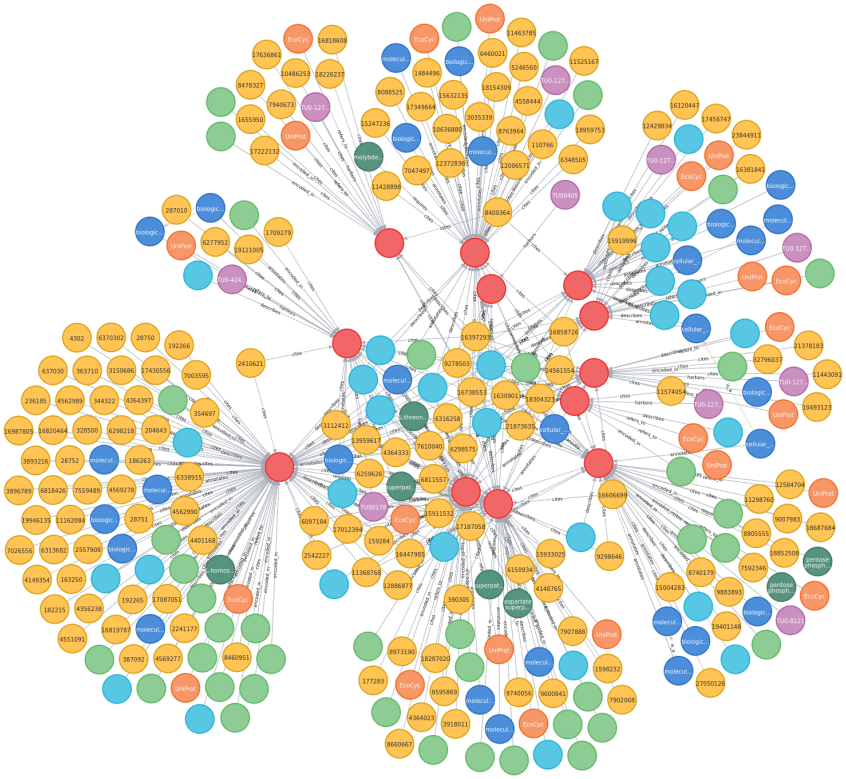
Escherichia coli
Gestion de données non structurées
Applications post-génomiques



Auteur : Lou Duron
Encadrant : Roland Bariot

Introduction :

Plusieurs ensemble de gènes d'intérêt chez Escherichia Coli ont été identifiés. L'objectif de ce projet est de déterminer l'existence d'un lien, si tant est qu'il existe, entre ces gènes. Pour cela nous nous intéresserons aux processus biologiques, fonctions moléculaires ou localisations sub-cellulaires auxquels ils participent afin de trouver ce qui les relient.

Dans un premier temps, l'ensemble du protéome correspondant à Escherichia Coli à été récupéré sur UniProt. Puis, pour chaque gènes ainsi récupéré sont associées des informations concernant les domaines InterPro, les Keyword, les termes GO, les voies métaboliques, les unités de transcription et les publications citant le gène.

L'ensemble de ces données sont ensuite intégrées dans une base de données orientée graph (Neo4j)., dont voici une illustration :

