Mémoire

Auteur : Harmel, Marie
Promoteur(s) : Baurain, Denis; 25326
Faculté : Faculté des Sciences
Diplôme : Master en bioinformatique et modélisation, à finalité approfondie
Année académique : 2023-2024
URI/URL : http://hdl.handle.net/2268.2/21069

Université de Liège
Faculté des Sciences : Bioinformatique et Modélisation

Découverte de nouvelles espèces de cyanobactéries basales

Assemblage métagénomique d'échantillons environnementaux et annotation automatique de leurs biomes par Zero-Shot Classification

Marie HARMEL

Promoteur : Denis BAURAIN

Copromoteur : Luc CORNET

Unité de Phylogénomique des Eucaryotes

2023-2024

Mémoire présenté en vue de l'obtention du grade de Master en Bioinformatique et Modélisation, à finalité approfondie.

Table des matières

1 Remerciements

2 Liste des abréviations

3 Résumé

4 Introduction **1**

4.1 Cyanobactéries : écologie, rôle historique et innovations biotechnologiques 1

4.2 Emergence de la photosynthèse et phylogénie 3

4.3 Métagénomique et accession 7

4.4 Bases de données et STAT 8

4.5 Word embeddings et zero-shot classification 11

5 Objectifs **14**

6 Matériel et méthodes : Données brutes **14**

6.1 BigQuery 14

6.1.1 Sélection des identifiants de cyano basales 14

6.2 parse.pl 15

6.3 Assemblages publics 16

6.4 EDirect 16

6.5 Concaténation des champs 16

6.6 tags_auto.pl 17

7 Matériel et méthodes : Zero-Shot Classification **17**

7.1 Choix du dictionnaire initial : ENVO 17

7.2 Clusterisation 17

7.2.1 Transformation en *word emeddings* 17

7.2.2 Réduction de la dimensionalité 18

7.2.3 Graphes et détermination de k 18

7.2.4 Clustering hiérarchique et *K-Means* 19

7.2.5 Choix du meilleur représentant de chaque *cluster* 20

7.3 Choix du modèle de *Zero-shot Classification* 20

7.3.1 Identification des étiquettes non environmentales 20

7.3.2 Détermination du seuil de score 21

7.3.3 ZSC sur totalité des textes 22

8 Matériel et méthodes : Phylogénie **22**

8.1 GENERA 22

8.1.1 Download, assemblage, binning et annotation des génomes de cyanobactéries publics . . . 24
8.1.2 ANI . . . 24
8.1.3 Orthology . . . 24
8.1.4 Phylogénies . . . 25
8.1.5 Choix des génomes et MAGs pour la deuxième phylogénie . . . 25
8.2 ORPER . . . 25
8.3 Machines . . . 26

9 Résultat : Récupération des données brutes 26
9.1 Récupération d'accessions susceptibles de contenir des cyanobactéries basales . . . 26
9.1.1 Récupération des identifiants taxonomiques . . . 28
9.1.2 Ajout de conditions à la requête . . . 29
9.1.3 Champs supplémentaires environnementaux . . . 29
9.2 Suppression d'accessions retrouvées dans les assemblages publics . . . 30
9.3 Récupération de métadonnées complémentaires . . . 30
9.4 Gestion des données . . . 31
9.4.1 Concaténation et nettoyage des champs . . . 31
9.4.2 Standardisation des valeurs numériques . . . 32

10 Résultat : Classification automatique 33
10.1 Choix du dictionnaire initial . . . 33
10.2 Prérequis afin d'obtenir un bon *clustering* . . . 34
10.2.1 Recherche du nombre optimal de *clusters* . . . 34
10.2.2 Choix du type de *clustering* . . . 38
10.3 Relecture du dictionnaire : Curation des termes non pertinents . . . 41
10.3.1 Identification des termes non environnementaux par ZSC . . . 41
10.3.2 Curation manuelle . . . 41
10.4 Classification automatique sur l'entièreté des données brutes . . . 42
10.4.1 Détermination du score minimum qu'un tag doit obtenir pour être retenu dans la classification . . . 42
10.4.2 Résultats de l'étiquetage automatique . . . 42

11 Résultat : assemblage et phylogénie 44
11.1 Assemblage et identification GTDB . . . 44
11.2 Génomes de référence . . . 45
11.3 Contrôle qualité . . . 45
11.4 ANI . . . 45
11.5 Orthologie . . . 46

11.6 Phylogénie . . . 47
11.7 Seconde phylogénie . . . 48
11.7.1 ORPER . . . 48

12 Discussion **52**
12.1 Assemblages publics . . . 52
12.2 Clusterisation des termes environnementaux . . . 52
12.3 Filtrage des bins . . . 52
12.4 ANI . . . 53
12.5 Arbres phylogénétiques . . . 53

13 Perspectives **54**

14 Références **56**

15 Annexes **57**
15.1 Annexe AAAA : Requête SQL . . . 57
15.2 Annexe BBBB : infos sur identifiant, lignée et clade . . . 59
15.3 Annexe A : download_make_tree_Cyanobacteriota_NCBI_taxo.py . . . 60
15.4 Annexe B : parse_nested_structure.pl . . . 61
15.5 Annexe C : EDirect_metadata_retrieval.pl . . . 63
15.6 Annexe D : fuse_columns.pl . . . 64
15.7 Annexe E : predined_tags.pl . . . 65
15.8 Annexe F : elbow_silhouette_plot.py . . . 69
15.9 Annexe G : gap_standard_error.R . . . 71
15.10 Annexe H : cluster_dendro_best.py . . . 72
15.11 Annexe I : Word_cloud.R . . . 76
15.12 Annexe J : choose_best_representive.py . . . 77
15.13 Annexe K : identify_non_environmental_tag.py . . . 79
15.14 Annexe L : generate_dynamic_tags_score.py . . . 80
15.15 Annexe M : download_assembly_gtdb.sh . . . 81
15.16 Annexe N : premier arbre phylogénétique . . . 84
15.17 Annexe P : arbre ORPER . . . 85
15.18 Annexe Q : librairies utilisées . . . 86

Références **87**

1 Remerciements

Je souhaite exprimer tout d'abord ma profonde gratitude envers le Professeur Denis Baurain pour son écoute, sa bienveillance et ses précieux conseils tout au long de mon mémoire. Mes remerciements vont également à Luc Cornet, mon co-promoteur, qui m'a offert une vision éclairée des nouveaux enjeux bioinformatiques. Ensemble, ils m'ont fourni un cadre de travail idéal et un encadrement irréprochable.

Je tiens également à remercier chaleureusement tous les membres de l'unité de Phylogénomique des Eucaryotes pour leur expertise, en particulier Coralie, Valérian, Mick et Benoît, toujours prêts à offrir leur aide.

Enfin, je ne saurais oublier mes proches, mes très chers parents pour leur soutien indéfectible tout au long de l'année, ainsi que mon frère François, Camille, et mes amis Kiril et Cléa pour leur encouragement et leur assistance.

2 Liste des abréviations

— ADN : Acide DésoxyriboNucléique
— ANI : Average Nucleotide Identity
— ARN : Acide RiboNucléique
— BFO : Basic Formal Ontology
— CSV : Comma-Separated Values
— ChatGPT : Chat Generative Pre-Trained Transformer
— EDirect : Entrez Direct
— ENVO : EnVironment Ontology
— EPS : ExoPolySaccharides
— ES : Endosymbiose secondaire
— FoodOn : Food Ontology
— Ga : Giga-Annum (milliard d'années)
— GCP : Google Cloud Platform
— GOE : Great Oxidation Event
— GTDB : Genome Taxonomy DataBase
— Go : giga octet
— HGT : Horizontal Gene Transfer
— HMM : Hidden Markov Model
— LLM : Large Language Model
— LPCA : Last Photosynthetic Common Ancestor
— MAG : Metagenome Assembled Genome
— MNLI : Multi Natural Lnaguage Inference
— NCBI : National Center for Biotechnology Information
— NEO : Neo Oxidation Event
— NER : Named Entity Recognition
— NLI : Natural Language Inference
— NLP : Natural Language Inference
— OG : Orthologous Gene
— ORPER : ORganism PlacER
— PAL : Pressure Atmospheric Level
— PC : Principal Component
— PCA : Principal Component Analysis
— PS : PhotoSystème
— RC : Reactional Center
— SLM : Statistical Language Model
— SQL : Structured Query Language

— SRA : Sequence Read Archive
— STAT : Sra Taxonomy Analysis Tool
— UBERON : UBER-anatomy ONtology
— VGT : Vertical Gene Transfer
— ZSC : Zero-Shot Classification
— iTOL : interactive Tree Of Life
— mB : mega Base
— pb : paire de bases
— WGS : Whole Genome Sequencing

3 Résumé

Les cyanobactéries constituent un phylum extrêmement diversifié de procaryotes photosynthétiques, présentes dans une vaste gamme d'environnements. Intensivement étudiées, elles jouent un rôle crucial non seulement sur le plan écologique, de par leur rôles de producteurs primaires et de fixateurs de l'azote atmosphérique, et évolutif, mais également dans diverses applications biotechnologiques. À l'origine de la photosynthèse il y a au moins 2,4 milliards d'années (Sánchez-Baracaldo et al., 2022 ; Stirbet et al., 2019), elles sont également responsables de l'apparition de ce processus chez les eucaryotes photosynthétiques (Cardona et al., 2015 ; Sibbald & Archibald, 2020).

