1 PPCIC Dissertação - 2021

DETECÇÃO DE EVENTOS ADVERSOS ATRAVÉS DO TWITTER UTILIZANDO O METAMAP PARA O PORTUGUÊS DO BRASIL

1.1 Matéria: CIC1229 - TÓPICOS ESPECIAIS EM ALGORITMOS

1.2 Aluno: Perciliano

1.3 Orientadora: Kele Belloze

1.3.1 Data: 11/06/2021

Autor: Luiz Perciliano - ***luiz.perciliano@eic.cefet-rj.br (mailto:luiz.perciliano@eic.cefet-rj.br)***

- Dados da pesquisa na base SCOPUS dia 31-05-2021
  - String: "metamap" OR ("natural language processing" AND "pharmacovigilance") OR ("text mining" AND (pharmacovigilance OR "adverse event" OR "adverse effect"))
  - 574 artigos
- Download de dados da pesquisa do Qualis dia 06-06-2021

Requisitos

- RF01 - Analisar bases de dados
- RF02 - Mesclar bases para identificar estrato do artigo
- RF03 - Remover duplicatas
- RF03 - Identificar artigos para leitura de resumo e conclusão
- RF04 - Identificar artigos para leitura completa

Site do projeto: https:git.com
Endereço da dissertação: https://pt.overleaf.com/project/60731ca2bcfa0afce8ae0cd1 (https://pt.overleaf.com/project/60731ca2bcfa0afce8ae0cd1)

---

2 Preparar Infraesrtutura

In [1]:

```
print(f'Importar as bibliotecas necessárias e mapear a pasta do projeto.')
import os, re
import sys
import pandas as pd
import datetime

import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

import nltk
from unidecode import unidecode
from wordcloud import WordCloud, STOPWORDS, ImageColorGenerator
```

```
Importar as bibliotecas necessárias e mapear a pasta do projeto.
```

In [2]:

```
data_inicio = pd.Timestamp.now()
print(data_inicio)
```

```
2021-06-10 11:24:38.137884
```

In [3]:

```
print(f'Lista do conteúdo da pasta ...')
os.listdir(os.path.join('..','data'))
```

```
Lista do conteúdo da pasta ...
```

Out[3]:

```
['base-qualis-06-06-2021.xlsx',
 'classificacao_qualis_06-06-2021.xls',
 'requiremets.txt',
 'resultado_geral.xlsx',
 'resultado_geral_06-06-2021.xlsx',
 'resultado_scopus.xlsx',
 'scopus-31-05-2021.csv',
 'stop_word_projeto.csv',
 'stop_word_pt.csv']
```

In [4]:

```
os.listdir('../Image')
```

Out[4]:

```
['logo-cefet.png', 'Logo_CVM.png', 'wordcloud.pdf']
```

2.1 Verificar, atualizar e instalar se necessário python e módulos

In [ ]:

```
## versao 3,9,0 estava funcionando
print('Local de instalação do Python:       ',sys.executable)
print('Versão do Python instalado e em uso: ',sys.version)
print('Informações da versão do Python:     ',sys.version_info)
print(f'Quantidade de CPU:                    {os.cpu_count()}')
```

In [ ]:

```
print('Atualizando os módulos Python.')
#!pip install --upgrade pip
```

In [ ]:

```
print('Local de instalação do Python:       ',sys.executable)
print('Versão do Python instalado e em uso: ',sys.version)
print('Informações da versão do Python:     ',sys.version_info)
print(f'Quantidade de CPU:                    {os.cpu_count()}')
```

In [ ]:

```
print('Instalando os módulos necessários.')
# print('')
#!pip install Unidecode -q
#!pip install sklearn
#!pip install wordcloud
#!pip install wget
#!pip install opencv-python #import cv2
#!pip install wand #wand=0.6.5
#!pip install jupyter_contrib_nbextensions
#!pip install pip-chill ## para verificar todos os mõdulos instalados para uma nova insta
```

```
print('Gerar arquivo com as dependencias necessárias para o projeto.')
#import pip-chill
#pip install -r > requirements.txt
pip freeze
```

```
print('Criando as variáveis para o projeto.')
## ????testar com novo caminho xxxx ????? -- path_pasta_trab =
os.path.join('SisCRI','data')

#pasta_trab = os.path.join('SisCRI','data','')
pasta_trab = 'C:\\Users\\luizp\\jupyter-notebook\\SisCRI\\data\\'

#print(f'Importar documentos necessários para o projeto. (stopwords, etc)')

# carregar stop_words padrao no git e sempre pegar de lá para demais projetos (eng e pt-
Br)
```

3 Preparar e carregar base de dados

3.1 Conexão e consulta ao SQL Server

3.2 Carregar Planilha - Base Qualis

In [5]:

```
print('Carregando pesquisa realizada na base Scopus.')
'''
Fonte: https://sucupira.capes.gov.br/sucupira/public/consultas/coleta/veiculoPublicacaoQu
Periódico do quadriênio 2013-2016
'''
arquivo_base_qualis = os.path.join('../data','base-qualis-06-06-2021.xlsx')
raw_data_qualis = pd.read_excel(arquivo_base_qualis)
raw_data_qualis
```

```
Carregando pesquisa realizada na base Scopus.
```

Out[5]:

	ISSN	Título	Área de Avaliação	Estrato
0	1981-030X	19&20 (RIO DE JANEIRO)	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	C
1	2236-6695	A BARRIGUDA: REVISTA CIENTÍFICA	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	B4
2	1413-6090	A ECONOMIA EM REVISTA	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	B4
3	1516-3210	A&C. REVISTA DE DIREITO ADMINISTRATIVO & CONST...	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	B4
4	0001-3072	ABACUS (SYDNEY. PRINT)	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	A2
...	...	...	...	...
131269	1175-5326	ZOOTAXA (AUCKLAND. PRINT)	ZOOTECNIA / RECURSOS PESQUEIROS ...	B1
131270	1175-5334	ZOOTAXA (ONLINE)	ZOOTECNIA / RECURSOS PESQUEIROS ...	B1
131271	2358-3576	ZOOTECNIA	ZOOTECNIA / RECURSOS PESQUEIROS ...	C
131272	0798-7269	ZOOTECNIA TROPICAL - FONAIAP	ZOOTECNIA / RECURSOS PESQUEIROS ...	B3
131273	0967-1994	ZYGOTE (CAMBRIDGE. PRINT)	ZOOTECNIA / RECURSOS PESQUEIROS ...	B2

131274 rows × 4 columns

3.3 Carregar CSV - Busca Scopus

In [6]:

```
print('Visualização do dataframe carregado da base scopus.')
arquivo_base_scopus = os.path.join('../data','scopus-31-05-2021.csv')
raw_data_scopus = pd.read_csv(arquivo_base_scopus, sep=",", delimiter=None, encoding='utf
raw_data_scopus.head()
```

```
Visualização do dataframe carregado da base scopus.
```

Out[6]:

	Authors	Author(s) ID	Title	Year	Sou
0	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	R
1	Aronson A.R.	17933416200;	Effective mapping of biomedical text to the UM...	2001	Proc / Sym
2	Aronson A.R., Lang F.-M.	17933416200;13612282900;	An overview of MetaMap: Historical perspective...	2010	J A Inf
3	Piñero J., Bravo A., Queralt-Rosinach N., Guti...	55220852900;56374942700;35766634600;5672880770...	DisGeNET: A comprehensive platform integrating...	2017	R
4	Nikfarjam A., Sarker A., O'Connor k., Ginn R.,...	36069663700;36976315000;56596185000;5659652430...	Pharmacovigilance from social media: Mining ad...	2015	J A Inf

5 rows × 47 columns

In [108]:

```
print('Carga e Visualização do dataframe')
arquivo_base_scopus = os.path.join('../data','pubmed-10-06-2021.csv')
raw_data_pubmed = pd.read_csv(arquivo_base_scopus, sep=",", delimiter=None, encoding='utf
raw_data_pubmed.head()
```

```
Carga e Visualização do dataframe
```

Out[108]:

	PMID	Title	Authors	Citation	First Author	Journal/Book	Publication Year	
0	30649735	Overview of the First Natural Language Process...	Jagannatha A, Liu F, Liu W, Yu H.	Drug Saf. 2019 Jan;42(1):99-111. doi: 10.1007/...	Jagannatha A	Drug Saf	2019	2
1	33245290	Identification of Adverse Drug Event-Related J...	Ujiie S, Yada S, Wakamiya S, Aramaki E.	JMIR Med Inform. 2020 Nov 27;8(11):e22661. doi...	Ujiie S	JMIR Med Inform	2020	2
2	31630063	A systematic review of natural language proces...	Young IJB, Luz S, Lone N.	Int J Med Inform. 2019 Dec;132:103971. doi: 10...	Young IJB	Int J Med Inform	2019	2
3	33278631	Natural language processing with deep learning...	Borjali A, Magnéli M, Shin D, Malchau H, Murat...	Comput Biol Med. 2021 Feb;129:104140. doi: 10....	Borjali A	Comput Biol Med	2021	2
4	26394725	Can Natural Language Processing Improve the Ef...	Baer B, Nguyen M, Woo EJ, Winiecki S, Scott J,...	Methods Inf Med. 2016;55(2):144-50. doi: 10.34...	Baer B	Methods Inf Med	2016	2

In [ ]:

```
### https://www.gov.br/capes/pt-br
```

3.4 Analisar dados brutos original

3.4.1 Analisando base SCOPUS

In [7]:

```
#pd.set_option("max_colwidth", 100)
```

In [8]:

```
print('Verificando tipos e se tem dados nulos')
raw_data_scopus.info()
```

```
Verificando tipos e se tem dados nulos
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 574 entries, 0 to 573
Data columns (total 47 columns):
 #   Column                          Non-Null Count  Dtype  
---  ------                          --------------  -----  
 0   Authors                         574 non-null    object 
 1   Author(s) ID                    574 non-null    object 
 2   Title                           574 non-null    object 
 3   Year                            574 non-null    int64  
 4   Source title                    574 non-null    object 
 5   Volume                          481 non-null    object 
 6   Issue                           242 non-null    object 
 7   Art. No.                        154 non-null    object 
 8   Page start                      420 non-null    object 
 9   Page end                        410 non-null    object 
 10  Page count                      6 non-null      float64
 11  Cited by                        448 non-null    float64
 12  DOI                             459 non-null    object 
 13  Link                            574 non-null    object 
 14  Affiliations                    567 non-null    object 
 15  Authors with affiliations       571 non-null    object 
 16  Abstract                        574 non-null    object 
 17  Author Keywords                 396 non-null    object 
 18  Index Keywords                  520 non-null    object 
 19  Molecular Sequence Numbers      1 non-null      object 
 20  Chemicals/CAS                   108 non-null    object 
 21  Tradenames                      20 non-null     object 
 22  Manufacturers                   2 non-null      object 
 23  Funding Details                 240 non-null    object 
 24  Funding Text 1                  188 non-null    object 
 25  Funding Text 2                  18 non-null     object 
 26  Funding Text 3                  2 non-null      object 
 27  References                      521 non-null    object 
 28  Correspondence Address          457 non-null    object 
 29  Editors                         115 non-null    object 
 30  Sponsors                        75 non-null     object 
 31  Publisher                       419 non-null    object 
 32  Conference name                 202 non-null    object 
 33  Conference date                 202 non-null    object 
 34  Conference location             65 non-null     object 
 35  Conference code                 193 non-null    float64
 36  ISSN                            493 non-null    object 
 37  ISBN                            153 non-null    object 
 38  CODEN                           207 non-null    object 
 39  PubMed ID                       314 non-null    float64
 40  Language of Original Document   574 non-null    object 
 41  Abbreviated Source Title        574 non-null    object 
 42  Document Type                   574 non-null    object 
 43  Publication Stage               574 non-null    object 
 44  Open Access                     245 non-null    object 
 45  Source                          574 non-null    object 
 46  EID                             574 non-null    object 
dtypes: float64(4), int64(1), object(42)
memory usage: 210.9+ KB
```

In [9]:

```
raw_data_scopus.nunique()
```

Out[9]:

```
Authors                          549
Author(s) ID                     546
Title                            572
Year                              25
Source title                     220
Volume                           208
Issue                             39
Art. No.                         147
Page start                       340
Page end                         342
Page count                         5
Cited by                          81
DOI                              459
Link                             574
Affiliations                     552
Authors with affiliations        567
Abstract                         565
Author Keywords                  394
Index Keywords                   520
Molecular Sequence Numbers         1
Chemicals/CAS                     94
Tradenames                        16
Manufacturers                      2
Funding Details                  228
Funding Text 1                   187
Funding Text 2                    18
Funding Text 3                     2
References                       521
Correspondence Address           448
Editors                           77
Sponsors                          64
Publisher                         85
Conference name                  157
Conference date                  155
Conference location               56
Conference code                  150
ISSN                             150
ISBN                             134
CODEN                             60
PubMed ID                        314
Language of Original Document      4
Abbreviated Source Title         213
Document Type                      9
Publication Stage                  2
Open Access                        7
Source                             1
EID                              574
dtype: int64
```

In [10]:

```
# verificar campos nulos
raw_data_scopus.isnull().sum()
```

Out[10]:

```
Authors                          0
Author(s) ID                     0
Title                            0
Year                             0
Source title                     0
Volume                          93
Issue                          332
Art. No.                       420
Page start                     154
Page end                       164
Page count                     568
Cited by                       126
DOI                            115
Link                             0
Affiliations                     7
Authors with affiliations        3
Abstract                         0
Author Keywords                178
Index Keywords                  54
Molecular Sequence Numbers     573
Chemicals/CAS                  466
Tradenames                     554
Manufacturers                  572
Funding Details                334
Funding Text 1                 386
Funding Text 2                 556
Funding Text 3                 572
References                      53
Correspondence Address         117
Editors                        459
Sponsors                       499
Publisher                      155
Conference name                372
Conference date                372
Conference location            509
Conference code                381
ISSN                            81
ISBN                           421
CODEN                          367
PubMed ID                      260
Language of Original Document    0
Abbreviated Source Title         0
Document Type                    0
Publication Stage                0
Open Access                    329
Source                           0
EID                              0
dtype: int64
```

3.4.2 Analisando base Qualis

In [11]:

```
print('Verificando tipos e se tem dados nulos')
raw_data_qualis.info()
```

```
Verificando tipos e se tem dados nulos
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 131274 entries, 0 to 131273
Data columns (total 4 columns):
 #   Column             Non-Null Count   Dtype 
---  ------             --------------   ----- 
 0   ISSN               131271 non-null  object
 1   Título             131274 non-null  object
 2   Área de Avaliação  131274 non-null  object
 3   Estrato            131274 non-null  object
dtypes: object(4)
memory usage: 4.0+ MB
```

In [12]:

```
raw_data_qualis.nunique()
```

Out[12]:

```
ISSN                27618
Título              29838
Área de Avaliação      49
Estrato                 8
dtype: int64
```

In [13]:

```
raw_data_qualis.isnull().sum()
```

Out[13]:

```
ISSN                3
Título              0
Área de Avaliação   0
Estrato             0
dtype: int64
```

In [ ]:

```

```

3.5 Copiar dataframe para ajustes

In [14]:

```
df_scopus = raw_data_scopus.copy()
df_qualis = raw_data_qualis.copy()
```

In [15]:

```
## Verificar se os dataframes nao s"ao espelhos
print(id(df_scopus),id(df_qualis), id(raw_data_scopus),id(raw_data_qualis))
```

```
2504434480840 2504503299144 2504503065544 2504500494536
```

