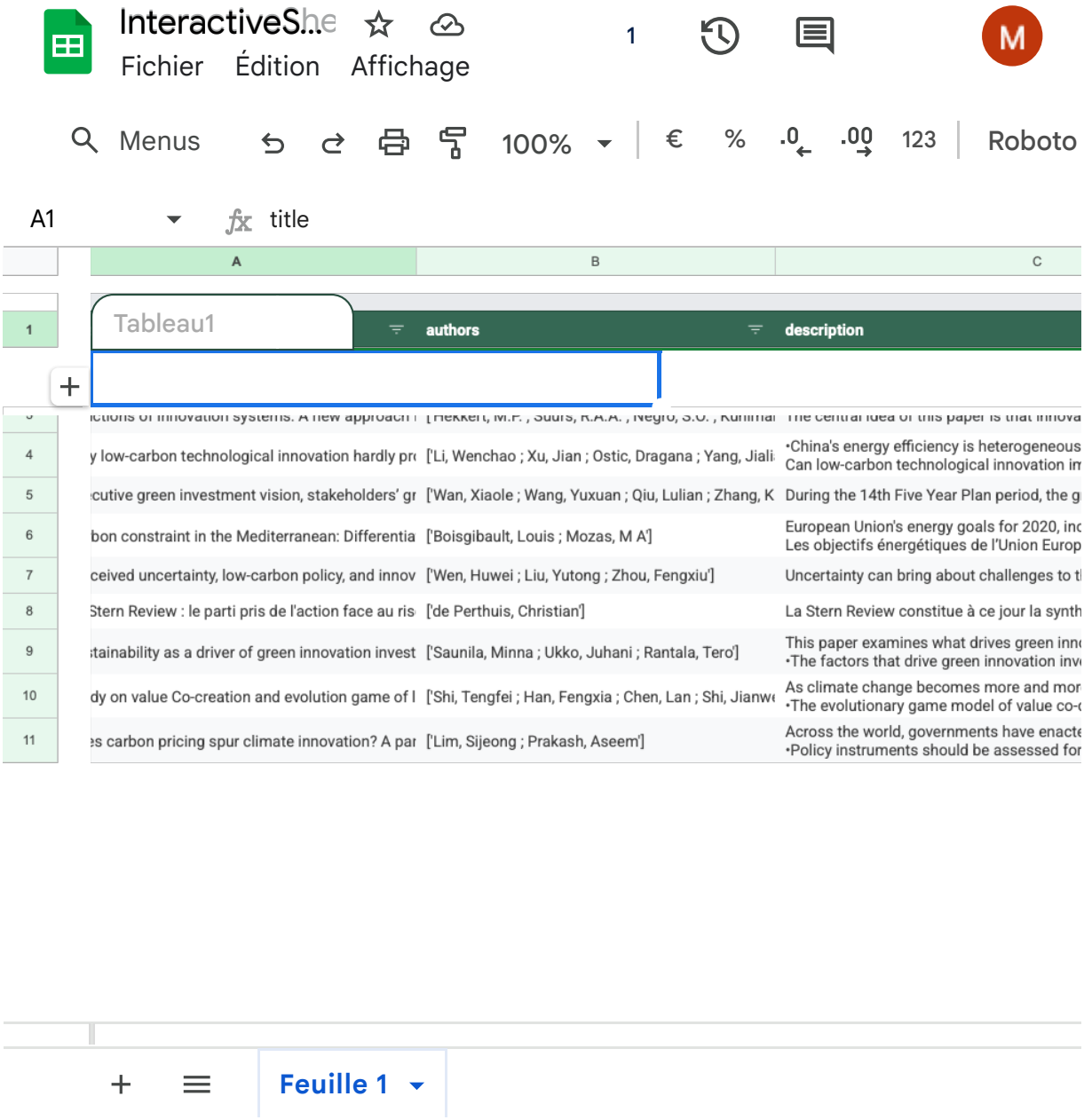
très complique de trier automatiquement les articles par phases puisque les dates peuvent apparaître plusieurs fois. pas très pertinent, mieux de les trier par creation date.

In [ ]:
```
```

google sheet

In [ ]:
```
from google.colab import sheets
sheet = sheets.InteractiveSheet(df=df_articles_filtres_nodoublons)
```

https://docs.google.com/spreadsheets/d/1Vap9V5xaQ3fOjJKxBph7Wvp3N62dx6hE_mJRKSpiPEY#gid=0

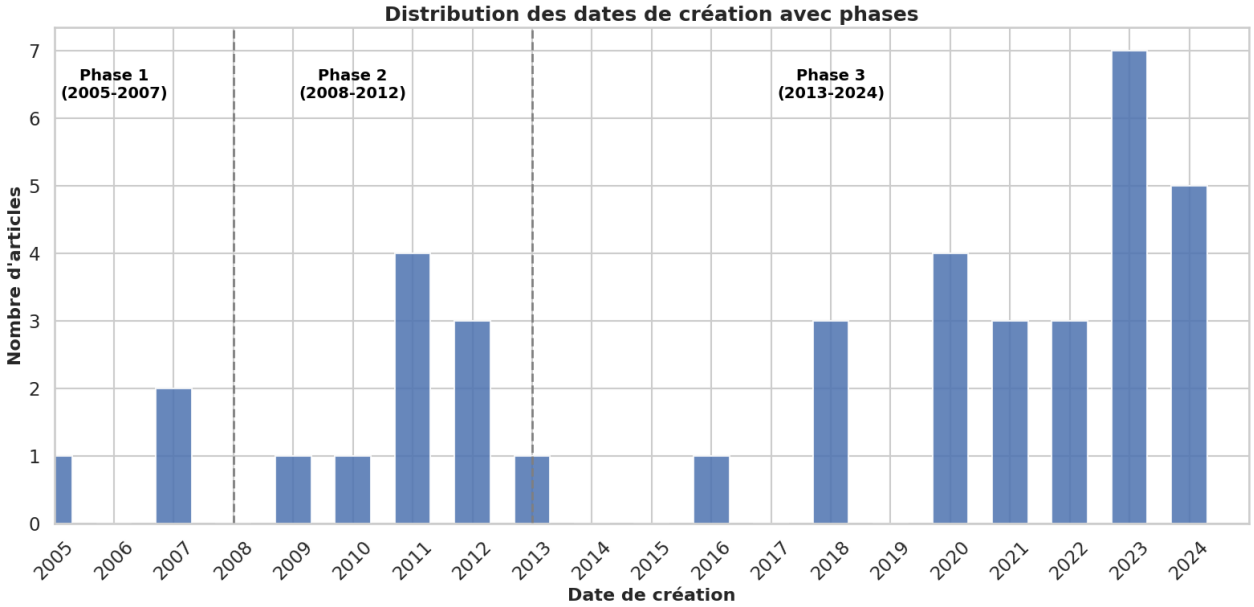


Data visualisation

In [ ]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import matplotlib.dates as mdates

# Conversion de la colonne en datetime
df_articles_filtres_nodoublons['creation_date'] = pd.to_datetime(df_artic

# Agrégation du nombre d'articles par année
df_yearly_counts = df_articles_filtres_nodoublons.groupby(pd.Grouper(key=
```

```
# Création d'une figure avec Seaborn
sns.set(style="whitegrid", context="talk")
fig, ax = plt.subplots(figsize=(16, 8))

# Tracé des barres avec un alignement au centre des années
years = df_yearly_counts.index.year
ax.bar(years, df_yearly_counts, width=0.6, color='#4c72b0', edgecolor='wh

# Définition des limites et labels de l'axe X
ax.set_xlim(2005, 2025)
ax.set_xticks(range(2005, 2025))
ax.set_xticklabels(range(2005, 2025), rotation=45)

# Ajout des lignes verticales pointillées pour séparer les phases
ax.axvline(2008, color='grey', linestyle='--', linewidth=2)
ax.axvline(2013, color='grey', linestyle='--', linewidth=2)

# Positionnement des annotations pour chaque phase
ymax = df_yearly_counts.max()
ax.text(2006, ymax*0.9, "Phase 1\n(2005-2007)", ha='center', fontsize=14,
ax.text(2010, ymax*0.9, "Phase 2\n(2008-2012)", ha='center', fontsize=14,
ax.text(2018, ymax*0.9, "Phase 3\n(2013-2024)", ha='center', fontsize=14,

# Personnalisation des axes et du titre
ax.set_xlabel("Date de création", fontsize=16, weight='bold')
ax.set_ylabel("Nombre d'articles", fontsize=16, weight='bold')
ax.set_title("Distribution des dates de création avec phases", fontsize=1

