



# Rapport hackathon de la Banque de France

Pichon Raphaël

pichon.raphael@icloud.com

2020



#### Introduction

Dans le cadre de ce hackathon, l'ensemble du travail a été réalisé en python. Nous nous arrêterons ici sur un descriptif de la démarche répondant à la problématique du hackathon.

<u>1 - we</u>	eb scrapingp	<u>.3</u>
	1.1 - Profil détaillé d'un économiste et de ses publications p	<u>.3</u>
	1.2 - Liste d'économistes par compétence et institution p	<u>.4-6</u>
<u>2 - UI</u>	p	<u>.7</u>
3 - Lir	mites et atouts p.	.8

# Bibliothèques utilisées

**UI:** Tkinter

Web scraping: Requests, bs4

OS: string, sys, os

PDF: PdfFileReader, Tabula

### **Utilisation**

python3 UI.py ou chmod +x UI.py puis ./UI.py ou python3 UI.py -update

Ajouter l'option -update pour mettre à jour les données utilisées dans le moteur de recherche

# Descriptif de la problématique

La problématique étant de créer un moteur de recherche avec UI. Ce moteur de recherche devra répondre aux critères suivants :

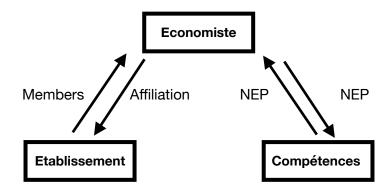
- 1 Rechercher un nom d'économiste, et afficher un bref descriptif de l'économiste
- 2 Rechercher un économiste par domaine de compétence (ex : inflation) et/ou par institution (ex : Banque de France, FMI ... )
  - 3 Recherche et accès aux articles de la sélection



## 1 - Web scraping

Le web scraping se fait uniquement sur le site web suivant : https://ideas.repec.org/

Afin de créer le moteur de recherche il faut délimiter les connexions qui existent entre les pages qui nous intéressent (économiste, institution et des compétences). Ci-dessous un schéma résumant les connexions que je vais utiliser pour le moteur :



Connecter les institutions et les compétences n'est pas ici nécessaire, le moteur étant centré sur l'économiste. Une recherche par institution et/ou compétence se fera par 2 requêtes et une jonction (ou non) des économistes trouvés. On gagne donc du temps en évitant de recroiser des informations.

## 1.1 Profil d'un économiste par son nom

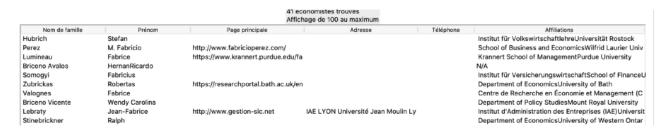
La recherche d'un économiste par nom se fait à l'aide d'un fichier regroupant l'ensemble des noms et leurs liens scrapés en amont du programme. Ce fichier peut être mis à jour au choix à l'aide de la fonction **update\_economist()**. Une fois l'adresse obtenue, deux fonctions permettent d'obtenir les noms des articles et leurs liens ainsi que les informations personnelles de l'économiste (respectivement **economist\_articles(url = False,soup = False)** et **economist personal informations(url = False,soup = False)**)

Je viens ainsi scraper les informations des onglets « personal details », « affiliation » et « research ». Ces informations feront office de bref descriptif de l'économiste. L'ensemble des publications de l'économiste est également affiché à l'aide du scraping de l'onglet « Research ».





A l'aide de l'interface utilisateur, on peut donc obtenir les résultats d'une recherche par nom. Cidessous la recherche d'un nom contenant « bric » :



Le temps de réponse d'une requête est d'environ 0,2 seconde pour obtenir les informations personnelles des économistes. Il n'est pas possible d'envisager un temps de réponse qui évolue de manière linéaire avec le temps (pour n économistes, le temps de réponse sera de n\*0,2 secondes). De plus pour avoir un programme léger, nous devons optimiser les requêtes et éviter de télécharger toute la base de données. En revanche nous téléchargeons les liens et les descriptions permettant d'accéder rapidement aux catégories sans requêtes. Nous allons alors utiliser une liste de thread et effectuer les requêtes en parallèles. Le temps d'une recherche est alors considérablement réduit.