4 Dataframe SCOPUS

4.1 Ajustar de Colunas

4.1.1 Alterar nome de Colunas - SCOPUS

In [16]:

```
print('Verificar colunas')
df_scopus.columns
```

```
Verificar colunas
```

Out[16]:

```
Index(['Authors', 'Author(s) ID', 'Title', 'Year', 'Source title', 'Volume',
       'Issue', 'Art. No.', 'Page start', 'Page end', 'Page count', 'Cited b
y',
       'DOI', 'Link', 'Affiliations', 'Authors with affiliations', 'Abstrac
t',
       'Author Keywords', 'Index Keywords', 'Molecular Sequence Numbers',
       'Chemicals/CAS', 'Tradenames', 'Manufacturers', 'Funding Details',
       'Funding Text 1', 'Funding Text 2', 'Funding Text 3', 'References',
       'Correspondence Address', 'Editors', 'Sponsors', 'Publisher',
       'Conference name', 'Conference date', 'Conference location',
       'Conference code', 'ISSN', 'ISBN', 'CODEN', 'PubMed ID',
       'Language of Original Document', 'Abbreviated Source Title',
       'Document Type', 'Publication Stage', 'Open Access', 'Source', 'EI
D'],
      dtype='object')
```

In [17]:

```
columns_scopus = {
    'Authors':'autores',
    'Author(s) ID':'id_autores',
    'Title':'titulo_artigo',
    'Year':'ano',
    'Source title':'titulo_fonte',
    'Volume':'volume',
    'Issue':'publicado',
    'Art. No.':'numero_artigo',
    'Page start':'inicio_pagina',
    'Page end':'fim_pagina',
    'Page count':'quantidade_paginas',
    'Cited by':'quantidade_citacoes',
    'DOI':'doi',
    'Link':'link_scopus',
    'Affiliations':'afiliacoes',
    'Authors with affiliations':'autores_com_filiacoes',
    'Abstract':'resumo',
    'Author Keywords':'palavras_chaves_autor',
    'Index Keywords':'palavras_chave_index',
    'Molecular Sequence Numbers':'numeros_sequencia_molecular' ,
    'Chemicals/CAS':'chemicals_cas',
    'Tradenames':'nomes_comerciais',
    'Manufacturers':'fabricantes',
    'Funding Details':'detalhes_financiamento',
    'Funding Text 1':'texto_financiamento_1',
    'Funding Text 2':'texto_financiamento_2',
    'Funding Text 3':'texto_financiamento_3',
    'References':'referencias',
    'Correspondence Address':'endereco_correspondencia',
    'Editors':'editores',
    'Sponsors':'patrocinadores',
    'Publisher':'editor',
    'Conference name':'nome_conferencia',
    'Conference date':'data_conferencia',
    'Conference location':'local_conferencia',
    'Conference code':'codigo_conferencia',
    'ISSN':'issn_scopus',
    'ISBN':'isbn',
    'CODEN':'coden',
    'PubMed ID':'id_pubmed',
    'Language of Original Document':'idioma_original',
    'Abbreviated Source Title':'titulo_abreviado_fonte',
    'Document Type':'tipo_documento',
    'Publication Stage':'etapa_publicacao',
    'Open Access':'acesso_livre',
    'Source':'fonte',
    'EID':'eid',
}
```

In [18]:

```
df_scopus = df_scopus.rename(columns=columns_scopus)
df_scopus.head(1)
```

Out[18]:

	autores	id_autores	titulo_artigo	ano	titulo_fonte	volume	publicado	numero_artigo
0	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nucleic Acids Research	32	DATABASE ISS.	NaN

1 rows × 47 columns

4.1.2 Ajustar Colunas de Páginas

In [19]:

```
df_scopus["inicio_pagina_"] = df_scopus["inicio_pagina"]
df_scopus["fim_pagina_"] = df_scopus["fim_pagina"]
```

In [20]:

```
## Expressao que pega só dígitos
r = re.compile(r'\D')
```

In [21]:

```
df_scopus.loc[df_scopus['quantidade_citacoes'] == 1912] [['issn_scopus','inicio_pagina','
```

Out[21]:

	issn_scopus	inicio_pagina	fim_pagina
0	03051048	D267	D270

In [22]:

```
df_scopus.loc[[0,1,2,513], ['inicio_pagina','fim_pagina','quantidade_paginas']]
```

Out[22]:

	inicio_pagina	fim_pagina	quantidade_paginas
0	D267	D270	NaN
1	17	21	NaN
2	229	236	NaN
513	NaN	NaN	837.0

In [23]:

```
## Criar novas colunas de paginas e + uma para calcular qtde paginas
```

```
df_scopus_1 = df_scopus.copy()
print(id(df_scopus), id(df_scopus_1))
```

In [24]:

```
df_scopus.inicio_pagina_.replace(r, '', regex = True, inplace=True)
df_scopus.fim_pagina_.replace(r, '', regex = True, inplace=True)
```

In [25]:

```
df_scopus.loc[[0,1,2,513], ['inicio_pagina','fim_pagina','quantidade_paginas','inicio_pag
```

Out[25]:

	inicio_pagina	fim_pagina	quantidade_paginas	inicio_pagina_	fim_pagina_
0	D267	D270	NaN	267	270
1	17	21	NaN	17	21
2	229	236	NaN	229	236
513	NaN	NaN	837.0	NaN	NaN

In [26]:

```
print('Ajustar tipagem dos dados')
df_scopus['inicio_pagina_'] = df_scopus['inicio_pagina_'].astype('float64')
df_scopus['fim_pagina_'] = df_scopus['fim_pagina_'].astype('float64')
```

```
Ajustar tipagem dos dados
```

In [27]:

```
df_scopus['quantidade_paginas_'] = df_scopus['fim_pagina_'] - df_scopus['inicio_pagina_']
```

In [28]:

```
df_scopus.loc[[0,1,2,513], ['inicio_pagina','fim_pagina','quantidade_paginas','inicio_pag
```

Out[28]:

	inicio_pagina	fim_pagina	quantidade_paginas	inicio_pagina_	fim_pagina_	quantidade_pa
0	D267	D270	NaN	267.0	270.0	
1	17	21	NaN	17.0	21.0	
2	229	236	NaN	229.0	236.0	
513	NaN	NaN	837.0	NaN	NaN	

In [29]:

```
df_scopus.sort_values('quantidade_paginas_',ascending=False, )[['issn_scopus','quantidade
```

Out[29]:

	issn_scopus	quantidade_paginas_
410	01492918	8836.0
192	NaN	60.0
320	09534814	30.0
480	13673270	26.0
502	18650929	26.0
...	...	...
562	NaN	NaN
564	16130073	NaN
566	1942597X	NaN
568	1048776X	NaN
571	10829873	NaN

574 rows × 2 columns

In [30]:

```
## Pegar documentos com paginas nao nulas menor q 3 páginas
```

4.1.3 Criar nova coluna com data de conferencia

In [31]:

```
# tratar data - criar novo campo de data
df_scopus['data_conferencia']
```

Out[31]:

```
0                                                NaN
1                                                NaN
2                                                NaN
3                                                NaN
4                                                NaN
                          ...
569    21 September 2005 through 23 September 2005
570                                              NaN
571                                              NaN
572        13 October 1999 through 15 October 1999
573        7 November 1990 through 8 November 1990
Name: data_conferencia, Length: 574, dtype: object
```

4.2 Ajustar tipagem de dados

4.2.1 Convertendo as colunas de datas para o formato datetime

```
#Convertendo as colunas de datas para o formato datetime
colunas_datas = df_scopus.columns[df_scopus.columns.str.contains('DT_|dt\|data|DATA|Data',
regex=True)]
colunas_datas
```

```
#convertendo cada coluna de colunas_datas para o formato datetime
for coluna in colunas_datas:
    df_scopus[coluna] = pd.to_datetime(df_scopus[coluna], format='%Y-%m-%d')
```

4.2.2 Convertendo as colunas para categóricas

In [32]:

```
print('Ajustar tipagem dos dados')
df_scopus['fabricantes'] = df_scopus['fabricantes'].astype('category')
df_scopus['acesso_livre'] = df_scopus['acesso_livre'].astype('category')
```

```
Ajustar tipagem dos dados
```

4.3 Filtrar dataframe

In [111]:

```
df_scopus.head()
```

Out[111]:

	autores	id_autores	titulo_artigo	ano	titul
0	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	R
1	Aronson A.R.	17933416200;	Effective mapping of biomedical text to the UM...	2001	Proc / Sym
2	Aronson A.R., Lang F.-M.	17933416200;13612282900;	An overview of MetaMap: Historical perspective...	2010	J A Inf
3	Piñero J., Bravo A., Queralt-Rosinach N., Guti...	55220852900;56374942700;35766634600;5672880770...	DisGeNET: A comprehensive platform integrating...	2017	R
4	Nikfarjam A., Sarker A., O'Connor k., Ginn R.,...	36069663700;36976315000;56596185000;5659652430...	Pharmacovigilance from social media: Mining ad...	2015	J A Inf

5 rows × 50 columns

In [ ]:

```

```

4.4 Analisar dataframe tratado

In [33]:

```
print('Verificando tipos e se tem dados nulos')
df_scopus.info()
```

```
Verificando tipos e se tem dados nulos
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 574 entries, 0 to 573
Data columns (total 50 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   autores                       574 non-null    object
 1   id_autores                    574 non-null    object
 2   titulo_artigo                 574 non-null    object
 3   ano                           574 non-null    int64
 4   titulo_fonte                  574 non-null    object
 5   volume                        481 non-null    object
 6   publicado                     242 non-null    object
 7   numero_artigo                 154 non-null    object
 8   inicio_pagina                 420 non-null    object
 9   fim_pagina                    410 non-null    object
 10  quantidade_paginas            6 non-null      float64
 11  quantidade_citacoes           448 non-null    float64
 12  doi                           459 non-null    object
 13  link_scopus                   574 non-null    object
 14  afiliacoes                    567 non-null    object
 15  autores_com_filiacoes         571 non-null    object
 16  resumo                        574 non-null    object
 17  palavras_chaves_autor         396 non-null    object
 18  palavras_chave_index          520 non-null    object
 19  numeros_sequencia_molecular   1 non-null      object
 20  chemicals_cas                 108 non-null    object
 21  nomes_comerciais              20 non-null     object
 22  fabricantes                   2 non-null      category
 23  detalhes_financiamento        240 non-null    object
 24  texto_financiamento_1         188 non-null    object
 25  texto_financiamento_2         18 non-null     object
 26  texto_financiamento_3         2 non-null      object
 27  referencias                   521 non-null    object
 28  endereco_correspondencia      457 non-null    object
 29  editores                      115 non-null    object
 30  patrocinadores                75 non-null     object
 31  editor                        419 non-null    object
 32  nome_conferencia              202 non-null    object
 33  data_conferencia              202 non-null    object
 34  local_conferencia             65 non-null     object
 35  codigo_conferencia            193 non-null    float64
 36  issn_scopus                   493 non-null    object
 37  isbn                          153 non-null    object
 38  coden                         207 non-null    object
 39  id_pubmed                     314 non-null    float64
 40  idioma_original               574 non-null    object
 41  titulo_abreviado_fonte        574 non-null    object
 42  tipo_documento                574 non-null    object
 43  etapa_publicacao              574 non-null    object
 44  acesso_livre                  245 non-null    category
 45  fonte                         574 non-null    object
 46  eid                           574 non-null    object
 47  inicio_pagina_                420 non-null    float64
 48  fim_pagina_                   410 non-null    float64
 49  quantidade_paginas_           410 non-null    float64
```

```
dtypes: category(2), float64(7), int64(1), object(40)
memory usage: 217.0+ KB
```

In [34]:

```
### analisar estes números máximos e mínimos ?????????
print('Resumo Estatístico de Campos Numéricos')
df_scopus.describe()
```

```
Resumo Estatístico de Campos Numéricos
```

Out[34]:

	ano	quantidade_paginas	quantidade_citacoes	codigo_conferencia	id_pubmed
count	574.000000	6.000000	448.000000	193.000000	3.140000e+02
mean	2014.632404	158.666667	28.962054	126570.849741	2.599832e+07
std	4.584182	334.274538	119.388752	39969.995470	5.110147e+06
min	1991.000000	2.000000	1.000000	14685.000000	8.947691e+06
25%	2012.000000	2.750000	3.000000	102305.000000	2.295241e+07
50%	2015.000000	8.000000	8.000000	117981.000000	2.637784e+07
75%	2018.000000	74.000000	23.000000	140679.000000	3.024264e+07
max	2021.000000	837.000000	1912.000000	253829.000000	3.393644e+07

In [35]:

```
# ver qtd no excel =NÚM.CARACT(O2)
print('Coluna com maior qtde de caracteres')
df_scopus['titulo_artigo'].apply(str).map(len).max()
```

```
Coluna com maior qtde de caracteres
```

Out[35]:

```
319
```

In [36]:

```
# analisar alguns campos
```

```
574 entries - scopus
81 sem issn
493 entradas
```

In [37]:

```
## tirar os espaços das colunas strings a serem trabalhadas
```

4.5 Visualizações

4.5.1 Analisando tipos de acessos dos arquivos

In [38]:

```
## pegar os 10 + patrocinadores
df_scopus.acesso_livre.unique()
```

Out[38]:

```
['All Open Access, Bronze, Green', NaN, 'All Open Access, Gold, Green', 'All
Open Access, Hybrid Gold, Green', 'All Open Access, Green', 'All Open Acces
s, Bronze', 'All Open Access, Hybrid Gold', 'All Open Access, Gold']
Categories (7, object): ['All Open Access, Bronze, Green', 'All Open Access,
Gold, Green', 'All Open Access, Hybrid Gold, Green', 'All Open Access, Gree
n', 'All Open Access, Bronze', 'All Open Access, Hybrid Gold', 'All Open Acc
ess, Gold']
```

In [39]:

```
# Groupby by
acesso_livre = df_scopus.groupby("acesso_livre")

# Summary statistic of all
acesso_livre.describe().head()
```