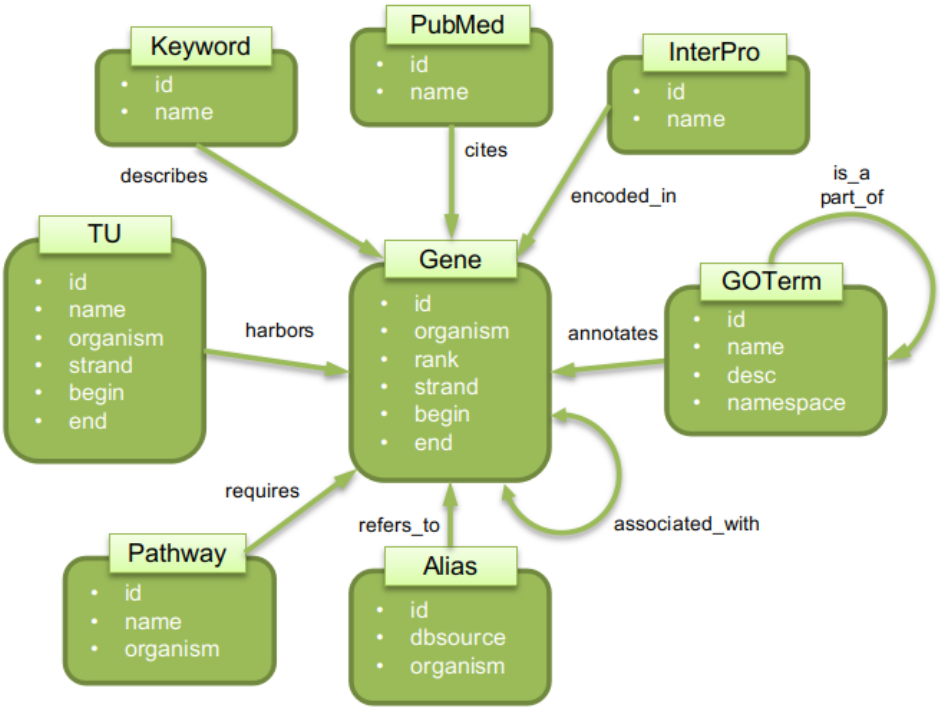


Fig.1 : Organisation de la base de données

Ensuite, un script codé en Python permet d'effectuer la recherche d'enrichissement d'un ensemble de gènes donnés en interrogeant la base de données précédemment créée. Cette méthode inclut 4 mesures différentes pour évaluer l'enrichissement du set , la loi binomial, la couverture, le chi 2 d'indépendance et la loi hypergéométrique.

Afin de comparer ces différentes méthodes, un benchmark a été effectué sur un jeu de données dont l'enrichissement était connu.

Enfin, une synthèse de l'analyse sur un ensemble de gènes d'intérêt a été réalisée.

Organisation du projet :

La première partie du projet, consistant à l'intégration des données dans la base de données, a été réalisée en TP et en binôme avec Victoria Fathi. Un GitLab commun de l'ensemble du travail effectué dans l'UE d'Intégration de Données Hétérogène est disponible en public (Ici).

La seconde partie du projet, l'analyse de l'enrichissement et comparaison des méthodes, est personnelle. Afin de simplifier l'accès au données et de pratiquer l'utilisation de branches git, il a été décidé de créer deux branches au projet, chacune correspondant aux travaux d'un élève.

Les instructions pour naviguer entre les branches sont disponibles dans la documentation de la branche principale. Pour accéder aux données concernant ce rapport veuillez vous reporter à la branche "lou".

Intégration et préparation des données :

Dans cette partie, la méthode utilisée pour préparer et intégrer les données à la base de données est décrite.

Génération des ensemble cibles

Uniprot :

Le protéome correspondant à E. coli K-12 MG1665 à été récupéré sur Uniprot (release 2021_03) au format Tab-separated en sélectionnant les colonnes suivantes :

- Entry name
- Gene names (ordered locus)
- Gene Ontology IDs
- Interpro
- Keywords

EcoCyc :

Les voies métaboliques (pathways) et les unités de transcription (TUs) associées au protéome de E. coli K-12 MG1665 ont été récupérées sur EcoCyc au format tabulé.

PubMed :

Les publications relatives à chacun des gènes ont été récupérées à partir de l'identifiant des gènes auprès du PubMed à l'aide d'un script Python fourni en TP.

Génération des .tsv :

La génération des fichiers tabulé permettant l'intégration des données dans Neo4J à partir des données d'Uniprot, EcoCyc et PubMed a été effectué via un script R. Le R MarkDown décrivant toutes les étapes ainsi que les fichiers de sorties sont disponibles sur le GitLab du projet (cf. respectivement **get_data.rmd** et dossier **import**).

La version des logiciels et librairies utilisés sont les suivante :

- **R 4.0.5**
- **Rstudio 1.4.1717**
- **Tidyverse 1.3.1**
- **Jsonlite 1.7.2**

Intégration dans Neo4j

Pour l'intégration des données, nous utilisons Neo4j, un système de gestion de base de données orienté graphe. Les données générées dans la partie précédente ont été intégrées directement via **Neo4j Desktop 1.4.8** à l'aide de l'interface Neo4j Browser. Le système de gestion de base de données utilisé est **Graph DBMS 4.3.1**. Un document résumant toute les étapes de l'intégration ainsi que l'ensemble des données au format .dump sont disponible sur le GitLab du projet (cf. respectivement **Intégration_des_donées_Neo4j.html** et **BDD.dump**).

Statistiques descriptives

La base de données est constituée de 104 463 noeuds répartis de la manière suivante :

- Gene : 4302
- Keyword : 385
- GOTerm : 43850
- Interpro : 6589
- Pathway : 442
- TU : 3697
- PubMed : 35320
- Alias : 8879

Les liens entre les noeuds décrits dans l'introduction sont au nombre de 1 348 064 et sont répartis de la manière suivante :

- annotates : 21093
- cites : 101526
- describes : 27006
- encoded_in : 15099
- harbors : 6112
- is_a : 70881
- part_of : 7096
- refers_to : 8549
- requires : 2682
- assodiated_with : 1088020

En moyenne, chaque gène est associé à :

- 6.29 keywords (maximum : 21)
- 5.44 GOTerms (maximum : 21)
- 3.73 domaines Interpros (maximum : 17)
- 2.55 voies métaboliques (maximum : 21)
- 1.48 unités de transcription (maximum : 13)
- 23.60 publications PubMed (maximum : 437)
- 263.76 autre gènes [Co-expression] (maximum : 1350)

Voici un exemple schématisant l'ensemble des nœuds associé à une gène, à l'exception des autres gènes co-exprimés avec celui-ci, trop nombreux pour avoir un affichage interprétable.