plt.tight_layout()
plt.show()
```

```
<ipython-input-32-86761f59e932>:7: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df_articles_filtres_nodoublons['creation_date'] = pd.to_datetime(df_arti
cles_filtres_nodoublons['creation_date'])
<ipython-input-32-86761f59e932>:10: FutureWarning: 'Y' is deprecated and w
ill be removed in a future version, please use 'YE' instead.
  df_yearly_counts = df_articles_filtres_nodoublons.groupby(pd.Grouper(key
='creation_date', freq='Y')).size()
```



Il ne faut pas comparer avec les articles de mandaroux le graphique faisant référence aux phases, il s'agit uniquement d'une représentation graphique de la distribution des articles selon leur date de création.

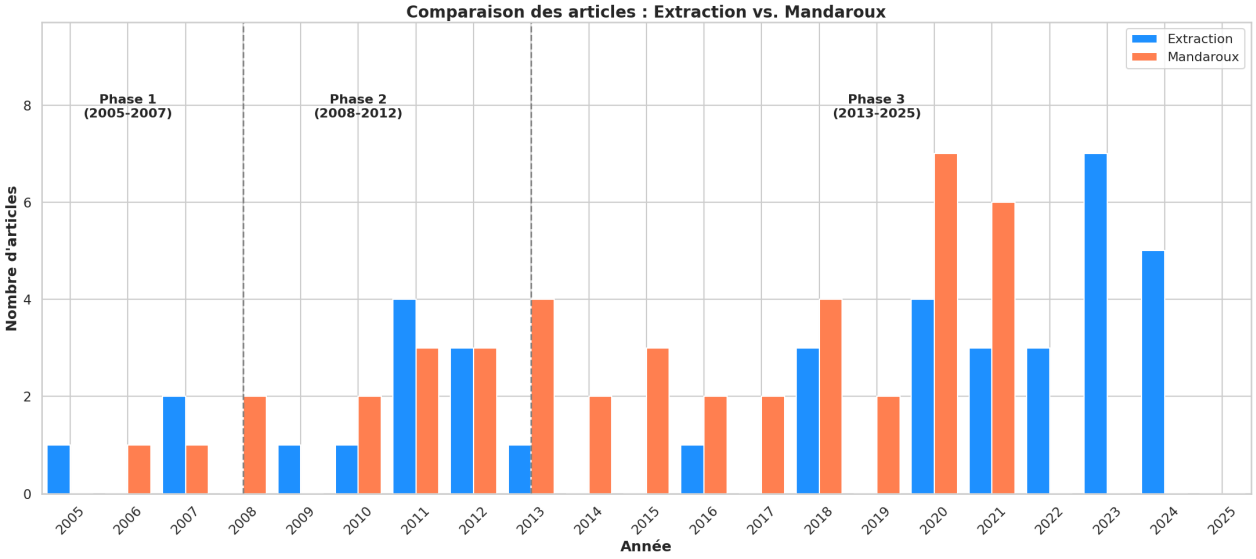
superposer avec les articles de mandaroux

In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns

# ------------------------------------------------------------------------
# DONNÉES 1 : Issues de l'extraction
# ------------------------------------------------------------------------
# On suppose que df_articles_filtres_nodoublons contient une colonne 'cre
df_articles_filtres_nodoublons['creation_date'] = pd.to_datetime(df_artic
df_yearly_counts = df_articles_filtres_nodoublons.groupby(pd.Grouper(key=
years_extraction = df_yearly_counts.index.year
extraction_values = df_yearly_counts.values

# On crée un dictionnaire pour faciliter l'accès aux valeurs par année
extraction_dict = {year: count for year, count in zip(years_extraction, e

# ------------------------------------------------------------------------
# DONNÉES 2 : D'après Mandaroux
# ------------------------------------------------------------------------
# Exemple de données (à remplacer par vos vraies valeurs)
positive_link = np.array([0, 1, 1, 2, 0, 2, 3, 3, 4, 2, 3, 2, 2, 4, 2, 7,
no_substantial_link = np.array([7, 8, 7, 9, 9, 6, 5, 4, 2, 1, 1, 1, 0, 1,
mandaroux_values_array = positive_link
years_mandaroux = np.arange(2005, 2005 + len(mandaroux_values_array))
mandaroux_dict = {year: val for year, val in zip(years_mandaroux, mandaro
```

```
# ------------------------------------------------------------------------
# COMBINAISON DES DONNÉES SUR UN SEUL GRAPHIQUE
# ------------------------------------------------------------------------
# On définit une plage d'années commune (2005 à 2025)
all_years = np.arange(2005, 2026)
extraction_combined = [extraction_dict.get(year, 0) for year in all_years
mandaroux_combined = [mandaroux_dict.get(year, 0) for year in all_years]

# ------------------------------------------------------------------------
# CONFIGURATION ET AFFICHAGE DU GRAPHIQUE
# ------------------------------------------------------------------------
sns.set_style("whitegrid")
sns.set_context("talk")

fig, ax = plt.subplots(figsize=(22, 10))

# Paramétrage de la largeur et des positions des barres
width = 0.4
x_extraction = all_years - width/2
x_mandaroux  = all_years + width/2

# Barres pour les articles issus de l'extraction
ax.bar(x_extraction, extraction_combined, width=width, color='dodgerblue'
       edgecolor='white', label="Extraction")

# Barres pour les articles d'après Mandaroux
ax.bar(x_mandaroux, mandaroux_combined, width=width, color='coral',
       edgecolor='white', label="Mandaroux")

# Tracer des lignes verticales pour délimiter les phases
ax.axvline(2008, color='grey', linestyle='--', linewidth=2)
ax.axvline(2013, color='grey', linestyle='--', linewidth=2)

# Calculer une hauteur maximale pour positionner correctement les annotat
ymax = max(max(extraction_combined), max(mandaroux_combined)) * 1.1

# Annoter les phases aux positions moyennes :
#   Phase 1 : 2005–2007  (moyenne = 2006)
#   Phase 2 : 2008–2012  (moyenne = 2010)
#   Phase 3 : 2013–2025  (moyenne = 2019)
ax.text(2006, ymax, "Phase 1\n(2005–2007)", ha='center', va='bottom', fon
ax.text(2010, ymax, "Phase 2\n(2008–2012)", ha='center', va='bottom', fon
ax.text(2019, ymax, "Phase 3\n(2013–2025)", ha='center', va='bottom', fon

# Configuration des axes, titre et légende
ax.set_xlabel("Année", fontsize=18, weight='bold')
ax.set_ylabel("Nombre d'articles", fontsize=18, weight='bold')
ax.set_title("Comparaison des articles : Extraction vs. Mandaroux", fonts
ax.set_xticks(all_years)
ax.set_xticklabels(all_years, rotation=45)
ax.set_xlim(2004.5, 2025.5)
ax.set_ylim(0, ymax + 2)
ax.legend(fontsize=16)
```

```
plt.tight_layout()
plt.show()
```

```
<ipython-input-33-be1a46d7192d>:10: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-doc
s/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df_articles_filtres_nodoublons['creation_date'] = pd.to_datetime(df_arti
cles_filtres_nodoublons['creation_date'])
<ipython-input-33-be1a46d7192d>:11: FutureWarning: 'Y' is deprecated and w
ill be removed in a future version, please use 'YE' instead.
  df_yearly_counts = df_articles_filtres_nodoublons.groupby(pd.Grouper(key
='creation_date', freq='Y')).size()
```



Matrice de confusion (lsite des articles de Mandaroux à parie des tables)

In [ ]:
```
!pip install fuzzywuzzy
```

```
Requirement already satisfied: fuzzywuzzy in /usr/local/lib/python3.11/dis
t-packages (0.18.0)
```

In [ ]:
```
# Récupérer les titres des articles marqués comme doublons dans df_articl
titles_doublon_true = df_articles_filtres.loc[df_articles_filtres["doublo

# Filtrer df_articles pour retirer les titres présents dans titles_doublo
df_articles_filtered = df_articles[~df_articles["title"].isin(titles_doub

print("Nombre d'articles après suppression des doublons :", len(df_articl
```

```
Nombre d'articles après suppression des doublons : 2699
```

