Indexation :

1. Extraction automatique de termes

```
>>> ExpReg = nltk.RegexpTokenizer('(?:[A-Za-z]\.)+|\d+(?:\.\d+)?%?|\w+(?:\-\w+)*')
```

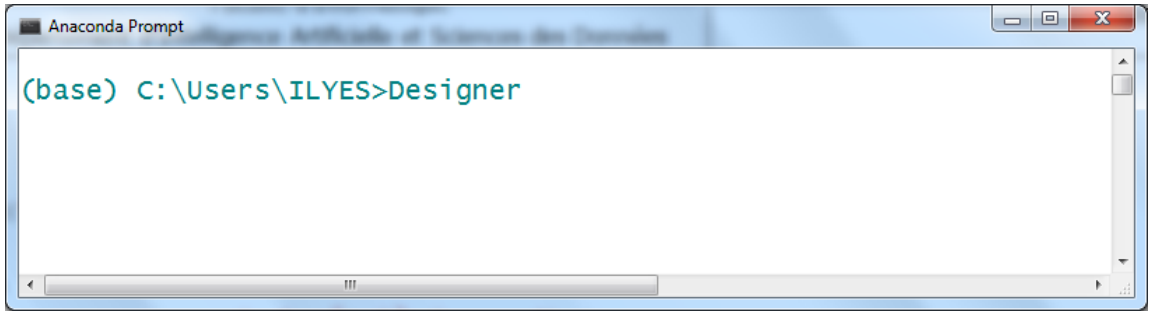
5. Pondération des termes normalisés

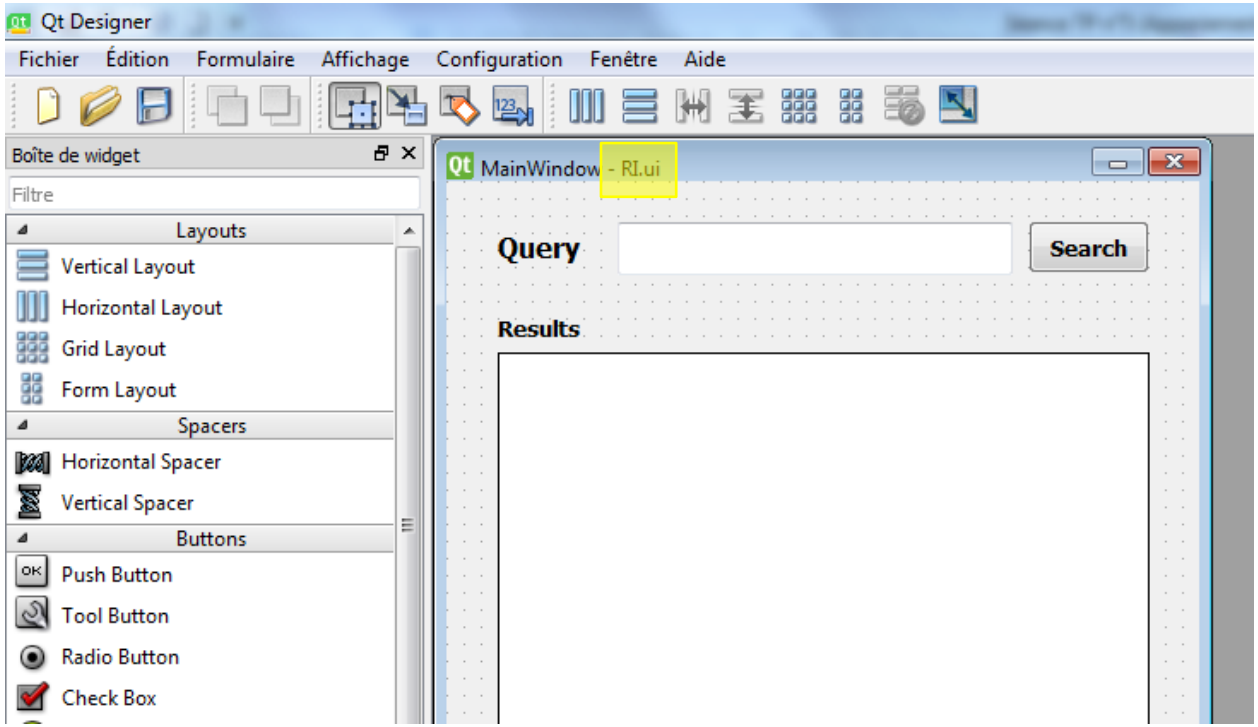
$$poids(t_i, d_j) = \left(\frac{freq(t_i, d_j)}{Max(freq(t, d_j)}\right) * log\left(\frac{N}{n_i} + 1\right)$$