Ce phylum a d'abord été classifié sur la base de critères morphologiques, conduisant à l'émergence de noms décrivant ces différents groupements (Stanier et al., 1979). Cependant, l'utilisation de données génétiques (ARNr 16S) puis génomiques plutôt que morphologiques pour définir les espèces et leur classification a entraîné des révisions successives de cette taxonomie. Ces révisions ont mis en évidence des incohérences entre la taxonomie et la phylogénie et ont posé des problèmes de compatibilité entre différents systèmes (NCBI, GTDB). En raison de la complexité liée au placement des clades les plus basaux, plusieurs taxonomies des cyanobactéries coexistent actuellement, dont une partie reste encore inexplorée, dissimulant ainsi des informations importantes sur le métabolisme des premières cyanobactéries (Cornet et al., 2021).

Dans le cadre de ce travail, une recherche systématique d'échantillons métagénomiques a été entreprise dans les bases de données publiques. Ces échantillons ont été filtrés de manière à ne garder que ceux avec une proportion minimale de *reads* de cyanobactéries basales, cette identification ayant été réalisée par l'outil de classification taxonomique STAT (K. S. Katz et al., 2021).

Les métadonnées associées à ces échantillons ont été récupérées à l'aide de l'outil EDirect (Kans, 2024) et d'une requête SQL. Ces métadonnées, sous forme de textes bruts, ont ensuite été soumises à **bart-large-mnli**, un *Large Language Model* capable de réaliser de la classification Zero-Shot (Wang et al., 2023). Cela a permis de générer des étiquettes décrivant les données environnementales, surpassant d'autres modèles de *Natural Language Processing* grâce à sa capacité à comprendre le sens des textes. Le dictionnaire d'étiquettes a été développé à partir de l'ontologie environnementale ENVO (Mungall, 2015/2024).

Parallèlement, les assemblages de ces échantillons menant à une phylogénie des cyanobactéries présentes a été construite grâce à l'outil GENERA (Cornet et al., 2023). L'objectif étant de combler le fossé évolutif observé entre le phylum des cyanobactéries et son groupe frère non photosynthétique, les Melainabacteria.

4 Introduction

4.1 Cyanobactéries : écologie, rôle historique et innovations biotechnologiques

Les cyanobactéries, souvent désignées sous le nom d'"algues bleues", représentent, contrairement à ce que leur nom pourrait laisser penser, un groupe extrêmement diversifié de bactéries phototrophes. Depuis 2023, elles sont officiellement désignées sous le nom de Cyanobacteriota (Oren et al., 2022) et comptent actuellement 1 classe, 23 ordres, 78 familles, 374 genres et 19 618 espèces [1] au sein de la taxonomie du National Center for Biotechnology Information. Elles font l'objet d'études intensives en raison de leur impact écologique, évolutif et biotechnologique.

Ces organismes sont uniques parmi les procaryotes en raison de leur capacité à réaliser la photosynthèse oxygénique, ce qui les place au début de la chaîne trophique en tant qu'abondants producteurs primaires. Présentes dans une large variété d'écosystèmes, ces bactéries colonisent aussi bien les milieux aquatiques que certains milieux terrestres (Guljamow et al., 2017). Elles peuvent être unicellulaires ou multicellulaires, vivant seules ou en colonies (Vidal & Ballot, 2021), capables de former des structures appelées stromatolithes [2] (Whitton, 2012), ainsi que des proliférations massives dans les milieux aquatiques, appelées blooms (Huisman et al., 2018). Bien que peuplant principalement océans et lacs, les cyanobactéries se retrouvent également dans des environnements plus extrêmes, tels que les sources chaudes (Keshari et al., 2022), les environnements hypersalins (De Philippis et al., 1998) et les roches volcaniques antarctiques (Hidalgo-Arias et al., 2023), grâce à leur haute tolérance face aux conditions environnementales (Barsanti et al., 2008).

Leur impact écologique est considérable. En effet, les cyanobactéries influencent fortement la qualité des écosystèmes aquatiques car, en plus de jouer un rôle clé dans divers cycles biochimiques tels que la production d'oxygène et la fixation de l'azote atomosphérique (Álvarez et al., 2023 ; Capone et al., 2005), elles établissent un certain nombre de symbioses avec des protozoaires, macroalgues, champignons, éponges, etc(Mutalipassi et al., 2021).

Leur prolifération excessive dans des milieux eutrophiques peut entraîner la formation de blooms, provoquant la mort d'organismes en raison des toxines libérées (Piontek et al., 2023) ou, ne laissant pas passer de lumière, provoque la turbidité de l'eau. Par ailleurs, bien qu'elles contribuent à l'oxygénation des milieux via la photosynthèse, cette production d'oxygène peut s'inverser la nuit, créant alors des conditions hypoxiques (Zhang et al., 2022).

Historiquement, les cyanobactéries ont également joué un rôle majeur dans l'évolution de la Terre

1. récupéré à partir de la librairie python ete3 :
ncbi = NCBITaxa()

2. Les stromatolithes sont des formations rocheuses anciennes, souvent fossiles, qui résultent de l'activité collaborative entre les cyanobactéries et d'autres microbes, créant des structures lamellaires ou en couches au fil du temps grâce à leurs processus métaboliques combinés.

telle que nous la connaissons aujourd'hui, étant peut-être responsables de la première extinction de masse et de l'oxygénation massive de l'atmosphère (*Great Oxidation Event*, GOE). Cet événement s'est produit il y a environ 2,4 milliards d'années (Sánchez-Baracaldo et al., 2022), quand la photosynthèse oxygénique que les cyanobactéries avaient acquise a permis de libérer une quantité massive d'oxygène. Celui-ci s'est d'abord vu séquestré dans des minéraux et éléments, comme le fer, qui l'a fait rouiller, donnant naissance aux *banded iron formations* (Yin et al., 2023).

Par la suite, après saturation des minéraux, l'oxygène s'est mélangé à l'eau des océans jusqu'à l'oxyder, ce qui a sûrement entraîné la mort de nombreuses formes de vie procaryotiques aquatiques pour lesquelles cet élément était toxique. Lorsque la concentration d'oxygène est devenue trop élevée, il s'est également diffusé dans l'atmosphère, diluant son principal constituant à ce moment, le méthane (Sessions et al., 2009).

Les preuves de cet événement sont visibles, notamment à travers la concentration accrue en oxygène détectée dans les roches datant de 3 à 2,4 milliards d'années (Murphy & Cardona, 2022). Il est également possible que cet événement ait eu lieu plus tôt, vers 3,3 milliards d'années, comme l'indiquent les découvertes de tapis microbiens, bien que leur lien direct avec l'oxygénation atmosphérique reste encore à préciser (Nishihara et al., 2024).

Cette augmentation ne s'est pas produite immédiatement après l'acquisition de la photosynthèse oxygénique. Les cyanobactéries étaient probablement limitées par la disponibilité du phosphore et la concurrence avec les photoautotrophes anoxygéniques dans les environnements où d'autres donneurs d'électrons étaient disponibles (Sánchez-Baracaldo et al., 2022). Avec le temps, la photosynthèse oxygénique est devenue la principale source d'énergie pour la vie sur Terre, éliminant en grande partie les donneurs d'électrons alternatifs dans la zone photique des océans et créant de nouvelles niches écologiques.

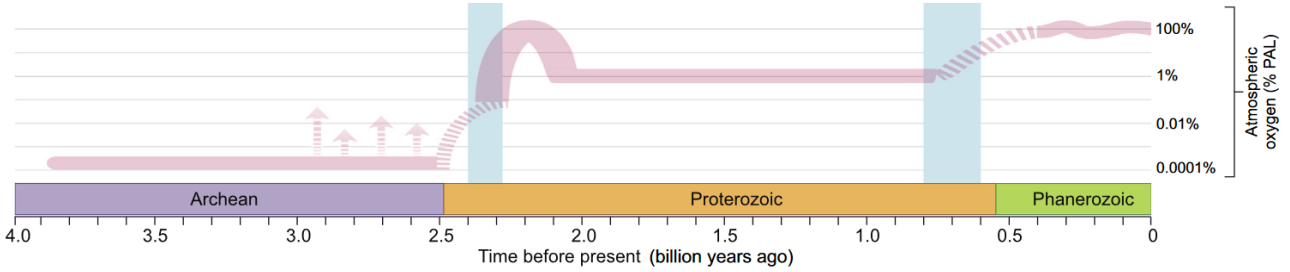

FIGURE 1 : **Niveau d'oxygène dans l'atmosphère par rapport aux concentrations actuelles.** Première bande bleue : GOE, deuxième bande : NEO (neo oxydation event). Augmentation rapide vers 2.4 Ga, atteignant un plateau à 1% . Les flèches roses claires indiquent une montée possible avant 2.4 Ga (Sanchez-Baracaldo et al., 2022).