In [ ]:
```
import re
```

```
import pandas as pd
from sklearn.metrics import confusion_matrix
from fuzzywuzzy import fuzz

# Exemple de liste de références
references = [
    "[George, Gerard ; Howard-Grenville, Jennifer ; Joshi, Aparna ; Tihan
    "[Gersbach, H. ; Riekhof, M.C. – « Permit markets, carbon prices and
    "[Groenenberg, H. ; Coninck, H. de – « Effective EU and member state
    "[Grubb, M. ; Drummond, P. ; Poncia, A. ; McDowall, W. ; Popp, D. ; S
    "[Gulbrandsen, L.H. ; Stenqvist, C. – « The limited effect of EU emis
    "[Hoffmann, A.J. – « Institutional evolution and change: environmenta
    "[He, R. ; Le Luo, Shamsuddin, A. ; Tang, Q. – « Corporate carbon acc
    "[Hobbie, H. ; Schmidt, M. ; Möst, D. – « Windfall profits in the pow
    "[Kline, S.J. ; Rosenberg, N. – « An overview of innovation. In: The
    "[Koch, N. ; Basse Mama, H. – « Does the EU Emissions Trading System
    "[Li, Z. ; Pan, Y. ; Yang, W. ; Ma, J. ; Zhou, M. – « Effects of gove
    "[Lilliestam, J. ; Patt, A. ; Bersalli, G. – « The Effect of Carbon P
    "[Lin, W.M. ; Chen, J.L. ; Zheng, Y. ; Dai, Y.W. – « Effects of the E
    "[Lise, W. ; Sijm, J. ; Hobbs, B.F. – « The impact of the EU ETS on p
    "[Löfgren, Å. ; Wråke, M. ; Hagberg, T. ; Roth, S. – « Why the EU ETS
    "[Lundgren, T. ; Marklund, P.-O. ; Samakovlis, E. ; Zhou, W. – « Carb
    "[Markard, J. ; Rosenbloom, D. – « Political conflict and climate pol
    "[Martin, R. ; Muûls, M. ; Preux, L.B. de ; Wagner, U.J. – « On the e
    "[Cainelli, G. ; Mazzanti, M. ; Montresor, S. – « Environmental innov
    "[Calel, R. – « Adopt or innovate: understanding technological respon
    "[Calel, R. ; Dechezlepretre, A. – « Environmental policy and directe
    "[Cecere, G. ; Corrocher, N. ; Gossart, C. ; Ozman, M. – « Technologi
    "[Chiappinelli, O. ; Neuhoff, K. – « Time-consistent Carbon Pricing:
    "[Clò, S. – « Grandfathering, auctioning and carbon leakage: assessin
    "[Consoli, D. ; Marin, G. ; Marzucchi, A. ; Vona, F. – « Do green job
    "[Corradini, M. ; Costantini, V. ; Markandya, A. ; Paglialunga, E. ;
    "[Davoudi, S.M.M. ; Fartash, K. ; Zakirova, V.G. ; Belyalova, A.M. ;
    "[Downs Jr., G.W. ; Mohr, L.B. – « Conceptual issues in the study of
    "[Edenhofer, O. ; Flachsland, C. ; Wolff, C. ; Schmid, L.K. ; Leippra
    "[Edquist, C. – « Systems of innovation approaches – their emergence
    "[European Central Bank, 2022 – « The role of speculation during the
    "[European Commission, 2022a – « Allocation to industrial installatio
    "[European Commission, 2022b – « EU emissions trading system (EU ETS)
    "[European Commission, 2022c – « Innovation fund: policy development
    "[European Commission, 2022d – « Market stability reserve » :contentR
    "[European Commission, 2022e – « Modernisation fund » :contentReferen
    "[European Commission, 2022f – « Sustainability-related disclosure in
    "[Europex, 2021 – « Carbon Contracts for Difference (CCfDs) and their
    "[Martin, R. ; Muûls, M. ; Wagner, U.J. – « The impact of the Europea
    "[Matschoss, P. ; Welsch, H. – « International emissions trading and
    "[Mazzanti, M. ; Rizzo, U. – « Diversely moving towards a green econo
    "[McAndrew, R. ; Mulcahy, R. ; Gordon, R. ; Russell-Bennett, R. – « H
    "[Moher, D. ; Liberati, A. ; Tetzlaff, J. ; Altman, D.G. – « Preferre
    "[Mowery, D. ; Rosenberg, N. – « The influence of market demand upon
    "[Venmans, F.M.J., 2016 – « The effect of allocation above emissions
    "[Von Stechow, C. ; Watson, J. ; Praetorius, B., 2011 – « Policy ince
    "[Warner, K.E., 1974 – « The need for some innovative concepts of inn
```

```
    "[Winkler, D., 2022 – « Pollution for sale: lobbying, allowance alloc
    "[Xia, L. ; Gao, S. ; Wei, J. ; Ding, Q., 2022 – « Government subsidy
    "[Yang, Defeng ; Wang, Aric Xu ; Zhou, Kevin Zheng ; Jiang, Wei, 2019
    "[Stornelli, A. ; Ozcan, S. ; Simms, C., 2021 – « Advanced manufactur
    "[Teixido, J. ; Verde, S.F. ; Nicolli, F., 2019 – « The impact of the
    "[Tomás, R. ; Ramôa Ribeiro, F. ; Santos, V. ; Gomes, J. ; Bordado, J
    "[Tranfield, D. ; Denyer, D. ; Smart, P., 2003 – « Towards a methodol
    "[Utterback, J.M. ; Abernathy, W.J., 1975 – « A dynamic model of proc
]

# Affichage de la liste des références
for ref in references:
    print(ref)
len(references)

# Fonction pour extraire le titre d'une référence (entre « et »)
def extract_title(ref_string):
    match = re.search(r'«\s*(.*?)\s*»', ref_string)
    return match.group(1) if match else ""

# Extraction des titres des référencess
reference_titles = [extract_title(ref) for ref in references]

print("Titres extraits des références :")
for title in reference_titles:
    print(title)

# Fonction pour calculer la similarité maximale entre le titre d'un artic
def max_similarity(article_title, reference_titles):
    scores = [fuzz.ratio(article_title.lower(), ref_title.lower()) for re
    return max(scores) if scores else 0

# Pour chaque article, calculer le score de similarité maximal
df_articles_filtered['similarity_score'] = df_articles_filtered['title'].

# Définir la pertinence réelle (vérité terrain) :
# Ici, un article est considéré comme pertinent si son score de similarit
df_articles_filtered['is_relevant'] = df_articles_filtered['similarity_sc

# Définir la prédiction basée sur le filtrage :
# On considère que les 'nombre_articles_filtrés' articles ayant le score
# Nous trions d'abord le DataFrame par 'similarity_score' de façon décroi
df_articles_filtered = df_articles_filtered.sort_values('similarity_score

# Créer une colonne 'predicted' : True pour les top N articles, False pou
N = nombre_articles_filtrés  # nombre_articles_filtrés est un entier défi
df_articles_filtered['predicted'] = False
df_articles_filtered.loc[:N-1, 'predicted'] = True

# Calcul de la matrice de confusion à partir de 'is_relevant' (vérité ter
y_true = df_articles_filtered['is_relevant']
y_pred = df_articles_filtered['predicted']
cm = confusion_matrix(y_true, y_pred)
```

```
# Optionnel : afficher la matrice sous forme de DataFrame pour plus de li
cm_df = pd.DataFrame(cm, index=['Non Pertinent', 'Pertinent'], columns=['
print("\nMatrice de confusion (formatée) :")
print(cm_df)
```

[George, Gerard ; Howard-Grenville, Jennifer ; Joshi, Aparna ; Tihanyi, La szlo — « Understanding and tackling societal grand challenges through mana gement research » :contentReference[oaicite:0]{index=0}]
[Gersbach, H. ; Riekhof, M.C. — « Permit markets, carbon prices and the cr eation of innovation clusters » :contentReference[oaicite:1]{index=1}]
[Groenenberg, H. ; Coninck, H. de — « Effective EU and member state polici es for stimulating CCS » :contentReference[oaicite:2]{index=2}]
[Grubb, M. ; Drummond, P. ; Poncia, A. ; McDowall, W. ; Popp, D. ; Samadi, S. ; Penasco, C. ; Gillingham, K.T. ; Smulders, S. ; Glachant, M. ; Hassal l, G. ; Mizuno, E. ; Rubin, E.S. ; Dechezleprêtre, A. ; Pavan, G. — « Indu ced innovation in energy technologies and systems: a review of evidence an d potential implications for CO_2 mitigation » :contentReference[oaicite:3] {index=3}]
[Gulbrandsen, L.H. ; Stenqvist, C. — « The limited effect of EU emissions trading on corporate climate strategies: comparison of a Swedish and a Nor wegian pulp and paper company » :contentReference[oaicite:4]{index=4}]
[Hoffmann, A.J. — « Institutional evolution and change: environmentalism a nd the US chemical industry » :contentReference[oaicite:5]{index=5}]
[He, R. ; Le Luo, Shamsuddin, A. ; Tang, Q. — « Corporate carbon accountin g: a literature review of carbon accounting research from the Kyoto Protoc ol to the Paris Agreement » :contentReference[oaicite:6]{index=6}]
[Hobbie, H. ; Schmidt, M. ; Möst, D. — « Windfall profits in the power sec tor during phase III of the EU ETS: interplay and effects of renewables an d carbon prices » :contentReference[oaicite:7]{index=7}]
[Kline, S.J. ; Rosenberg, N. — « An overview of innovation. In: The Positi ve Sum Strategy: Harnessing Technology for Economic Growth » :contentRefer ence[oaicite:8]{index=8}]
[Koch, N. ; Basse Mama, H. — « Does the EU Emissions Trading System induce investment leakage? Evidence from German multinational firms » :contentRef erence[oaicite:9]{index=9}]
[Li, Z. ; Pan, Y. ; Yang, W. ; Ma, J. ; Zhou, M. — « Effects of government subsidies on green technology investment and green marketing coordination of supply chain under the cap-and-trade mechanism » :contentReference[oaic ite:10]{index=10}]
[Lilliestam, J. ; Patt, A. ; Bersalli, G. — « The Effect of Carbon Pricing on Technological Change for Full Energy Decarbonization: A Review of Empir ical Ex-Post Evidence » :contentReference[oaicite:11]{index=11}]
[Lin, W.M. ; Chen, J.L. ; Zheng, Y. ; Dai, Y.W. — « Effects of the EU Emis sion Trading Scheme on the international competitiveness of pulp-and-paper industry » :contentReference[oaicite:12]{index=12}]
[Lise, W. ; Sijm, J. ; Hobbs, B.F. — « The impact of the EU ETS on prices, profits and emissions in the power sector: simulation results with the COM PETES EU20 model » :contentReference[oaicite:13]{index=13}]
[Löfgren, Å. ; Wråke, M. ; Hagberg, T. ; Roth, S. — « Why the EU ETS needs reforming: an empirical analysis of the impact on company investments » :c ontentReference[oaicite:14]{index=14}]
[Lundgren, T. ; Marklund, P.-O. ; Samakovlis, E. ; Zhou, W. — « Carbon pri

ces and incentives for technological development » :contentReference[oaicite:15]{index=15}]
[Markard, J. ; Rosenbloom, D. – « Political conflict and climate policy: the European emissions trading system as a Trojan Horse for the low-carbon transition? » :contentReference[oaicite:16]{index=16}]
[Martin, R. ; Muûls, M. ; Preux, L.B. de ; Wagner, U.J. – « On the empirical content of carbon leakage criteria in the EU Emissions Trading Scheme » :contentReference[oaicite:17]{index=17}]
[Cainelli, G. ; Mazzanti, M. ; Montresor, S. – « Environmental innovations, local networks and internationalization » :contentReference[oaicite:18]{index=18}]
[Calel, R. – « Adopt or innovate: understanding technological responses to cap-and-trade » :contentReference[oaicite:19]{index=19}]
[Calel, R. ; Dechezlepretre, A. – « Environmental policy and directed technological change: evidence from the European carbon market » :contentReference[oaicite:20]{index=20}]
[Cecere, G. ; Corrocher, N. ; Gossart, C. ; Ozman, M. – « Technological pervasiveness and variety of innovators in Green ICT: a patent-based analysis » :contentReference[oaicite:21]{index=21}]
[Chiappinelli, O. ; Neuhoff, K. – « Time-consistent Carbon Pricing: the Role of Carbon Contracts for Differences » :contentReference[oaicite:22]{index=22}]
[Clò, S. – « Grandfathering, auctioning and carbon leakage: assessing the inconsistencies of the new ETS directive » :contentReference[oaicite:23]{index=23}]
[Consoli, D. ; Marin, G. ; Marzucchi, A. ; Vona, F. – « Do green jobs differ from non-green jobs in terms of skills and human capital? » :contentReference[oaicite:24]{index=24}]
[Corradini, M. ; Costantini, V. ; Markandya, A. ; Paglialunga, E. ; Sforna, G. – « A dynamic assessment of instrument interaction and timing alternatives in the EU low-carbon policy mix design » :contentReference[oaicite:25]{index=25}]
[Davoudi, S.M.M. ; Fartash, K. ; Zakirova, V.G. ; Belyalova, A.M. ; Kurbanov, R.A. ; Boiarchuk, A.V. ; Sizova, Z.M. – « Testing the mediating role of open innovation on the relationship between intellectual property rights and organizational… » :contentReference[oaicite:26]{index=26}]
[Downs Jr., G.W. ; Mohr, L.B. – « Conceptual issues in the study of innovation » :contentReference[oaicite:27]{index=27}]
[Edenhofer, O. ; Flachsland, C. ; Wolff, C. ; Schmid, L.K. ; Leipprand, A. ; Koch, N. ; Kornek, U. ; Pahle, M. – « Decarbonization and EU ETS Reform: Introducing a Price Floor to Drive Low-Carbon Investments » :contentReference[oaicite:28]{index=28}]
[Edquist, C. – « Systems of innovation approaches – their emergence and characteristics » :contentReference[oaicite:29]{index=29}]
[European Central Bank, 2022 – « The role of speculation during the recent increase in EU emissions allowance prices: issue 3/2022 » :contentReference[oaicite:30]{index=30}]
[European Commission, 2022a – « Allocation to industrial installations » :contentReference[oaicite:31]{index=31}]
[European Commission, 2022b – « EU emissions trading system (EU ETS) » :contentReference[oaicite:32]{index=32}]
[European Commission, 2022c – « Innovation fund: policy development » :contentReference[oaicite:33]{index=33}]

