Développement d'une application pour l'automatisation de recherche de jeux de données en bioinformatique

Fabien Duchateau, Guillaume Launay
Université Claude Bernard Lyon 1
prénom.nom@univ-lyon1.fr

Contexte : dans le cadre de l'UE *base de données pour la bioinformatique* enseignée en Master BIOINFO, les étudiant.e.s réalisent un projet qui consiste à intégrer différents jeux de données dans une même base de données, par exemple pour produire ou vérifier des connaissances. Ces jeux de données sont hétérogènes (modèles, formats, vocabulaires différents) et leur intégration s'apparente à un *workflow*. Par exemple, en partant du génome d'un virus (format GenBank converti en SQL), il faut intégrer son protéome (format XML), les interactions avec des protéines humaines (format JSON), puis intégrer des informations supplémentaires sur les protéines humaines (format RDF) et enfin sur le génome humain (requêtes REST). Pour concevoir de nouveaux sujets de projet, l'une des difficultés est la recherche des différents jeux de données pour l'ensemble du *workflow* (e.g., on peut trouver un génome et son protéome, mais aucune interaction, et donc le workflow est incomplet). Une étude de faisabilité, réalisée l'an dernier par des étudiantes de L3 en projet TER, a montré que l'automatisation de recherche de jeux de données est possible. Cependant, le script produit pendant ce TER est inutilisable.

Objectif : développer une application Python qui permet d'automatiser la recherche de jeux de données cohérents sur les interactions entre protéines. Les fonctionnalités attendues sont les suivantes (liste non exhaustive) :

- Détection de workflows (avec un maximum de formats et modèles) à partir d'une source de données (fichier d'interraction, ou fichier d'un génome). Les tentatives, quel que soit le résultat (workflow complet, partiel, etc.), seront stockées (fichier JSON ou base de données);
- Développement d'une interface web pour visualiser et gérer les workflows détectés;
- Implémentation de script(s) de conversion (e.g., format GenBank vers SQL ou JSON);

Langages et outils :

- Programmation en Python
- SGBD relationnel (e.g., SQLite)
- Utilisation de git pour le suivi