IHM :

1. **Qt Designer**

```
>>> Designer
```





Extension du fichier sur Qt Designer :

<nom>.ui

2. Convertir le fichier <nom>.ui en <nom>.py

```
>>> pyuic5 -x RI.ui -o RI.py
```

Anaconda Prompt

```
(base) C:\Users\ILYES>Designer

(base) C:\Users\ILYES>pyuic5 -x  RI.ui -o RI.py

(base) C:\Users\ILYES>
```

C:\Users\ILYES\RI.py - Notepad++

Fichier Édition Recherche Affichage Encodage Langage Paramétrage Macro Exécution Compléments

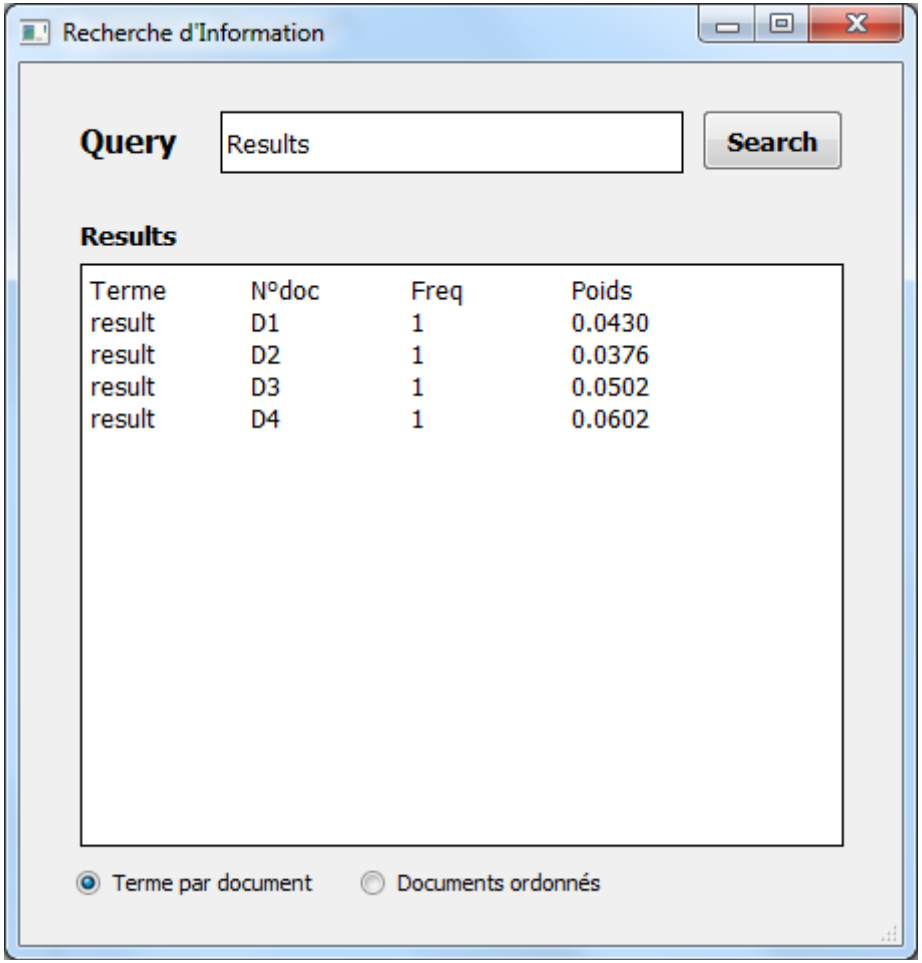
RI.py

```
# -*- coding: utf-8 -*-

# Form implementation generated from reading ui file 'RI.ui'
#
# Created by: PyQt5 UI code generator 5.11.3
#
# WARNING! All changes made in this file will be lost!

from PyQt5 import QtCore, QtGui, QtWidgets

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(440, 640)
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.label = QtWidgets.QLabel(self.centralwidget)
        self.label.setGeometry(QtCore.QRect(30, 28, 61, 21))
        font = QtGui.QFont()
        font.setPointSize(12)
        font.setBold(True)
        font.setWeight(75)
        self.label.setFont(font)
```