[European Commission, 2022d – « Market stability reserve » :contentReference[oaicite:34]{index=34}]
[European Commission, 2022e – « Modernisation fund » :contentReference[oaicite:35]{index=35}]
[European Commission, 2022f – « Sustainability-related disclosure in the financial services sector: what the obligations are for manufacturers of financial products and financial advisers towards end-investors » :contentReference[oaicite:36]{index=36}]
[Europex, 2021 – « Carbon Contracts for Difference (CCfDs) and their potentially distortive effects on emission markets: call for a comprehensive impact assessment » :contentReference[oaicite:37]{index=37}]
[Martin, R. ; Muûls, M. ; Wagner, U.J. – « The impact of the European union emissions trading scheme on regulated firms: what is the evidence after ten years? » :contentReference[oaicite:38]{index=38}]
[Matschoss, P. ; Welsch, H. – « International emissions trading and induced carbon-saving technological change: effects of restricting the trade in carbon rights » :contentReference[oaicite:39]{index=39}]
[Mazzanti, M. ; Rizzo, U. – « Diversely moving towards a green economy: techno-organisational decarbonisation trajectories and environmental policy in EU sectors » :contentReference[oaicite:40]{index=40}]
[McAndrew, R. ; Mulcahy, R. ; Gordon, R. ; Russell-Bennett, R. – « Household energy efficiency interventions: a systematic literature review » :contentReference[oaicite:41]{index=41}]
[Moher, D. ; Liberati, A. ; Tetzlaff, J. ; Altman, D.G. – « Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement » :contentReference[oaicite:42]{index=42}]
[Mowery, D. ; Rosenberg, N. – « The influence of market demand upon innovation: a critical review of some recent empirical studies » :contentReference[oaicite:43]{index=43}]
[Venmans, F.M.J., 2016 – « The effect of allocation above emissions and price uncertainty on abatement investments under the EU ETS » :contentReference[oaicite:44]{index=44}]
[Von Stechow, C. ; Watson, J. ; Praetorius, B., 2011 – « Policy incentives for carbon capture and storage technologies in Europe: a qualitative multi-criteria analysis » :contentReference[oaicite:45]{index=45}]
[Warner, K.E., 1974 – « The need for some innovative concepts of innovation: an examination of research on the diffusion of innovations » :contentReference[oaicite:46]{index=46}]
[Winkler, D., 2022 – « Pollution for sale: lobbying, allowance allocation and firm outcomes in the EU ETS » :contentReference[oaicite:47]{index=47}]
[Xia, L. ; Gao, S. ; Wei, J. ; Ding, Q., 2022 – « Government subsidy and corporate green innovation – does board governance play a role? » :contentReference[oaicite:48]{index=48}]
[Yang, Defeng ; Wang, Aric Xu ; Zhou, Kevin Zheng ; Jiang, Wei, 2019 – « Environmental strategy, institutional force, and innovation capability: A managerial cognition perspective » :contentReference[oaicite:49]{index=49}]
[Stornelli, A. ; Ozcan, S. ; Simms, C., 2021 – « Advanced manufacturing technology adoption and innovation: a systematic literature review on barriers, enablers, and innovation types » :contentReference[oaicite:50]{index=50}]
[Teixido, J. ; Verde, S.F. ; Nicolli, F., 2019 – « The impact of the EU Emissions Trading System on low-carbon technological change: the empirical evidence » :contentReference[oaicite:51]{index=51}]

[Tomás, R. ; Ramôa Ribeiro, F. ; Santos, V. ; Gomes, J. ; Bordado, J., 2010 – « Assessment of the impact of the European CO_2 emissions trading scheme on the Portuguese chemical industry » :contentReference[oaicite:52]{index=52}]
[Tranfield, D. ; Denyer, D. ; Smart, P., 2003 – « Towards a methodology for developing evidence-informed management knowledge by means of systematic review » :contentReference[oaicite:53]{index=53}]
[Utterback, J.M. ; Abernathy, W.J., 1975 – « A dynamic model of process and product innovation » :contentReference[oaicite:54]{index=54}]
Titres extraits des références :
Understanding and tackling societal grand challenges through management research
Permit markets, carbon prices and the creation of innovation clusters
Effective EU and member state policies for stimulating CCS
Induced innovation in energy technologies and systems: a review of evidence and potential implications for CO_2 mitigation
The limited effect of EU emissions trading on corporate climate strategies: comparison of a Swedish and a Norwegian pulp and paper company
Institutional evolution and change: environmentalism and the US chemical industry
Corporate carbon accounting: a literature review of carbon accounting research from the Kyoto Protocol to the Paris Agreement
Windfall profits in the power sector during phase III of the EU ETS: interplay and effects of renewables and carbon prices
An overview of innovation. In: The Positive Sum Strategy: Harnessing Technology for Economic Growth
Does the EU Emissions Trading System induce investment leakage? Evidence from German multinational firms
Effects of government subsidies on green technology investment and green marketing coordination of supply chain under the cap-and-trade mechanism
The Effect of Carbon Pricing on Technological Change for Full Energy Decarbonization: A Review of Empirical Ex-Post Evidence
Effects of the EU Emission Trading Scheme on the international competitiveness of pulp-and-paper industry
The impact of the EU ETS on prices, profits and emissions in the power sector: simulation results with the COMPETES EU20 model
Why the EU ETS needs reforming: an empirical analysis of the impact on company investments
Carbon prices and incentives for technological development
Political conflict and climate policy: the European emissions trading system as a Trojan Horse for the low-carbon transition?
On the empirical content of carbon leakage criteria in the EU Emissions Trading Scheme
Environmental innovations, local networks and internationalization
Adopt or innovate: understanding technological responses to cap-and-trade
Environmental policy and directed technological change: evidence from the European carbon market
Technological pervasiveness and variety of innovators in Green ICT: a patent-based analysis
Time-consistent Carbon Pricing: the Role of Carbon Contracts for Differences
Grandfathering, auctioning and carbon leakage: assessing the inconsistencies of the new ETS directive

Do green jobs differ from non-green jobs in terms of skills and human capital?
A dynamic assessment of instrument interaction and timing alternatives in the EU low-carbon policy mix design
Testing the mediating role of open innovation on the relationship between intellectual property rights and organizational…
Conceptual issues in the study of innovation
Decarbonization and EU ETS Reform: Introducing a Price Floor to Drive Low-Carbon Investments
Systems of innovation approaches – their emergence and characteristics
The role of speculation during the recent increase in EU emissions allowance prices: issue 3/2022
Allocation to industrial installations
EU emissions trading system (EU ETS)
Innovation fund: policy development
Market stability reserve
Modernisation fund
Sustainability-related disclosure in the financial services sector: what the obligations are for manufacturers of financial products and financial advisers towards end-investors
Carbon Contracts for Difference (CCfDs) and their potentially distortive effects on emission markets: call for a comprehensive impact assessment
The impact of the European union emissions trading scheme on regulated firms: what is the evidence after ten years?
International emissions trading and induced carbon-saving technological change: effects of restricting the trade in carbon rights
Diversely moving towards a green economy: techno-organisational decarbonisation trajectories and environmental policy in EU sectors
Household energy efficiency interventions: a systematic literature review
Preferred reporting items for systematic reviews and meta-analyses: the PRISMA statement
The influence of market demand upon innovation: a critical review of some recent empirical studies
The effect of allocation above emissions and price uncertainty on abatement investments under the EU ETS
Policy incentives for carbon capture and storage technologies in Europe: a qualitative multi-criteria analysis
The need for some innovative concepts of innovation: an examination of research on the diffusion of innovations
Pollution for sale: lobbying, allowance allocation and firm outcomes in the EU ETS
Government subsidy and corporate green innovation – does board governance play a role?
Environmental strategy, institutional force, and innovation capability: A managerial cognition perspective
Advanced manufacturing technology adoption and innovation: a systematic literature review on barriers, enablers, and innovation types
The impact of the EU Emissions Trading System on low-carbon technological change: the empirical evidence
Assessment of the impact of the European CO_2 emissions trading scheme on the Portuguese chemical industry
Towards a methodology for developing evidence-informed management knowledge by means of systematic review

```
A dynamic model of process and product innovation

Matrice de confusion (formatée) :
               Non Retenu  Retenu
Non Pertinent        2653      37
Pertinent               0       9
```

catégoriser les groueps

In [51]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, c
all drive.mount("/content/drive", force_remount=True).
```

In [52]:
```
import pandas as pd

# Chemin complet vers votre fichier
file_path = '/content/drive/MyDrive/M2 MASTER/Projet tuteuré/extraction a

# Lecture du CSV
df = pd.read_csv(file_path)

print(df.head())
```

```
                                               title  \
0  Measuring regional innovation: A critical insp...
1  Functions of innovation systems: A new approac...
2  Why low-carbon technological innovation hardly...
3  Executive green investment vision, stakeholder...
4  Carbon constraint in the Mediterranean: Differ...