Au-delà de leur rôle écologique et historique, les cyanobactéries offrent un potentiel biotechnologique remarquable. Elles sont en effet exploitées dans divers domaines, notamment en cosmétique, où elles entrent dans la composition de crèmes solaires, d'hydratants et de soins anti-âge (Morone

et al., 2019). Certaines espèces halophiles sont particulièrement intéressantes en biorestauration, grâce à leur capacité à produire de grandes quantités d'exopolysaccharides (EPS). Ces composés réduisent la tension superficielle de l'huile, améliorant ainsi sa solubilité et sa mobilité dans le sol, ce qui facilite la dégradation des composants pétroliers et le nettoyage des teintures (Moreno et al., 1998). Par ailleurs, les cyanobactéries peuvent accumuler des polyhydroxyalkanoates, qui représentent une alternative écologique aux plastiques non biodégradables (Verlinden et al., 2007).

Enfin, les métabolites secondaires des cyanobactéries possèdent des propriétés antimicrobiennes, antifongiques, algicides et antivirales (Burja et al., 2002 ; Murakami et al., 1991). Elles sont également une source précieuse de composés antioxydants et de coenzymes (Stunda-Zujeva et al., 2023).

4.2 Emergence de la photosynthèse et phylogénie

Leur histoire évolutive, marquée par des événements majeurs comme l'aquisition de photosystèmes et par l'endosymbiose primaire conduisant à la propagation de la photosynthèse dans les lignées eucaryotes, souligne leur intérêt majeur en recherche fondamentale.

La présentation suivante se concentre sur les aspects essentiels de la phylogénie des organismes photosynthétiques, en décrivant les différents clades et leur relation avec la photosynthèse.

L'établissement d'une phylogénie globale et précise des bactéries a longtemps été limité par la difficulté de cultiver de nombreuses espèces en laboratoire. L'émergence des technologies de séquençage et de métagénomique, par opposition à la culture traditionnelle des souches en laboratoire, a permis au cours de la dernière décennie l'accumulation d'une vaste quantité de séquences génomiques, offrant ainsi des analyses phylogénétiques plus précises grâce à une couverture plus riche et plus fidèle de la diversité bactérienne (Parks et al., 2018). Cependant, bien que les phylogénies au sein des phyla soient relativement robustes, des questions demeurent au niveau interphylum, notamment concernant la transmission de la photosynthèse sur deux de ses aspects (Cardona, 2015 ; Nishihara et al., 2024 ; Sousa et al., 2013) :

— **Acquisition verticale ou horizontale** : Les organismes photosynthétiques actuels ont-ils acquis leur machinerie principalement par transfert vertical (modèle de perte sélective) ou par transfert horizontal (modèle de protocyanobactérie) ?
— **Origine de la photosynthèse oxygénique** : La photosynthèse oxygénique dérive-t-elle d'un seul ancêtre anoxygénique ou de deux lignées anoxygéniques distinctes (modèle de fusion) ?

Pour aborder ces questions, une présentation de la phylogénie de différents clades sera faite, représentant la diversité photosynthétique au sein du vivant.

Les procaryotes se divisent en deux super-clades majeurs : les Terrabacteria et les Hydrobacteria

(ou Gracilicutes).

Parmi les neuf clades capables de chlorophototrophie (utilisation du soleil pour synthétiser des sucres par l'intermédiaire de (bacterio)chlorophylle), quatre appartiennent aux Terrabacteria : les Cyanobacteriota, les Chloroflexi (Chloroflexia), les Firmicutes (Heliobacteriaceae), et les Vulcanimicrobiota. Ce regroupement phototrophique forme un cluster phylogénétique distinct, reprenant très peu de phyla non chlorophototrophiques (Nishihara et al., 2024). Cette capacité semble résulter de transferts latéraux de gènes (HGT), pour les cinq phyla restant, plutôt que d'une évolution verticale directe, comme pour les autres (Saini et al., 2021). Il s'agit des Gemmatimonadetes (*Gemmatimonas phototrophica* (Koblížek et al., 2020 ; Y. Zeng et al., 2014)), Myxococcota (Li et al., 2023), Acidobacteriota (Bryant et al., 2007), Chlorobi (Ward & Shih, 2022) et Proteobacteria (alpha, beta, gamma (Brinkmann et al., 2018 ; Bryant & Frigaard, 2006)).

Les Melainabacteriota et les Cyanobacteria sont des phyla frères, partageant un ancêtre commun photosynthétique, bien que les Melainabacteria ne possèdent pas cette capacité. On les retrouve dans des environnements aphotiques tels que les tubes digestifs (humains, termites, koalas, etc), les milieux aquatiques organiques de haute teneur ou les eaux souterraines de bioréacteurs (Soo et al., 2014). Ce phylum inclut plusieurs groupes tels que Vampirovibrionia, Sericytochromatia, et Margulisbacteria (Oliver et al., 2021).

La phylogénie des Cyanobacteria présentée ici s'inspire des travaux de L. Cornet, C. Chen, S. Bettina et J. Komárek (Bettina et al., 2015 ; Chen et al., 2021 ; Cornet et al., 2021 ; Jiri et al., 2014) car les clades basaux étant particulièrement difficiles à positionner avec précision dans une phylogénie globale, plusieurs phylogénies sont toujours actuellement débattues. Bien que leurs résultats montrent des variations, l'ordre évolutif consensuel des clades basaux est le suivant : Gloeobacterales, certaines souches thermophiles de Synechococcales, Gloeomargaritales, Pseudanabaenales, Thermosynechococcus, Acaryochloris, Synechococcus, Prochlorococcus, Leptolyngbya, etc

Une hypothèse pour expliquer cette difficulté, même si elle manque de preuves actuellement, est la suivante : ces clades basaux habitent des niches écologiques extrêmement particulières, comme les lacs de Yellowstone, les surfaces rocheuses ou les environnements à températures extrêmes. Éyant peu en concurrence dans ces niches, ces cyanobactéries ont pu diverger de manière marquée par rapport aux autres cyanobactéries, n'ayant pas la même pression de sélection, rendant leur placement précis dans l'arbre phylogénétique global plus complexe.

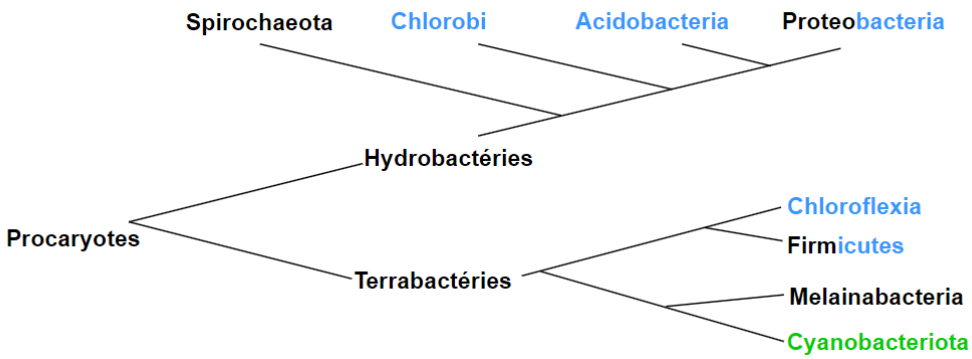

FIGURE 2 : **Phylogénie non exhaustive des différents phyla aux sein des 2 super-clades bactériens présentée sur base des sources du texte précédent.** Les termes bicolores signifient que seule une fraction du clade est photosynthétique. Les Spirochaeota ont été ajoutés pour montrer le fait que l'entièreté du clade des Hydrobactéries n'est pas photosynthétique. Les Gemmatimonadetes et Myxococcota ne sont pas représentés.

Sur les neuf clades de bactéries phototrophes, huit sont anoxygéniques et se caractérisent par l'utilisation d'un seul centre réactionnel (RCI ou RCII) pour la photosynthèse. Cette répartition n'est pas homogène parmi les phyla concernés :

— Le RCI est présent chez les Chlorobi, les Firmicutes et les Acidobacteria.
— Le RCII est utilisé par les Protéobactéries, les Gemmatimonadetes, les Chloroflexi, les Myxococcota et les Vulcanimicrobiota.

Cette distribution non uniforme des centres réactionnels indique une fois de plus qu'il n'y a pas eu de transmission verticale uniforme (VGT) de ces structures à travers les différents groupes bactériens. La photosynthèse chez ces bactéries n'est pas un caractère ancestral mais plutôt une innovation évolutive tardive, résultant de nombreux transferts horizontaux de gènes (HGT) (Ward et al., 2018).