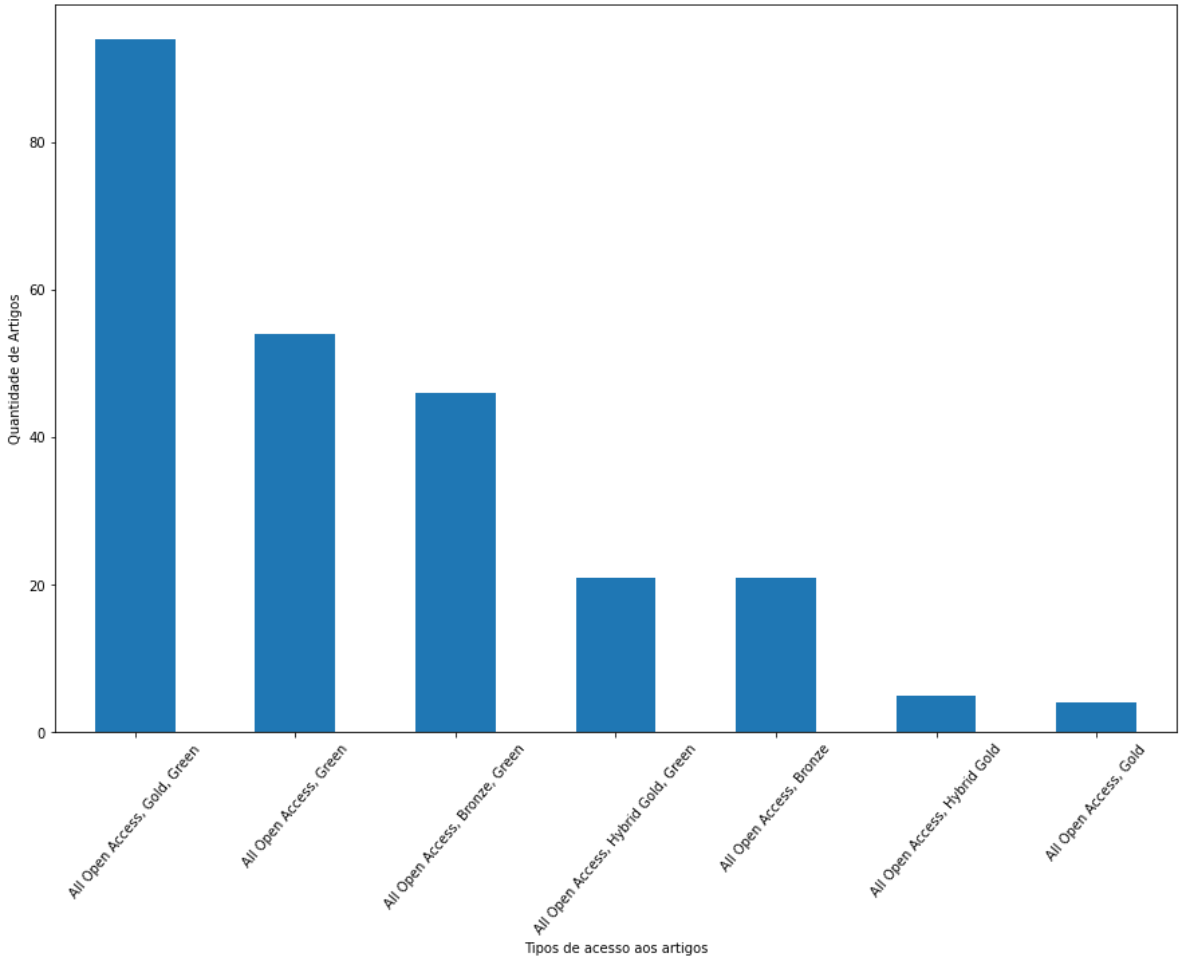
Out[39]:

	ano								quantidade
	count	mean	std	min	25%	50%	75%	max	count
acesso_livre									
All Open Access, Bronze	21.0	2016.380952	2.060975	2014.0	2015.0	2016.0	2019.00	2019.0	0.0
All Open Access, Bronze, Green	46.0	2013.630435	3.573723	2003.0	2012.0	2014.0	2017.00	2019.0	0.0
All Open Access, Gold	4.0	2019.500000	0.577350	2019.0	2019.0	2019.5	2020.00	2020.0	0.0
All Open Access, Gold, Green	94.0	2016.244681	3.339842	2006.0	2014.0	2017.0	2019.00	2021.0	0.0
All Open Access, Green	54.0	2013.925926	4.321302	2005.0	2010.0	2013.5	2017.75	2021.0	0.0

5 rows × 64 columns

In [40]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
acesso_livre.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Tipos de acesso aos artigos")
plt.ylabel("Quantidade de Artigos")
plt.show()
```



4.5.2 Analisando tipos de documentos

In [41]:

```
# Groupby by
tipo_documento = df_scopus.groupby("tipo_documento")

# Summary statistic of all
tipo_documento.describe().head()
```

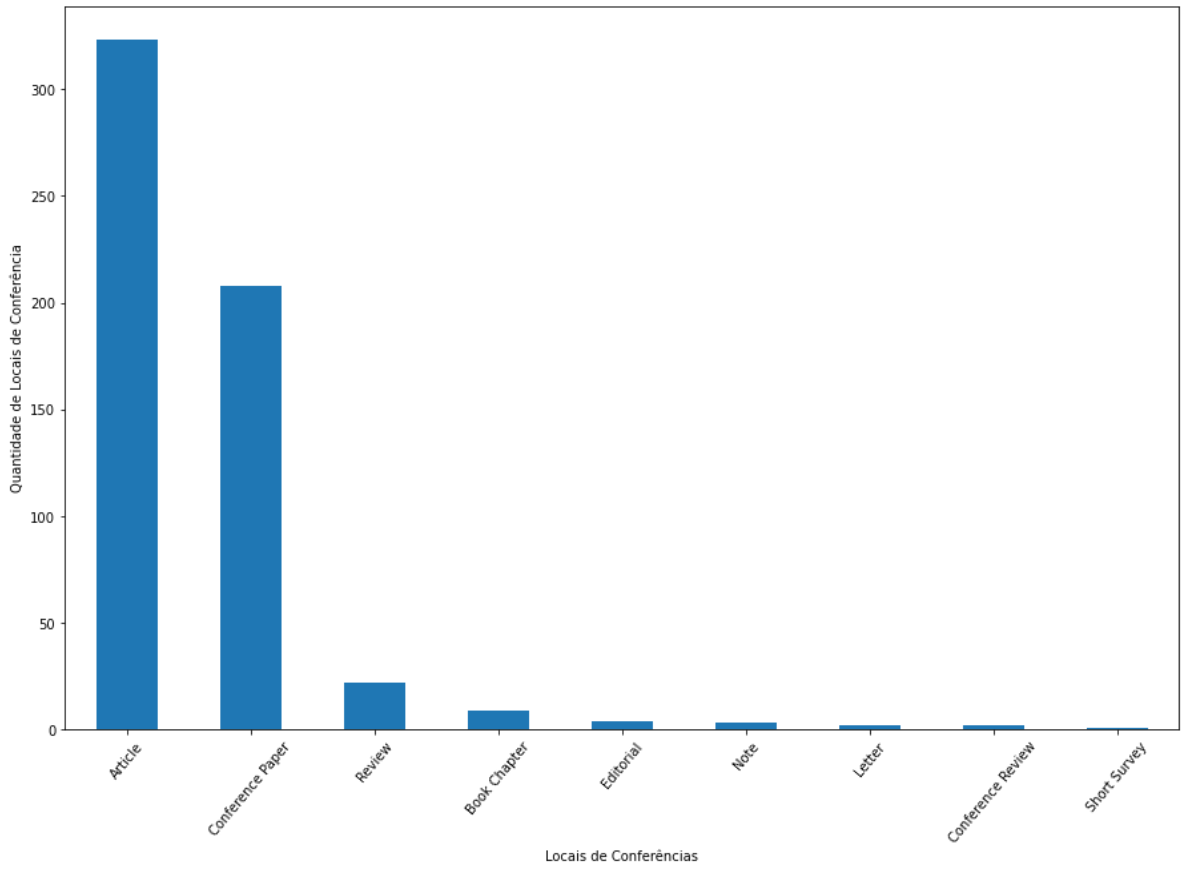
Out[41]:

								ano	quanti
	count	mean	std	min	25%	50%	75%	max	co
tipo_documento									
Article	323.0	2014.770898	4.659541	1996.0	2012.00	2016.0	2018.00	2021.0	
Book Chapter	9.0	2016.444444	3.811532	2009.0	2014.00	2017.0	2019.00	2021.0	
Conference Paper	208.0	2014.096154	4.509827	1991.0	2012.00	2014.0	2018.00	2021.0	
Conference Review	2.0	2014.500000	3.535534	2012.0	2013.25	2014.5	2015.75	2017.0	
Editorial	4.0	2016.500000	3.696846	2012.0	2014.25	2017.0	2019.25	2020.0	

5 rows × 64 columns

In [42]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
tipo_documento.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Locais de Conferências")
plt.ylabel("Quantidade de Locais de Conferência")
plt.show()
```



In [ ]:

In [ ]:

4.5.3 Analisando Locais de Conferencia

In [43]:

```
# Groupby by
local_conferencia = df_scopus.groupby("local_conferencia")

# Summary statistic of all
local_conferencia.describe().head()
```

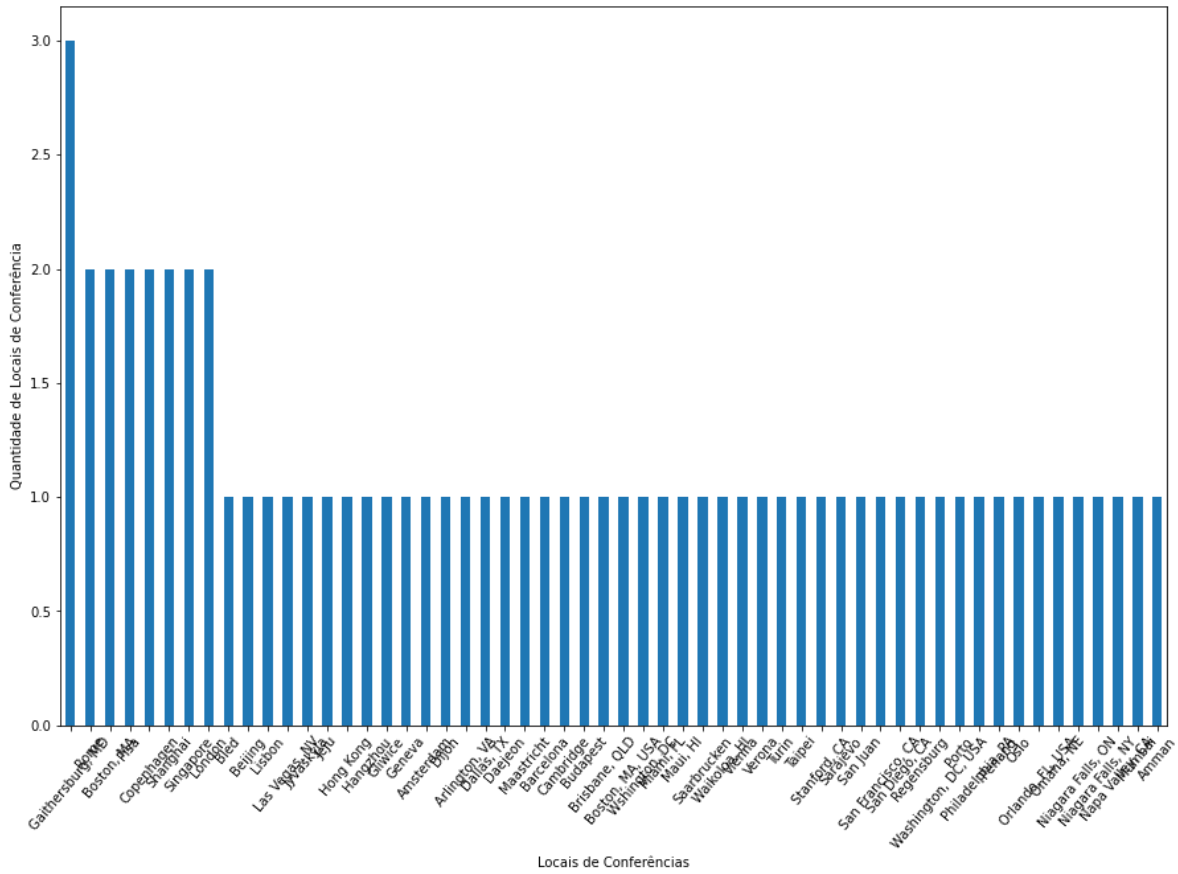
Out[43]:

	ano								quantidade_pagin	
	count	mean	std	min	25%	50%	75%	max	count	me
local_conferencia										
Amman	1.0	2011.0	NaN	2011.0	2011.0	2011.0	2011.0	2011.0	0.0	N
Amsterdam	1.0	2007.0	NaN	2007.0	2007.0	2007.0	2007.0	2007.0	0.0	N
Arlington, VA	1.0	2010.0	NaN	2010.0	2010.0	2010.0	2010.0	2010.0	0.0	N
Barcelona	1.0	2013.0	NaN	2013.0	2013.0	2013.0	2013.0	2013.0	0.0	N
Beijing	1.0	2012.0	NaN	2012.0	2012.0	2012.0	2012.0	2012.0	0.0	N

5 rows × 64 columns

In [44]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
local_conferencia.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Locais de Conferências")
plt.ylabel("Quantidade de Locais de Conferência")
plt.show()
```



In [ ]:

```
```

4.5.4 Analisando Editores

In [45]:

```
# Groupby by
editor = df_scopus.groupby("editor")

# Summary statistic of all
editor.describe().head()
```

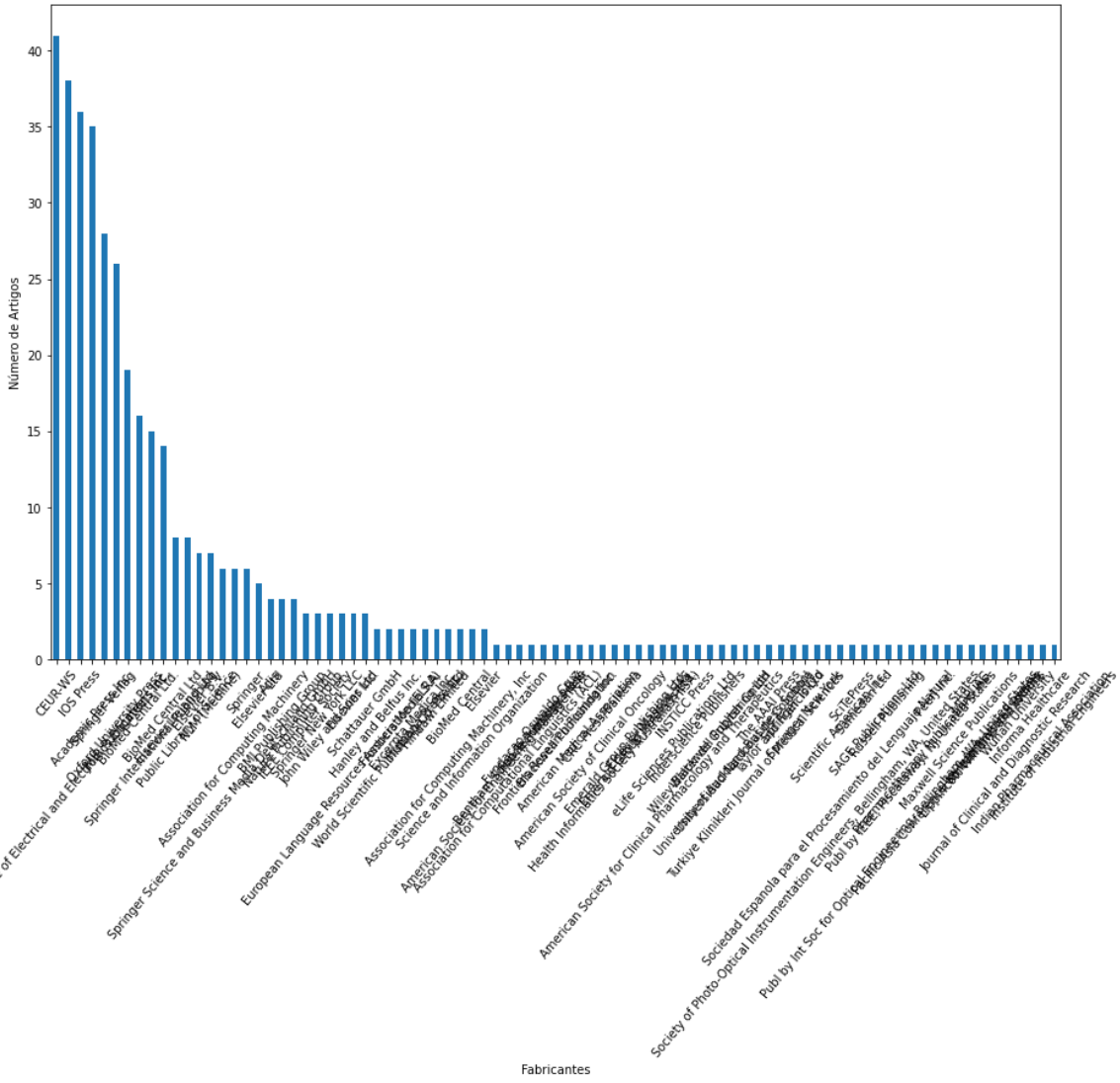
Out[45]:

	ano								quantidad
	count	mean	std	min	25%	50%	75%	max	count
editor									
Academic Press Inc.	35.0	2016.857143	2.116363	2014.0	2015.00	2017.0	2019.0	2021.0	0.0
Adis	4.0	2020.250000	0.957427	2019.0	2019.75	2020.5	2021.0	2021.0	0.0
American Medical Association	1.0	2019.000000	NaN	2019.0	2019.00	2019.0	2019.0	2019.0	0.0
American Society for Clinical Pharmacology and Therapeutics	1.0	2018.000000	NaN	2018.0	2018.00	2018.0	2018.0	2018.0	0.0
American Society for Engineering Management	1.0	2019.000000	NaN	2019.0	2019.00	2019.0	2019.0	2019.0	0.0

5 rows × 64 columns

In [46]:

```
plt.figure(figsize=(15,10))
editor.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Fabricantes")
plt.ylabel("Número de Artigos")
plt.show()
```



In [ ]:

```
```

4.5.5 Analisando artigos por Ano

In [47]:

```
# Groupby by
ano = df_scopus.groupby("ano")

# Summary statistic of all
ano.describe().head()
```

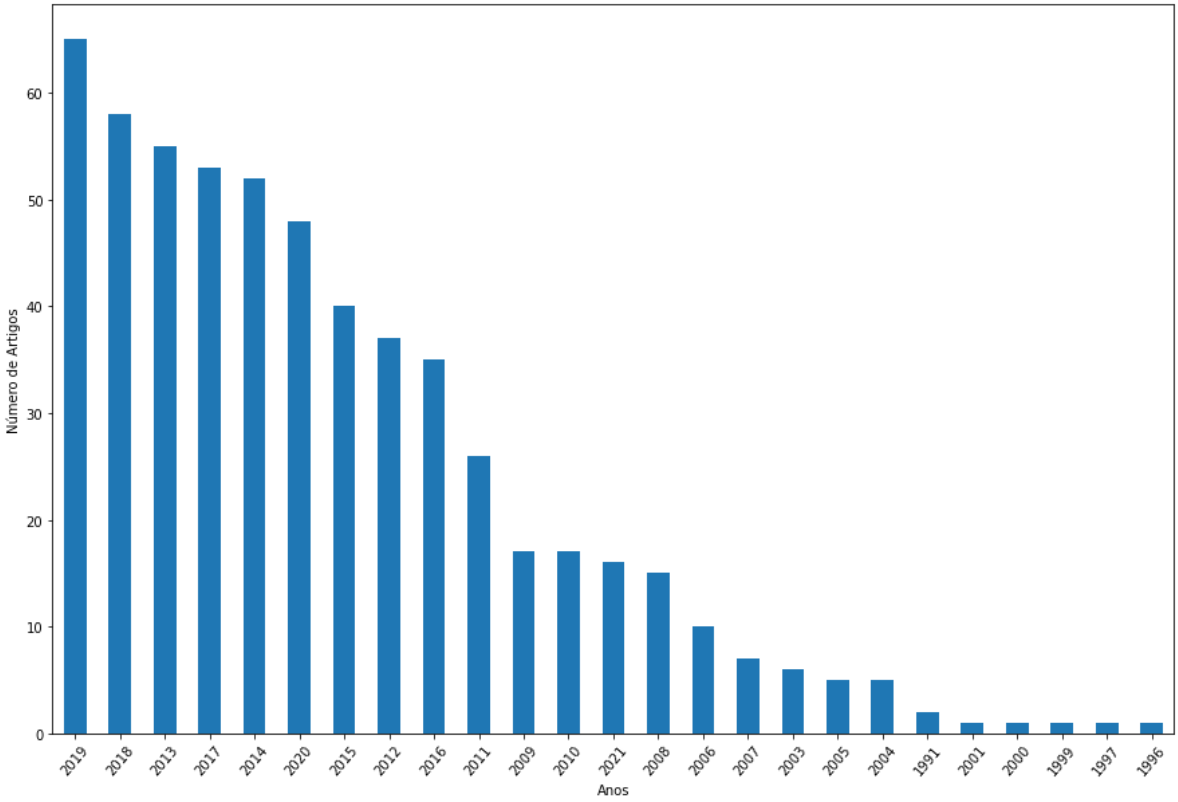
Out[47]:

	quantidade_paginas								quantidade_citacoes		...	fim_pagina_	
	count	mean	std	min	25%	50%	75%	max	count	mean	...	75%	max
ano													
1991	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	1.0	...	809.0	1047.0
1996	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	20.0	...	377.0	377.0
1997	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	69.0	...	489.0	489.0
1999	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	1.0	...	312.0	312.0
2000	0.0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	0.0	NaN	...	182.0	182.0

5 rows × 56 columns

In [48]:

```
plt.figure(figsize=(15,10))
ano.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Anos")
plt.ylabel("Número de Artigos")
plt.show()
```



4.5.6 Analisando os idiomas

In [49]:

```
# Groupby by
idioma = df_scopus.groupby("idioma_original")

# Summary statistic of all
idioma.describe().head()
```

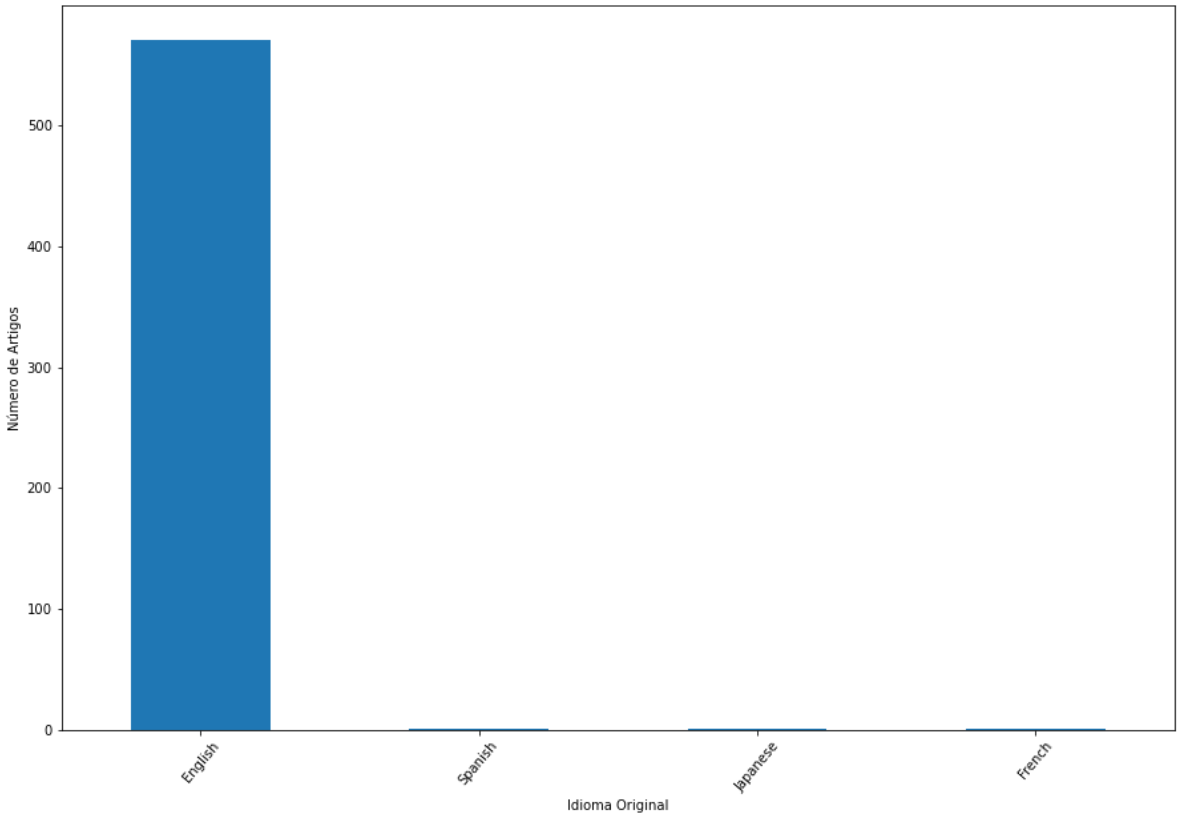
Out[49]:

								ano	quantidac
	count	mean	std	min	25%	50%	75%	max	count
idioma_original									
English	571.0	2014.633975	4.591468	1991.0	2012.0	2015.0	2018.0	2021.0	6.0
French	1.0	2011.000000	NaN	2011.0	2011.0	2011.0	2011.0	2011.0	0.0
Japanese	1.0	2014.000000	NaN	2014.0	2014.0	2014.0	2014.0	2014.0	0.0
Spanish	1.0	2018.000000	NaN	2018.0	2018.0	2018.0	2018.0	2018.0	0.0

4 rows × 64 columns

In [50]:

```
plt.figure(figsize=(15,10))
idioma.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Idioma Original")
plt.ylabel("Número de Artigos")
plt.show()
```



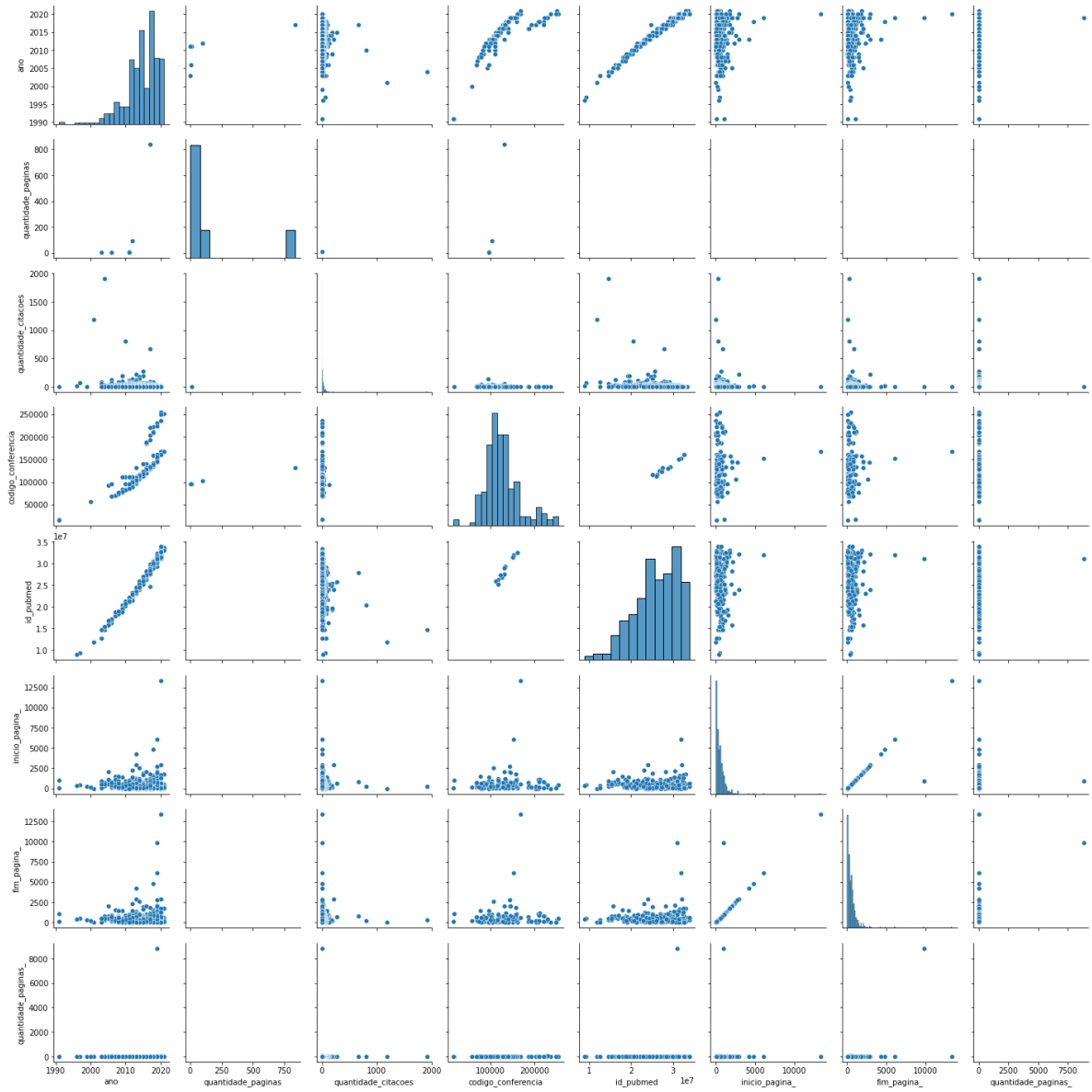
In [51]:

```
print('Visão geral em gráfico')
sns.pairplot(df_scopus)
```

```
Visão geral em gráfico
```

Out[51]:

```
<seaborn.axisgrid.PairGrid at 0x2471f344f88>
```



In [52]:

```
## qtde de artigos financiados e nao financiados - criar coluna
```

In [53]:

```
## qtde de artigos patrociandores e nao patrociandores - criar coluna
```

In [54]:

```
## criar regex para identificar emails da coluna editores
```

In [55]:

```
## qtde de artigos por editores - unicos
```

In [56]:

```
## qtde de artigos por local de conferencia - unicos
```

In [57]:

```
## qtde de artigos por idioma_original - unicos
```

In [58]:

```
## qtde de artigos por tipo de documentos - unicos
```

4.6 Nuvem de Palavras

4.6.1 Nuvem de Palavras dos Títulos

In [59]:

```
import nltk
from nltk.corpus import stopwords

stopwords = stopwords.words('english')

print(stopwords)

# apend outras palavras
```

```
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you'r
e", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves',
'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'i
t', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselve
s', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'tho
se', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has',
'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'bu
t', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for',
'with', 'about', 'against', 'between', 'into', 'through', 'during', 'befor
e', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'o
n', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'the
re', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'mo
re', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'sa
me', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "d
on't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y',
'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "d
oesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "is
n't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 's
han', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "were
n't", 'won', "won't", 'wouldn', "wouldn't"]
```

In [60]:

```
print('Carregar novo dataframe de palavras apenas com dados da coluna texto, ou seja, uma
#df_scopus['issn_scopus'] = df_scopus.issn_scopus.str.upper()
palavras_titulo = df_scopus['titulo_artigo'].str.lower()
palavras_titulo
```

```
Carregar novo dataframe de palavras apenas com dados da coluna texto, ou sej
a, uma série.
```

Out[60]:

```
0      the unified medical language system (umls): in...
1      effective mapping of biomedical text to the um...
2      an overview of metamap: historical perspective...
3      disgenet: a comprehensive platform integrating...
4      pharmacovigilance from social media: mining ad...
                             ...
569    ub at clef 2005: bilingual clir and medical im...
570    automated indexing of the hazardous substances...
571                                              metamap
572    multivistm: a web-based interactive remote vis...
573    new method for identifying features of an imag...
Name: titulo_artigo, Length: 574, dtype: object
```

In [61]:

```
print(f'O objeto palavras é do tipo {type(palavras_titulo)} e tem o shape de {palavras_ti
```

```
O objeto palavras é do tipo <class 'pandas.core.series.Series'> e tem o shap
e de (574,)
```

In [62]:

```
print(f'Criando Nuvem de Palavras sem tratamento para ter ideia do corpus.')
# Variável recebe conteúdo do dataframe palavras concatenando cada conteúdo do texto sepa
wordcloud_palavras_titulo = " ".join(s for s in palavras_titulo)
#wordcloud_palavras
```

```
Criando Nuvem de Palavras sem tratamento para ter ideia do corpus.
```

In [63]:

```
print(f'Quantidade de palavras no corpus: {len(wordcloud_palavras_titulo)} e seu tipo atu
```

```
Quantidade de palavras no corpus: 53282 e seu tipo atual é<class 'str'>
```

In [64]:

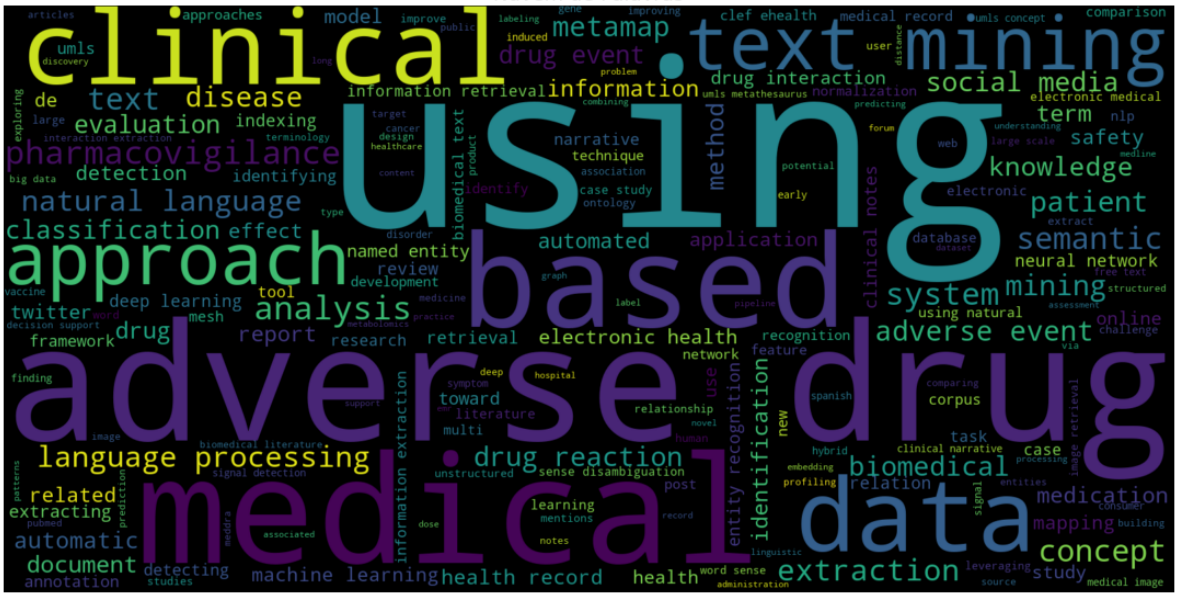
```
print(f'Criação da nuvem de palavras sem tratamento nos textos')
wordcloud_titulo = WordCloud(stopwords=stopwords,
                      background_color='black', width=1600,
                      height=800).generate(wordcloud_palavras_titulo)
```

```
Criação da nuvem de palavras sem tratamento nos textos
```

In [65]:

```
#Gráfico Nuvem de Palavras
# Gerando o grafico
# Variáveis do gráfico
path_image = '../image/'

titulo = 'Nuvem de Palavras'
#eixo_x = ''
#eixo_y = ''
image = path_image+'wordcloud'
extensao_arquivo = '.pdf'

#Gráfico
fig, ax = plt.subplots(figsize=(16,8))
ax.imshow(wordcloud_titulo, interpolation='bilinear')
ax.set_axis_off()

#Legendas
#ax.legend(title='Legenda', loc=4, fontsize=9)
ax.set_title(titulo, fontsize=18)
#ax.set_xlabel(eixo_x, fontsize=9)
#ax.set_ylabel(eixo_y, fontsize=9)

#salvar imagens
plt.savefig(image+extensao_arquivo) #, format='pdf', dpi=300, transparent=True)
plt.tight_layout()
plt.show()
```

Nuvem de Palavras



4.6.2 Nuvem de Palavras dos Resumos

In [66]:

```
print('Carregar novo dataframe de palavras apenas com dados da coluna texto, ou seja, uma
palavras_resumo = df_scopus['resumo'].str.lower()
palavras_resumo
```

```
Carregar novo dataframe de palavras apenas com dados da coluna texto, ou sej
a, uma série.
```

Out[66]:

```
0      the unified medical language system (http://um...
1      the umls metathesaurus, the largest thesaurus ...
2      metamap is a widely available program providin...
3      the information about the genetic basis of hum...
4      objective social media is becoming increasingl...
                           ...
569    this paper presents the results of the state u...
570    the hazardous substances data bank (hsdb), a f...
571                              [no abstract available]
572    the evolution of hypermedia imagemap technolog...
573    the metamap process extends the concept of dir...
Name: resumo, Length: 574, dtype: object
```

In [67]:

```
print(f'Criando Nuvem de Palavras sem tratamento para ter ideia do corpus.')
# Variável recebe conteúdo do dataframe palavras concatenando cada conteúdo do texto sepa
wordcloud_palavras_resumo = " ".join(s for s in palavras_resumo)
#wordcloud_palavras
```

```
Criando Nuvem de Palavras sem tratamento para ter ideia do corpus.
```

In [68]:

```
print(f'Quantidade de palavras no corpus: {len(wordcloud_palavras_resumo)} e seu tipo atu
```

```
Quantidade de palavras no corpus: 799380 e seu tipo atual é<class 'str'>
```

In [69]:

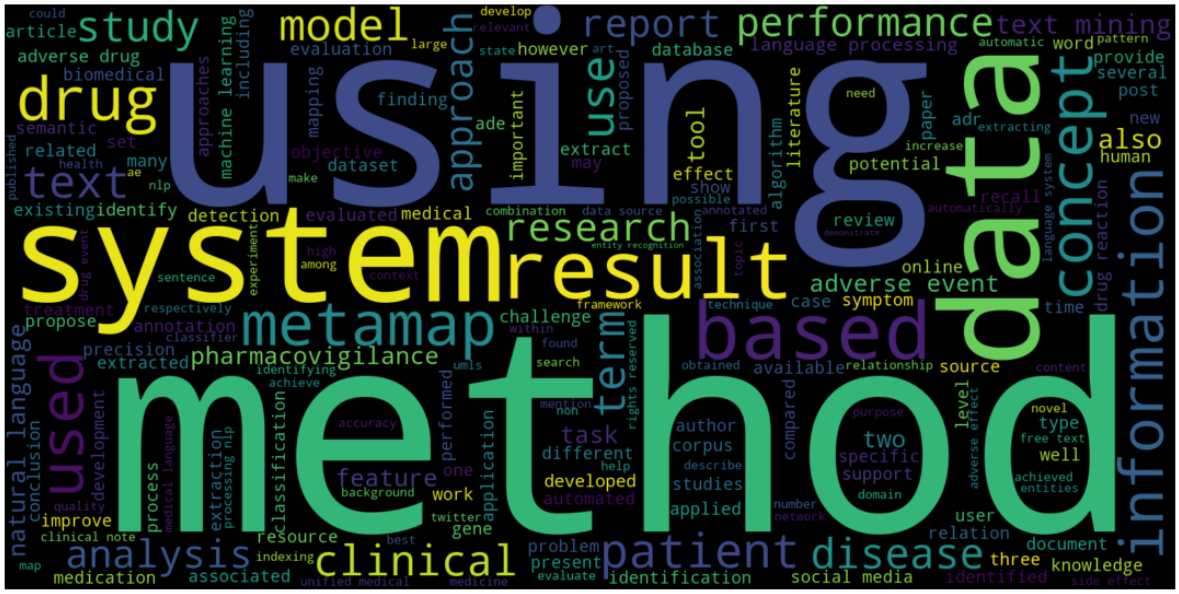
```
print(f'Criação da nuvem de palavras sem tratamento nos textos')
wordcloud_resumo = WordCloud(stopwords=stopwords,
                      background_color='black', width=1600,
                      height=800).generate(wordcloud_palavras_resumo)
```

```
Criação da nuvem de palavras sem tratamento nos textos
```

In [70]:

```
#Gráfico Nuvem de Palavras
path_image = '../image/'

titulo = 'Nuvem de Palavras'
#eixo_x = ''
#eixo_y = ''
image = path_image+'wordcloud'
extensao_arquivo = '.pdf'

#Gráfico
fig, ax = plt.subplots(figsize=(16,8))
ax.imshow(wordcloud_resumo, interpolation='bilinear')
ax.set_axis_off()

#Legendas
#ax.legend(title='Legenda', loc=4, fontsize=9)
ax.set_title(titulo, fontsize=18)
#ax.set_xlabel(eixo_x, fontsize=9)
#ax.set_ylabel(eixo_y, fontsize=9)

#salvar imagens
plt.savefig(image+extensao_arquivo) #, format='pdf', dpi=300, transparent=True)
plt.tight_layout()
plt.show()
```

Nuvem de Palavras



4.6.3 Nuvem de Palavras do Conteúdo dos textos

5 Dataframe Qualis

5.1 Ajustar de Colunas

5.1.1 Alterar nome de colunas

In [71]:

```
print('Verificar colunas')
df_qualis.columns
```

```
Verificar colunas
```

Out[71]:

```
Index(['ISSN', 'Título', 'Área de Avaliação', 'Estrato'], dtype='object')
```

In [72]:

```
columns_qualis = {
    'ISSN': 'issn_qualis',
    'Título':'titulo_periodico',
    'Área de Avaliação':'area_avaliacao',
    'Estrato':'estrato',
}
```

In [73]:

```
df_qualis = df_qualis.rename(columns=columns_qualis)
df_qualis.head(1)
```

Out[73]:

	issn_qualis	titulo_periodico	area_avaliacao	estrato
0	1981-030X	19&20 (RIO DE JANEIRO)	ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS ...	C

5.2 Ajustar tipagem de dados

5.2.1 Converter colunas para categórica

In [74]:

```
print('Ajustar tipagem dos dados')
df_qualis['estrato'] = df_qualis['estrato'].astype('category')
```

```
Ajustar tipagem dos dados
```

5.2.2 Filtrar base Qualis

In [75]:

```
print(df_qualis['area_avaliacao'].unique())
```

```
['ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS CONTÁBEIS E TURISMO'
 'ANTROPOLOGIA / ARQUEOLOGIA                                '
 'ARQUITETURA, URBANISMO E DESIGN' 'ARTES'
 'ASTRONOMIA / FÍSICA                                ' 'BIODIVERSIDADE'
 'BIOTECNOLOGIA                                      '
 'CIÊNCIA DA COMPUTAÇÃO                              '
 'CIÊNCIA DE ALIMENTOS                               '
 'CIÊNCIA POLÍTICA E RELAÇÕES INTERNACIONAIS         '
 'CIÊNCIAS AGRÁRIAS I                                '
 'CIÊNCIAS AMBIENTAIS'
 'CIÊNCIAS BIOLÓGICAS I                              '
 'CIÊNCIAS BIOLÓGICAS II                             '
 'CIÊNCIAS BIOLÓGICAS III                            '
 'CIÊNCIAS DA RELIGIÃO E TEOLOGIA' 'COMUNICAÇÃO E INFORMAÇÃO'
 'DIREITO                                            '
 'ECONOMIA                                           '
 'EDUCAÇÃO                                           '
 'EDUCAÇÃO FÍSICA                                    '
 'ENFERMAGEM                                         '
 'ENGENHARIAS I                                      '
 'ENGENHARIAS II                                     '
 'ENGENHARIAS III                                    '
 'ENGENHARIAS IV                                     ' 'ENSINO'
 'FARMÁCIA                                           ' 'FILOSOFIA'
 'GEOCIÊNCIAS                                        '
 'GEOGRAFIA                                          '
 'HISTÓRIA                                           '
 'INTERDISCIPLINAR                                   '
 'LINGUíSTICA E LITERATURA'
 'MATEMÁTICA / PROBABILIDADE E ESTATÍSTICA           '
 'MATERIAIS                                          '
 'MEDICINA I                                         '
 'MEDICINA II                                        '
 'MEDICINA III                                       '
 'MEDICINA VETERINÁRIA                               ' 'NUTRIÇÃO'
 'ODONTOLOGIA                                        '
 'PLANEJAMENTO URBANO E REGIONAL / DEMOGRAFIA        '
 'PSICOLOGIA                                         '
 'QUÍMICA                                            '
 'SAÚDE COLETIVA                                     '
 'SERVIÇO SOCIAL                                     '
 'SOCIOLOGIA                                         '
 'ZOOTECNIA / RECURSOS PESQUEIROS                    ']
```

5.2.3 Remover as areas abaixo

In [76]:

```
#%%timeit
df_remove = df_qualis.loc[
    (df_qualis['area_avaliacao'] == 'ADMINISTRAÇÃO PÚBLICA E DE EMPRESAS, CIÊNCIAS CONTÁB
    (df_qualis['area_avaliacao'] == 'ARTES') |
    (df_qualis['area_avaliacao'] == 'ANTROPOLOGIA / ARQUEOLOGIA                         ')
    (df_qualis['area_avaliacao'] == 'ARQUITETURA, URBANISMO E DESIGN') |
    (df_qualis['area_avaliacao'] == 'ASTRONOMIA / FÍSICA                                ')
    (df_qualis['area_avaliacao'] == 'BIODIVERSIDADE') |
    (df_qualis['area_avaliacao'] == 'CIÊNCIAS AGRÁRIAS I                                ')
    (df_qualis['area_avaliacao'] == 'CIÊNCIA DE ALIMENTOS                               ')
    (df_qualis['area_avaliacao'] == 'CIÊNCIA POLÍTICA E RELAÇÕES INTERNACIONAIS         ')
    (df_qualis['area_avaliacao'] == 'CIÊNCIAS AGRÁRIAS I                                ')
    (df_qualis['area_avaliacao'] == 'CIÊNCIAS AMBIENTAIS') |
    (df_qualis['area_avaliacao'] == 'CIÊNCIAS DA RELIGIÃO E TEOLOGIA') |
    (df_qualis['area_avaliacao'] == 'COMUNICAÇÃO E INFORMAÇÃO') |
    (df_qualis['area_avaliacao'] == 'DIREITO                                            ')
    (df_qualis['area_avaliacao'] == 'ECONOMIA                                           ')
    (df_qualis['area_avaliacao'] == 'EDUCAÇÃO                                           ')
    (df_qualis['area_avaliacao'] == 'ENSINO') |
    (df_qualis['area_avaliacao'] == 'EDUCAÇÃO FÍSICA                                    ')
    (df_qualis['area_avaliacao'] == 'ENGENHARIAS I                                      ')
    (df_qualis['area_avaliacao'] == 'ENGENHARIAS II                                     ')
    (df_qualis['area_avaliacao'] == 'ENGENHARIAS III                                    ')
    (df_qualis['area_avaliacao'] == 'ENGENHARIAS IV                                     '
    (df_qualis['area_avaliacao'] == 'GEOCIÊNCIAS                                        '
    (df_qualis['area_avaliacao'] == 'FILOSOFIA') |
    (df_qualis['area_avaliacao'] == 'GEOGRAFIA                                          ')
    (df_qualis['area_avaliacao'] == 'HISTÓRIA                                           ')
    (df_qualis['area_avaliacao'] == 'INTERDISCIPLINAR                                   ')
    (df_qualis['area_avaliacao'] == 'LINGUíSTICA E LITERATURA') |
    (df_qualis['area_avaliacao'] == 'MATERIAIS                                          ')
    (df_qualis['area_avaliacao'] ==  'MEDICINA VETERINÁRIA                              '
    (df_qualis['area_avaliacao'] ==  'NUTRIÇÃO') |
    (df_qualis['area_avaliacao'] == 'PLANEJAMENTO URBANO E REGIONAL / DEMOGRAFIA        ')
    (df_qualis['area_avaliacao'] == 'SERVIÇO SOCIAL                                     ')
    (df_qualis['area_avaliacao'] == 'SOCIOLOGIA                                         ')
    (df_qualis['area_avaliacao'] ==  'ZOOTECNIA / RECURSOS PESQUEIROS                   '