                                             authors  \
0  ['Hauser, Christoph ; Siller, Matthias ; Schat...
1  ['Hekkert, M.P. ; Suurs, R.A.A. ; Negro, S.O. ...
2  ['Li, Wenchao ; Xu, Jian ; Ostic, Dragana ; Ya...
3  ['Wan, Xiaole ; Wang, Yuxuan ; Qiu, Lulian ; Z...
4                ['Boisgibault, Louis ; Mozas, M A']

                                         description  \
0  The disparities in regional innovation are oft...
1  The central idea of this paper is that innovat...
2  •China's energy efficiency is heterogeneous in...
3  During the 14th Five Year Plan period, the gre...
4  European Union's energy goals for 2020, inclus...

                                            subjects        creation_date
\
0  ['Community Innovation Survey', 'Indexes', 'In...  2018-01-01T00:00:00
1  ['Determinants', 'Dynamical systems', 'Dynamic...  2007-01-01T00:00:00
2  ['Agglomeration', 'Energy efficiency', 'FCM', ...  2021-01-01T00:00:00
3  ['business ecosystem', 'enterprise green innov...  2022-01-01T00:00:00
```

```
4  ['Economics and Finance', 'Geography', 'Humani...  2012-01-01T00:00:00

                  source                                          full_text_li
nk  \
0  New York: Elsevier Inc                                                   N
aN
1            Elsevier Inc                                                   N
aN
2            Elsevier Ltd                                                   N
aN
3     Frontiers Media S.A                                                   N
aN
4                    NaN  $$Uhttps://shs.hal.science/halshs-00942495$$EV.
..

                                        record_id rsrctype langue  ...
\
0               TN_cdi_proquest_journals_2084459774  article     en  ...
1            TN_cdi_proquest_miscellaneous_743050087  article     en  ...
2  TN_cdi_crossref_primary_10_1016_j_cie_2021_107566  article     en  ...
3  TN_cdi_doaj_primary_oai_doaj_org_article_fb954...  article     en  ...
4       TN_cdi_hal_primary_oai_HAL_halshs_00942495v1  article     fr  ...

  methode_quantitative                groupe_methodes  \
0                 True  Regression Models, Time Series
1                False                            NaN
2                 True              Regression Models
3                 True  Regression Models, Time Series
4                 True  Regression Models, Time Series

                           extracted_years phase_description  \
0                                       []               NaN
1                                       []               NaN
2                                       []               NaN
3                                       []               NaN
4  [2020, 2012, 2013, 2020, 2012, 2005, 2013]           Phase 1

                                    first_6_words  doublon Identifiant
\
0  measuring regional innovation: a critical insp...    False           1
1             functions of innovation systems: a new    False           2
2  why low-carbon technological innovation hardly...    False           3
3  executive green investment vision, stakeholder...    False           4
4  carbon constraint in the mediterranean: differ...    False           5

                                  secteur                   effet  \
0  secteur de l'innovation technologique    Effet non significatif
1  secteur de l'innovation technologique              Effet positif
2  secteur de l'innovation technologique               Effet positif
3                  secteur de l'industrie             Effet positif
4                          secteur aérien             Effet négatif

                                    rédumé effet
```

```
0  les indices d’innovations populaires ne sont p...
1  es systèmes d’innovations sont déterminants da...
2  Malgré les disparités selon les régions en chi...
3  les investissements verts permettent notamment...
4  Les quotas carbones peuvent générer à court ou...

[5 rows x 25 columns]
```

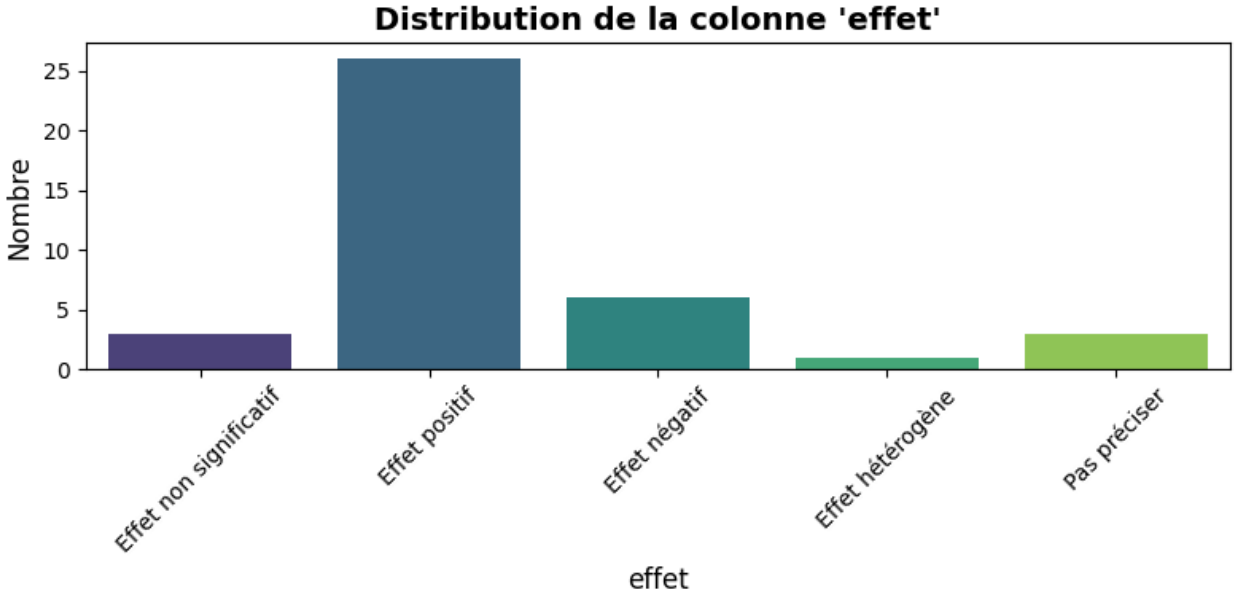
In [53]:
```
def clean_and_remove_duplicates(cell):
    # Si la cellule est manquante, on la renvoie telle quelle
    if pd.isnull(cell):
        return cell
    # On convertit la cellule en chaîne de caractères et on la découpe pa
    items = [item.strip() for item in str(cell).split(',')]
    # On retire les doublons en préservant l'ordre
    seen = set()
    unique_items = []
    for item in items:
        if item not in seen:
            unique_items.append(item)
            seen.add(item)
    # On reconstitue la chaîne, avec les éléments uniques séparés par une
    return ', '.join(unique_items)

# Liste des colonnes à nettoyer
cols_to_clean = ["effet", "secteur"]

# Appliquer la fonction à chacune de ces colonnes
for col in cols_to_clean:
    df[col] = df[col].apply(clean_and_remove_duplicates)

# Vérifier le résultat
print(df[cols_to_clean].head())
```

```
                   effet                                secteur
0  Effet non significatif  secteur de l'innovation technologique
1           Effet positif  secteur de l'innovation technologique
2           Effet positif  secteur de l'innovation technologique
3           Effet positif                secteur de l'industrie
4           Effet négatif                        secteur aérien
```

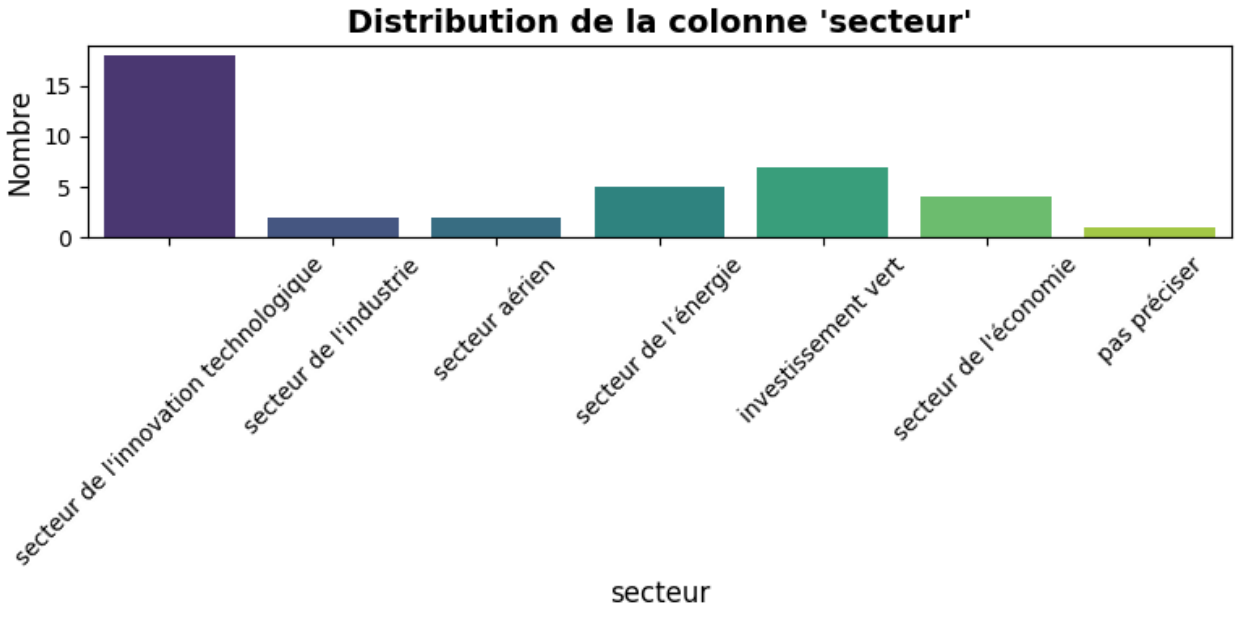
In [54]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
colonnes_a_visualiser = ["effet", "secteur"]

# 5. Création d'un histogramme (diagramme en barres) pour chaque colonne
for col in colonnes_a_visualiser:
    plt.figure(figsize=(8, 4))
    sns.countplot(x=col, data=df, palette="viridis")
    plt.title(f"Distribution de la colonne '{col}'", fontsize=14, fontwei
    plt.xlabel(col, fontsize=12)
    plt.ylabel("Nombre", fontsize=12)
    plt.xticks(rotation=45)  # Faites pivoter les étiquettes si nécessair
```

```
    plt.tight_layout()
    plt.show()
```

```
<ipython-input-54-c27bd62b3458>:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be remove
d in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for
the same effect.

  sns.countplot(x=col, data=df, palette="viridis")
```




```
<ipython-input-54-c27bd62b3458>:9: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be remove
d in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for
the same effect.

  sns.countplot(x=col, data=df, palette="viridis")
```