Un article récemment publié offre des perspectives prometteuses afin de mieux comprendre la transmission de la photosynthèse. Selon cette étude (Nishihara et al., 2024), le Dernier Ancêtre Commun Photosynthétique (LPCA) était probablement une Terrabacterie qui possédait deux types de centres réactionnels : un type I similaire aux formes actuelles et un type II primitif. Le papier suggère de manière contre-intuitive que, si la photosynthèse avait été transmise verticalement au sein des Terrabacteria, la branche possèdant des chlorophylles (photosynthèse oxygénique) n'aurait essuyé que trois pertes. En revanche, la branche possèdant des bactériochlorophylles (photosynthèse anoxygénique) en aurait connu quatorze. Ces pertes fréquentes seraient attribuées aux désavantages associés aux conditions oxiques qui auraient pu entraîner la perte de la capacité photosynthétique.

En ce qui concerne les centres réactionnels (RC), le LPCA possédait deux types de RC homodimériques avec des pigments ancestraux. Chez les Bactériochlorophylles, les RC ont évolué au sein de la branche ancestrale, de sorte qu'à présent, chaque organisme photosynthétique n'en possède plus qu'un seul. Les autres espèces hors des terrabactéries ont acquis leurs RC par transfert horizontal.

Pour les Chlorophylles, les RC ont évolué différemment : ils ont perdu leur partie centrale, se sont réorganisés en hétérodimères et ont développé la capacité d'utiliser l'eau comme accepteur d'électrons, ce qui a conduit à l'émergence des photosystèmes PSI et PSII.

L'endosymbiose primaire a permis aux eucaryotes[3] d'acquérir la photosynthèse (Cheong, 2010 ; McFadden, 2014) en incorporant une cyanobactérie, *Gloeomargarita lithophora*, trouvée principalement en eau douce.

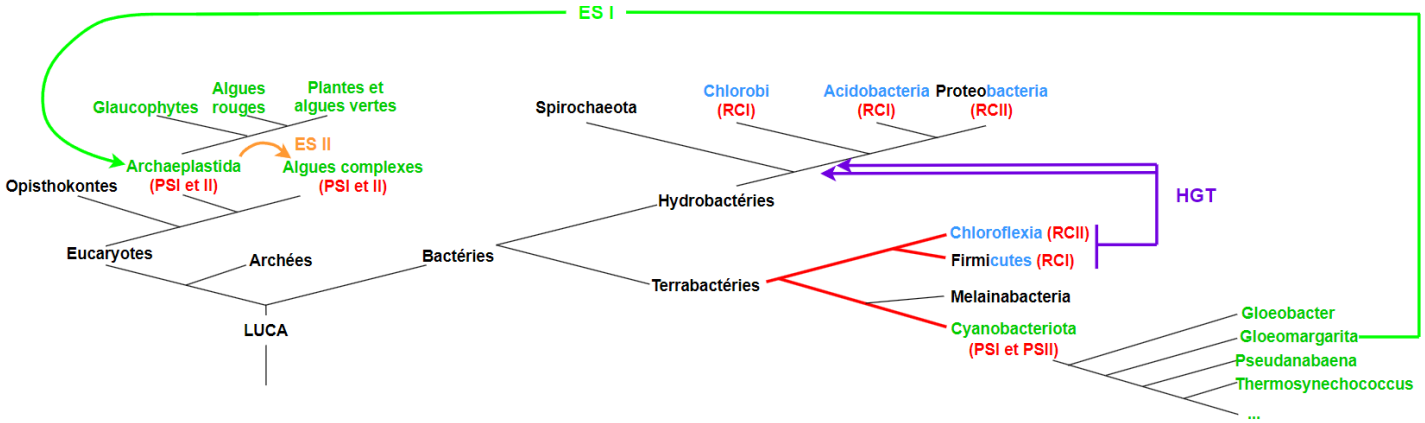

FIGURE 3 : **Phylogénie non exhaustive et mise en évidence de l'origine de la photosynthèse au sein des procaryotes et eucaryotes.** En bleu, organismes photosynthétiques anoxygéniques. En vert, les photosynthétiques oxygéniques. La flèche verte représente l'endosymbiose primaire,avec l'hôte eucaryote et le donneur procaryote. En orange, une des endosymbioses secondaires. En rouge, l'évolution des centres réactionnels. En mauve, les transferts de gènes horizontaux

L'exploration phylogénétique des cyanobactéries basales est toujours un *hot-topic* pour comprendre l'évolution des systèmes photosynthétiques. La structure hétérodimérique des photosystèmes, contrastant avec les centres réactionnels homodimériques des autres photosynthétiques anoxygéniques, souligne une évolution différente, passant par une 'complexification' du centre réactionnel ancestral, sûrement homodimérique(Nishihara et al., 2024).

L'évolution par perte de la photosynthèse chez ces bactéries, plutôt que par ajout, suggère que le modèle de 2024 présenté pourrait ne pas être complet ou vrai, car assez contre-intuitif.

L'écart entre Gloeobacter, dépourvu de thylakoïdes (Mareš, 2013), et Thermosynechococcus, avec une architecture thylakoïdienne complète (Heinz et al., 2016), indique probablement l'existence d'un clade intermédiaire encore non découvert. La découverte de ce clade pourrait clarifier les mécanismes évolutifs des cyanobactéries et affiner notre compréhension des plastes. De plus, il reste encore certaines zones d'ombre également entre les Melainabacteria et Gloeobacter.

3. Parmi les clades photosynthétiques des eucaryotes se trouvent les **Archaeplastida** (glaucophytes, algues rouges, plantes et algues vertes) et les **algues complexes**. Ces derniers possèdent des plastes entourés de trois ou quatre membranes, résultant d'évènements d'endosymbioses secondaires entre un eucaryote hétérotrophe et un eucaryote déjà doté d'un plastide. (Keeling, 2010 ; Pfanzagl et al., 1996)

4.3 Métagénomique et accession

L'approche commence par le prélèvement d'un échantillon environnemental, qui contient une diversité de micro-organismes provenant de milieu étudié. Une fois l'échantillon recueilli, le matériel génétique des organismes présents est extrait, ce qui inclut l'ADN (ou ARN) de tous les micro-organismes dans l'échantillon. Ensuite, cet ADN est séquencé, produisant un grand nombre de reads qui sont assemblés en contigs. Les contigs obtenus sont ensuite analysés en utilisant des techniques bioinformatiques, telles que l'analyse des profils de k-mer—une méthode qui compare des motifs de nucléotides de longueur fixe pour regrouper des séquences similaires (Zielezinski et al., 2017). Grâce à cette analyse, les contigs peuvent être regroupés en unités plus grandes et cohérentes appelées *"bins"*. Chaque bin regroupe des contigs identifiés comme appartenant normalement au même organisme. Idéalement, une *bin* représente un génome assemblé à partir de métagénomes, également appelé **Metagenome Assembled Genome (MAG)**(**Setubal, 2021**).

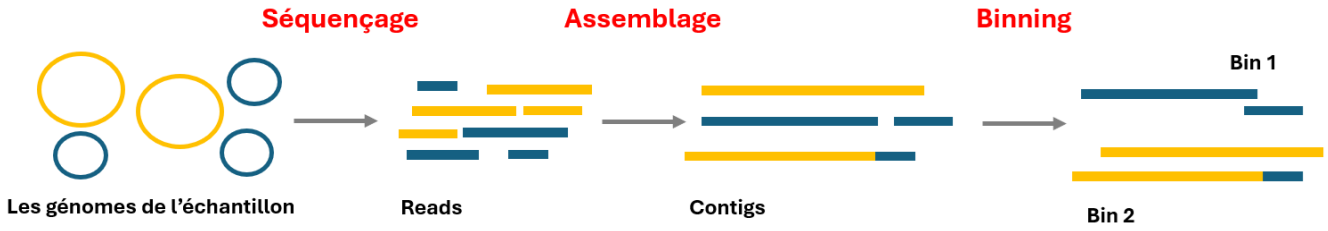

FIGURE 4 : **Étapes de création des MAG.** Adapté du site carpentries-lab.github.io, deux bins sont créées avec des degrés de contamination différents. Les MAGs seront les bins retenues après être passées dans des filtres qualités. Une bin ayant seulement près peu de complétude et un taux élevé de contamination n'est généralement pas considérée comme un génome valide.

Ces données publiques sont disponibles sur le site du NCBI, un groupe de recherche multidisciplinaire qui gère de nombreuses bases de données, telles que celles concernant la taxonomie, les échantillons, les génomes de référence, et les assemblages. Chaque run de séquençage, qu'il soit métagénomique ou non, est stocké dans la SRA (Sequence Read Archive) et se voit attribuer une accession unique sous la forme d'un identifiant SRR, ERR ou DRR [4].

Un run est associé à des métadonnées décrivant ses expérience, échantillon et projet (*BioProject Frequently Asked Questions*, n.d. ; *DRA Update*, n.d. ; *SRA Metadata and Submission Overview*, n.d.). Ci-dessous, l'architecture des différents composants.