]

df_qualis_filtrado = df_qualis.drop(df_remove.index)
df_qualis_filtrado
```

Out[76]:

	issn_qualis	titulo_periodico	area_avaliacao	estrato
12853	2328-0662	# ISOJ JOURNAL	BIOTECNOLOGIA ...	C
12854	2190-5738	3 BIOTECH	BIOTECNOLOGIA ...	B3
12855	0101-9163	A HORA VETERINÁRIA	BIOTECNOLOGIA ...	C
12856	1232-1966	AAEM. ANNALS OF AGRICULTURAL AND ENVIRONMENTAL...	BIOTECNOLOGIA ...	B3
12857	1530-9932	AAPS PHARMSCITECH	BIOTECNOLOGIA ...	B1

	issn_qualis	titulo_periodico	area_avaliacao	estrato
...	...	...	...	...
126668	1696-3202	ÁTOPOS - SALUD MENTAL, COMUNIDAD Y CULTURA	SAÚDE COLETIVA ...	B5
126669	2316-4360	ÉLISÉE - REVISTA DE GEOGRAFIA DA UEG	SAÚDE COLETIVA ...	B5
126670	1929-7017	ÉTHIQUE PUBLIQUE - REVUE INTERNATIONALE D'ÉTHI...	SAÚDE COLETIVA ...	C
126671	1415-899X	ÚLTIMO ANDAR (PUCSP. IMPRESSO)	SAÚDE COLETIVA ...	B5
126672	1980-8305	ÚLTIMO ANDAR (PUCSP. ONLINE)	SAÚDE COLETIVA ...	B5

47063 rows × 4 columns

5.3 Analisar dataframe tratado

In [77]:

```
print('Verificando tipos e se tem dados nulos')
df_qualis_filtrado.info()
```

```
Verificando tipos e se tem dados nulos
<class 'pandas.core.frame.DataFrame'>
Int64Index: 47063 entries, 12853 to 126672
Data columns (total 4 columns):
 #   Column            Non-Null Count  Dtype   
---  ------            --------------  -----   
 0   issn_qualis       47063 non-null  object  
 1   titulo_periodico  47063 non-null  object  
 2   area_avaliacao    47063 non-null  object  
 3   estrato           47063 non-null  category
dtypes: category(1), object(3)
memory usage: 1.5+ MB
```

In [78]:

```
print('Resumo Estatístico de Campos Numéricos')
df_qualis_filtrado.describe()
```

```
Resumo Estatístico de Campos Numéricos
```

Out[78]:

	issn_qualis	titulo_periodico	area_avaliacao	estrato
count	47063	47063	47063	47063
unique	14975	15603	15	8
top	1932-6203	PLOS ONE	MEDICINA I ...	B1
freq	60	57	5181	9020

In [79]:

```
# ver qtd no excel =NÚM.CARACT(O2)
print('Coluna com maior qtde de caracteres')
df_qualis_filtrado['titulo_periodico'].apply(str).map(len).max()
```

```
Coluna com maior qtde de caracteres
```

Out[79]:

```
254
```

5.4 Visualizações Qualis

5.4.1 Anáilise Area de Avaliação

In [80]:

```
df_qualis_filtrado.columns
```

Out[80]:

```
Index(['issn_qualis', 'titulo_periodico', 'area_avaliacao', 'estrato'], dtyp
e='object')
```

```
#colocar appos definir este df
## pegar os 10 + patrocinadores
df_bases.area_avaliacao.unique()
```

In [81]:

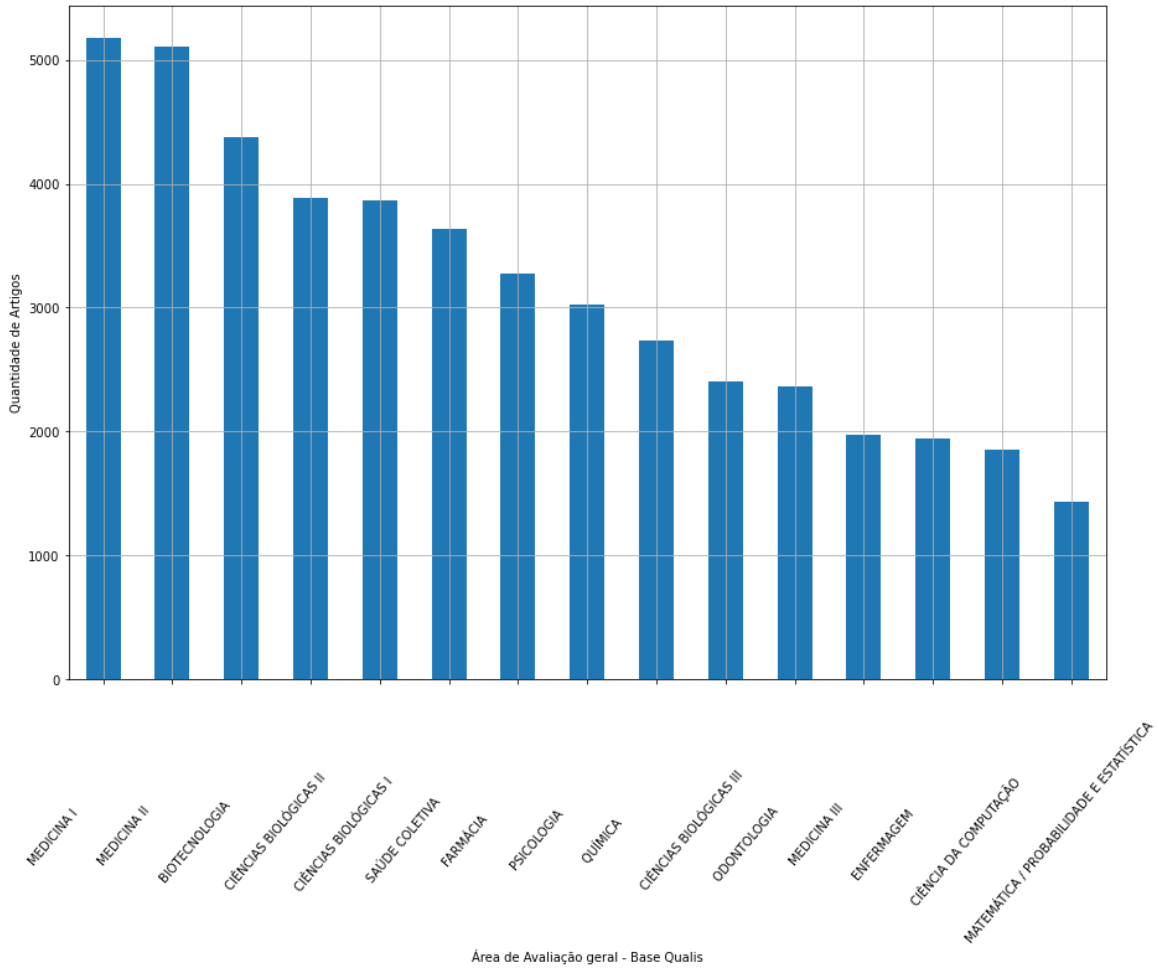
```
# Groupby by
area_avaliacao_all = df_qualis_filtrado.groupby("area_avaliacao")

# Summary statistic of all
area_avaliacao_all.describe().head()
```

Out[81]:

	issn_qualis				titulo_periodico					u
	count	unique	top	freq	count	unique	top	freq	count	
area_avaliacao										
BIOTECNOLOGIA	4376	4137	0100-4042	5	4376	4211	PLOS ONE	4	4376	
CIÊNCIA DA COMPUTAÇÃO	1850	1775	1932-6203	3	1850	1794	REVISTA DE SISTEMAS E COMPUTAÇÃO - RSC	3	1850	
CIÊNCIAS BIOLÓGICAS I	3870	3677	1932-6203	4	3870	3750	PLOS ONE	4	3870	
CIÊNCIAS BIOLÓGICAS II	3889	3681	1932-6203	5	3889	3760	TRENDS IN PSYCHIATRY AND PSYCHOTHERAPY	4	3889	
CIÊNCIAS BIOLÓGICAS III	2403	2276	1932-6203	8	2403	2330	PLOS ONE	7	2403	

In [82]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
area_avaliacao_all.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Área de Avaliação geral - Base Qualis")
plt.ylabel("Quantidade de Artigos")
plt.grid()
plt.show()
```



In [ ]:

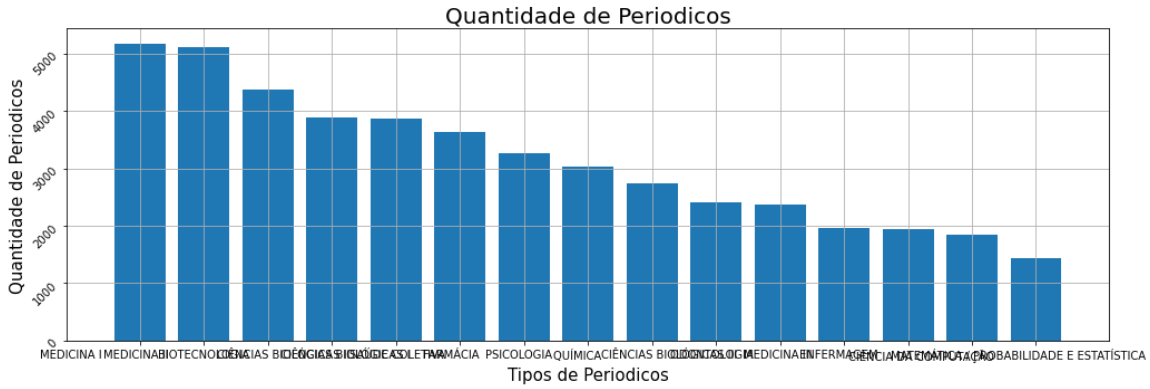
```
## muito lento
# Groupby by
titulo_periodico = df_qualis_filtrado.groupby("titulo_periodico")

# Summary statistic of all
titulo_periodico.describe().head()
```

```
# pegar as 10 +
plt.figure(figsize=(15,10))
titulo_periodico.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Título dos Periódicos")
plt.ylabel("Quantidade de Periodicos")
plt.show()
```

In [ ]:

In [83]:

```
### Gerando gráfico Tipo de Extensões

#variáveis do gráfico
titulo = 'Quantidade de Periodicos'
eixo_x = 'Tipos de Periodicos'
eixo_y = 'Quantidade de Periodicos'
data_hora = datetime.datetime.now()
path = os.path.join('..','image')
img_extensao = '.png'
image = ''

# Matplotlib
fig, ax = plt.subplots(figsize=(15,5))
plt.bar(df_qualis_filtrado['area_avaliacao'].value_counts().index, df_qualis_filtrado['ar

#Legendas e eixos
#ax.legend(title='Legenda', loc=4, fontsize=20)
ax.set_title(titulo, fontsize=20)
ax.set_xlabel(eixo_x, fontsize=15)
ax.set_ylabel(eixo_y, fontsize=15)
plt.yticks(rotation=45) ## agrupar datas por meses, trimestre ou ano, etc

plt.grid()

#Salvar imagem
#plt.savefig(path+image, dpi=300)
plt.tight_layout()
plt.show()

plt.show()
```



6 Dataframe PubMEd

In [ ]:

```

```

7 Dataframe CAPES

In [ ]:

```

```

8 Unir / Merge de dataframes

In [84]:

```
#df_qualis["issn_qualis_ajustado"] =
df_qualis_filtrado["issn_qualis"].replace('-','',regex=True, inplace=True)
```

In [85]:

```
df_qualis_filtrado["issn_qualis"]
```

Out[85]:

```
12853     23280662
12854     21905738
12855     01019163
12856     12321966
12857     15309932
            ...
126668    16963202
126669    23164360
126670    19297017
126671    1415899X
126672    19808305
Name: issn_qualis, Length: 47063, dtype: object
```

In [86]:

```
df_scopus['issn_scopus'] = df_scopus.issn_scopus.str.upper()
df_qualis_filtrado['issn_qualis'] = df_qualis_filtrado.issn_qualis.str.upper()
```

In [87]:

```
df_scopus['issn_scopus']
```

Out[87]:

```
0      03051048
1      1531605X
2      10675027
3      03051048
4      10675027
         ...
569    03029743
570    15508390
571    10829873
572    0277786X
573    0277786X
Name: issn_scopus, Length: 574, dtype: object
```

In [88]:

```
df_qualis_filtrado['issn_qualis']
```

Out[88]:

```
12853     23280662
12854     21905738
12855     01019163
12856     12321966
12857     15309932
            ...
126668    16963202
126669    23164360
126670    19297017
126671    1415899X
126672    19808305
Name: issn_qualis, Length: 47063, dtype: object
```

In [ ]:

```

```

```
## criar um dataframe q une o links_regulamentos_fundos + allLines, ligados pelo nome do
arquivo
df_bases = pd.merge(left=df_scopus, right=df_qualis_filtrado, left_on='issn_scopus',
right_on='issn_qualis', suffixes=["_df_scopus", "_df_qualis"])
df_bases
```

In [89]:

```
## criar um dataframe q une o links_regulamentos_fundos + allLines, ligados pelo nome do
df_bases = df_scopus.merge(df_qualis_filtrado, left_on='issn_scopus', right_on='issn_qual
df_bases
```

Out[89]:

	autores	id_autores	titulo_artigo	ano	titulo_fo
0	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nuc Ac Resea
1	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nuc Ac Resea
2	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nuc Ac Resea
3	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nuc Ac Resea
4	Bodenreider O.	6603893164;	The Unified Medical Language System (UMLS): In...	2004	Nuc Ac Resea
...	...	...	...	...	
2331	Doyle M., Klein G., Hussaini F., Pescitelli M.	57197355375;57199985005;6603483180;6508147369;	MultiVISTM: A Web-based interactive remote vis...	2000	Proceedi of SP Internatic Soci
2332	Doyle Michael	36804804600;	New method for identifying features of an imag...	1991	Proceedi of SP Internatic Soci
2333	Doyle Michael	36804804600;	New method for identifying features of an imag...	1991	Proceedi of SP Internatic Soci
2334	Doyle Michael	36804804600;	New method for identifying features of an imag...	1991	Proceedi of SP Internatic Soci
2335	Doyle Michael	36804804600;	New method for identifying features of an imag...	1991	Proceedi of SP Internatic Soci