In [57]:
```
df.secteur.value_counts()
```

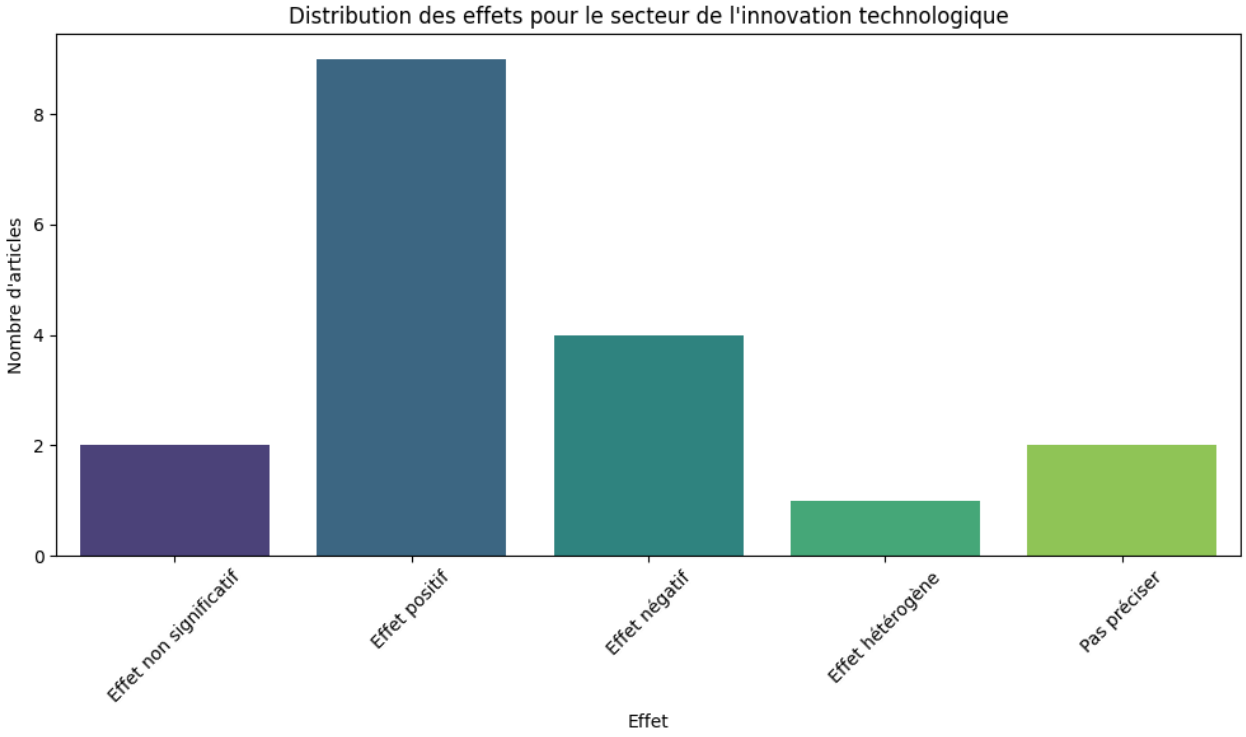
Out[57]:

secteur	count
secteur de l'innovation technologique	18
investissement vert	7
secteur de l'énergie	5
secteur de l'économie	4
secteur de l'industrie	2
secteur aérien	2
pas préciser	1

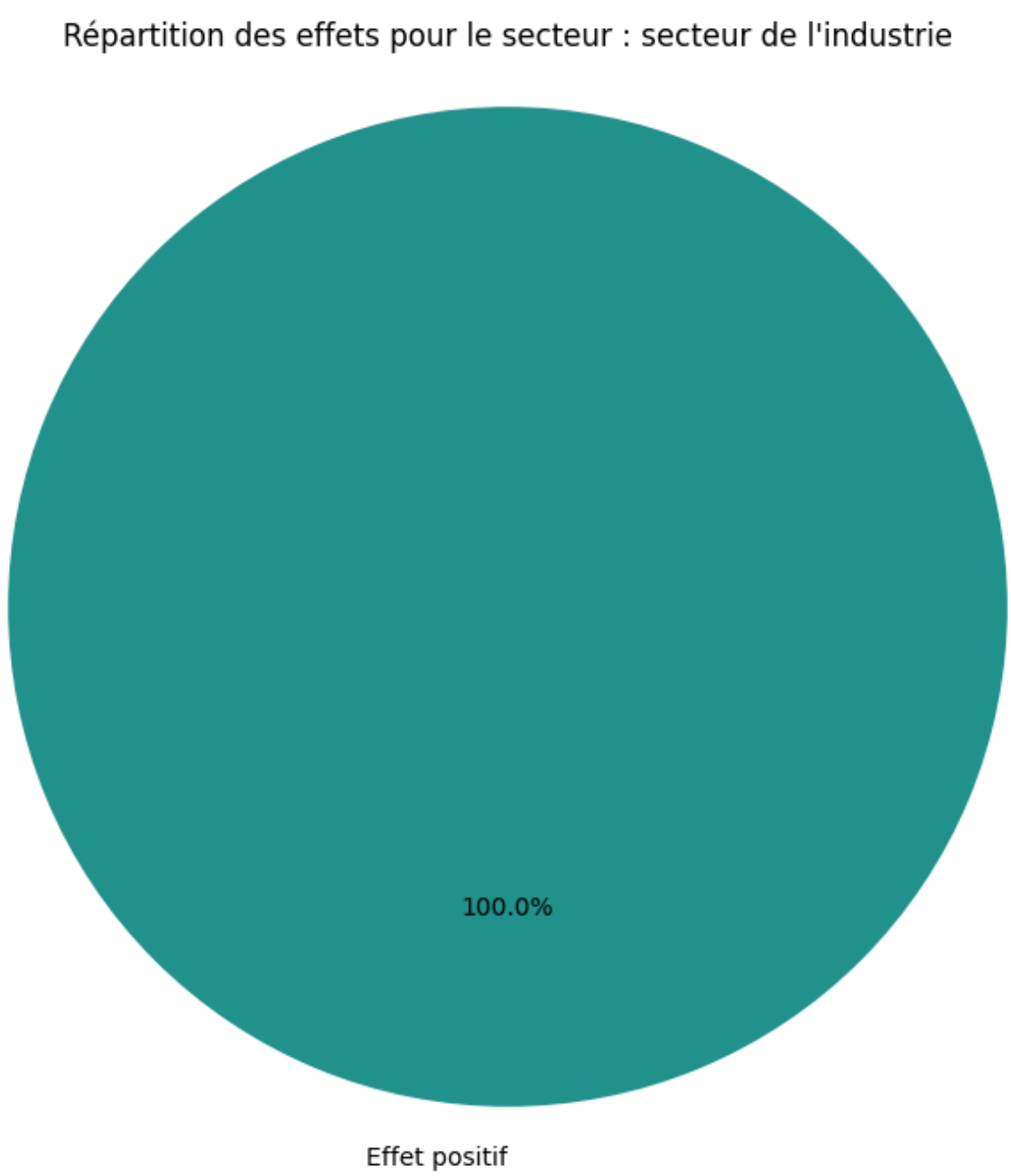
dtype: int64

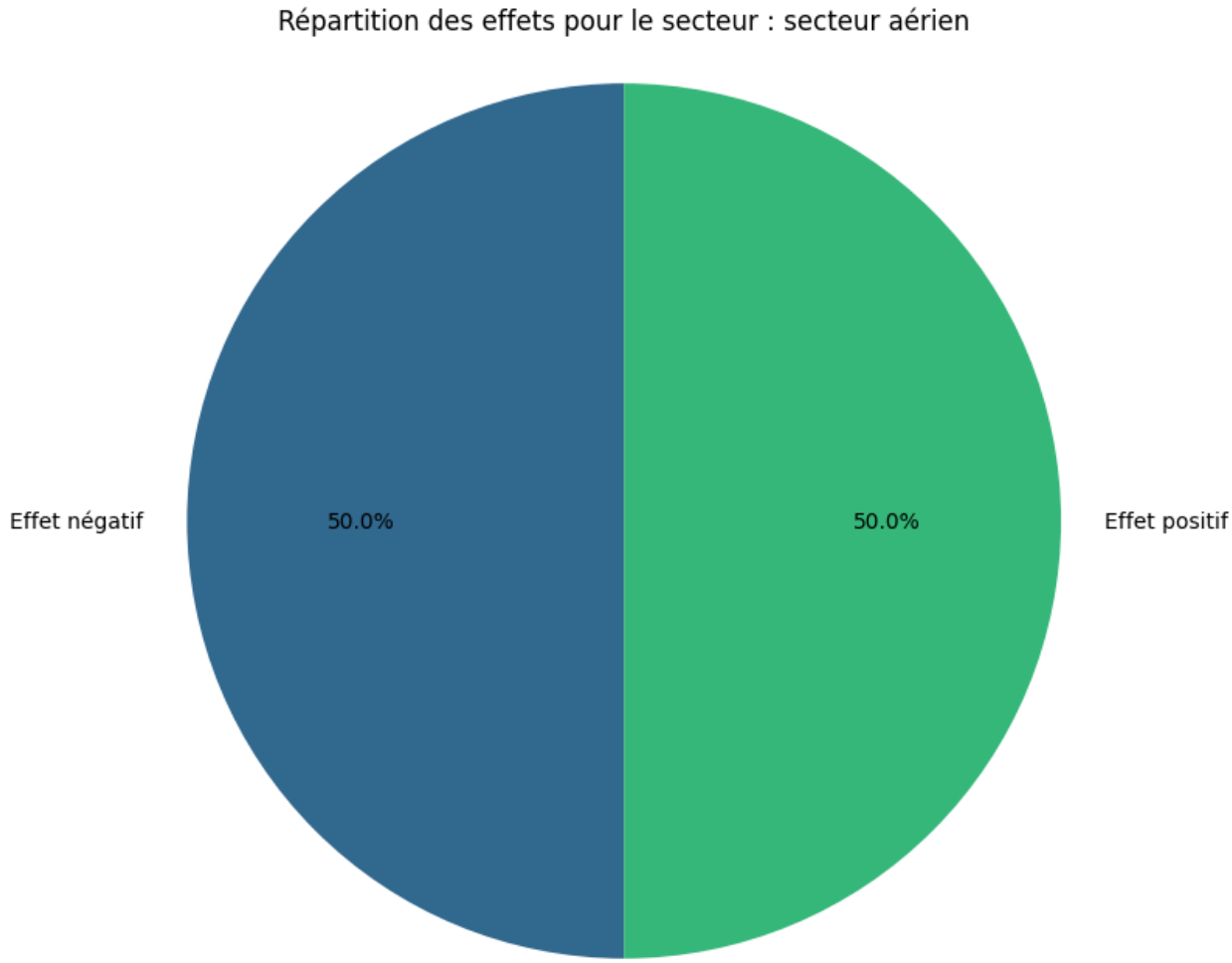
In [59]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Supposons que df est le DataFrame contenant les colonnes "effet" et "se
# Par exemple : df = pd.read_csv("votre_fichier.csv")