— Les **Expériences** sont spécifiques à chaque échantillon et représentent l'unité principale dans la base de données. Elles combinent l'ensemble des métadonnées décrivant un type d'expérience pour un échantillon donné.
— Les **Biosamples** décrivent l'échantillon biologique et peuvent être partagés entre différents projets et expériences.

4. La première lettre de l'accession indique la base de données source : S = NCBI-SRA, E = EMBL-SRA, D = DDBJ-SRA.

— Les **BioProjects** sont des descriptions faites sur un ensemble de données biologiques provenant d'une même organisation. Ils permettent d'ajouter du contexte en formulant le but global de l'ensemble de ses expériences.
— Les **Runs** correspondent aux données dérivées du séquençage liés à l'expérience. Ce sont eux qui seront utilisés pour downloader les reads de l'expérience. Une expérience peut engendrer plusieurs runs dans le cas de réplicats techniques et non biologiques.

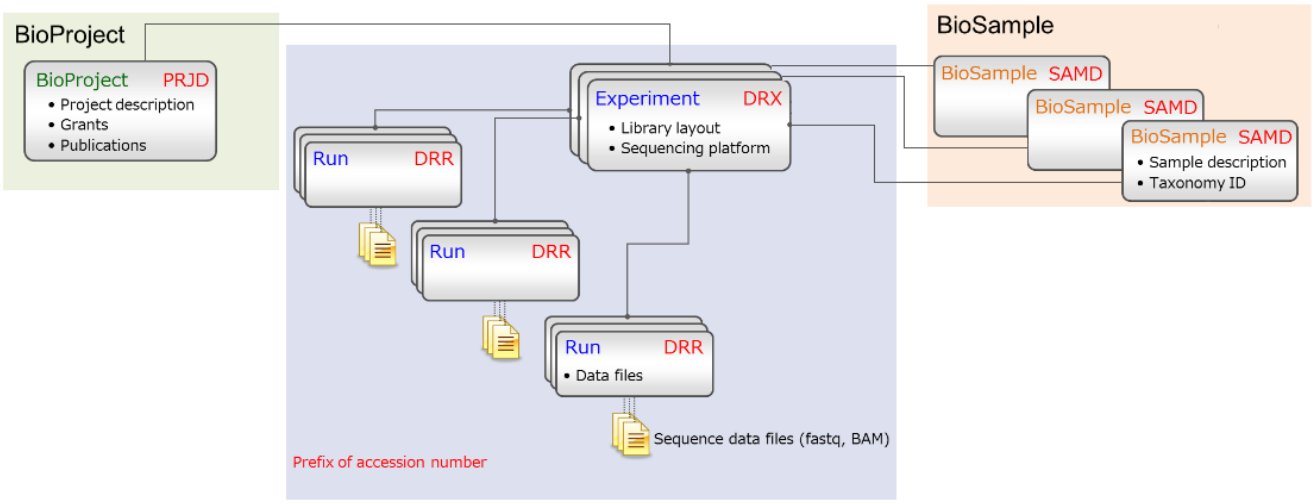

FIGURE 5 : **Schéma repris du site sur le SRA représentant les liens entre les différentes tables.** Une expérience possède un ou plusieurs runs. L'ensemble des expériences peut être placé sous un numéro de bioproject. Les accessions sont les clés primaires.

4.4 Bases de données et STAT

D'autres bases de données supportées par la NCBI font référence aux résultats obtenus par leur outil STAT (K. S. Katz et al., 2021), un programme de taxonomie. *Metadata* reprend les métadonnées liées aux séquençage des différents runs tandis que les deux autres font directement référence à cet outil.

— **nih-sra-datastore.sra.metadata**
— **nih-sra-datastore.sra__tax__analysis__tool.tax__analysis**
— **nih-sra-datastore.sra__tax__analysis__tool.tax__analysis__info**

Ces trois tables assurent la cohérence de leur résutat par la jointure des tables au niveau de la clé primaire *acc*, le numéro d'accession du run.

— *Metadata* fait référence aux données d'expérience de chaque run, aussi bien actuelles (99%) que inutilisables[5].
— *Tax_analysis* reprend les informations taxonomiques de chaque run révélées par l'outil

5. SELECT DISTINCT acc FROM `nih-sra-datastore.sra.metadata` WHERE bioproject IS NULL; affiche 394 672 accessions qui ne renvoient à aucun résultat sur le site du NCBI.

STAT, comme l'identifiant taxonomique (tax_id) et le nombre de reads associé à cet id (self_count).

— *Tax_analysis_info* fournit des informations sur le traitement effectué par STAT, notamment la version du programme et sa date d'utilisation.

STAT (SRA Taxonomy Analysis Tool) est un outil développé par le NCBI dans le cadre de l'analyse automatique des données de séquençage (K. S. Katz et al., 2021), déployé sur l'ensemble des SRRs (Katz, 2021). Cet outil génère une taxonomie de l'échantillon peu de temps avant ou après sa mise à disposition publique. Il repose sur l'utilisation de MinHash (Broder, 2000), une technique permettant d'estimer rapidement la similarité entre ensembles en les représentant sous forme d'une signature, plus efficace à comparer, apellé ici k-mer diagnostique.

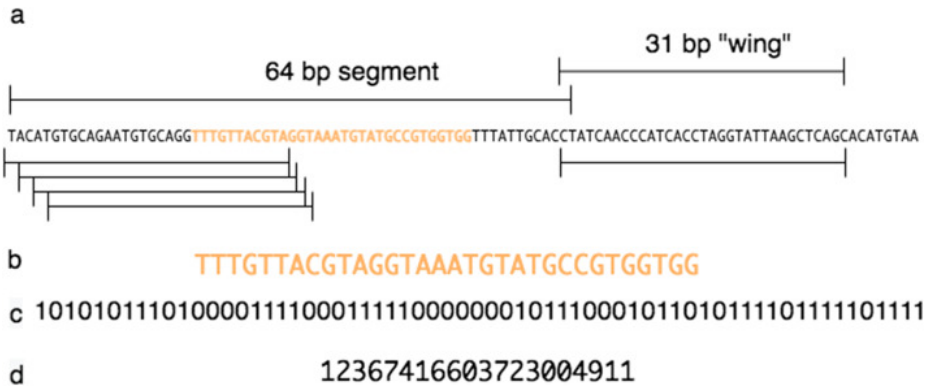

FIGURE 6 : **Détermination du k-mer diagnostique pour une séquence de 64 pb. a.** Chaque séquence est divisée en k-mers de 32 pb avec un décalage de 1 à chaque fois. Une séquence de 64 pb génère ainsi 64 k-mers de 32 pb chacun. **b.** Chacun de ces k-mers est encodé en binaire par paquet de deux (*2-bit encoding*). **c.** Chaque k-mer est ensuite converti en valeur décimale sur 64 bits. **d.** Enfin, le k-mer avec la plus petite valeur est choisi pour représenter le segment.

Ces k-mers couvrent au maximum la taxonomie (qui, en mars 2020, comptait 2 383 364 nœuds) et permettent d'assigner chaque read sur une partie de l'arbre taxonomique (K. Katz et al., 2022).

Sur le site du NCBI, chaque run dispose d'un onglet *Analysis* où les résultats taxonomiques sont présentés sous forme de représentations en *piecharts* et liste hiérarchique. Ces données sont également enregistrées dans une base de données dédiée, nih-sra-datastore.sra_tax_analysis_tool.tax_analysis, accessible via des requêtes SQL pour une analyse approfondie.

a)

acc	tax_id	name	self_count	total_count
DRR014159	909930	Acidaminococcaceae	10	755
DRR014159	1843488	Acidaminococcales	0	755
DRR014159	904	Acidaminococcus	123	719
DRR014159	905	Acidaminococcus fermentans	0	1
DRR014159	591001	Acidaminococcus fermentans DSM 20731	1	1
DRR014159	187327	Acidaminococcus intestini	21	583
DRR014159	568816	Acidaminococcus intestini RyC-MR95	562	562
DRR014159	563191	Acidaminococcus sp. D21	12	12
DRR014159	33024	Phascolarctobacterium	0	26
DRR014159	626940	Phascolarctobacterium succinatutens	0	26
DRR014159	626939	Phascolarctobacterium succinatutens YIT 12067	26	26

b)

- Acidaminococcales: **0.01%**
  - Acidaminococcaceae: **0.01%**
    - Acidaminococcus: **0.01%**
      - Acidaminococcus intestini: **0.01%**
        - Acidaminococcus intestini RyC-MR95: **0.01%**
      - Acidaminococcus fermentans: **<0.01%**
        - Acidaminococcus fermentans DSM 20731: **<0.01%**
      - Acidaminococcus sp. D21: **<0.01%** (1 Kbp)
    - Phascolarctobacterium: **<0.01%**
      - Phascolarctobacterium succinatutens YIT 12067: **<0.01%** (3 Kbp)

FIGURE 7 : **Contenu taxonomique de l'échantillon DRR014159. a.** La requête SQL sur BigQuery montre 12 lignes correspondant aux nœuds taxonomiques descendant de la famille des *Acidaminococcales*. Cette famille a un *total_count* de 755, représentant le nombre de reads assignés à la famille et à ses nœuds enfants (genres, espèces, etc.), dont la distibution en visible à travers les *self_counts* (10+123+...+26=755). **b.** Le même échantillon est visualisé sous forme de liste hiérarchique sur la page du NCBI, dans l'onglet *Analysis*.

