ECOLE SUPÉRIEURE PRIVÉE D'INGÉNIERIE ET DE TECHNOLOGIES

www.esprit.tn - E-mail : contact@esprit.tn

Siége Social : 18 rue de l'Usine - Charguia II - 2035 - Tél. : +216 71 941 541 - Fax. : +216 71 941 889

Annexe : Z.I. Chotrana II - B.P. 160 - 2083 - Pôle Technologique - El Ghazala - Tél. : +216 70 685 685 - Fax. : +216 70 685 454

esprit
Se former autrement

Stage d'ingénieur

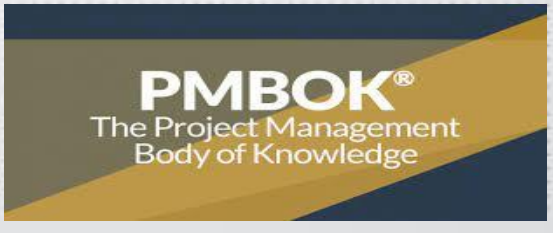
DIPLÔME NATIONAL D'INGÉNIEUR

SPÉCIALITÉ : DATA SCIENCE

Projet IA et Cognition :

PROJECT MANAGEMENT BODY OF KNOWLEDGE - PMBOK

Réalsié par:
- **LABIDI Ons**
- **FARAH Bilel**
- **DJEBBY WIEM**
- **LAMINE Kais**
- **CHEBBI Chaima**
- **SLIMANI Taoufik**



Extraction du texte à partir d'un document PDF :

Importation des bibliothèque :

```
# Useful libraries
from pdf2image import convert_from_path
from pytesseract import image_to_string
from PIL import Image
import csv
import re
```

Transformer le PDF en image ensuite les images en texte en utilisant Pytesseract :

```
def convert_pdf_to_img(pdf_file):

    return convert_from_path(pdf_file)


def convert_image_to_text(file):


    text = image_to_string(file)
    return text


def get_text_from_any_pdf(pdf_file):

    images = convert_pdf_to_img(pdf_file)
    final_text = ""
    for pg, img in enumerate(images):

        final_text += convert_image_to_text(img)
        #print("Page n°{}".format(pg))
        #print(convert_image_to_text(img))

    return final_text
```

```
Entrée [3]: path_to_pdf = 'Scope.pdf'

Entrée [4]: textextract = get_text_from_any_pdf(path_to_pdf)
            print(textextract)
```

```
PROJECT SCOPE MANAGEMENT

Project Scope Management includes the processes required to ensure that the project includes all the work required,
and only the work required, to complete the project successfully. Managing the project scope is primarily concerned with
defining and controlling what is and is not included in the project.

The Project Scope Management processes are:

5.1 Plan Scope Management—The process of creating a scope management plan that documents how the project
and product scope will be defined, validated, and controlled.

5.2 Collect Requirements—The process of determining, documenting, and managing stakeholder needs and
requirements to meet project objectives.

5.3 Define Scope—the process of developing a detailed description of the project and product.

5.4 Create WBS—The process of subdividing project deliverables and project work into smaller, more manageable
components.
```

Remplacer certains caractères par un espace vide :

```
textextract=textextract.replace("@","")
textextract=textextract.replace("¢","")
textextract=textextract.replace("©","")
textextract=textextract.replace("*","")
#textextract=textextract.replace(".","")
textextract=textextract.replace("©","")
textextract=textextract.replace("*","")
textextract=textextract.replace("-","")
textextract=textextract.replace("Part 1  Guide","")
textextract=textextract.replace("'","")
#textextract=textextract.replace("71","souschapitre1")
#textextract=textextract.replace("72","souschapitre2")
#textextract=textextract.replace("73","souschapitre3")
#textextract=textextract.replace("74","souschapitre4")

textextract = textextract.replace('; 7A', '')
textextract = textextract.replace('eeccccccccscscccccoe >', '')
textextract = textextract.replace('\nm', '')
textextract = textextract.replace('SS', '')
textextract = textextract.replace('XR J', '')
textextract = textextract.replace('« ', '')
textextract = textextract.replace('m=>', '')
textextract = textextract.replace('= ', '')
textextract = textextract.replace('wy', '')
textextract = textextract.replace('ececccecencccccccccccecceceseeg|', '')
textextract = textextract.replace('| seeerceccccscccccscn de Documents', '')
textextract = textextract.replace('=e', '')
textextract = textextract.replace('ez', '')
textextract = textextract.replace('——>', '')
textextract = textextract.replace('\no', '')
with open('CosttextextractSansFigure.txt', 'w', encoding="utf-8") as fp:
    fp.write(textextract)

```

Extraction des figures :

```
Entrée [7]: FIG = re.findall(r"Figure\s[0-9]-[0-9]{1,2}.\s[A-Z]{1}.*", textextract)
            #FIG.sort()
            FIG

Out[7]: ['Figure 5-1. Project Scope Management Overview',
         'Figure 5-2. Figure 5-3 depicts',
         'Figure 5-2. Plan Scope Management: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-3. Plan Scope Management: Data Flow Diagram',
         'Figure 5-4. Figure 5-5 depicts the data flow diagram of the process.',
         'Figure 5-4. Collect Requirements: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-5. Collect Requirements: Data Flow Diagram',
         'Figure 5-6. Context Diagram',
         'Figure 5-7. Example of a Requirements Traceability Matrix',
         'Figure 5-8. Figure 5-9 depicts the data flow diagram',
         'Figure 5-8. Define Scope: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-9. Define Scope: Data Flow Diagram',
         'Figure 5-10. Figure 5-11 depicts the data flow diagram of the process.',
         'Figure 5-10. Create WBS: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-11. Create WBS: Data Flow Diagram',
         'Figure 5-12. Sample WBS Decomposed Down Through Work Packages',
         'Figure 5-13. Sample WBS Organized by Phase',
         'Figure 5-14. Sample WBS with Major Deliverables',
         'Figure 5-15. Figure 5-16 depicts',
         'Figure 5-15. Validate Scope: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-16. Validate Scope: Data Flow Diagram',
         'Figure 5-17. Figure 5-18 depicts the data flow diagram of the process.',
         'Figure 5-17. Control Scope: Inputs, Tools & Techniques, and Outputs',
         'Figure 5-18. Control Scope: Data Flow Diagram']
```

Extraction des titres et les mettre dans un data frame appelé ‘CONCEPT’ :

Entrée [10]:
```
Titre = re.findall(r"[0-9].*[A-Z]\n", textextract)
Titre.sort()
Titre
```

Out[10]:
```
['5.1 PLAN SCOPE MANAGEMENT\n',
 '5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n',
 '5.1.1.1 PROJECT CHARTER\n',
 '5.1.1.2 PROJECT MANAGEMENT PLAN\n',
 '5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n',
 '5.1.1.4 ORGANIZATIONAL PROCE AETS\n',
 '5.1.2 PLAN SCOPE MANAGEMENT: TOOLS AND TECHNIQUES\n',
 '5.1.2.1 EXPERT JUDGMENT\n',
 '5.1.2.2 DATA ANALYSIS\n',
 '5.1.2.3 MEETINGS\n',
 '5.1.3 PLAN SCOPE MANAGEMENT: OUTPUTS\n',
 '5.1.3.1 SCOPE MANAGEMENT PLAN\n',
 '5.1.3.2 REQUIREMENTS MANAGEMENT PLAN\n',
 '5.2 COLLECT REQUIREMENTS\n',
 '5.2.1 COLLECT REQUIREMENTS: INPUTS\n',
 '5.2.1.1 PROJECT CHARTER\n',
 '5.2.1.2 PROJECT MANAGEMENT PLAN\n',
 '5.2.1.3 PROJECT DOCUMENTS\n',
 '5.2.1.4 BUSINE DOCUMENTS\n',
 '5.2.1.5 AGREEMENTS\n',
 '5.2.1.6 ENTERPRISE ENVIRONMENTAL FACTORS\n',
 '5.2.1.7 ORGANIZATIONAL PROCE AETS\n',
 '5.2.2 COLLECT REQUIREMENTS: TOOLS AND TECHNIQUES\n',
 '5.2.2.1 EXPERT JUDGMENT\n',
 '5.2.2.2 DATA GATHERING\n',
 '5.2.2.3 DATA ANALYSIS\n',
 '5.2.2.4 DECISION MAKING\n',
 '5.2.2.5 DATA REPRESENTATION\n',
 '5.2.2.6 INTERPERSONAL AND TEAM SKILLS\n',
```