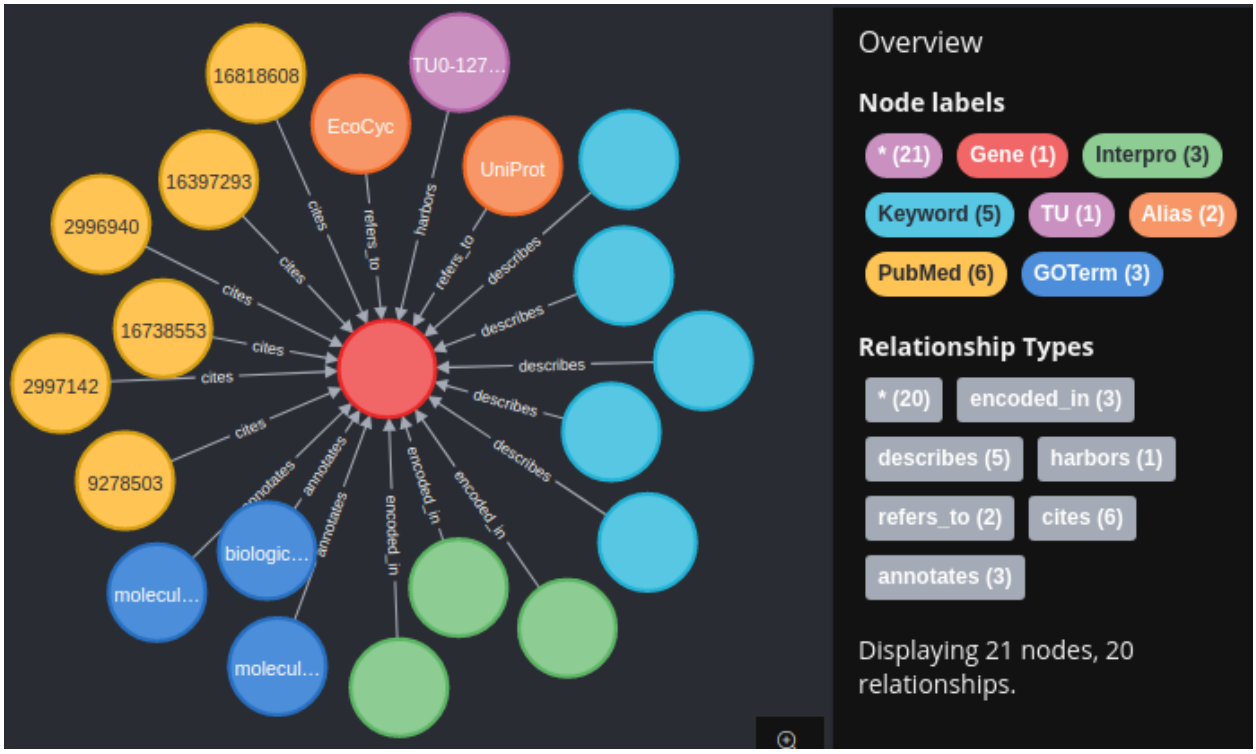


Fig.2 : Exemple d'un gène et ces nœuds associés. Résultat de la requête :
"**MATCH (g:Gene{bnumber:'b0016'})<-[r]-(n) WHERE type(r) <> 'associated_with' RETURN g, r, n**"

Ajout de fonctionnalités :

Afin d'effectuer la recherche d'enrichissement, un script en Python a été fourni en TP. Ce dernier a été modifié afin d'y ajouter plusieurs fonctionnalités. Ce script est écrit en **Python 3.6.13** et utilise les librairies suivantes :

- **argparse** (gestionnaire d'argument)
- **numpy** (calcul scientifique)
- **pprint** (meilleur print)
- **time** (temps d'execution)
- **scipy 1.5.3** (analyse statistique)
- **py2neo 2021.2.3** (Client Neo4j)

Le script (**Enrichment.py**), une documentation détaillée et un exemple d'utilisation sont disponibles sur la branche 'lou' du GitLab du projet. De plus, le script a été commenté pour faciliter sa compréhension.