STAT a été appliqué à une grande partie des runs, mais ce processus s'est déroulé progressivement. Une portion significative des échantillons a été traitée entre 2017 et 2019, au cours de son développement, et a été analysée avec la taxonomie et k-mers diagnostiques de cette période. La phylogénie des cyanobactéries étant assez dynamique, il est possible que certains échantillons ne reflètent pas fidèlement la diversité que STAT leur a attribuée.

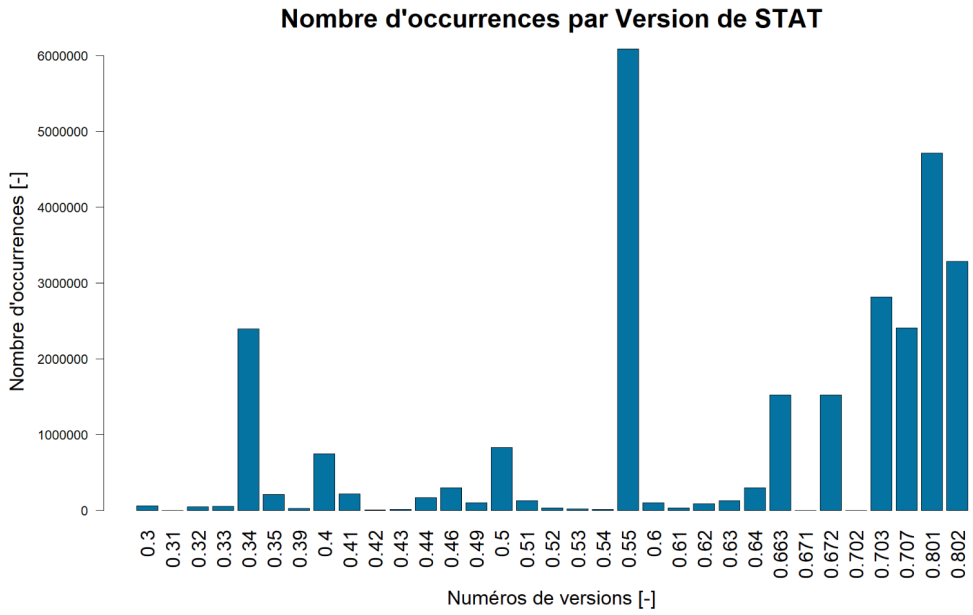

FIGURE 8 : **Barplot représentant le nombre d'échantillons analysés par différentes versions de STAT.** Une partie importante des analyses a été réalisée avec la version 0.55 de STAT, surtout utilisé en 2019.

Il semble donc pertinent d'explorer l'ensemble des archives SRA pour identifier les échantillons particulièrement riches en cyanobactéries basales déjà connues. Si une espèce est réellement nouvelle, STAT, au vu son fonctionnement, va assigner l'espèce ou le genre le plus proche.

4.5 Word embeddings et zero-shot classification

La base de données *metadata* contient un certain nombre de métadonnées standardisées, telles que nom d'organisme, date, pays, mais également une partie assez conséquente, non standardisée, appelée *attributes*, qui est une structure imbriquée regroupant des paires de clé-valeur dont le contenu est variable selon les chercheurs. Une autre base de données, accessible grâce à EDirect (Kans, 2013), permet d'extraire des informations plus contextualisées sous forme de phrases complètes, décrivant le but et le déroulement des expériences. Bien que l'ensemble de ces sources offrent une quantité de données suffisantes, ces dernières restent relativement désorganisées, ce qui limite leur exploitabilité automatique dans le cadre d'outils de *screening*, notamment pour rechercher des échantillons présentant des caractéristiques similaires à d'autres.

Le manque de rigueur dans leur structuration (chaque utilisateur pouvant personnaliser ses uploads) a déjà été signalé par de nombreuses personnes, tant en ce qui concerne la qualité des données (Gonçalves & Musen, 2019 ; Warren, 2008) que leur manque d'homogénéité globale (Hull, 1997). Bien que des efforts de centralisation aient été entrepris, ils restent insuffisants (Barrett et al., 2012 ; O'Leary et al., 2024 ; Sheffield et al., 2023).

Plusieurs solutions ont été proposées pour tenter d'exploiter ces données désorganisées, en les classifiant a *posteriori*. La méthode la plus directe consiste à classer manuellement les informations, soit en se basant sur une connaissance globale du sujet, soit en utilisant des ontologies, des sortes de dictionnaires relationnels, alliant définitions et liens entre termes [6]. Ces approches demeurent très chronophages et demandent un certain degré d'expertise lorsque le sujet est complexe.

Une autre voie passe par l'automatisation.

Dans la reconnaissance de motifs (gestalt pattern matching) (Ratclif, 1988), l'algorithme identifie les séquences les plus longues dans un texte et les compare à des termes prédéfinis (Blurock, 2021 ; Sanino, 2020). Cette méthode est cependant limitée car elle se concentre principalement sur la similarité orthographique plutôt que sémantique et est à présent utilisée dans la correction orthographique (Sudo et al., 2023).

D'autres méthodes utilisent des outils de traitement du langage naturel (NLP), comme le NER (Named Entity Recognition) (Rau, 1991), qui identifie les informations du texte qu'il a appris à reconnaître durant son entraînement. Cette méthode d'apprentissage en fait un outil hyper spécialisé et il est souvent nécessaire de former son propre modèle pour l'adapter au contexte voulu. Les modèles N-gram sont également utilisés et permettent également de saisir le contexte des phrases en regardant par tranche de N-mot (sorte de k-mer) la phrase. 'Je mange une glace', en bi-gram, se lira 'Je mange' 'mange une', 'une glace'.

6. id: ENVO:00000402, nom: volcan de boue, déf: "Volcan formé par des liquides et gaz géologiques, à des températures plus basses que les volcans igneux.", réf: https://en.wikipedia.org/wiki/Mud_volcano, SOUS-ENSEMBLE DE: ENVO:00000247 ! volcan

D'autres méthodes utilisent des outils de traitement du langage naturel (NLP), comme le NER (Named Entity Recognition) (Rau, 1991), qui identifie les informations dans un texte qu'il a appris à reconnaître durant son entraînement. Cette méthode d'apprentissage en fait un outil hyper spécialisé, et il est souvent nécessaire de former son propre modèle pour l'adapter au contexte voulu. Les modèles N-gram sont également utilisés et permettent de saisir le contexte des phrases en analysant des séquences de N mots (similaires aux k-mers en génomique). Par exemple, pour la phrase "Je mange une glace", un modèle bi-gram générerait les séquences "Je mange", "mange une", et "une glace" durant son entraînement (Brown et al., 1992).

Ces dernières années, les modèles d'intelligence artificielle générative, tels qu'utilisés pour créer des images (DALL-E(OpenAI, 2021)) et mener des conversations (ChatGPT (OpenAI, 2022)), ont suscité un grand intérêt auprès du public [7].

Ces Large Language Model (LLM) ont démontré une capacité impressionnante à traiter des sujets variés et à saisir le contexte des demandes, fournissant, la plupart du temps, des réponses pertinentes (Bubeck et al., 2023). Cette aptitude provient d'un apprentissage sur de vastes ensembles de données, au cours duquel ils ont ajusté des millions - voire des milliards - de paramètres, dans le but de former des modèles de langage statistiques (SLM). Les LLMs utilisent une architecture d'encodeur-décodeur, où l'encodeur convertit l'entrée en une représentation vectorielle dense, et le décodeur génère une réponse pertinente à partir de cette représentation encodée (Bahdanau et al., 2016 ; Cho et al., 2014 ; Sutskever et al., 2014).

Lors de leur entraînement, les Large Language Models (LLM) apprennent à représenter chaque token (unité de texte, souvent une partie d'un mot) sous forme de vecteur multidimensionnel dense appelé **word embedding**. En plus de stocker le mot, il encode également des informations supplémentaires sur ses caractéristiques sémantiques et contextuelles. Par exemple, les termes de même thème auront cette notion de similarité reflétée dans le vecteur, permettant ainsi leur visualisation en 2 ou 3D suite à une PCA (Analyse en Composantes Principales), où ils apparaîtront groupés. Cette réduction de dimensionnalité permet de ne garder que les composantes maximisant la variance. Par exemple, si on compare des termes de fruits à des animaux, une des dimensions pourrait exprimer le "mouvement" car elle permettrait de distinguer efficacement les termes, et serait donc retenue comme composante principale.