Définir le type de chaque concept :

Entrée [12]:
```
for i in range(0,90):
    if ((df.loc[i,['Concept']].str.contains('INPUTS')).bool()):
        df.loc[i,['TYPE']]='INPUTS'
    elif ((df.loc[i,['Concept']].str.contains('TOOLS AND TECHNIQUES')).bool()):
        df.loc[i,['TYPE']]='TOOLS AND TECHNIQUES'
    elif ((df.loc[i,['Concept']].str.contains('OUTPUTS')).bool()):
        df.loc[i,['TYPE']]='OUTPUTS'
df.head(51)
```

Out[12]:

	Concept	TYPE
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS
2	5.1.1.1 PROJECT CHARTER\n	NaN
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN
5	5.1.1.4 ORGANIZATIONAL PROCE AETS\n	NaN
6	5.1.2 PLAN SCOPE MANAGEMENT: TOOLS AND TECHNIQ...	TOOLS AND TECHNIQUES
7	5.1.2.1 EXPERT JUDGMENT\n	NaN
8	5.1.2.2 DATA ANALYSIS\n	NaN
9	5.1.2.3 MEETINGS\n	NaN
10	5.1.3 PLAN SCOPE MANAGEMENT: OUTPUTS\n	OUTPUTS
11	5.1.3.1 SCOPE MANAGEMENT PLAN\n	NaN
12	5.1.3.2 REQUIREMENTS MANAGEMENT PLAN\n	NaN
13	5.2 COLLECT REQUIREMENTS\n	NaN
14	5.2.1 COLLECT REQUIREMENTS: INPUTS\n	INPUTS
15	5.2.1.1 PROJECT CHARTER\n	NaN
16	5.2.1.2 PROJECT MANAGEMENT PLAN\n	NaN
17	5.2.1.3 PROJECT DOCUMENTS\n	NaN
18	5.2.1.4 BUSINE DOCUMENTS\n	NaN

Extraction du texte :

Entrée [14]:
```
a = []
for i in Content:
    if i not in a:
        a.append(i)
Content= a
print(Content)
```

```
['5.1.1.1 PROJECT CHARTER\n\nDescribed in Section 4.1.3.1. The project charter documents the project purpose, highlevel proje
ct description,\nassumptions, constraints, and highlevel requirements that the project is intended to satisfy.\n\n5.1.1.2 PRO
JECT MANAGEMENT PLAN\nDescribed in Section 4.2.3.1. Project management plan components include but are not limited to:\n\n Qu
ality management plan. Described in Section 8.1.3.1. The way the project and product scope will beanaged can be influenced by
how the organization's quality policy, methodologies, and standards are\nimplemented on the project.\n\n Project life cycle d
escription. The project life cycle determines the series of phases that a project passes\nthrough from its inception to the e
nd of the project.\n\n Development approach. The development approach defines whether waterfall, iterative, adaptive, agile,
or a\nhybrid development approach will be used.\n\n5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n\nThe enterprise environmental f
actors that can influence the Plan Scope Management process include but are not\nlimited to:\n\n Organization's culture,\n In
frastructure,\n Personnel administration, and\n\n Marketplace conditions.\n\n135\n5.1.1.4 ORGANIZATIONAL PROCE AETS\n\nThe or
ganizational process assets that can influence the Plan Scope Management process include but are not\nlimited to:\n\n Policie
s and procedures, and\n\n Historical information and lessons learned repositories.\n\n5.1.2 PLAN SCOPE MANAGEMENT: TOOLS AND
TECHNIQUES\n\n5.1.2.1 EXPERT JUDGMENT\n\nDescribed in Section 4.1.2.1 Expertise should be considered from individuals or grou
ps with specialized knowledger training in the following topics:\n\n Previous similar projects, and\n\n Information in the in
dustry, discipline, and application area.\n\n5.1.2.2 DATA ANALYSIS\n\nA data analysis technique that can be used for this pro
cess includes but is not limited to alternatives analysis.\nVarious ways of collecting requirements, elaborating the project
and product scope, creating the product, validating\nthe scope, and controlling the scope are evaluated.\n\n5.1.2.3 MEETINGS
\n\nProject teams may attend project meetings to develop the scope management plan. Attendees may include the\nproject manage
r, the project sponsor, selected project team members, selected stakeholders, anyone with responsibility\nfor any of the scop
```

Transformation le texte en data frame :

Entrée [21]:
```
import pandas as pd

dict = {'Content' : one}
df10000 = pd.DataFrame(dict)
df10000
```

Out[21]:

	Content
0	.1 PROJECT CHARTER Described in Section 4.1.3...
1	EXPERT JUDGMENT Described in Section 4.1.2....
2	DATA ANALYSIS A data analysis technique tha...
3	MEETINGS Project teams may attend project m...
4	SCOPE MANAGEMENT PLAN The scope management ...
...	...
86	. The scope management plan may be updated to...
87	. Changes to the scope baseline are incorpora...
88	PROJECT DOCUMENTS UPDATES Project documents ...
89	. Requirements documentation may be updated w...
90	. The requirements traceability matrix may be...

91 rows × 1 columns

Extraction des références de chaque ligne de la colonne 'Content' :

Entrée [23]:
```
df10000['REF']=df10000['Content'].str.findall(r'Described in Section \d.\d.\d.\d+')
df10000['REF']=df10000['REF']+df10000['Content'].str.findall(r'depicted in Figure \d+')
df10000['REF']=df10000['REF']+df10000['Content'].str.findall(r'Figure \d\d+')
df10000
```

Out[23]:

	Content	REF
0	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...
1	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]
2	DATA ANALYSIS A data analysis technique tha...	[]
3	MEETINGS Project teams may attend project m...	[]
4	SCOPE MANAGEMENT PLAN The scope management ...	[]
...	...	...
86	. The scope management plan may be updated to...	[]
87	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...
88	PROJECT DOCUMENTS UPDATES Project documents ...	[Described in Section 4.4.3.1]
89	. Requirements documentation may be updated w...	[]
90	. The requirements traceability matrix may be...	[]

91 rows × 2 columns

Appliquer les méthodes de pre-processing de NLP :

- Lowercase.
- Remove text in square brackets.
- Remove links.
- Remove punctuations.
- Remove words containing numbers.

Entrée [24]:
```
def clean_text(text):
    '''Make text lowercase, remove text in square brackets,remove links,remove punctuation
    and remove words containing numbers.'''
    text = str(text).lower()
    text = re.sub('\[.*?\]', '', text)
    text = re.sub('https?://\S+|www\.\S+', '', text)
    text = re.sub('<.*?>+', '', text)
    text = re.sub(r'[^a-zA-Z0-9]', ' ', text)
    text = re.sub('\n', '', text)
    text = re.sub('\w*\d\w*', '', text)
    return text
```

Entrée [25]:
```
df10000['Content_clean'] = df10000['Content'].apply(lambda x : clean_text(x))
df10000
```

Out[25]:

	Content	REF	Content_clean
0	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...
1	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...
2	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...
3	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...
4	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...
...	...	...	...
86	. The scope management plan may be updated to...	[]	the scope management plan may be updated to...
87	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...	changes to the scope baseline are incorpora...
88	PROJECT DOCUMENTS UPDATES Project documents ...	[Described in Section 4.4.3.1]	project documents updates project documents ...
89	. Requirements documentation may be updated w...	[]	requirements documentation may be updated w...
90	. The requirements traceability matrix may be...	[]	the requirements traceability matrix may be...