2336 rows × 55 columns

In [90]:

```
#df_scopus.merge(df_qualis, left_on='issn_scopus', right_on='issn_qualis', how='left', va
```

In [91]:

```
df_bases[['autores','titulo_artigo','area_avaliacao','issn_scopus','issn_qualis','estrato
```

Out[91]:

	autores	titulo_artigo	area_avaliacao	issn_scopus	issn_qualis	estrato
0	Bodenreider O.	The Unified Medical Language System (UMLS): In...	BIOTECNOLOGIA ...	03051048	03051048	A1
1	Bodenreider O.	The Unified Medical Language System (UMLS): In...	CIÊNCIA DA COMPUTAÇÃO ...	03051048	03051048	B2
2	Bodenreider O.	The Unified Medical Language System (UMLS): In...	CIÊNCIAS BIOLÓGICAS I ...	03051048	03051048	A1
3	Bodenreider O.	The Unified Medical Language System (UMLS): In...	CIÊNCIAS BIOLÓGICAS II ...	03051048	03051048	A1
4	Bodenreider O.	The Unified Medical Language System (UMLS): In...	CIÊNCIAS BIOLÓGICAS III ...	03051048	03051048	A1
...	...	...	...	...	...	...
2331	Doyle M., Klein G., Hussaini F., Pescitelli M.	MultiVISTM: A Web-based interactive remote vis...	QUÍMICA ...	0277786X	0277786X	C
2332	Doyle Michael	New method for identifying features of an imag...	MEDICINA II ...	0277786X	0277786X	B4
2333	Doyle Michael	New method for identifying features of an imag...	MEDICINA III ...	0277786X	0277786X	B4
2334	Doyle Michael	New method for identifying features of an imag...	ODONTOLOGIA ...	0277786X	0277786X	B4
2335	Doyle Michael	New method for identifying features of an imag...	QUÍMICA ...	0277786X	0277786X	C

2336 rows × 6 columns

In [92]:

```
print(type(df_bases['_merge'].unique()))
```

```
<class 'pandas.core.arrays.categorical.Categorical'>
```

In [93]:

```
print(df_bases['_merge'].unique())
```

```
['both', 'left_only']
Categories (2, object): ['both', 'left_only']
```

In [94]:

```
df_bases.nunique()
```

Out[94]:

```
autores                          549
id_autores                       546
titulo_artigo                    572
ano                               25
titulo_fonte                     220
volume                           208
publicado                         39
numero_artigo                    147
inicio_pagina                    340
fim_pagina                       342
quantidade_paginas                 5
quantidade_citacoes               81
doi                              459
link_scopus                      574
afiliacoes                       552
autores_com_filiacoes            567
resumo                           565
palavras_chaves_autor            394
palavras_chave_index             520
numeros_sequencia_molecular        1
chemicals_cas                     94
nomes_comerciais                  16
fabricantes                        2
detalhes_financiamento           228
texto_financiamento_1            187
texto_financiamento_2             18
texto_financiamento_3              2
referencias                      521
endereco_correspondencia         448
editores                          77
patrocinadores                    64
editor                            85
nome_conferencia                 157
data_conferencia                 155
local_conferencia                 56
codigo_conferencia               150
issn_scopus                      150
isbn                             134
coden                             60
id_pubmed                        314
idioma_original                    4
titulo_abreviado_fonte           213
tipo_documento                     9
etapa_publicacao                   2
acesso_livre                       7
fonte                              1
eid                              574
inicio_pagina_                   340
fim_pagina_                      338
quantidade_paginas_               26
issn_qualis                      104
titulo_periodico                 111
area_avaliacao                    15
estrato                            8
```

```
_merge
```

```
2
```

In [95]:

```
df_bases.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2336 entries, 0 to 2335
Data columns (total 55 columns):
 #   Column                          Non-Null Count  Dtype
---  ------                          --------------  -----
 0   autores                         2336 non-null   object
 1   id_autores                      2336 non-null   object
 2   titulo_artigo                   2336 non-null   object
 3   ano                             2336 non-null   int64
 4   titulo_fonte                    2336 non-null   object
 5   volume                          2238 non-null   object
 6   publicado                       1419 non-null   object
 7   numero_artigo                   1147 non-null   object
 8   inicio_pagina                   1187 non-null   object
 9   fim_pagina                      1157 non-null   object
 10  quantidade_paginas              10 non-null     float64
 11  quantidade_citacoes             1960 non-null   float64
 12  doi                             2191 non-null   object
 13  link_scopus                     2336 non-null   object
 14  afiliacoes                      2317 non-null   object
 15  autores_com_filiacoes           2325 non-null   object
 16  resumo                          2336 non-null   object
 17  palavras_chaves_autor           1269 non-null   object
 18  palavras_chave_index            2239 non-null   object
 19  numeros_sequencia_molecular     3 non-null      object
 20  chemicals_cas                   824 non-null    object
 21  nomes_comerciais                118 non-null    object
 22  fabricantes                     13 non-null     category
 23  detalhes_financiamento          1174 non-null   object
 24  texto_financiamento_1           873 non-null    object
 25  texto_financiamento_2           71 non-null     object
 26  texto_financiamento_3           16 non-null     object
 27  referencias                     2245 non-null   object
 28  endereco_correspondencia        2115 non-null   object
 29  editores                        280 non-null    object
 30  patrocinadores                  129 non-null    object
 31  editor                          1790 non-null   object
 32  nome_conferencia                473 non-null    object
 33  data_conferencia                473 non-null    object
 34  local_conferencia               182 non-null    object
 35  codigo_conferencia              428 non-null    float64
 36  issn_scopus                     2255 non-null   object
 37  isbn                            421 non-null    object
 38  coden                           1267 non-null   object
 39  id_pubmed                       1780 non-null   float64
 40  idioma_original                 2336 non-null   object
 41  titulo_abreviado_fonte          2336 non-null   object
 42  tipo_documento                  2336 non-null   object
 43  etapa_publicacao                2336 non-null   object
 44  acesso_livre                    1444 non-null   category
 45  fonte                           2336 non-null   object
 46  eid                             2336 non-null   object
 47  inicio_pagina_                  1187 non-null   float64
 48  fim_pagina_                     1157 non-null   float64
 49  quantidade_paginas_             1157 non-null   float64
 50  issn_qualis                     2163 non-null   object
 51  titulo_periodico                2163 non-null   object
```

```
 52  area_avaliacao               2163 non-null   object
 53  estrato                      2163 non-null   category
 54  _merge                       2336 non-null   category
dtypes: category(4), float64(7), int64(1), object(43)
memory usage: 959.1+ KB
```

In [ ]:

```
```

In [96]:

```
# Criar colunas
df_bases['leitura_Resumo']=None
df_bases['leitura_Conclusao']=None
df_bases['leitura_completa']=None
```

In [ ]:

```
```

8.1 Visualizações Base Geral

8.1.1 Analisando Nomes Comerciais

In [97]:

```
#Colocar + para baixo após criar este df

# Groupby by
nomes_comerciais = df_bases.groupby("nomes_comerciais")

# Summary statistic of all
nomes_comerciais.describe().head()
```

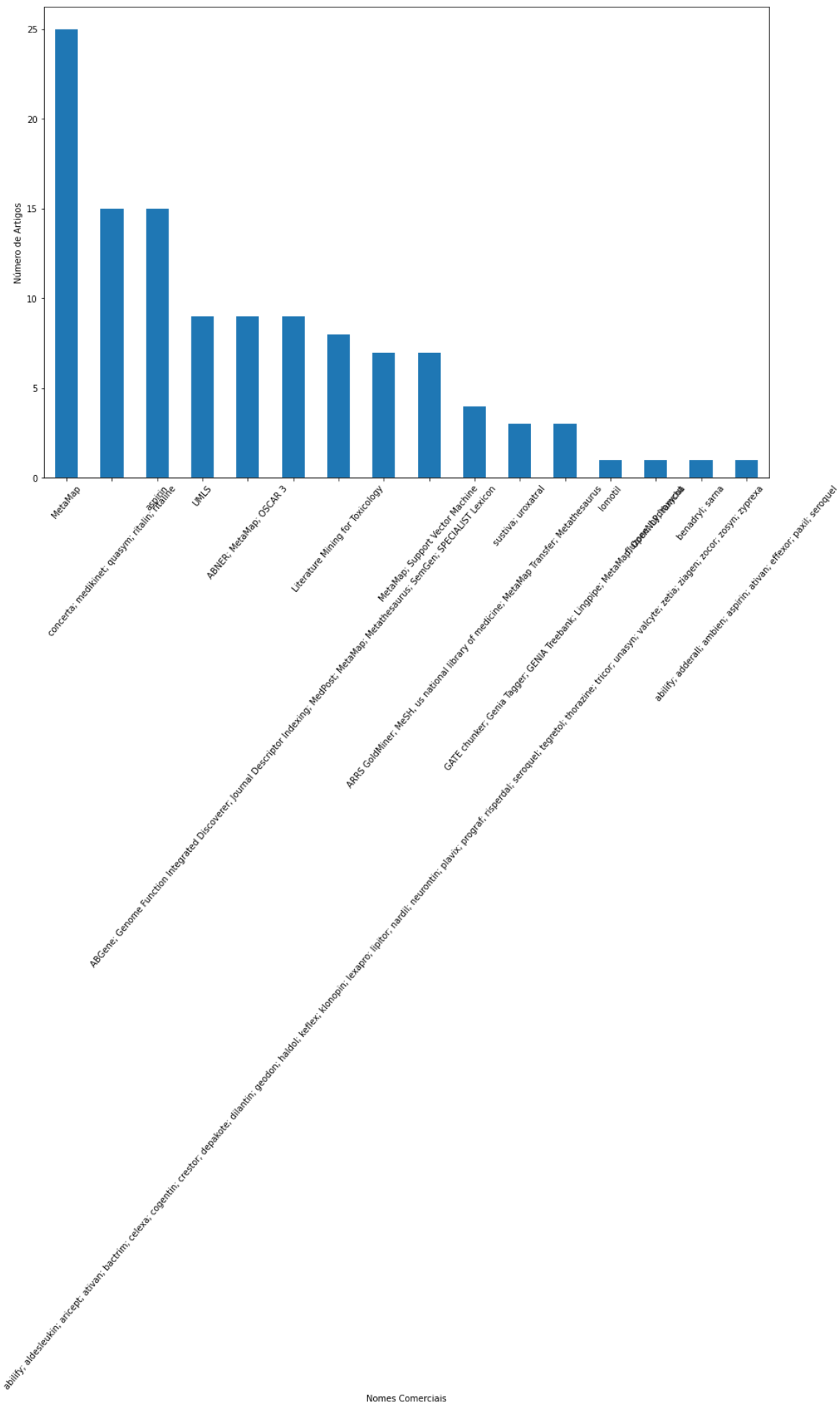
Out[97]:

	ano								quantidade_pagin	
	count	mean	std	min	25%	50%	75%	max	count	me
nomes_comerciais										
ABGene; Genome Function Integrated Discoverer; Journal Descriptor Indexing; MedPost; MetaMap; Metathesaurus; SemGen; SPECIALIST Lexicon	9.0	2006.0	0.0	2006.0	2006.0	2006.0	2006.0	2006.0	0.0	N
ABNER; MetaMap; OSCAR 3	9.0	2009.0	0.0	2009.0	2009.0	2009.0	2009.0	2009.0	0.0	N
ARRS GoldMiner; MeSH, us national library of medicine; MetaMap Transfer; Metathesaurus	4.0	2008.0	0.0	2008.0	2008.0	2008.0	2008.0	2008.0	0.0	N
GATE chunker; Genia Tagger; GENIA Treebank; Lingpipe; MetaMap; OpenNLP; Yamcha	3.0	2011.0	0.0	2011.0	2011.0	2011.0	2011.0	2011.0	0.0	N
Literature Mining for Toxicology	8.0	2017.0	0.0	2017.0	2017.0	2017.0	2017.0	2017.0	0.0	N

5 rows × 64 columns

In [98]:

```
plt.figure(figsize=(15,10))
nomes_comerciais.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Nomes Comerciais")
plt.ylabel("Número de Artigos")
plt.show()
```


Número de Artigos
0
5
10
15
20
25
MetaMap
concerta; medikinet; quasym; ritalin; ritalin
UMLS
ABNER, MetaMap; OSCAR 3
Literature Mining for Toxicology
MetaMap; Support Vector Machine
sustiva; uroxatral
lomotil
benadryl; sarna
abilify; adderall; ambien; aspirin; ativan; effexor; paxil; seroquel
Nomes Comerciais

In [ ]:

In [99]:

```
# Groupby by
area_avaliacao = df_bases.groupby("area_avaliacao")

# Summary statistic of all
area_avaliacao.describe().head()
```

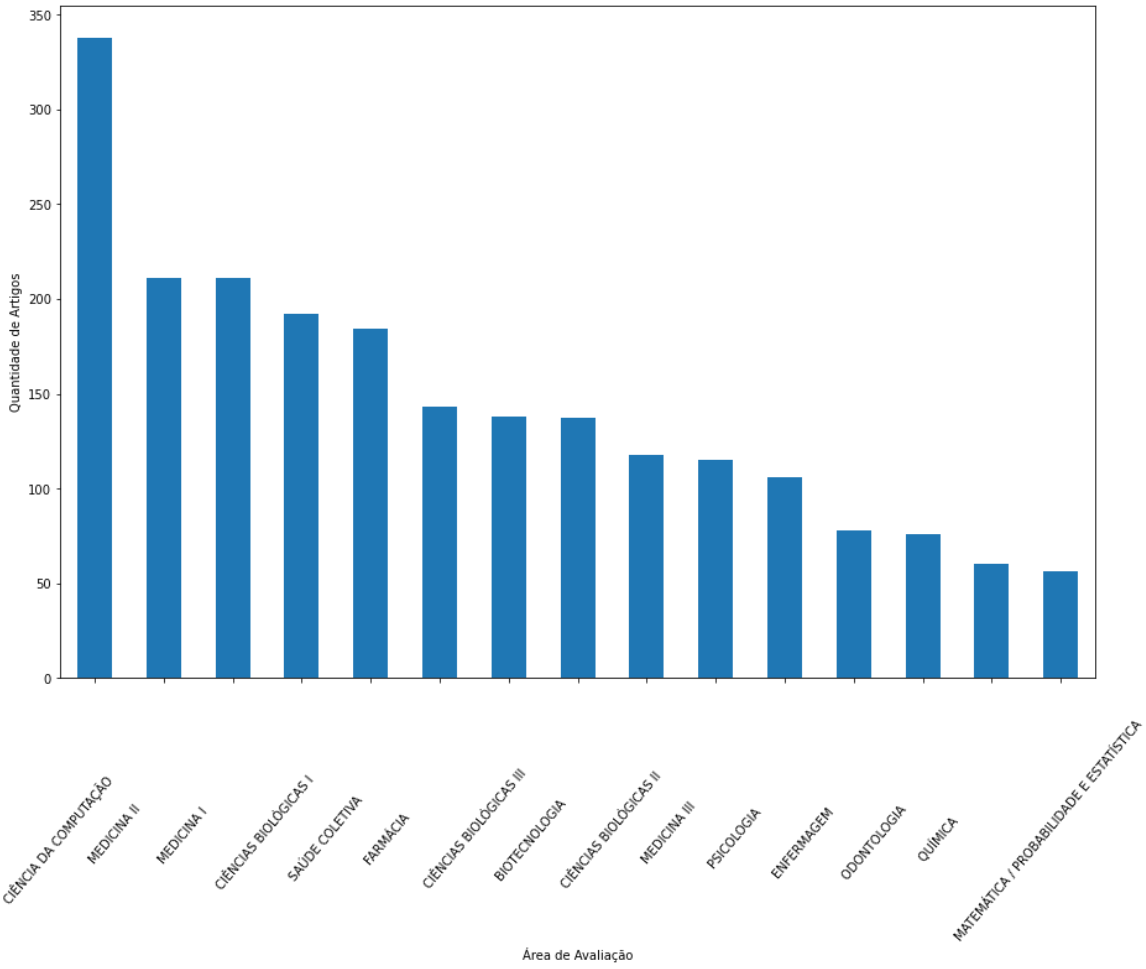
Out[99]:

								ano	quant
	count	mean	std	min	25%	50%	75%	max	count
area_avaliacao									
BIOTECNOLOGIA	137.0	2015.043796	4.061781	2004.0	2013.00	2016.0	2018.0	2021.0	0.0
CIÊNCIA DA COMPUTAÇÃO	338.0	2014.920118	4.136701	1997.0	2013.00	2015.0	2018.0	2021.0	3.0
CIÊNCIAS BIOLÓGICAS I	192.0	2015.218750	3.876910	2004.0	2013.00	2016.0	2018.0	2021.0	0.0
CIÊNCIAS BIOLÓGICAS II	118.0	2015.898305	3.944898	2004.0	2013.25	2017.0	2019.0	2021.0	0.0
CIÊNCIAS BIOLÓGICAS III	138.0	2016.021739	3.796786	2004.0	2014.00	2017.0	2019.0	2021.0	0.0