Les relations entre les termes sont également encodées dans ces vecteurs. Par exemple, il est possible d'effectuer des calculs arithmétiques vectoriels pour déduire un terme à partir d'autres (analogie de mot).

7. ChatGPT a atteint 180 millions d'utilisateurs au mois d'août 2024, avec un million d'inscriptions dans les cinq jours suivant son lancement. Le nombre de visites durant le mois d'avril 2024 a culminé à 1,2 milliard (Sources : OpenAI, SimilarWeb).

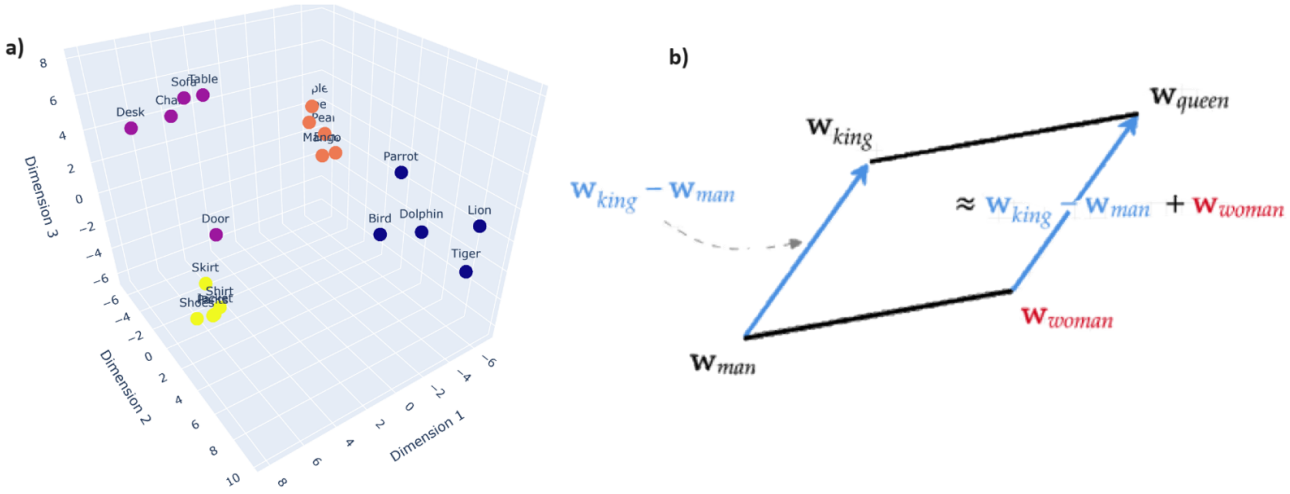

FIGURE 9 : **Représentation de word embeddings. a.** Visualisation des termes dont le clustering est bien visible entre les différents thèmes après une réduction de dimensionnalité n=3. **b.** Exemple classique de la littérature montrant la relation logique entre différents termes dans leur représentation spatiale. Le vecteur roi - homme + femme se rapproche du vecteur reine.

Ces vecteurs d'embedding sont générés par l'entraînement de réseaux de neurones. Le modèle apprend à prédire le prochain token en se basant sur le précédent dans un corpus de données auquel il a eu accès. Les poids des noeuds à l'intérieur du réseau, souvent initialement aléatoires (Editor, 2024), s'ajustent jusqu'à ce que le modèle prédise correctement les séquences de tokens dans l'ensemble d'entraînement. Dans l'exemple, "I love walking my dog", l'entraînement s'arrêtera dès lors que 'I' prédira 'love', 'love' prédira 'walk', etc (Naveed et al., 2024 ; StatQuest with Josh Starmer, 2023 ; Talamadupula, 2024) [8].

Cette compréhension approfondie du langage rend les LLM particulièrement adaptés à une technique de classification appelée **zero-shot classification** (Wang et al., 2023). Cette méthode permet aux modèles de faire des prédictions sur des tâches pour lesquelles ils n'ont pas été explicitement entraînés (Chaudhary, 2020), en utilisant leur connaissance contextuelle du langage acquise au préalable et à l'inférence de langage naturel (Tae, 2021).

L'inférence de langage naturel (NLI) consiste à comprendre le lien logique entre une prémisse (la phrase d'intérêt à classifier) et une hypothèse (une affirmation concernant la prémisse), en se basant sur des labels fournis dans le dataset : correspondance (*entailment*), contradiction ou neutre. Une fois réentraînés de cette manière, les LLM sont capables de classifier des textes, même sans avoir rencontré les catégories données par l'utilisateur.

8. Ce processus implique l'utilisation de fonctions d'activation, de la rétropropagation (*backpropagation*), de la fonction softmax qui transforme les logits en probabilité, et du mécanisme d'attention. Toutefois, ces concepts ne seront pas expliqués ici, car cela nécessiterait une discussion beaucoup plus approfondie, n'ayant pas sa place dans une introduction.

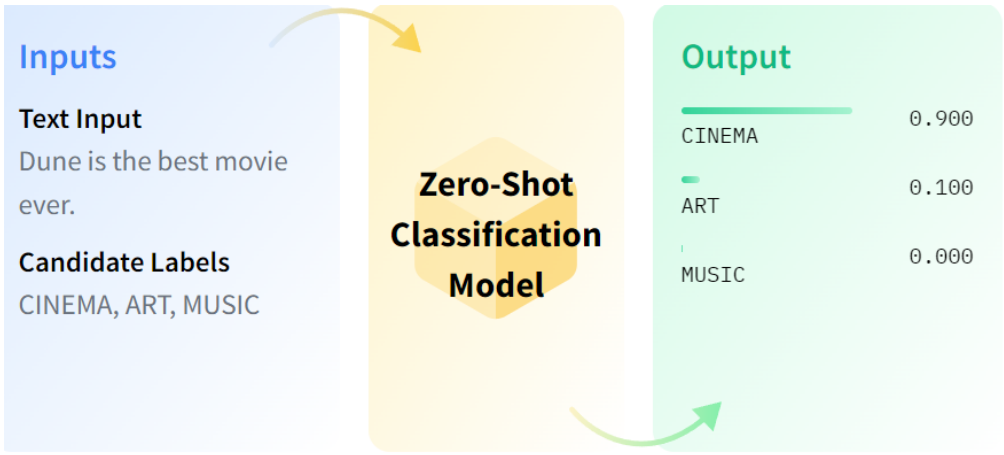

FIGURE 10 : **Schéma expliquant les inputs et outputs d'une ZSC.**

5 Objectifs

— Récupérer les échantillons publics provenant d'expériences métagénomiques contenant des cyanobactéries basales.
— Créer un dictionnaire d'étiquettes environnementales utilisées pour la classification automatique de textes et annoter les échantillons d'intérêts.
— Réaliser une phylogénie axée sur la partie basale de l'arbre évolutif des cyanobactéries afin de détecter des nouvelles espèces s'insérant entre les Mélainabactéries et les Gloeobacterales.

6 Matériel et méthodes : Données brutes

6.1 BigQuery

Les identifiants d'expériences méta-génomiques ainsi que leurs métadonnées associées ont été récupérés via des requêtes SQL sur BigQuery, la plateforme de data warehouse de la Google Cloud Platform (GCP) (Husen, 2021 ; Thallam, 2020). Ce dernier héberge les trois bases de données de la NCBI déjà abordées dans l'introduction.

Le taxon des Cyanobacteriota est référencé par le numéro **1117**. Un tableau récapitulant les différents numéros et noms scientifiques des autres taxons est disponible en **Annexe BBBB**.

6.1.1 Sélection des identifiants de cyano basales

Les cyanobactéries d'intérêt se situent à la base de la phylogénie des Cyanobacteriota. Comme mentionné précédemment dans l'introduction, plusieurs phylogénies sont actuellement proposées, et elles divergent sur certains aspects, en particulier au niveau des branches plus éloignées. Il est

nécessaire de choisir une phylogénie de référence, car la sélection négative exige d'identifier les taxons non intéressants, c'est-à-dire ceux qui ne sont pas à la base de la phylogénie.

Les taxons correspondant aux cyanobactéries, qu'ils soient d'intérêt ou non, sont déterminés en suivant la phylogénie définie par L. Cornet (Cornet et al., 2021). Les cyanobactéries basales sont définies par les clades 1 à 4 ainsi que celui intitulé "*missing*" dans le papier. Les clades jugés moins pertinents se retrouvent dans la partie restante de l'arbre.