91 rows × 3 columns

Extraction des définitions de chaque ligne de la colonne ‘Content’ :

Entrée [33]:
```
df1=df1.replace({'Content_clean':'described in section [0-9] [0-9] [0-9] [0-9]'},{'Content_clean':''}, regex=True)
```

Entrée [34]:
```
df1
```

Out[34]:

	Concept	TYPE	Content	REF	Content_clean	Definition
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...
...	...	...	...	...	...	...
85	5.6.3 CONTROL SCOPE: OUTPUTS\n	OUTPUTS	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[]	project management plan updates any change ...	PROJECT MANAGEMENT PLAN UPDATES Any change ...
86	5.6.3.1 WORK PERFORMANCE INFORMATION\n	NaN	. The scope management plan may be updated to...	[]	the scope management plan may be updated to...	. The scope management plan may be updated to...
87	5.6.3.2 CHANGE REQUESTS\n	NaN	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...	changes to the scope baseline are incorpora...	. Changes to the scope baseline are incorpora...
88	5.6.3.3 PROJECT MANAGEMENT PLAN UPDATES\n	NaN	PROJECT DOCUMENTS UPDATES Project documents ...	[Described in Section 4.4.3.1]	project documents updates project documents ...	PROJECT DOCUMENTS UPDATES Project documents ...
89	5.6.3.4 PROJECT DOCUMENTS UPDATES\n	NaN	. Requirements documentation may be updated w...	[]	requirements documentation may be updated w...	. Requirements documentation may be updated w...

90 rows × 6 columns

- Segmentation des données

```
Entrée [35]: from nltk.tokenize import word_tokenize

df1['new'] = df1['Content_clean'].apply(word_tokenize)
df1.head(40)
```

Out[35]:

	Concept	TYPE	Content	REF	Content_clean	Definition	new
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER	[project, charter, described, in, section, the...
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT	[expert, judgment, described, in, section, exp...
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...	[data, analysis, a, data, analysis, technique,...
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...	[meetings, project, teams, may, attend, projec...
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...	[scope, management, plan, the, scope, manageme...
	REQUIREMENTS			[depicted in	requirements	REQUIREMENTS	[requirements

- Stop-Words

```
Entrée [37]: import nltk
nltk.download("stopwords")
from nltk.corpus import stopwords

stop = set(stopwords.words('english'))
df1['new'] = df1['new'].apply(lambda x: [word for word in x if word not in stop])
df1
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\MSI\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[37]:

	Concept	TYPE	Content	REF	Content_clean	Definition	new
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER	[project, charter, described, section, project...
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT	[expert, judgment, described, section, experti...
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...	[data, analysis, data, analysis, technique, us...
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...	[meetings, project, teams, may, attend, projec...
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...	[scope, management, plan, scope, management, p...
...	...	...	...	...	...	...	...
85	5.6.3 CONTROL SCOPE: OUTPUTS\n	OUTPUTS	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[]	project management plan updates any change ...	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[project, management, plan, updates, change, p...
86	5.6.3.1 WORK PERFORMANCE INFORMATION\n	NaN	. The scope management plan may be updated to...	[]	the scope management plan may be updated to...	. The scope management plan may be updated to...	[scope, management, plan, may, updated, reflec...
87	5.6.3.2 CHANGE REQUESTS\n	NaN	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...	changes to the scope baseline are incorpora...	. Changes to the scope baseline are incorpora...	[changes, scope, baseline, incorporated, respo...
88	5.6.3.3 PROJECT MANAGEMENT PLAN UPDATES\n	NaN	PROJECT DOCUMENTS UPDATES Project documents ...	[Described in Section 4.4.3.1]	project documents updates project documents ...	PROJECT DOCUMENTS UPDATES Project documents ...	[project, documents, updates, project, documen...
89	5.6.3.4 PROJECT DOCUMENTS UPDATES\n	NaN	. Requirements documentation may be updated w...	[]	requirements documentation may be updated w...	. Requirements documentation may be updated w...	[requirements, documentation, may, updated, ad...

- Bigrame+Trigramme+post tagging

```
from nltk.corpus import wordnet
from nltk.corpus import brown

wordnet_map = {"N":wordnet.NOUN,
               "V":wordnet.VERB,
               "J":wordnet.ADJ,
               "R":wordnet.ADV
              }

train_sents = brown.tagged_sents(categories='news')
t0 = nltk.DefaultTagger('NN')
t1 = nltk.UnigramTagger(train_sents, backoff=t0)
t2 = nltk.BigramTagger(train_sents, backoff=t1)

def pos_tag_wordnet(text, pos_tag_type="pos_tag"):
    """
        Create pos_tag with wordnet format
    """
    pos_tagged_text = t2.tag(text)

    # map the pos tagging output with wordnet output
    pos_tagged_text = [(word, wordnet_map.get(pos_tag[0])) if pos_tag[0] in wordnet_map.keys() else (word, wordnet.NOUN) for (wor
    return pos_tagged_text
```

Entrée [43]:
```
df1['new1'] = df1['new'].apply(lambda x: pos_tag_wordnet(x))
df1.head()
```

Out[43]:

	Concept	TYPE	Content	REF	Content_clean	Definition	new	new1
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER	[project, charter, described, section, project...	[(project, n), (charter, n), (described, v), (...
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT	[expert, judgment, described, section, experti...	[(expert, n), (judgment, n), (described, v), (...
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...	[data, analysis, data, analysis, technique, us...	[(data, n), (analysis, n), (data, n), (analysi...
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...	[meetings, project, teams, may, attend, projec...	[(meetings, n), (project, n), (teams, n), (may...
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...	[scope, management, plan, scope, management, p...	[(scope, n), (management, n), (plan, n), (scop...

- Synonym

	Concept	TYPE	Content	REF	Content_clean	Definition	new	new1	new2	synonym
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER	[project, charter, described, section, project...	[(project, n), (charter, n), (described, v), (...	[(project, n), (charter, n), (described, v), (...	{factor, plan, status, intended, standard, sen...
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT	[expert, judgment, described, section, experti...	[(expert, n), (judgment, n), (described, v), (...	[(expert, n), (judgment, n), (described, v), (...	{cost, specialised, report, distinguish, plan,...
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...	[data, analysis, data, analysis, technique, us...	[(data, n), (analysis, n), (data, n), (analysi...	[(data, n), (analysis, n), (data, n), (analysi...	{plan, send_away, pile_up, flesh_out, tooshie,...
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...	[meetings, project, teams, may, attend, projec...	[(meetings, n), (project, n), (teams, n), (may...	[(meetings, n), (project, n), (teams, n), (may...	{whitethorn, plan, arise, extremity, required,...
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...	[scope, management, plan, scope, management, p...	[(scope, n), (management, n), (plan, n), (scop...	[(scope, n), (management, n), (plan, n), (scop...	{factor, acceptation, plan, validated, send_aw...
...	...	...	...	...	...	...	...	...	...	...
85	5.6.3 CONTROL SCOPE: OUTPUTS\n	OUTPUTS	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[]	project management plan updates any change ...	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[project, management, plan, updates, change, p...	[(project, n), (management, n), (plan, n), (up...	[(project, n), (management, n), (plan, n), (up...	{factor, plan, disco_biscuit, break_down, cogn...
86	5.6.3.1 WORK PERFORMANCE INFORMATION\n	NaN	. The scope management plan may be updated to...	[]	the scope management plan may be updated to...	.. The scope management plan may be updated to...	[scope, management, plan, may, updated, reflec...	[(scope, n), (management, n), (plan, n), (may,...	[(scope, n), (management, n), (plan, n), (may,...	{whitethorn, cost, report, distinguish, plan, ...
87	5.6.3.2 CHANGE REQUESTS\n	NaN	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...	changes to the scope baseline are incorpora...	. Changes to the scope baseline are incorpora...	[changes, scope, baseline, incorporated, respo...	[(changes, n), (scope, n), (baseline, n), (inc...	[(changes, n), (scope, n), (baseline, n), (inc...	{send_away, soh, couch, tooshie, group_A, Wb, ...

- Similarity

```
import spacy

nlp = spacy.load('en_core_web_sm')

for words in df1['Content_clean']:
    tokens = nlp(words)
    #for token in tokens:
    # Printing the following attributes of each token.
    # text: the word string, has_vector: if it contains a vector representation in the model,
    # vector_norm: the algebraic norm of the vector,
    # is_oov: if the word is out of vocabulary.
        #print(token.text, token.has_vector, token.vector_norm, token.is_oov)

    token1, token2 = tokens[0], tokens[1]
    if  token1.similarity(token2) :
        print("Similarity:", token1.similarity(token2),token1.text,token2.text)
```