## 1.2 Profils d'économistes par institution et compétence

Les institutions et compétences viennent, tout comme les économistes, être scrapées et stockées dans les fichiers institutions.txt et NEP.txt.

#### 1.2.1 Institutions

Les institutions seront cherchées par mots-clés dans le fichier institutions.txt avec leur lien respectif. Les institutions sont mises à jours avec la fonction **update\_all\_institution(categories\_urls)** à l'aide de la page les triant par catégories JEL. Afin d'optimiser le temps de recherche, nous ne ferons des requêtes que sur les institutions où figurent des économistes. J'impose donc que l'icône « Has members registered with the RePEc Author Service » soit présente lorsqu'on ajoute une institution au fichier texte. Exemple :



La recherche des économistes se fait parmi toutes les institutions trouvées avec l'onglet « members » dans une institution.

A l'aide de l'interface utilisateur on peut constater que la recherche par institution est fonctionnelle :





# 1.2.1 Compétences

Dans cette base de données, les économistes sont classés par NEP. Je vais donc utiliser la classification pour faire une recherche par compétence. Une brève description des catégories étant disponible, la recherche se fera par code NEP et mots-clés. Je viens scraper la page recensant les noms à partir de la page NEP. Exemple ci-dessous avec la catégorie ACC Accounting & Auditing:

## Authors in Accounting & Auditing

This list attempts to categories authors by fields. The procedure is to look at all their papers announced in a NEP report. If 5 or more have appeared in a report, authors are considered to be working within that field. For authors whose first paper is less than 10 years old, having 25% of the papers announced in NEP be announced in this report is sufficient. Note that a paper may appear in several reports.

Below are authors identified to be in the following field: NEP-ACC (Accounting & Auditing). See the top institutions and authors in this field. ? signals authors with whom we have lost contact and are seeking a current email address.

#### A|B|C|D|E|F|G|H|I|J|K|L|M|N|O|P|Q|R|S|T|U|V|W|X|Y|Z

A		
Abad, Jorge 🏏	Akcigit, Ufuk	
Aboutajdine, Reda	Akgun, Oguzhan	
Adhikari, Bibek 🏏	Allen, Franklin	
Adu, Kofi Osei	Alm, James	
Advani, Arun N. 🏏	Alstadsaeter, Annette	
Afonso, Sergio Pires	Altshuler, Rosanne	
Agostini, Marisa	Alvarez Martinez, Maria Teresa	
Ahsina, Khalifa	Amat, Oriol	
В		

Angrist, Noam
Anwar, Chowdhury Mohammad Sakib
Apps, Patricia Frances
Assa, Jacob
Atanasov, Atanas
Auerbach, Alan
Azimi, Mohammad Naim

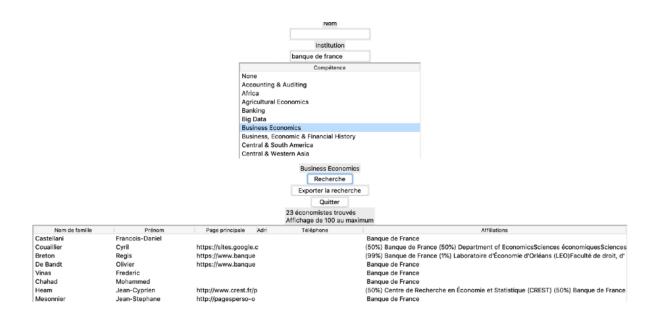
On constate que tous les économistes sont listés sur une seule page, avec cette méthode une requête suffit pour obtenir tous les économistes. La recherche par compétence est donc très rapide.