Les fonctionnalités ajoutées au script sont les suivantes :

- Nouvelles mesures de similarité :
    - Coverage
    - Chi 2 d'indépendance
    - Loi hypergéométrique
- Benchmark de comparaison des mesures de similarité (détaillé plus bas)
- Retour utilisateur et temps d'exécution
    - Permet à l'utilisateur d'avoir un retour sur la progression de l'exécution (qui peut prendre un certain temps selon les arguments) et d'avoir le temps d'exécution final.
- Gestion des exceptions
    - L'ensemble des exceptions (mauvais arguments, erreurs dans la base de données ou fichier d'entrée au mauvais format ou vide) ont été traitées.

Comparaison des mesures disponibles

Afin de comparer les mesures disponibles, il a été choisi de faire un "Benchmark". L'objectif étant de constituer des jeux de données dont l'enrichissement est connu, puis de comparer l'enrichissement de ces jeux de données avec les résultats des différentes méthodes.

Pour créer le jeu de donnés test, nous avons récupéré sur UniProt l'ensemble des séquences vérifiées correspondant à l'organisme Escherichia coli (strain K12) associés au mots clefs "Helicase" (**helicase AND reviewed:yes AND organism:"Escherichia coli (strain K12) [83333]**). À partir de ces données, un script R a permis de mapper les identifiants aux ensembles cibles et de générer des fichiers tabulé pour l'analyse. Ce script est disponible sur le GitLab (cf. **get_benchmark_data.r**).
L'analyse du jeu de données test à été fait sous R et est disponible sur le GitLab (cf. **benchmark_analysis.Rmd**).

Le jeu de test à ensuite été utilisé pour faire une analyse d'enrichissement avec le script Python. Afin d'appréhender l'impact de la taille du jeu de test, plusieurs analyses ont été réalisées sur des jeux de tailles différentes, soit en tronquant le jeu de données ou en ajoutant des gènes piochés aléatoirement dans la base de données. Voici les conclusions de cette analyse :

- Comme attendu, pour un grand nombre de gènes, la loi hypergéométrique s'approche de la loi binomiale.
- Pour les petits jeux de données, le coverage donne des résultats faussés.
- De manières générales, sur des très grand jeu de données (>100 gènes) les résultats des 4 méthodes sont identiques.
- Concernant la significativité des résultats : la p-value du chi 2 d'indépendance est immanquablement plus faible que celle des autres méthodes. Il est important de prendre cette information en considération lors du choix du seuil alpha.

Pour aller plus loin, il pourrait être intéressant d'implémenter d'autres mesures pour évaluer l'enrichissement. Des méthodes standardisées de benchmarking pour l'analyse de l'enrichissement de set de gêne commencent à voir le jour (*Ludwig et al. 2020* [1]) .

Analyse de l’ensemble de gènes fourni

Un ensemble de gènes d’intérêt a été fourni à chaque étudiant dans l’optique d’être analysé par les méthodes d’enrichissement implémentées. Ce rapport traitera l’analyse de l’ensemble **set.M2.21.txt**.

Dans un premier temps, si l’on se restreint au 10 premiers résultats significatifs, l’analyse montre que les 4 méthodes donnent globalement les mêmes résultats concernant l'enrichissement en Keyword, domaines InterPro, les voies métaboliques, les unités de transcription et les publications PubMed. (cf.Table 1). Cependant, l’enrichissement en Termes GO donne des résultats différents selon la méthode utilisée (cf Table 2). Pour pousser l’analyse plus loin, il a été choisi de garder l’ensemble les termes GO les plus significatifs pour l’ensemble des méthodes, soit 18 termes GO.

Une analyse visuelle obtenue avec Neo4j Desktop permet de représenter les liens entre l’ensemble de gènes d’intérêts et les résultats les plus significatifs en termes d'enrichissement. (cf. Figure supp. 1 à 6). Cette approche permet de se rendre compte que l’enrichissement en domaine InterPro (Figure supp. 2) et en unité de transcription (Figure supp. 5) est limité et que la plupart des gènes n’ont pas de lien avec les résultats les plus significatifs. D’autre part, l’analyse d’enrichissement en termes de Keyword (Figure supp. 1), montre un cluster de gènes qui inclut la majorité de l’ensemble de gènes d'intérêt fourni. Enfin, l’enrichissement en Pathway, PubMed et GOTerm montre une structure en cluster composé à chaque fois de 2 clusters.
Il est possible d'émettre l’hypothèse que l’ensemble de gènes d'intérêt est caractérisé par 2 groupes de gènes reliés par un processus biologique commun.

Une analyse en profondeur de l’enrichissement en Keyword, termes GO, voies métaboliques et publications PubMed citant les gènes, permet d’identifier que le jeu de données est enrichie avec gènes liés à des processus biologique en relation avec le cycle de Krebs, la Glycolyse et la synthèse des Lipopolysaccharides.