```
Similarity: 0.18570521473884583    project
Similarity: 0.05067988112568855    expert
Similarity: 0.18677888810634613    data
Similarity: 0.09646563231945038    meetings
Similarity: 0.07418753206729889    scope
Similarity: 0.12520818412303925    requirements
Similarity: 0.18570521473884583    project
Similarity: 0.15778063237667084    project
Similarity: 0.010815915651619434     the
Similarity: 0.005686407908797264     the
Similarity: 0.11796143651008606    project
Similarity: 0.07132532447576523    busine
Similarity: 0.14040763676166534    agreements
Similarity: 0.03733307868242264    enterprise
Similarity: -0.09046298265457153    organizational
Similarity: 0.05067988112568855    expert
Similarity: 0.16039255261421204    data
Similarity: 0.1788518875837326    data
Similarity: 0.04805265739560127    decision
Similarity: 0.17617206275463104    data
Similarity: 0.01544579304754734    interpersonal
Similarity: 0.11939940601587296    context
Similarity: 0.018785659223794937    prototypes
Similarity: 0.08332089334726334    requirements
Similarity: 0.11349549889564514    requirements
Similarity: 0.18570521473884583    project
Similarity: 0.14565090835094452    project
Similarity: 0.1082460805773735     which
Similarity: 0.11796143651008606    project
Similarity: 0.09617016464471817     requirements
Similarity: 0.03733307868242264    enterprise
Similarity: -0.09046298265457153    organizational
Similarity: 0.05067988112568855    expert
Similarity: 0.1889413595199585    data
Similarity: 0.01093687117099762    decision
Similarity: 0.04958058521151543     a
Similarity: 0.01544579304754734    interpersonal
Similarity: 0.1281757950782776    product
Similarity: 0.18083687126636505    project
Similarity: 0.12821926176548004    project
Similarity: 0.14967992901802063     requirements
Similarity: -0.010793756693601608     the
Similarity: 0.1415296196937561    project
Similarity: 0.006941767875105143     the
Similarity: 0.10296080261468887    project
```

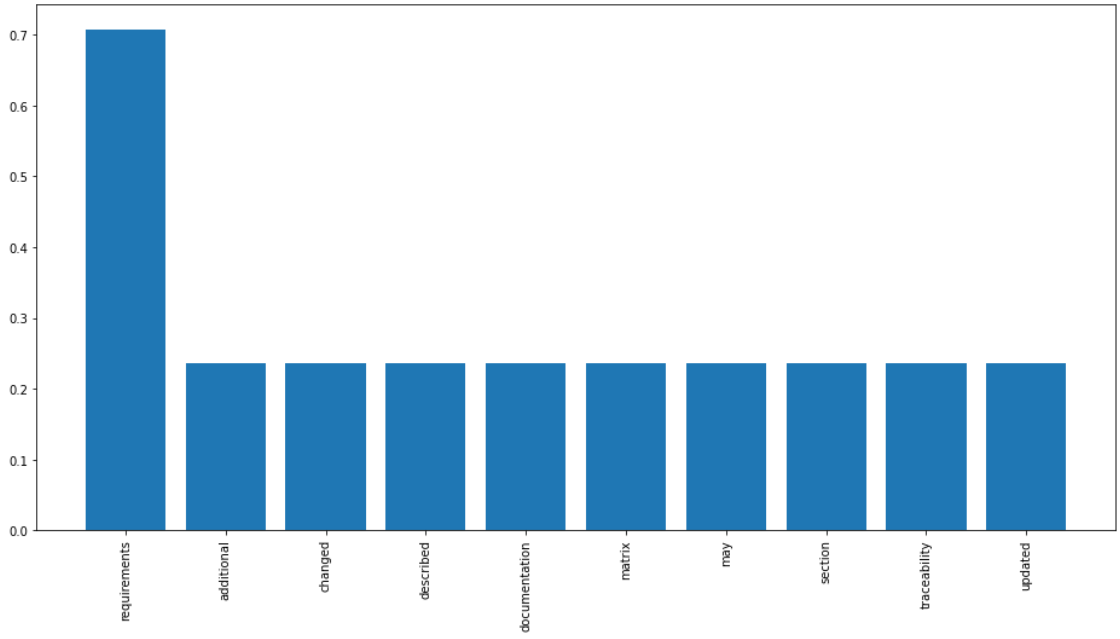
Extraire les mots clés de chaque contenu en utilisant le ‘TF-IDF’

```
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.feature_extraction.text import TfidfVectorizer


from nltk.corpus import stopwords
stop = stopwords.words('english')

df1['keywords'] = df1['Content_clean'].apply(lambda x: ' '.join([word for word in x.split() if word not in (stop)]))
#using the count vectorizer

for i in range(1,len(df1['keywords'])):

    count = CountVectorizer()
    word_count=count.fit_transform([df1['keywords'][i]])

    tfidf_transformer=TfidfTransformer(smooth_idf=True,use_idf=True)
    tfidf_transformer.fit(word_count)

    df_idf = pd.DataFrame(tfidf_transformer.idf_, index=count.get_feature_names_out(),columns=["idf_weights"])

    #inverse document frequency

    df_idf.sort_values(by=['idf_weights'])

    #tfidf

    tf_idf_vector=tfidf_transformer.transform(word_count)
    feature_names = count.get_feature_names_out()

    first_document_vector=tf_idf_vector[0]
    df_tfifd= pd.DataFrame(first_document_vector.T.todense(), index=feature_names, columns=["tfidf"])

    tf_idf = df_tfifd.sort_values(by=["tfidf"],ascending=False)
    tf_idf = tf_idf[tf_idf["tfidf"] != 0.0]
    df1['keywords'][i] = tf_idf.index.tolist()
```

Concept	TYPE	Content	REF	Content_clean	Definition	new	new1	new2	synonym	keywords
5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER Described in Section 4.1.3...	[Described in Section 4.1.3.1, Described in Se...	project charter described in section t...	.1 PROJECT CHARTER	[project, charter, described, section, project...	[(project, n), (charter, n), (described, v), (...	[(project, n), (charter, n), (described, v), (...	{factor, plan, status, intended, standard, sen...	project charter described section project char...
5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT Described in Section 4.1.2....	[Described in Section 4.1.2.1]	expert judgment described in section ex...	EXPERT JUDGMENT	[expert, judgment, described, section, experti...	[(expert, n), (judgment, n), (described, v), (...	[(expert, n), (judgment, n), (described, v), (...	{cost, specialised, report, distinguish, plan,...	[application, information, topics, specialized...
5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	data analysis a data analysis technique tha...	DATA ANALYSIS A data analysis technique tha...	[data, analysis, data, analysis, technique, us...	[(data, n), (analysis, n), (data, n), (analysi...	[(data, n), (analysis, n), (data, n), (analysi...	{plan, send_away, pile_up, flesh_out, tooshie,...	[scope, analysis, product, data, alternatives,...
5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	meetings project teams may attend project m...	MEETINGS Project teams may attend project m...	[meetings, project, teams, may, attend, projec...	[(meetings, n), (project, n), (teams, n), (may...	[(meetings, n), (project, n), (teams, n), (may...	{whitethorn, plan, arise, extremity, required,...	[project, management, scope, plan, selected, m...
5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	scope management plan the scope management ...	SCOPE MANAGEMENT PLAN The scope management ...	[scope, management, plan, scope, management, p...	[(scope, n), (management, n), (plan, n), (scop...	[(scope, n), (management, n), (plan, n), (scop...	{factor, acceptation, plan, validated, send_aw...	[scope, plan, project, management, process, fo...
...	...	...	...	...	...	...	...	...	...	...
5.6.3 CONTROL SCOPE: OUTPUTS\n	OUTPUTS	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[]	project management plan updates any change ...	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[project, management, plan, updates, change, p...	[(project, n), (management, n), (plan, n), (up...	[(project, n), (management, n), (plan, n), (up...	{factor, plan, disco_biscuit, break_down, cogn...	[change, plan, management, project, request, u...
5.6.3.1 WORK PERFORMANCE INFORMATION\n	NaN	. The scope management plan may be updated to...	[]	the scope management plan may be updated to...	. The scope management plan may be updated to...	[scope, management, plan, may, updated, reflec...	[(scope, n), (management, n), (plan, n), (may,...	[(scope, n), (management, n), (plan, n), (may,...	{whitethorn, cost, report, distinguish, plan, ...	[scope, baseline, change, described, managed, ...
5.6.3.2 CHANGE		. Changes to the scope	[Described in Section	changes to the	. Changes to the scope	[changes, scope,	[(changes, n), (scope, n)	[(changes, n), (scope, n)	{send_away, soh, couch,	[baseline, performance,

Visualisation des mots clés à partir d'un graphe :