Appariement :

```
self.pushButton.clicked.connect(self.search)
…
def search(self):
…
```

```
self.lineEdit.text()
```



Entrée :

Un terme

Sortie :

Pour chaque document de la collection, en retourne la fréquence et le poids du terme

```
self.textEdit.setText("")
…
self.textEdit.append("")
…
```

```
self.radioButton.setChecked(True)
…
self.radioButton.isChecked()==True
…
```

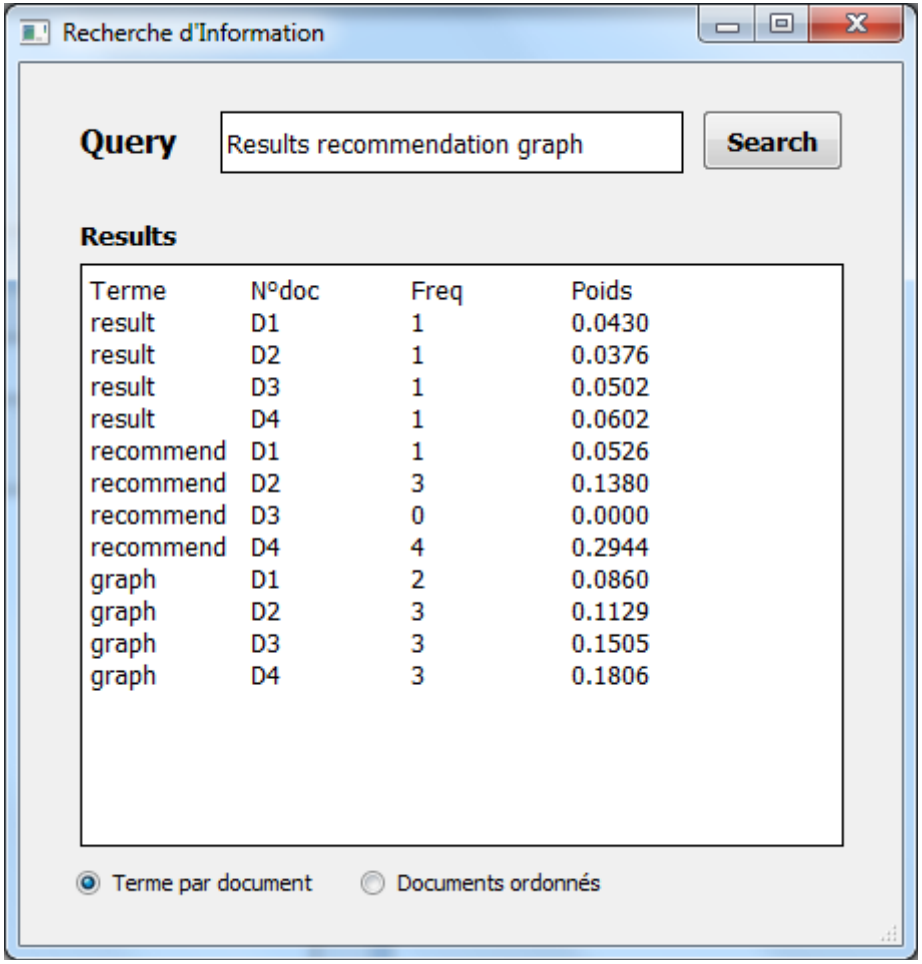
Appariement :

Entrée :

Un ensemble de termes

Sortie :

Pour chaque document de la collection, en retourne la fréquence et le poids de chaque terme de la requête



Appariement :

Entrée :

Un ensemble de termes

Sortie :

En retourne une liste de documents ordonnés selon leurs degrés de pertinences. Le degré de pertinence d'un document correspond à la somme des poids des termes de ce document appartenant à la requête