La glycolyse correspond à la voie métabolique qui convertit le glucose en acide pyruvique et alimente ainsi le cycle de Krebs, appelé aussi cycle des acides tricarboxyliques (*Rocha et all. 2010* [2]). De plus, les Lipopolysaccharides, composant majeur de la membrane externe des bactéries gram -, vient réguler l’activité des la glycolyse et indirectement le cycle de krebs (*Niamh C. & Luke A. 2018* [3], *Ruyuan et all. 2018* [4], *Fanta et all. 2020* [5])

Keyword	InterPro	TU	Pathway	PubMed
Tricarboxylic acid cycle	IPR002201	TU00524	GLYCOLYSIS-TCA-GLYOX-BYPASS	17362200
Glycolysis	IPR020557	TU0-7161	LPSSYN-PWY	12045108
Lipopolysaccharide biosynthesis	IPR015806	TU0-7162	TCA	7504166
Acetylation	IPR011167	TU0-6702	TCA-GLYOX-BYPASS	12963713
Metal-binding	IPR015793	TU00017	PWY-5484	9791168
Glycosyltransferase	IPR035474	TU00018	HEXITOLDEGSUPER-PWY	1385388
Direct protein sequencing	IPR000701	TU00103	GLYCOLYSIS	9004408
Transferase	IPR004800	TU0-42509	GLYCOLYSIS-E-D	1938935
3D-structure	IPR015795	TU0-42499	KDO-NAGLIPASYN-PWY	9720032
Allosteric enzyme	IPR002495	TU0-8081	LIPA-CORESYN-PWY	1624462

Table 1 : 10 Premiers résultats significatifs de l'enrichissement en Keyword, InterPro, TU, Pathway, PubMed. Ces résultats sont communs au 4 méthodes, cependant l'ordre d'apparition (significativité) peut varier d'une méthode à l'autre. L'ordre obtenu avec la Loi Binomiale a été conservé ici.

Loi Binomiale et Loi Hypergéométrique	Coverage et Chi 2 d'indépendance
GO:0005975	GO:0006099 *
GO:1901135	GO:0009312 *
GO:0016051	GO:0006096
GO:0009312 *	GO:0046364
GO:0009311	GO:0006757
GO:1903509	GO:0006165
GO:0003824	GO:0046939
GO:0006091	GO:0009135
GO:0006099 *	GO:0009179
GO:0044238	GO:0046031

Table 2 : 10 premiers résultats significatifs de l'enrichissement en termes GO en fonction des méthodes utilisées. Les astérisques indiquent les termes GO commun à l'ensemble des méthodes

Bilan personnel sur le projet et l'UE

J'ai trouvé que l'utilisation de base de données NoSQL pour intégrer des données hétérogène est très intéressante. En effet, avec le développement du Big-data, des bases de données non traditionnelles peuvent, dans certains cas, permettre d'analyser des données d'origine et de nature différentes. Notamment les bases de données orienté graph qui permettent une visualisation des données, apportant ainsi des informations sur des tendances globale, invisible sur les bases de données SQL.

Je trouve le projet en lui-même très complet car il balaie un grand nombre d'aspects différents d'une analyse d'enrichissement de gêne. De l'intégration des données à l'exploration d'un jeu de gènes, en passant par le conception d'un script d'analyse.

La seule suggestion/critique que j'aurais, serait de laisser le choix du langage de programmation à l'élève. En effet, bien que pratiquer plusieurs langages différents est un bon exercice, je pense qu'il préférable pour un élève de maîtriser en profondeur un langage plutôt que de connaître les bases de plusieurs.

Références

1. Ludwig ar all. Toward a gold standard for benchmarking gene set enrichment analysis 2020 *Brienfing in Bioinformatics*

2. Rocha et all. Glycolysis and the Tricarboxylic Acid Cycle Are Linked by Alanine Aminotransferase during Hypoxia Induced by Waterlogging of *Lotus japonicus 2010, Plant physiology*

3. Niamh C. & Luke A. A Role for the Krebs Cycle Intermediate Citrate in Metabolic Reprogramming in Innate Immunity and Inflammation. *Front. Immunol*

4. Ruyuan et all. Investigation into Cellular Glycolysis for the Mechanism Study of Energy Metabolism Disorder Triggered by Lipopolysaccharide - 2018, *Toxins*

5. Fanat et all. Metabolic reprograming of LPS-stimulated human lung macrophages involves tryptophan metabolism and the aspartate-arginosuccinate shunt. 2020 *Plos one.*