On peut constater ci-dessous une recherche par compétence :





Une recherche avec les deux conditions (compétence et institution) est réalisée en superposant tous les économistes ou en testant les économistes présents dans les deux sets de données. Cidessous un exemple comprenant deux conditions :



Le moteur de recherche comprenant la recherche d'économistes par noms ainsi qu'une recherche filtrée par compétences et/ou institutions est donc fonctionnelle. Afin de continuer et de rendre ce moteur utilisable, nous montrerons dans la prochaine partie l'interface utilisateur (UI) développée.

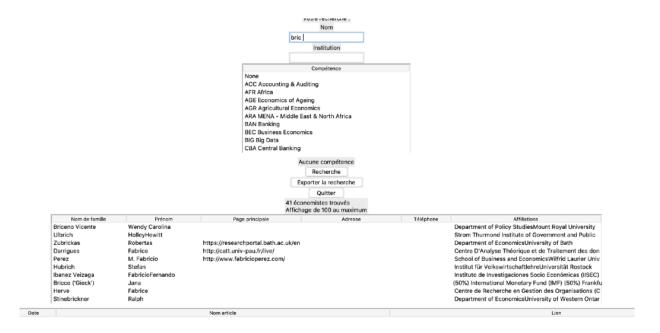


#### 2 - UI

L'Ul est faite avec la bibliothèque Tkinter. Deux barres de recherche et un menu déroulant (nom, institution et compétence), trois boutons (quitter, exporter et chercher) et les tableaux d'affichage sont les interfaces où l'utilisateur peut avoir une interaction.

La méthode d'affichage par liste de compétences permet de limiter la recherche par les compétences déjà présentes dans la base de données. Il est envisageable de rajouter la classification JEL et de fusionner l'ensemble.

L'ensemble des informations importantes sont affichées en deux parties: un tableau résumant les informations personnelles des économistes et un autre où sont affichés les articles et leurs liens. Ci-dessous un exemple de recherche d'économistes.



Les requêtes se font à l'aide des deux barres de cherche et de la liste des compétences. Les institutions étant très nombreuses, un passage de mots-clés est suffisant pour trouver l'institution et reste le plus efficace.

Pour faciliter la lecture du résultat de recherche, un clic sur le titre de la colonne classifie le contenu par ordre alphabétique.

Il est possible de double-cliquer sur un économiste lorsqu'une recherche mène à plusieurs économistes et qu'on souhaite consulter ses articles. En double-cliquant sur un article, une page web s'ouvre sur l'article. La navigation est simplifiée et évite de devoir copier le lien et chercher l'article soi-même.



Ci-dessous un double clic sur un économiste de la recherche précédente. Afin de faciliter la consultation de la liste des publications, l'information de la date de publication a été ajoutée.



Il est également possible d'enregistrer une recherche sous forme de fichier CSV en appuyant sur le bouton 'exporter la recherche'. Deux fichiers seront créés pour les articles et les économistes.

#### 3 - Limites et atouts

Les limites des solutions retenues sont les suivantes :

- Un grand nombre de requêtes est lancé sur le site hébergeant la base de données. Afin de ne pas se laisser saturer par un grand nombre de requêtes, il est impossible d'en faire plus de 100 en quelques secondes. Nous sommes donc restreints à l'affichage des données personnelles de 100 économistes maximum. Cependant une recherche plus précise permettra à l'utilisateur d'obtenir le résultat souhaité.

Les atouts des solutions retenues sont les suivants :

- Pour une recherche relativement précise, la recherche prend très peu de temps
- Seul les menus et les classements du site étant sauvegardés dans notre solution, le programme est **très léger**.
- La recherche n'ayant que 100 requêtes maximum à faire, le temps de recherche est borné entre moins d'une seconde à une dizaine de secondes
- La mise à jour des données téléchargées est faisable rapidement (~ 30 secondes) et peut être faite quotidiennement. Ainsi le **moteur de recherche est facilement et rapidement actualisable.**