```
import matplotlib.pyplot as plt
plt.figure(figsize=(16,8))
plt.bar(tf_idf.index,tf_idf["tfidf"])
plt.xticks(rotation=90)
plt.show()
```



- Chunkig

Entrée [56]:
```
def conll_tag_chunks(chunk_sents):
    tagged_sents = [tree2conlltags(tree) for tree in chunk_sents]
    return [[(t, c) for (w, t, c) in sent] for sent in tagged_sents]
def combined_tagger(train_data, taggers, backoff=None):
    for tagger in taggers:
        backoff = tagger(train_data, backoff=backoff)
    return backoff
#Define the chunker class
class NGramTagChunker(ChunkParserI):
    def __init__(self,train_sentences,tagger_classes=[UnigramTagger,BigramTagger]):
        train_sent_tags=conll_tag_chunks(train_sentences)
        self.chunk_tagger=combined_tagger(train_sent_tags,tagger_classes)
    def parse(self,tagged_sentence):
        if not tagged_sentence:
            return None
        pos_tags=[tag for word, tag in tagged_sentence]
        chunk_pos_tags=self.chunk_tagger.tag(pos_tags)
        chunk_tags=[chunk_tag for (pos_tag,chunk_tag) in chunk_pos_tags]
        wpc_tags=[(word,pos_tag,chunk_tag) for ((word,pos_tag),chunk_tag) in zip(tagged_sentence,chunk_tags)]
        return conlltags2tree(wpc_tags)
```

Entrée [74]:
```
nltk_pos_tagged = nltk.pos_tag(text.split())
chunk_tree = ntc.parse(nltk_pos_tagged)
```

Entrée [75]:
```
print(chunk_tree)
```

```
(S
  (NP scope/NN management/NN plan/NN Scope/NNP management/NN plan/NN)
  (VP be/VB)
  (NP component/JJ project/NN management/NN plan/NN)
  (VP describe/VBZ)
  (NP Scope/NNP)
  (VP will/MD be/VB)
  (NP define/JJ)
  ,/,
  (VP develop/VB)
  ,/,
  (NP monitor/NN)
  ,/,
  (NP control/NN)
  ,/,
  (NP validate/NN)
  ./.
  (NP component/NN Scope/NNP management/NN plan/NN)
  (VP include/VBP)
  (NP
    Process/NNP
    prepare/NN
    project/NN
    Scope/NNP
    statement/NN
    Process/NNP
    enable/JJ
    creation/NN
    WBS/NNP)
  (VP detailed/VBD)
  (NP project/NN Scope/NNP statement/NN Process/NNP)
  (VP establish/VB)
  (NP Scope/NNP baseline/NN)
  (VP will/MD be/VB)
  (NP
```

Transformer les résultats en Data Frame :

Entrée [80]:
```
import pandas as pd
df_relation_2_3_4 = pd.DataFrame(list(zip(NP1,VB,NP2)),columns=['noun1','VB','noun2'])
df_relation_2_3_4
```

Out[80]:

	noun1	VB	noun2
0	scope management plan Scope management plan	be	component project management plan
1	component project management plan	describe	Scope
2	Scope	will be	define
3	define	develop	monitor
4	component Scope management plan	include	Process prepare project Scope statement Proces...
5	Process prepare project Scope statement Proces...	detailed	project Scope statement Process
6	project Scope statement Process	establish	Scope baseline
7	Scope baseline	will be	maintain Process specify formal acceptance com...
8	maintain Process specify formal acceptance com...	will be obtain	Scope management plan
9	Scope management plan	can be	formal informal

OWL

Importation des données et la configuration :

Entrée [87]:
```
from rdflib.namespace import DC, DCTERMS, DOAP, FOAF, OWL, RDF, RDFS, SKOS, VOID, XMLNS
from rdflib import URIRef, BNode, Literal, Namespace, Graph
from rdflib.extras import describer
```

Entrée [88]:
```
g = Graph()
g.bind("owl",OWL)
g.bind("pr","http://example.org/projectOntology/")
ns_url = "http://example.org/projectOntology/"
g.add((URIRef('http://example.org/projectOntology/'), RDF.type, OWL.Ontology ))
```

Out[88]: <Graph identifier=N1998166bb50a4eb485f415c56907bc5c (<class 'rdflib.graph.Graph'>)>

Entrée [89]:
```
final_df_scope = df3[['Concept','TYPE','Definition','REF','Chapitre']]
final_df_scope
```

	Concept	TYPE	Definition	REF	Chapitre
0	5.1 PLAN SCOPE MANAGEMENT\n	NaN	.1 PROJECT CHARTER	[Described in Section 4.1.3.1, Described in Se...	5.1 PLAN SCOPE MANAGEMENT
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	INPUTS	EXPERT JUDGMENT	[Described in Section 4.1.2.1]	5.1 PLAN SCOPE MANAGEMENT
2	5.1.1.1 PROJECT CHARTER\n	NaN	DATA ANALYSIS A data analysis technique tha...	[]	5.1 PLAN SCOPE MANAGEMENT
3	5.1.1.2 PROJECT MANAGEMENT PLAN\n	NaN	MEETINGS Project teams may attend project m...	[]	5.1 PLAN SCOPE MANAGEMENT
4	5.1.1.3 ENTERPRISE ENVIRONMENTAL FACTORS\n	NaN	SCOPE MANAGEMENT PLAN The scope management ...	[]	5.1 PLAN SCOPE MANAGEMENT
...	...	...	...	...	...
85	5.6.3 CONTROL SCOPE: OUTPUTS\n	OUTPUTS	PROJECT MANAGEMENT PLAN UPDATES Any change ...	[]	5.6 CONTROL SCOPE
86	5.6.3.1 WORK PERFORMANCE INFORMATION\n	NaN	. The scope management plan may be updated to...	[]	5.6 CONTROL SCOPE
87	5.6.3.2 CHANGE REQUESTS\n	NaN	. Changes to the scope baseline are incorpora...	[Described in Section 6.5.3.1, Described in Se...	5.6 CONTROL SCOPE
88	5.6.3.3 PROJECT MANAGEMENT PLAN UPDATES\n	NaN	PROJECT DOCUMENTS UPDATES Project documents ...	[Described in Section 4.4.3.1]	5.6 CONTROL SCOPE
89	5.6.3.4 PROJECT DOCUMENTS UPDATES\n	NaN	. Requirements documentation may be updated w...	[]	5.6 CONTROL SCOPE

Associer à chaque contenu sa référence à travers le paramètre ‘DESCRIBED IN’

Entrée [93]:
```
df_section = final_df_scope[['Concept','REF']]
df_section['Data_Property'] = 'Described in'
df_section = df_section.dropna().reset_index(drop=True)
df_section = df_section[df_section['REF'].map(lambda x : 0 < len(x))]
df_section = df_section.reset_index(drop=True)
df_section
```