Figures supplémentaires

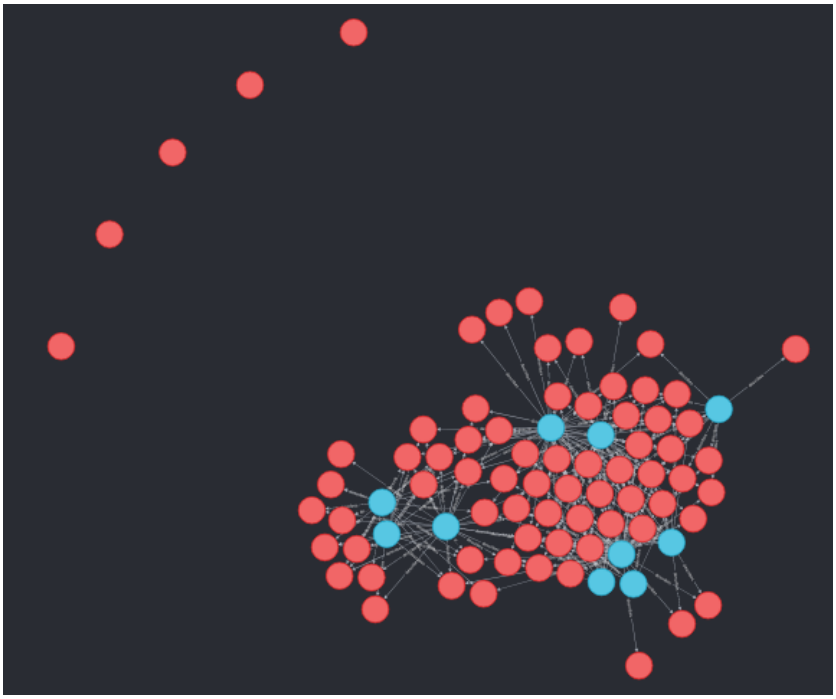


Figure supp. 1 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 10 premiers résultats les plus significatifs de l'enrichissement en Keyword (en bleu)

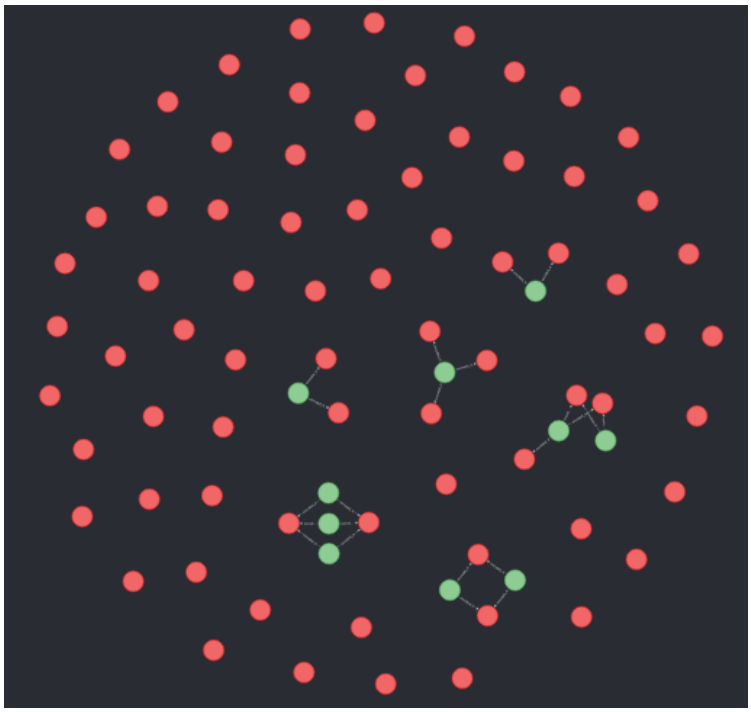


Figure supp. 2 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 10 premiers résultats les plus significatifs de l'enrichissement en domaine InterPro (en vert)