5 rows × 64 columns

In [100]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
area_avaliacao.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Área de Avaliação")
plt.ylabel("Quantidade de Artigos")
plt.show()
```



In [ ]:

8.1.2 Análise de Titulo Periodico

In [101]:

```
# Groupby by
titulo_periodico = df_bases.groupby("titulo_periodico")

# Summary statistic of all
titulo_periodico.describe().head()
```

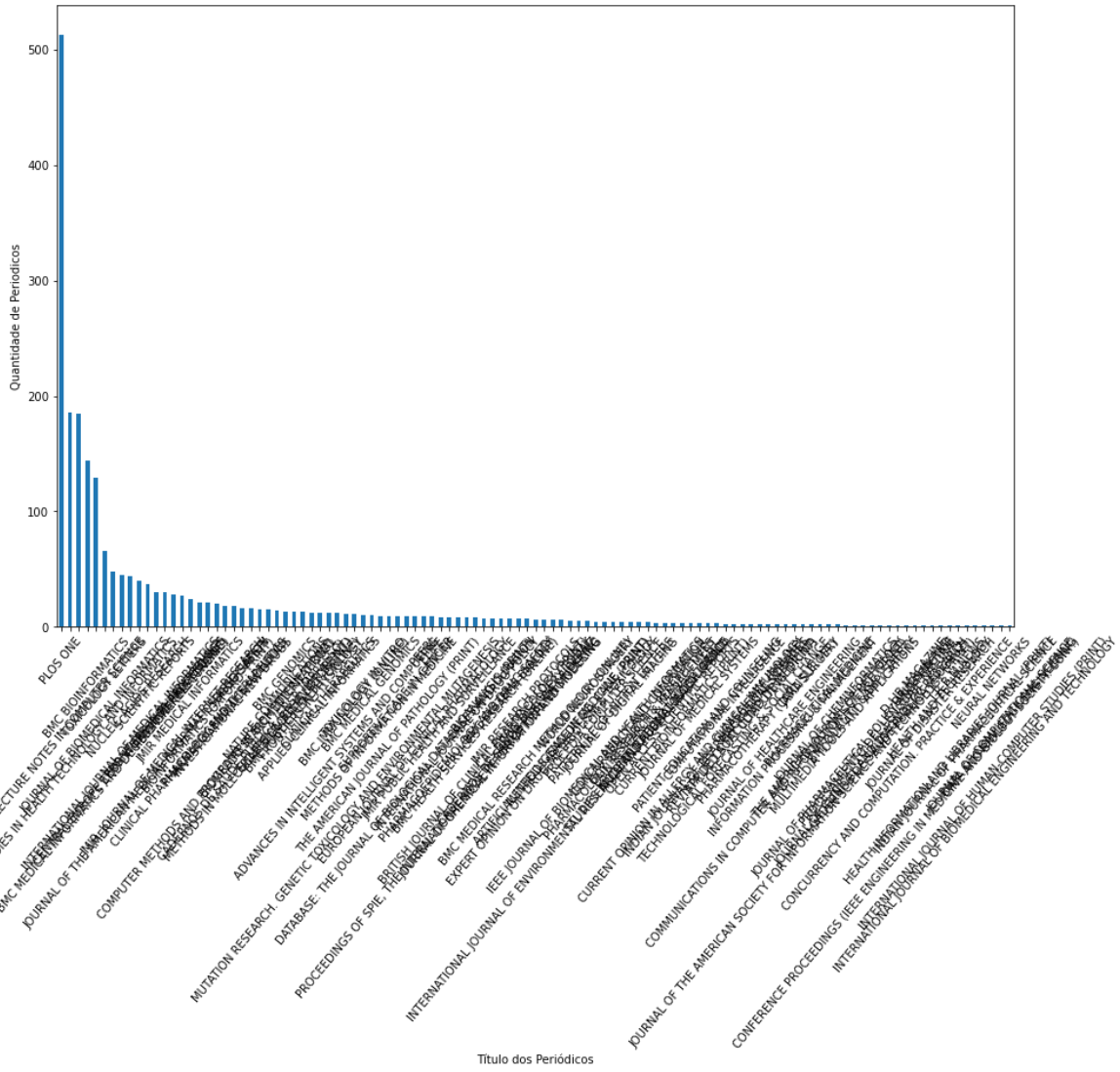
Out[101]:

	ano								quantidade_pag	
	count	mean	std	min	25%	50%	75%	max	count	
titulo_periodico										
ADVANCES IN INTELLIGENT SYSTEMS AND COMPUTING	12.0	2019.0	1.206045	2018.0	2018.0	2018.5	2020.0	2021.0	0.0	
APPLIED CLINICAL INFORMATICS	12.0	2015.0	2.088932	2013.0	2013.0	2015.0	2017.0	2017.0	0.0	
APPLIED SOFT COMPUTING (PRINT)	2.0	2019.0	0.000000	2019.0	2019.0	2019.0	2019.0	2019.0	0.0	
ARTIFICIAL INTELLIGENCE IN MEDICINE (PRINT)	6.0	2017.0	1.095445	2016.0	2016.0	2017.0	2018.0	2018.0	0.0	
BIOANALYSIS (PRINT)	7.0	2012.0	0.000000	2012.0	2012.0	2012.0	2012.0	2012.0	0.0	

5 rows × 64 columns

In [102]:

```
# pegar as 10 +
plt.figure(figsize=(15,10))
titulo_periodico.size().sort_values(ascending=False).plot.bar()
plt.xticks(rotation=50)
plt.xlabel("Título dos Periódicos")
plt.ylabel("Quantidade de Periodicos")
plt.show()
```



9 Exportação do resultado para Excel

In [103]:

```
df_scopus.columns
```

Out[103]:

```
Index(['autores', 'id_autores', 'titulo_artigo', 'ano', 'titulo_fonte',
       'volume', 'publicado', 'numero_artigo', 'inicio_pagina', 'fim_pagin
a',
       'quantidade_paginas', 'quantidade_citacoes', 'doi', 'link_scopus',
       'afiliacoes', 'autores_com_filiacoes', 'resumo',
       'palavras_chaves_autor', 'palavras_chave_index',
       'numeros_sequencia_molecular', 'chemicals_cas', 'nomes_comerciais',
       'fabricantes', 'detalhes_financiamento', 'texto_financiamento_1',
       'texto_financiamento_2', 'texto_financiamento_3', 'referencias',
       'endereco_correspondencia', 'editores', 'patrocinadores', 'editor',
       'nome_conferencia', 'data_conferencia', 'local_conferencia',
       'codigo_conferencia', 'issn_scopus', 'isbn', 'coden', 'id_pubmed',
       'idioma_original', 'titulo_abreviado_fonte', 'tipo_documento',
       'etapa_publicacao', 'acesso_livre', 'fonte', 'eid', 'inicio_pagina_',
       'fim_pagina_', 'quantidade_paginas_'],
      dtype='object')
```

In [104]:

```
## Gerar planilha com colunas especificas
#resultado_geral = os.path.join('../data','scopus-31-05-2021.csv')
resultado_scopus = os.path.join('../data','resultado_scopus.xlsx')

df_xlsx_scopus = pd.DataFrame(df_scopus, columns = [
       'issn_scopus','titulo_artigo', 'ano','quantidade_citacoes', 'tipo_documento', 'nom
       'editores', 'patrocinadores', 'editor',
       'autores', 'id_autores', 'titulo_fonte',
       'volume', 'publicado', 'numero_artigo',
       'inicio_pagina', 'fim_pagina','quantidade_paginas','inicio_pagina_','fim_pagina_',
       'doi', 'link_scopus',
       'afiliacoes', 'autores_com_filiacoes', 'resumo',
       'palavras_chaves_autor', 'palavras_chave_index',
       'numeros_sequencia_molecular', 'chemicals_cas', 'nomes_comerciais',
       'fabricantes',
       'detalhes_financiamento', 'texto_financiamento_1','texto_financiamento_2', 'texto_
       'referencias', 'endereco_correspondencia',
       'data_conferencia', 'local_conferencia',
       'codigo_conferencia', 'isbn', 'coden', 'id_pubmed',
       'idioma_original', 'titulo_abreviado_fonte',
       'etapa_publicacao', 'acesso_livre', 'fonte', 'eid'
    ,])
df_xlsx_scopus = df_xlsx_scopus.to_excel(resultado_scopus,index=False, encoding='utf-8',
```

```
C:\Users\luizp\anaconda3\lib\site-packages\xlsxwriter\worksheet.py:943: Us
erWarning: Ignoring URL 'http://www.ncbi.nlm.nih.gov/sites/entrez?db=OMI
M,%20OMIM;%20http://biocreative.sourceforge.net,%20BioCreAtIvE;%20Gene%20O
ntology,%20,%20http://www.geneontology.org;%20Kim,%20J.,%20Ohta,%20T.,%20T
suruoka,%20Y.,%20Tateisi,%20Y.,%20Collier,%20N.,%20Introduction%20to%20th
e%20bio-entity%20recognition%20task%20at%20JNLPBA%20(2004)%20Proceedings%2
0of%20the%20Joint%20Workshop%20on%20Natural%20Language%20Processing%20in%2
0Biomedicine%20and%20Its%20Applications%20(JNLPBA-2004),%20pp.%2070-75;%20
Kim,%20J.,%20GENIA%20corpus-semantically%20annotated%20corpus%20for%20bio-
textmining%20(2003)%20Bioinformatics,%2019%20(SUPPL.%201),%20pp.%20i180-i1
82.%20,%2012855455;%20Wilbur,%20W.J.,%20Hazard,%20G.F.,%20Divita,%20G.,%20
Mork,%20J.G.,%20Aronson,%20A.R.,%20Browne,%20A.C.,%20Analysis%20of%20biome
dical%20text%20for%20chemical%20names:%20A%20comparison%20of%20three%20met
hods%20(1999)%20Proc%20AMIA%20Symp,%20pp.%20176-180.%20,%2010566344;%20Cor
bett,%20P.,%20Batchelor,%20C.,%20Teufel,%20S.,%20Annotation%20of%20Chemica
l%20Named%20Entities%20(2007)%20Biological,%20Translational,%20and%20Clini
cal%20Language%20Processing,%20pp.%2057-64;%20http://pubchem.ncbi.nlm.nih.
gov,%20PubChem;%20http://www.ebi.ac.uk/chebi,%20ChEBI;%20Settles,%20B.,%20
ABNER:%20An%20open%20source%20tool%20for%20automatically%20tagging%20gene
```

In [105]:

```
df_bases.columns
```

Out[105]:

```
Index(['autores', 'id_autores', 'titulo_artigo', 'ano', 'titulo_fonte',
       'volume', 'publicado', 'numero_artigo', 'inicio_pagina', 'fim_pagin
a',
       'quantidade_paginas', 'quantidade_citacoes', 'doi', 'link_scopus',
       'afiliacoes', 'autores_com_filiacoes', 'resumo',
       'palavras_chaves_autor', 'palavras_chave_index',
       'numeros_sequencia_molecular', 'chemicals_cas', 'nomes_comerciais',
       'fabricantes', 'detalhes_financiamento', 'texto_financiamento_1',
       'texto_financiamento_2', 'texto_financiamento_3', 'referencias',
       'endereco_correspondencia', 'editores', 'patrocinadores', 'editor',
       'nome_conferencia', 'data_conferencia', 'local_conferencia',
       'codigo_conferencia', 'issn_scopus', 'isbn', 'coden', 'id_pubmed',
       'idioma_original', 'titulo_abreviado_fonte', 'tipo_documento',
       'etapa_publicacao', 'acesso_livre', 'fonte', 'eid', 'inicio_pagina_',
       'fim_pagina_', 'quantidade_paginas_', 'issn_qualis', 'titulo_periodic
o',
       'area_avaliacao', 'estrato', '_merge', 'leitura_Resumo',
       'leitura_Conclusao', 'leitura_completa'],
      dtype='object')
```

In [106]:

```
## Gerar planilha com colunas especificas
#resultado_geral = os.path.join('../data','scopus-31-05-2021.csv')
resultado_geral = os.path.join('../data','resultado_geral.xlsx')

df_xlsx = pd.DataFrame(df_bases, columns =
        ['titulo_artigo', 'ano','titulo_periodico', 'area_avaliacao', 'estrato','id_autor
       'volume', 'publicado', 'numero_artigo', 'inicio_pagina', 'fim_pagina',
       'quantidade_paginas', 'quantidade_citacoes', 'doi', 'link_scopus',
       'autores','afiliacoes', 'autores_com_filiacoes', 'resumo',
       'palavras_chaves_autor', 'palavras_chave_index',
       'numeros_sequencia_molecular', 'chemicals_cas', 'nomes_comerciais',
       'fabricantes', 'detalhes_financiamento', 'texto_financiamento_1',
       'texto_financiamento_2', 'texto_financiamento_3', 'referencias',
       'endereco_correspondencia', 'editores', 'patrocinadores', 'editor',
       'nome_conferencia', 'data_conferencia', 'local_conferencia',
       'codigo_conferencia', 'issn_scopus', 'isbn', 'coden', 'id_pubmed',
       'idioma_original', 'titulo_abreviado_fonte', 'tipo_documento',
       'etapa_publicacao', 'acesso_livre', 'fonte', 'eid', 'issn_qualis','_merge'
       'leitura_Resumo','leitura_Conclusao','leitura_completa'
         ,])
df_xlsx = df_xlsx.to_excel(resultado_geral,index=False, encoding='utf-8', header=True)
#df_xlsx_1 = df_xlsx_1.to_excel(r'C:/Users/luizp/jupyter-notebook/SisCRI-ML/data/REQ-002-
```

In [ ]:

```

```

10 Analises

In [ ]:

```
```

In [107]:

```
data_fim = pd.Timestamp.now()
print(data_inicio)
print(data_fim)
```

```
2021-06-10 11:24:38.137884
2021-06-10 11:25:42.959971
```