```
C:\Users\MSI\AppData\Local\Temp/ipykernel_22024/2813421293.py:2: 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-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df_section['Data_Property'] = 'Described in'
```

Out[93]:

	Concept	REF	Data_Property
0	5.1 PLAN SCOPE MANAGEMENT\n	[Described in Section 4.1.3.1, Described in Se...	Described in
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	[Described in Section 4.1.2.1]	Described in
2	5.1.1.4 ORGANIZATIONAL PROCE AETS\n	[depicted in Figure 54, Figure 54, Figure 55, ...	Described in
3	5.1.2 PLAN SCOPE MANAGEMENT: TOOLS AND TECHNIQ...	[Described in Section 4.1.3.1]	Described in
4	5.1.2.1 EXPERT JUDGMENT\n	[Described in Section 4.2.3.1]	Described in
5	5.1.3 PLAN SCOPE MANAGEMENT: OUTPUTS\n	[Described in Section 4.1.3.2, Described in Se...	Described in
6	5.2.1.1 PROJECT CHARTER\n	[Described in Section 4.1.2.1]	Described in

Extraction des sujets de chaque chapitres :

Entrée [96]:
```
for c in df_topic:
    project = URIRef(ns_url+"project_scope_management")
    topic = URIRef(ns_url+c.replace(" ","_"))
    for o in final_df_scope.index :
    #for c in df_topic:
        if c == final_df_scope.loc[o,'Chapitre'] and final_df_scope.loc[o,'TYPE'] == 'OUTPUTS':
            clo = URIRef(ns_url+c.replace(" ","_")+"_"+"OUTPUTS".replace(" ","_"))

            ind1 = URIRef(ns_url+final_df_scope.loc[o,'Concept'].replace(" ","_"))
            output = URIRef(ns_url+"OUTPUTS")


            g.add((ind1, RDFS.subClassOf, clo))
            g.add((clo, RDFS.subClassOf, topic))
            g.add((topic, RDFS.subClassOf, project))


        if c == final_df_scope.loc[o,'Chapitre'] and final_df_scope.loc[o,'TYPE'] == 'INPUTS':

            cli = URIRef(ns_url+c.replace(" ","_")+"_"+"INPUTS".replace(" ","_"))
            ind2 = URIRef(ns_url+final_df_scope.loc[o,'Concept'].replace(" ","_"))

            inputs = URIRef(ns_url+"INPUTS")


            g.add((ind2, RDFS.subClassOf, cli))
            g.add((cli, RDFS.subClassOf, topic))
            g.add((topic, RDFS.subClassOf, project))

        if c == final_df_scope.loc[o,'Chapitre'] and final_df_scope.loc[o,'TYPE'] == 'TOOLS AND TECHNIQUES':

            clt = URIRef(ns_url+c.replace(" ","_")+"_"+"TOOLS AND TECHNIQUES".replace(" ","_"))

            ind3 = URIRef(ns_url+final_df_scope.loc[o,'Concept'].replace(" ","_"))
            tool = URIRef(ns_url+"TOOLS AND TECHNIQUES")

            g.add((ind3, RDFS.subClassOf, clt))
            g.add((clt, RDFS.subClassOf, topic))
            g.add((topic, RDFS.subClassOf, project))
```

Entrée [98]:
```
from rdflib.namespace import XSD

for i in range(0,len(final_df_scope['Definition'])) :
    c = URIRef(ns_url+final_df_scope.loc[i,'Concept'].replace(" ","_"))
    desc=final_df_scope.loc[i,'Definition']
    #print(desc)
    definedby = Literal(desc,datatype=XSD.string)
    g.add((c, RDFS.isDefinedBy, definedby))
```

Entrée [99]:
```
#df_section['Concept']  = df_section['Concept'].apply(lambda x : remove_punct(x))
#df_section['Concept']  = df_section['Concept'].apply(lambda x : remove_number(x))
#df_section
```

Entrée [100]:
```
for i in range(1,len(df_section)):
    c = URIRef(ns_url+df_section.loc[i,'Data_Property'].replace(" ","_"))
    domaine = URIRef(ns_url+df_section.loc[i,'Concept'].replace(" ","_"))
    g.add((c, RDF.type, OWL.DatatypeProperty))
    g.add((c, RDFS.domain, domaine))
    g.add((c, RDFS.range, XSD.string))
```

Entrée [101]:
```
df_section
```

Out[101]:

	Concept	REF	Data_Property
0	5.1 PLAN SCOPE MANAGEMENT\n	[Described in Section 4.1.3.1, Described in Se...	Described in
1	5.1.1 PLAN SCOPE MANAGEMENT: INPUTS\n	[Described in Section 4.1.2.1]	Described in
2	5.1.1.4 ORGANIZATIONAL PROCE AETS\n	[depicted in Figure 54, Figure 54, Figure 55, ...	Described in
3	5.1.2 PLAN SCOPE MANAGEMENT: TOOLS AND TECHNIQ...	[Described in Section 4.1.3.1]	Described in
4	5.1.2.1 EXPERT JUDGMENT\n	[Described in Section 4.2.3.1]	Described in
5	5.1.3 PLAN SCOPE MANAGEMENT: OUTPUTS\n	[Described in Section 4.1.3.2, Described in Se...	Described in
6	5.2.1.1 PROJECT CHARTER\n	[Described in Section 4.1.2.1]	Described in
7	5.2.1.2 PROJECT MANAGEMENT PLAN\n	[Described in Section 4.1.2.2, Described in Se...	Described in
8	5.2.1.3 PROJECT DOCUMENTS\n	[Described in Section 4.5.2.2]	Described in
9	5.2.1.6 ENTERPRISE ENVIRONMENTAL FACTORS\n	[Described in Section 4.1.2.3, Described in Se...	Described in
10	5.2.1.7 ORGANIZATIONAL PROCE AETS\n	[Figure 56, Figure 56]	Described in
11	5.2.2.2 DATA GATHERING\n	[depicted in Figure 58, Figure 57, Figure 57, ...	Described in
12	5.2.2.3 DATA ANALYSIS\n	[Described in Section 4.1.3.1]	Described in
13	5.2.2.4 DECISION MAKING\n	[Described in Section 4.2.3.1]	Described in

Extraction des domaines et des ranges :

Entrée [105]:
```
for c in df_concept:
    #print(c.strip())
    autre = URIRef(ns_url+"_Autre_")
    for i in final_df_relation.index :
    #for c in df_topic[1:]:
        #print('work',c.strip(),final_df_relation.loc[i,'noun1'].strip())
        if c.strip() == final_df_relation.loc[i,'noun1'].strip():
            #print('work')
            cl = URIRef(ns_url+final_df_relation.loc[i,'VB'].replace(" ","_"))
            domaine = URIRef(ns_url+c.replace(" ","_"))
            rang = URIRef(ns_url+final_df_relation.loc[i,'noun2'].replace(" ","_"))
            g.add((cl, RDF.type, OWL.ObjectProperty))
            g.add((cl, RDFS.domain, domaine))
            g.add((cl, RDFS.range, rang))
            g.add((domaine,cl, rang))
        else :
            cl = URIRef(ns_url+final_df_relation.loc[i,'VB'].replace(" ","_"))
            domaine = URIRef(ns_url+final_df_relation.loc[i,'noun1'].replace(" ","_"))
            rang = URIRef(ns_url+final_df_relation.loc[i,'noun2'].replace(" ","_"))

            g.add((cl, RDF.type, OWL.ObjectProperty))
            g.add((cl, RDFS.domain, domaine))
            g.add((cl, RDFS.range, rang))
            g.add((domaine,cl, rang))
            g.add((domaine, RDFS.subClassOf, autre))
            g.add((rang, RDFS.subClassOf, autre))
```

Génération du fichier OWL :

Entrée [107]:
```
g.serialize(destination='OWL_TEST.owl', format='ntriples')
```