# 1. Histogramme pour les articles dont le secteur contient "EU ETS"
df_euets = df[df["secteur"].str.contains("secteur de l'innovation technol
plt.figure(figsize=(10,6))
sns.countplot(data=df_euets, x="effet", palette="viridis")
plt.title("Distribution des effets pour le secteur de l'innovation techno
plt.xlabel("Effet")
plt.ylabel("Nombre d'articles")
plt.xticks(rotation=45)
plt.tight_layout()
plt.show()
```

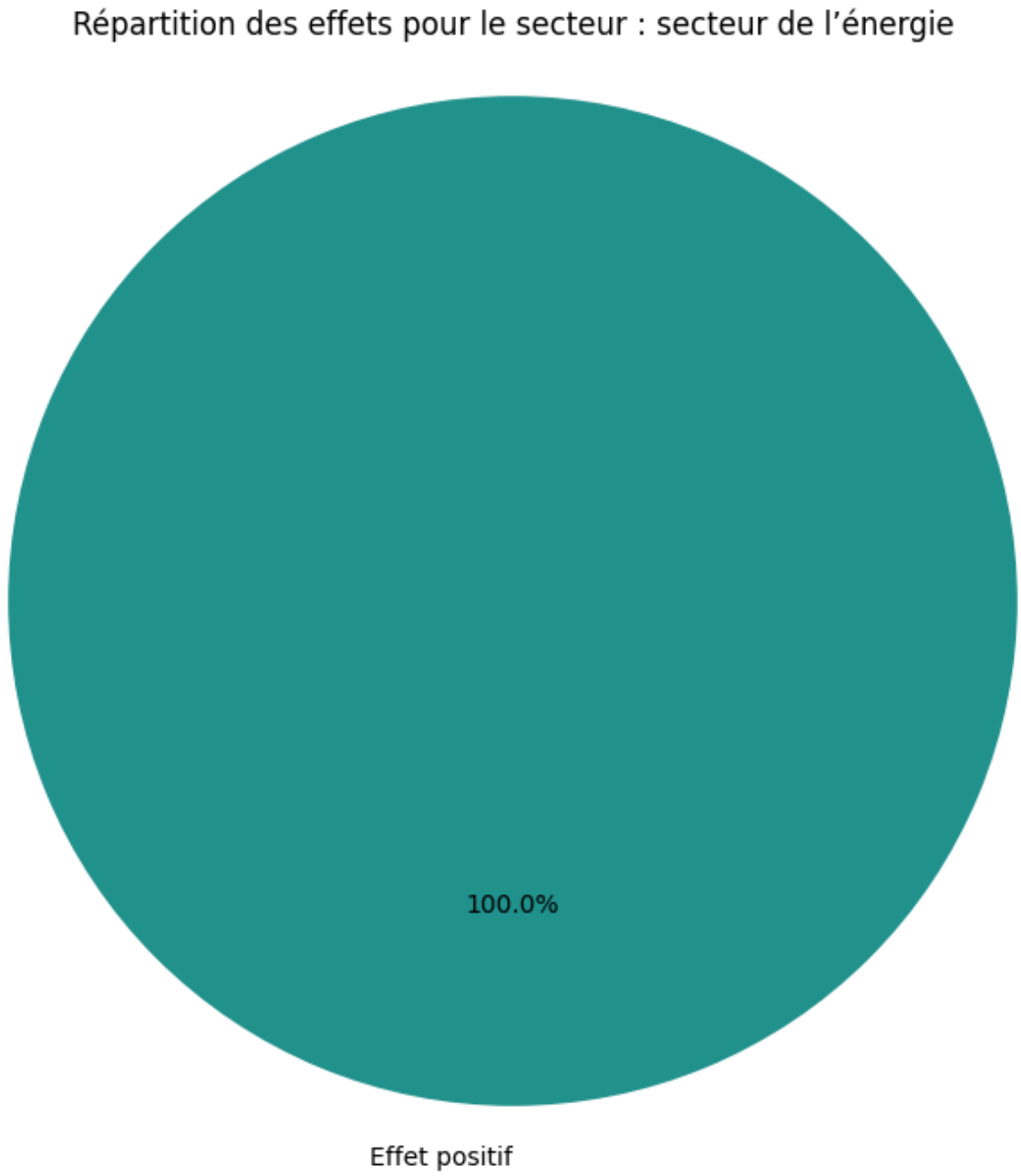
```
<ipython-input-59-7c1e258152ff>:11: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be remove
d in v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for
the same effect.