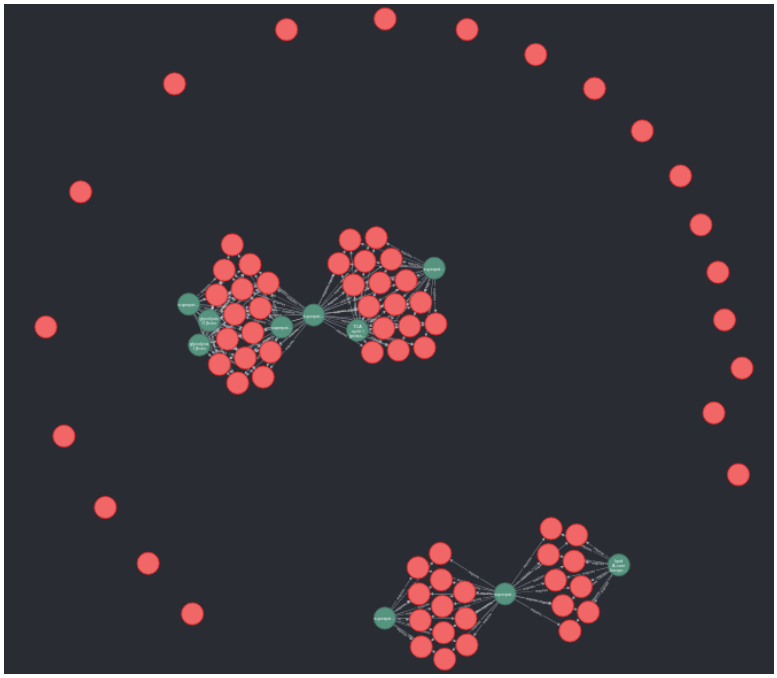


Figure supp. 3 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 10 premiers résultats les plus significatifs de l'enrichissement en Pathway (en vert)

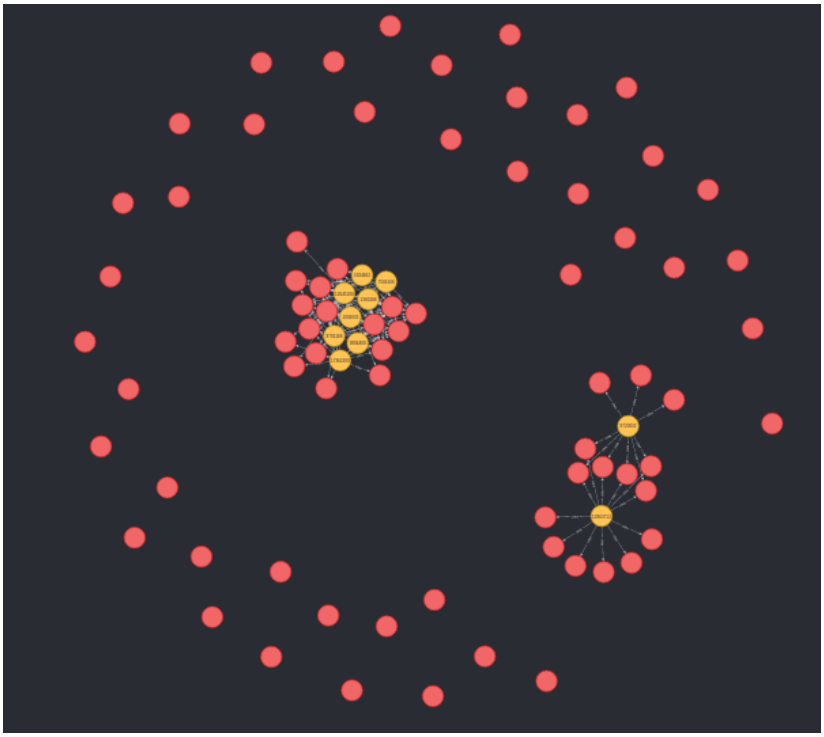


Figure supp. 4 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 10 premiers résultats les plus significatifs de l'enrichissement en Publications Pubmed (en jaune)

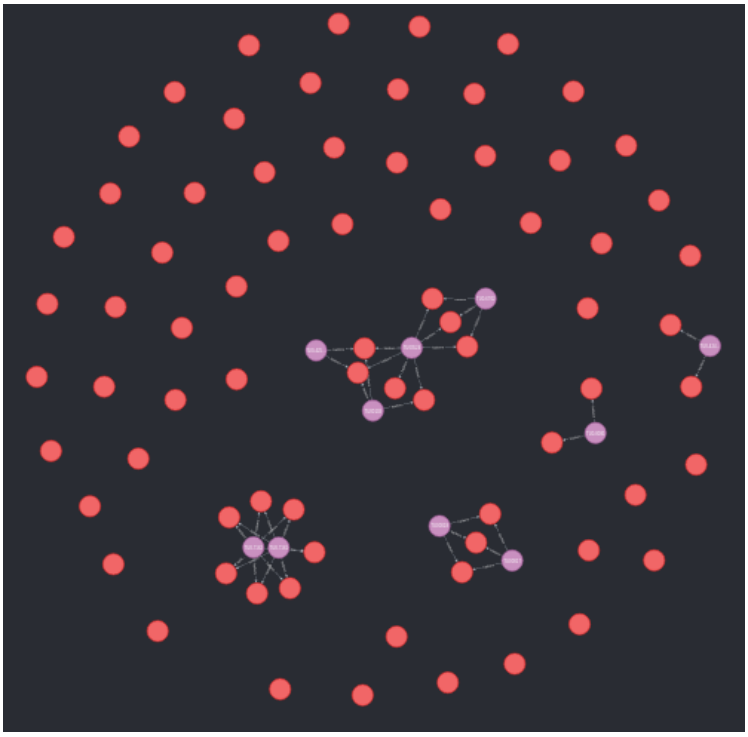


Figure supp. 5 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 10 premiers résultats les plus significatifs de l'enrichissement en TU (en violet)