```
C:\Users\MSI\anaconda3\lib\site-packages\rdflib\plugins\serializers\nt.py:35: UserWarning: NTSerializer always uses UTF-8 encod
ing. Given encoding was: None
  warnings.warn(
```

Out[107]:
```
<Graph identifier=N1998166bb50a4eb485f415c56907bc5c (<class 'rdflib.graph.Graph'>)>
```

Construction de l'ontologie :

:
```
ontology = final_df_['Concept'].tolist() + final_df_['TYPE'].tolist() + final_df_['Definition'].tolist() + final_df_['REF'].tolis
ontology
```

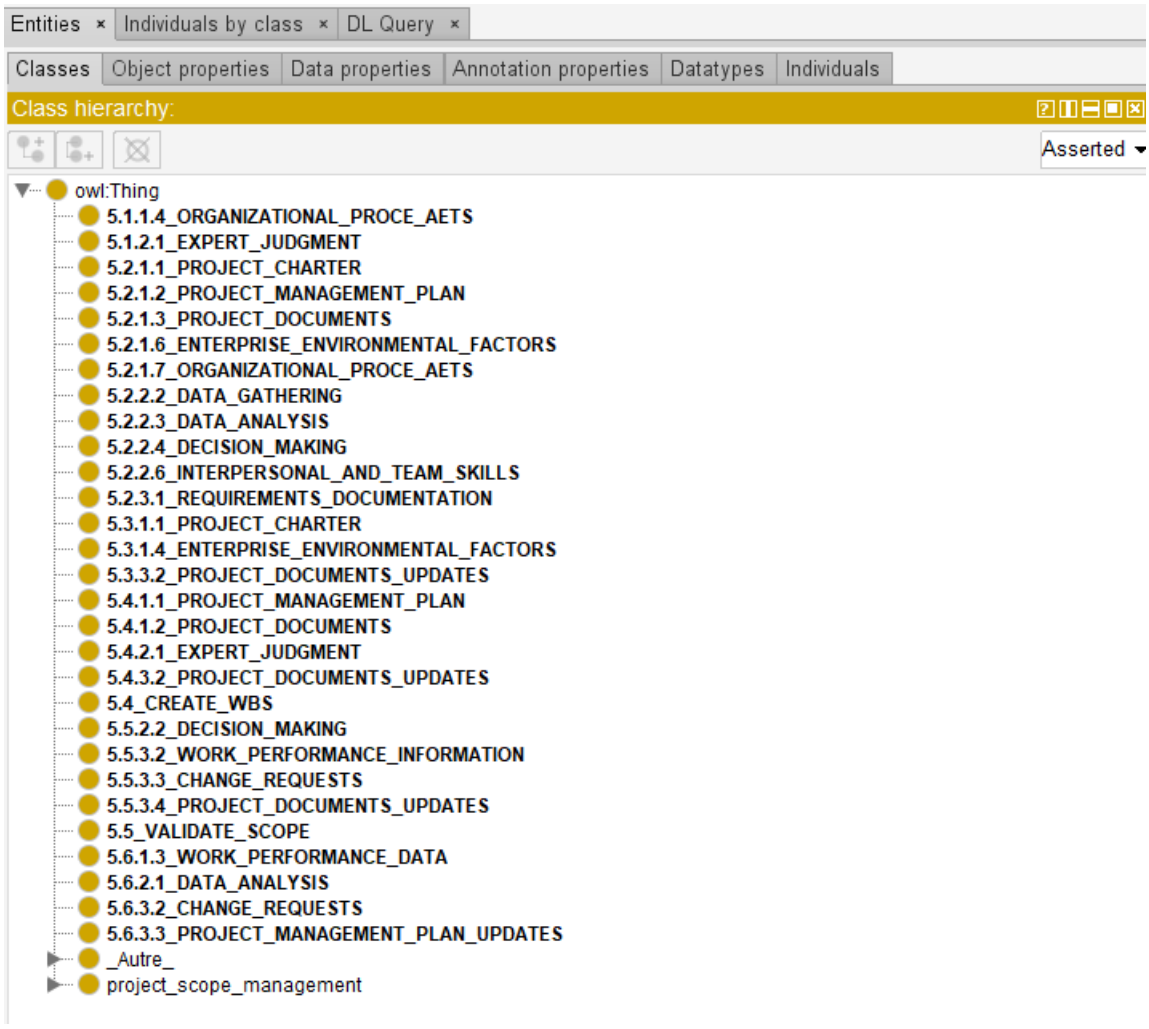
:
```
ch_ont=''
for e in ontology:
    ch_ont +='. '+str(e)
    ch_ont
```

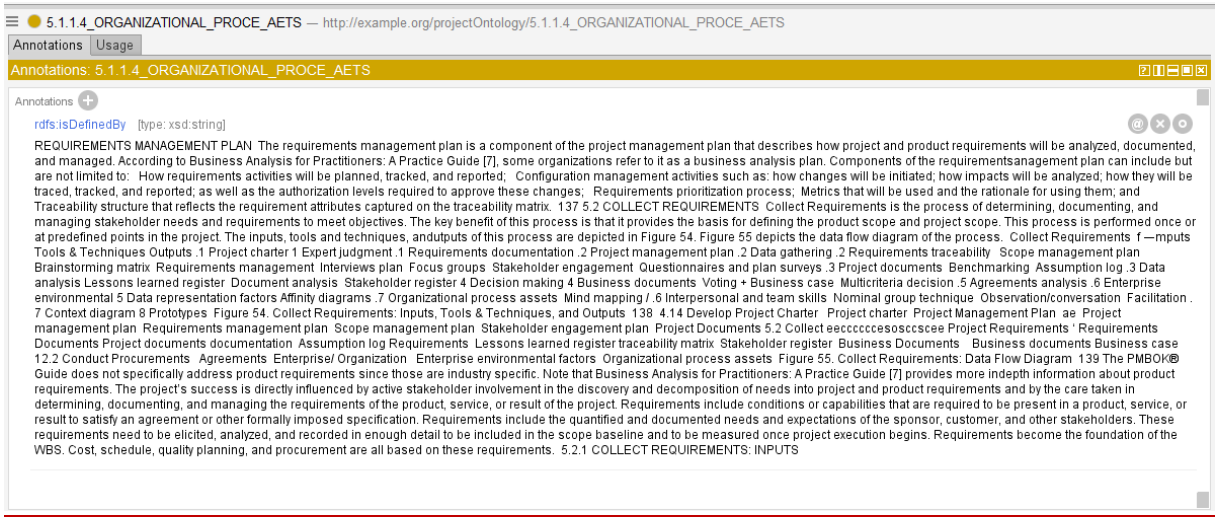
Evaluation du model :

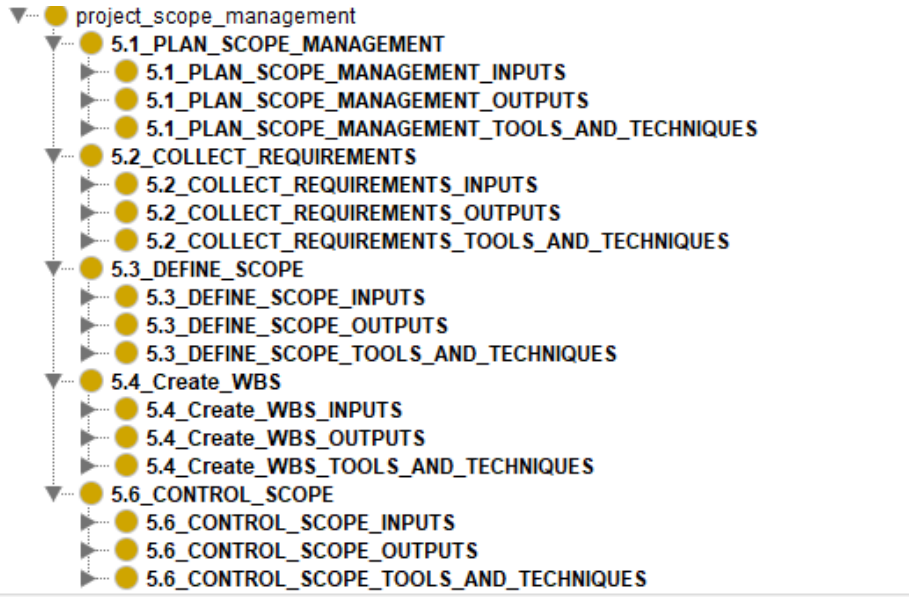
```
from rouge import Rouge

rouge = Rouge()
score=rouge.get_scores(ch_ont, pmbok, avg=True)