  sns.countplot(data=df_euets, x="effet", palette="viridis")
```

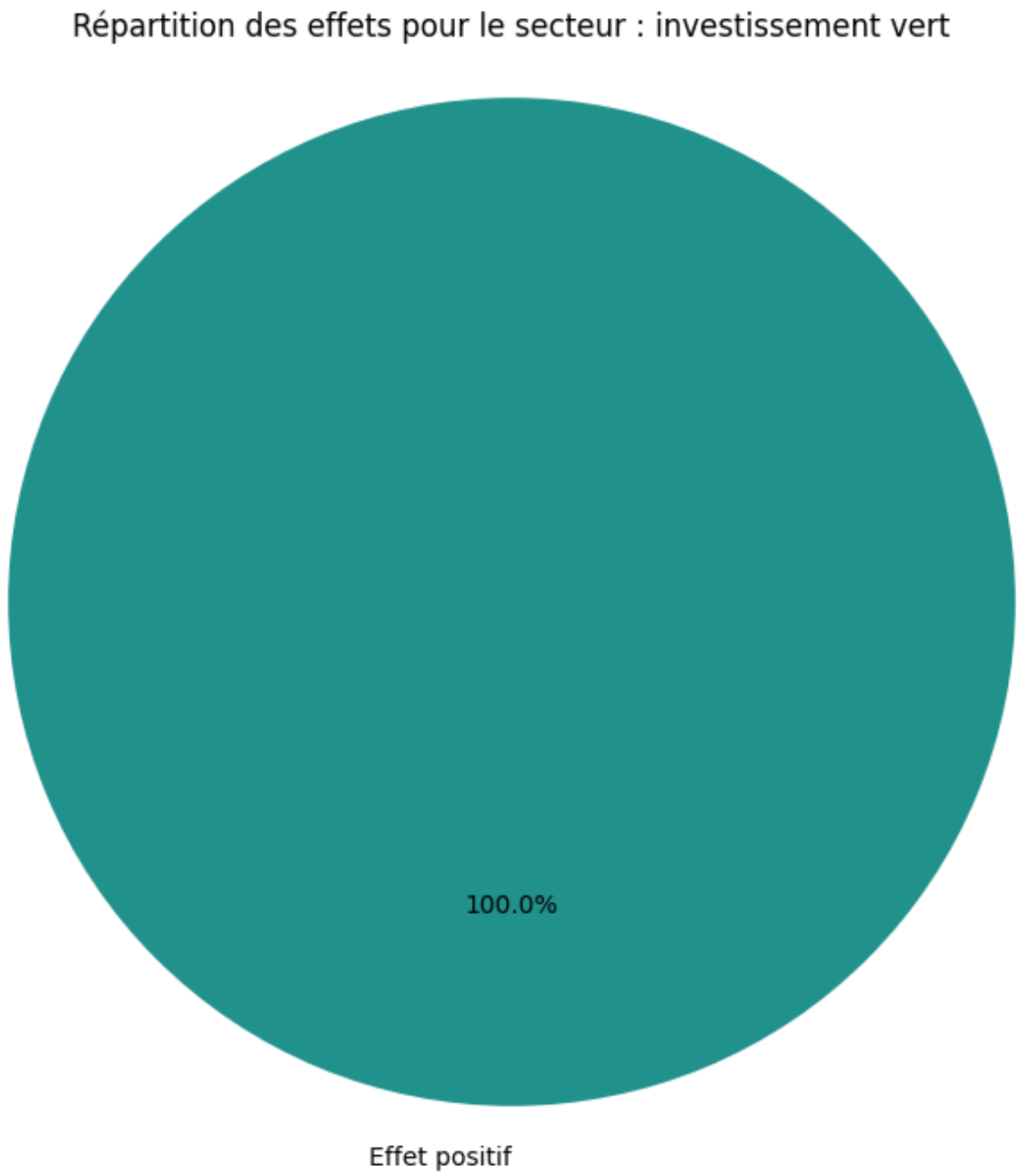


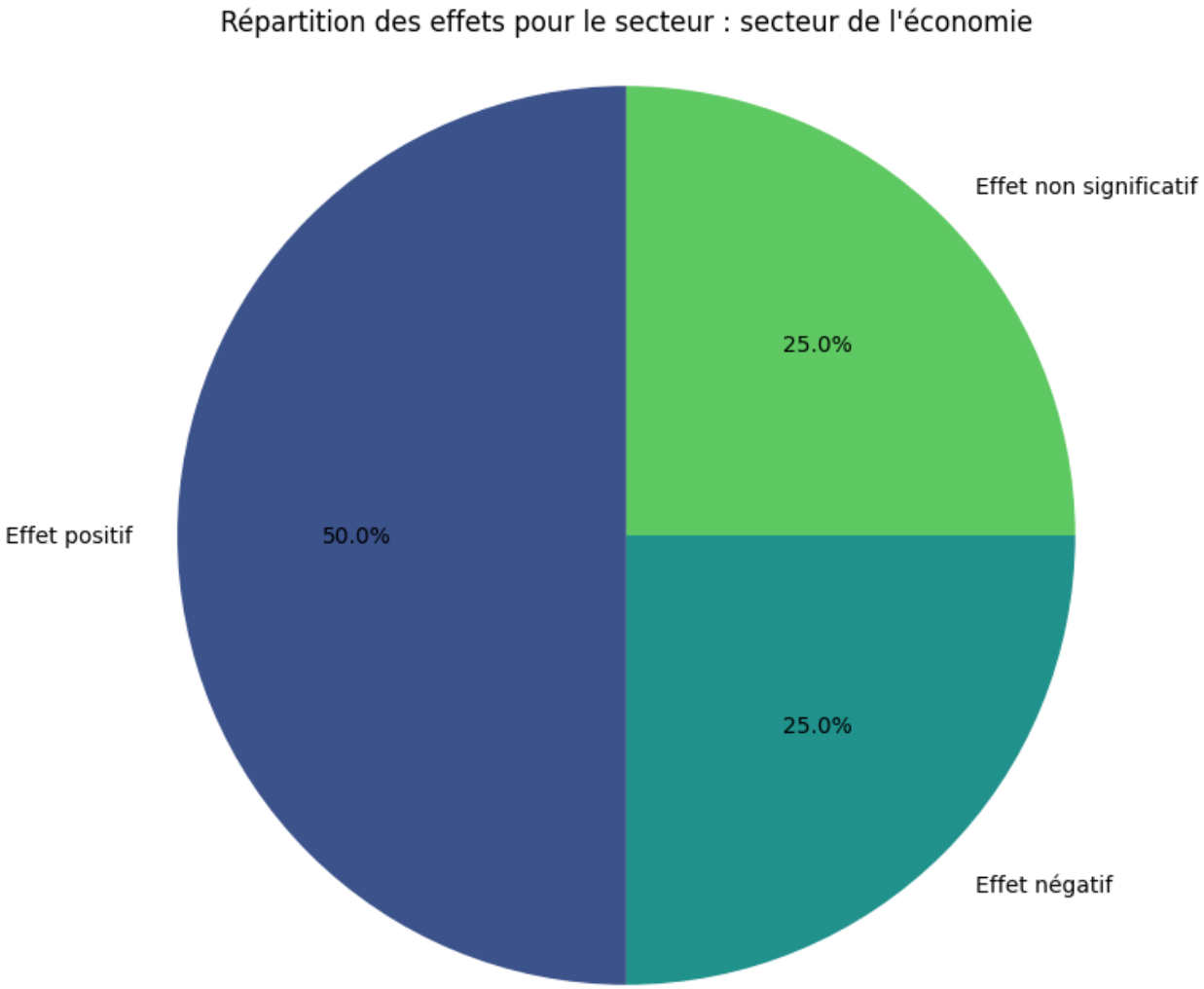
In [61]:
```python
import matplotlib.pyplot as plt
import seaborn as sns

# Filtrer les articles pour exclure "secteur de l'innovation technologiqu
df_filtered = df[~df["secteur"].str.contains("secteur de l'innovation tec

# Obtenir la liste des secteurs uniques dans ce sous-ensemble
secteurs = df_filtered["secteur"].unique()

# Pour chaque secteur, calculer la répartition des effets et tracer un ca
for secteur in secteurs:
    # Sélectionner les articles pour le secteur en cours
    df_secteur = df_filtered[df_filtered["secteur"] == secteur]

    # Calculer le nombre d'articles par effet
    effet_counts = df_secteur["effet"].value_counts()

    # Définir une palette de couleurs adaptée au nombre de modalités
    colors = sns.color_palette("viridis", len(effet_counts))

    # Créer le camembert
    plt.figure(figsize=(8,8))
    plt.pie(effet_counts, labels=effet_counts.index, autopct='%1.1f%%', s
    plt.title(f"Répartition des effets pour le secteur : {secteur}")
    plt.axis('equal')  # Pour que le camembert soit bien circulaire
    plt.show()
```

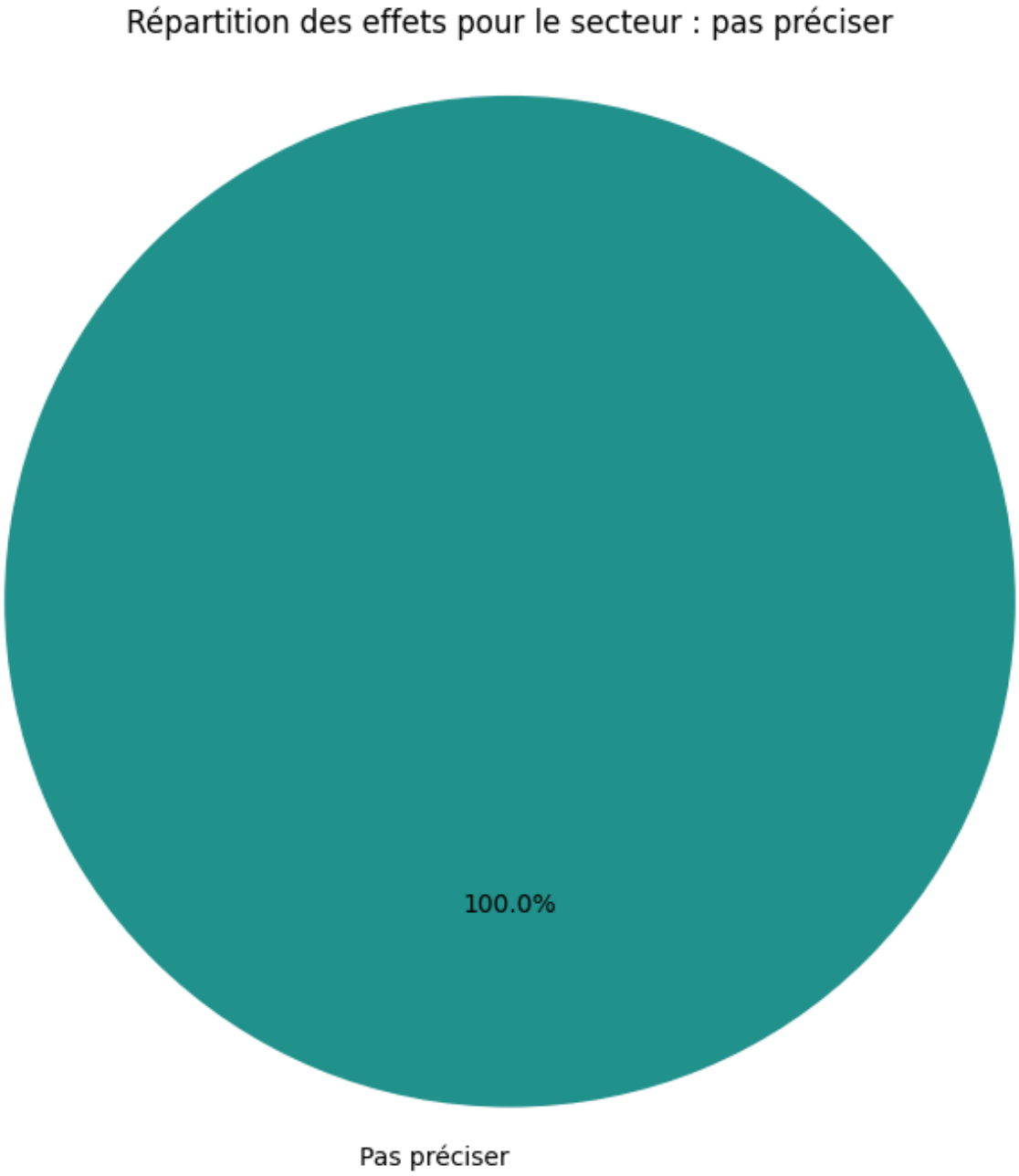

Répartition des effets pour le secteur : secteur de l'industrie
100.0%
Effet positif


Répartition des effets pour le secteur : secteur aérien
Effet négatif
50.0%
50.0%
Effet positif


Répartition des effets pour le secteur : secteur de l’énergie
100.0%
Effet positif


Répartition des effets pour le secteur : investissement vert
100.0%
Effet positif


Répartition des effets pour le secteur : secteur de l'économie
Effet non significatif
25.0%
Effet positif
50.0%
25.0%
Effet négatif



In [ ]:
```
!jupyter nbconvert --to html "/content/drive/MyDrive/M2 MASTER/Projet tut
```

```
[NbConvertApp] Converting notebook /content/drive/MyDrive/M2 MASTER/Projet
tuteuré/Scraping_fina_MAX_export.ipynb to html
[NbConvertApp] WARNING | Alternative text is missing on 8 image(s).
[NbConvertApp] Writing 977686 bytes to /content/drive/MyDrive/M2 MASTER/Pr
ojet tuteuré/Scraping_fina_MAX_export.html
```