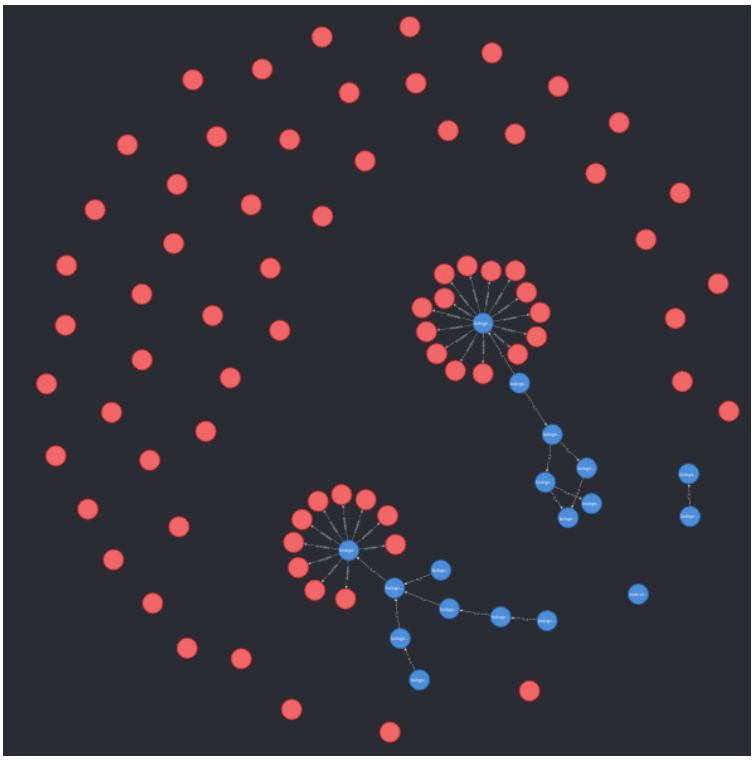


Figure supp. 6 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 18 premiers résultats les plus significatifs (toute méthodes) de l'enrichissement en GOTerm (en bleu)

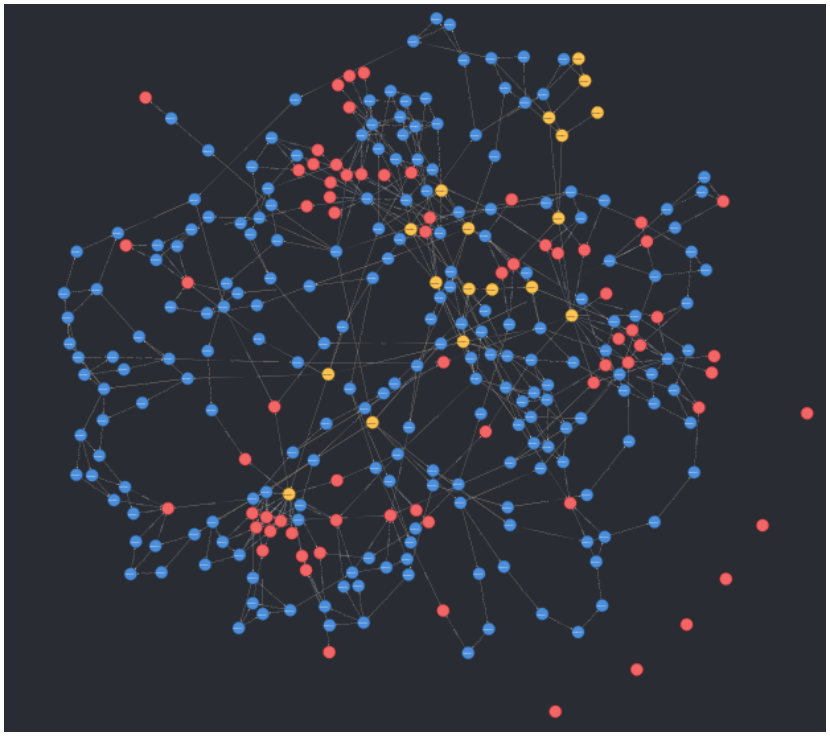


Figure supp. 7 : Représentation des relations entre les gènes (en rouge) du set fourni avec les 18 premiers résultats les plus significatifs (toute méthodes) de l'enrichissement en GOTerm (en jaune) et les GOTerm intermédiaires (en bleu)