print('Precision = ' +str(score['rouge-1']['r']))
print('f-measure = '+str(score['rouge-1']['f']))
```

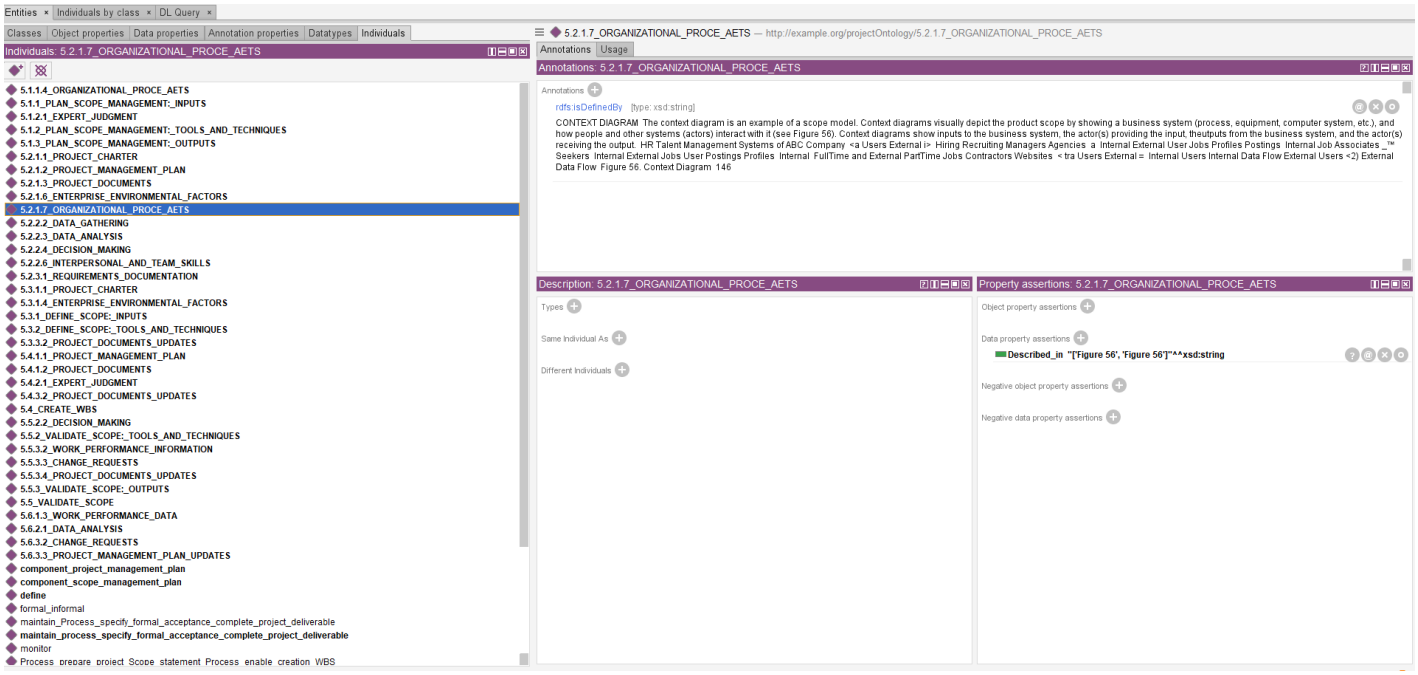
```
Precision = 0.868421052631579
f-measure = 0.04217252348782983
```

Visualisation du fichier OWL généré à partir du logiciel 'Protégé' :






project_scope_management
5.1_PLAN_SCOPE_MANAGEMENT
5.1_PLAN_SCOPE_MANAGEMENT_INPUTS
5.1_PLAN_SCOPE_MANAGEMENT_OUTPUTS
5.1_PLAN_SCOPE_MANAGEMENT_TOOLS_AND_TECHNIQUES
5.2_COLLECT_REQUIREMENTS
5.2_COLLECT_REQUIREMENTS_INPUTS
5.2_COLLECT_REQUIREMENTS_OUTPUTS
5.2_COLLECT_REQUIREMENTS_TOOLS_AND_TECHNIQUES
5.3_DEFINE_SCOPE
5.3_DEFINE_SCOPE_INPUTS
5.3_DEFINE_SCOPE_OUTPUTS
5.3_DEFINE_SCOPE_TOOLS_AND_TECHNIQUES
5.4_Create_WBS
5.4_Create_WBS_INPUTS
5.4_Create_WBS_OUTPUTS
5.4_Create_WBS_TOOLS_AND_TECHNIQUES
5.6_CONTROL_SCOPE
5.6_CONTROL_SCOPE_INPUTS
5.6_CONTROL_SCOPE_OUTPUTS
5.6_CONTROL_SCOPE_TOOLS_AND_TECHNIQUES


Entities
Individuals by class
DL Query
Classes
Object properties
Data properties
Annotation properties
Datatypes
Individuals
Individuals: 5.2.1.7_ORGANIZATIONAL_PROCE_AETS
5.1.1.4_ORGANIZATIONAL_PROCE_AETS
5.1.1_PLAN_SCOPE_MANAGEMENT:_INPUTS
5.1.2.1_EXPERT_JUDGMENT
5.1.2_PLAN_SCOPE_MANAGEMENT:_TOOLS_AND_TECHNIQUES
5.1.3_PLAN_SCOPE_MANAGEMENT:_OUTPUTS
5.2.1.1_PROJECT_CHARTER
5.2.1.2_PROJECT_MANAGEMENT_PLAN
5.2.1.3_PROJECT_DOCUMENTS
5.2.1.6_ENTERPRISE_ENVIRONMENTAL_FACTORS
5.2.1.7_ORGANIZATIONAL_PROCE_AETS
5.2.2.2_DATA_GATHERING
5.2.2.3_DATA_ANALYSIS
5.2.2.4_DECISION_MAKING
5.2.2.6_INTERPERSONAL_AND_TEAM_SKILLS
5.2.3.1_REQUIREMENTS_DOCUMENTATION
5.3.1.1_PROJECT_CHARTER
5.3.1.4_ENTERPRISE_ENVIRONMENTAL_FACTORS
5.3.1_DEFINE_SCOPE:_INPUTS
5.3.2_DEFINE_SCOPE:_TOOLS_AND_TECHNIQUES
5.3.3.2_PROJECT_DOCUMENTS_UPDATES
5.4.1.1_PROJECT_MANAGEMENT_PLAN
5.4.1.2_PROJECT_DOCUMENTS
5.4.2.1_EXPERT_JUDGMENT
5.4.3.2_PROJECT_DOCUMENTS_UPDATES
5.4_CREATE_WBS
5.5.2.2_DECISION_MAKING
5.5.2_VALIDATE_SCOPE:_TOOLS_AND_TECHNIQUES
5.5.3.2_WORK_PERFORMANCE_INFORMATION
5.5.3.3_CHANGE_REQUESTS
5.5.3.4_PROJECT_DOCUMENTS_UPDATES
5.5.3_VALIDATE_SCOPE:_OUTPUTS
5.5_VALIDATE_SCOPE
5.6.1.3_WORK_PERFORMANCE_DATA
5.6.2.1_DATA_ANALYSIS
5.6.3.2_CHANGE_REQUESTS
5.6.3.3_PROJECT_MANAGEMENT_PLAN_UPDATES
component_project_management_plan
component_scope_management_plan
define
formal_informal
maintain_Process_specify_formal_acceptance_complete_project_deliverable
maintain_process_specify_formal_acceptance_complete_project_deliverable
monitor
Process_prepare_project_Scope_statement_Process_enable_creation_WBS
5.2.1.7_ORGANIZATIONAL_PROCE_AETS — http://example.org/projectOntology/5.2.1.7_ORGANIZATIONAL_PROCE_AETS
Annotations
Usage
Annotations: 5.2.1.7_ORGANIZATIONAL_PROCE_AETS
rdfs:isDefinedBy [type: xsd:string]
CONTEXT DIAGRAM The context diagram is an example of a scope model. Context diagrams visually depict the product scope by showing a business system (process, equipment, computer system, etc.), and how people and other systems (actors) interact with it (see Figure 56). Context diagrams show inputs to the business system, the actor(s) providing the input, theutputs from the business system, and the actor(s) receiving the output. HR Talent Management Systems of ABC Company <a Users External i> Hiring Recruiting Managers Agencies a Internal External User Jobs Profiles Postings Internal Job Associates _™ Seekers Internal External Jobs User Postings Profiles Internal FullTime and External PartTime Jobs Contractors Websites < tra Users External = Internal Users Internal Data Flow External Users <2) External Data Flow Figure 56. Context Diagram 146
Description: 5.2.1.7_ORGANIZATIONAL_PROCE_AETS
Types
Same Individual As
Different Individuals
Property assertions: 5.2.1.7_ORGANIZATIONAL_PROCE_AETS
Object property assertions
Data property assertions
Described_in "[Figure 56', 'Figure 56']"^^xsd:string
Negative object property assertions
Negative data property assertions