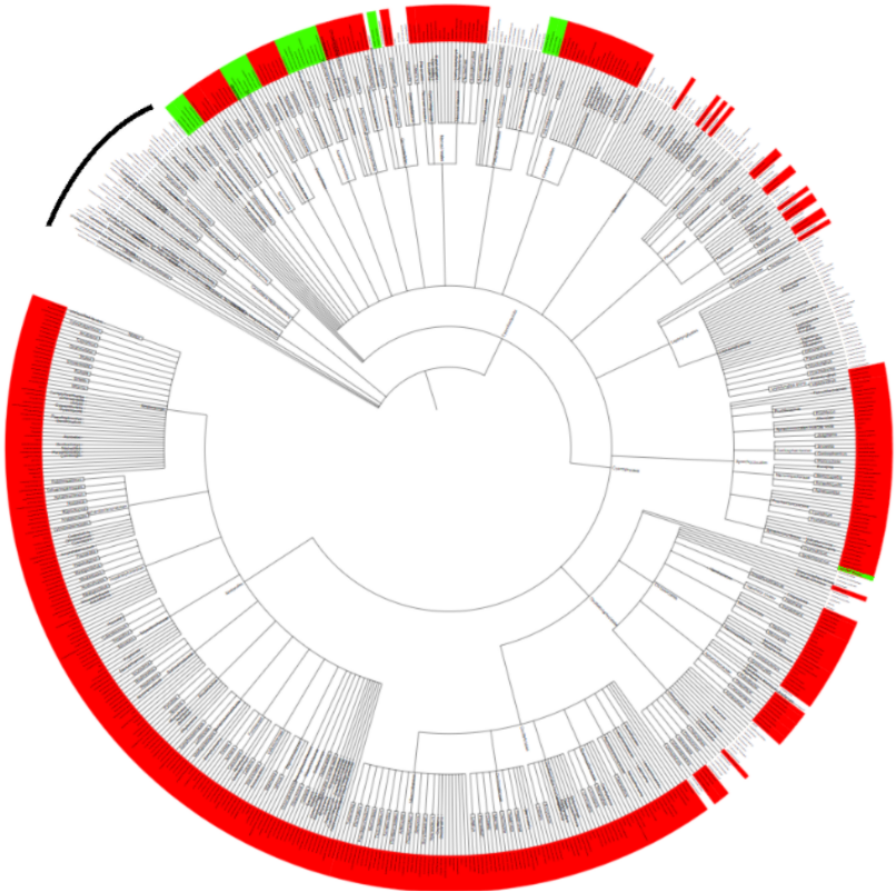

FIGURE 11 : **Représentation de la sélection vis-à-vis de la taxonomie de la NCBI sur iTOL. En rouge**, représentation des taxons de la sélection négative. **En vert**, tax_id de la positive. **L'arc de cercle noir** représente le phylum des mélainabacteria sans les cyanos.

La taxonomie a été téléchargée grâce à la bibliothèque **ete3**, le script est disponible en **Annexe A**.

6.2 parse.pl

Le script parse.pl en **Annexe B** permet de traiter correctement le format imbriqué généré par le champ *attributes* de la requête SQL. Il génère un fichier CSV utilisable, avec une seule ligne par accession et des valeurs nulles pour les clés n'ayant pas de valeurs.

6.3 Assemblages publics

L'ensemble des assemblages publics est disponible dans l'archive taxdump.tar.gz sur le site de la NCBI, et a été consulté le 12 avril 2024. Les fichiers "assembly_summary_genbank.txt" et "assembly_summary_genbank_historical.txt" ont été téléchargés, constituant un ensemble de 2 153 253 données d'assemblages GenBank.

6.4 EDirect

Les métadonnées de type 'titre' 'but du projet', 'description de l'expérience' seront récupérées grâce au programme Entrez Direct (Kans, 2024), version 20.9. Le script fetch_titles.pl est disponible en **Annexe C**

6.5 Concaténation des champs

La fusion des colonnes ayant des attributs de même nature se fera grâce au script merge_columns.pl disponible en **Annexe D**. Le tableau récapitulatif des colonnes fusionnées et de leurs noms finaux est disponible ci-dessous :

Colonne finale	Colonnes fusionnées
environment	broad_scale_environmental_context_sam, biome_sam, environment___biome___sam, env_broad_scale_sam, hix___proportion___sam, isolation_source_sam
temperature_C	temperature_sam, water_temp_____c___sam, temp_sam, wet_bulb_temp_sam
altitude	altitude_sam_s_dpl11, ele_sn___m___sam, elev_sam, elevation_sam_s_dpl25, geographic_location___elevation___sam, ele_sp___m___sam
depth	geographic_location___depth___sam, depth_sam, depth___mbsf___sam
latitude	sampling_event___latitude___start_sam, geographic___location___latitude_sam, lat_sn___dd___sam, latitude_start_sam, latitude_sam
longitude	geographic_location___longitude___sam, longitude_start_sam, sampling_event___longitude___start_sam, longitude_sam
country	geo_loc_name_country_calc, geographic_location___country_and_or_sea___sam
env_medium	environment___material___sam, env_material_sam, material_sam, env_medium_sam, env_package_sam, environmental_medium_sam
host	host_sam, common_name_sam, organism_sam, host_body_habitat_sam
local_environment	env_local_scale_sam, local_environ, env_biome_sam, environment___feature___sam, env_feature_sam

Colonne finale	Colonnes fusionnées
others	isolate_sam_ss_dpl100, collection_method_sam, body_site_sam, chemostat_treatment_sam, type_material_sam, tissue_sam_ss_dpl145, surface_material___sam, hydro_isotope___per_mil___sam, lab_conductivity___*s*cm___sam, samp_collect_device_sam, project_name_sam
others2	wastewater_sludge_environmental_package_sam, isolation_source_sam_ss_dpl261, secondary_treatment_sam, sample_description_sam, metagenome_source_sam_s_dpl289, microbial_mat_biofilm_environmental_package_sam, lap___nmol_g_1_h_1___sam, location___sam, title_sam, ground_level___sam, histological_type_sam

Fusionner les fichiers merged_columns.txt avec les 7 premières colonnes de metadata.csv et ajouter le texte brut de raw_text. Remplacer les virgules par des points-virgules pour éviter les problèmes d'importation dans Excel.

6.6 tags_auto.pl

Disponible en **Annexe E**. Ce script transforme les valeurs simples d'altitude, de profondeur et de température en plages de valeurs (par exemple, 0-1 mètre de profondeur, 20-25°C, etc.) et ne conserve que l'année des dates. Il convertit également les pieds (ft) en mètres et les centimètres (cm) en mètres.

7 Matériel et méthodes : Zero-Shot Classification

7.1 Choix du dictionnaire initial : ENVO

La base de vocabulaire utilisée pour l'étiquetage est extraite du fichier ontologique envo.obo (Mungall, 2015/2024), lequel représente l'ontologie ENVO (ENVironment Ontology) (Buttigieg et al., 2013). Ce fichier contient des classes/termes identifiés par une étiquette "name", qui sont eux-même accompagnés qu'une dizaine de métadonnées représentées par d'autres balises, telles que *"def"*, *"comment"*, *"relationship"*, etc.

7.2 Clusterisation

7.2.1 Transformation en *word emeddings*

Le *clustering* des termes a été réalisé en évaluant la similarité entre les objets afin de former des groupes de données similaires. Pour mesurer cette similarité, les données textuelles ont été

représentées sous forme numérique grâce au modèle de *Sentence Transformer* **all-MiniLM-L6-v2** (Reimers & Gurevych, 2021), qui transforme les mots en vecteurs multidimensionnels appelés *word embeddings*. Chaque terme a ainsi été vectorisé en un vecteur numérique dense dans un espace de 384 dimensions.

7.2.2 Réduction de la dimensionalité

Pour assurer une représentation significative des données, il est recommandé de conserver au minimum 70 à 80 % de la variance expliquée par les composantes principales (Bruin, 2006 ; Lindgren, 2020).

7.2.3 Graphes et détermination de k

Les calculs des différentes métriques ont été réalisés après vectorisation des termes et en appliquant l'algorithme *K-means*. Le script utilise les librairires sklearn, gap_statistic, matplotlib ainsi que le *Sentence Transformer* all-MiniLM-L6-v2. Il est disponible en **Annexe F**. Pour le calcul de la statistique de gap avec la méthode 1-SE, le script utilise le package R cluster, assisté par factoextra, et est disponible **Annexe G**.

L'Elbow Plot (Thorndike, 1953) génère un graphique représentant la somme des carrés des distances des points au centre du groupement le plus proche, également appelée **inertie**. Le "coude" du graphique indique le nombre optimal de *clusters*, car il représente le point où l'ajout de *clusters* supplémentaires ne réduit plus significativement l'inertie.

Le Score de Silhouette (Rousseeuw, 1987) mesure à la fois la compacité intracluster (distance moyenne au sein du cluster) et la séparation entre *clusters* (distance moyenne entre le *cluster* et le *cluster* le plus proche). Le score varie de -1 à 1 : Un score de 1 indique une bonne séparation et une bonne compacité du groupe, 0 signifie que les points se trouvent à la frontière entre deux , et un score négatif indique une mauvaise assignation des points.

La Statistique de Gap (Tibshirani, 2000) évalue la qualité des clusters en comparant l'inertie des données réelles à celle des données générées aléatoirement. Le "gap" est la différence entre ces deux inerties, et un écart important suggère un bon regroupement des données. Le nombre optimal de clusters est généralement celui où le *gap* est maximal.

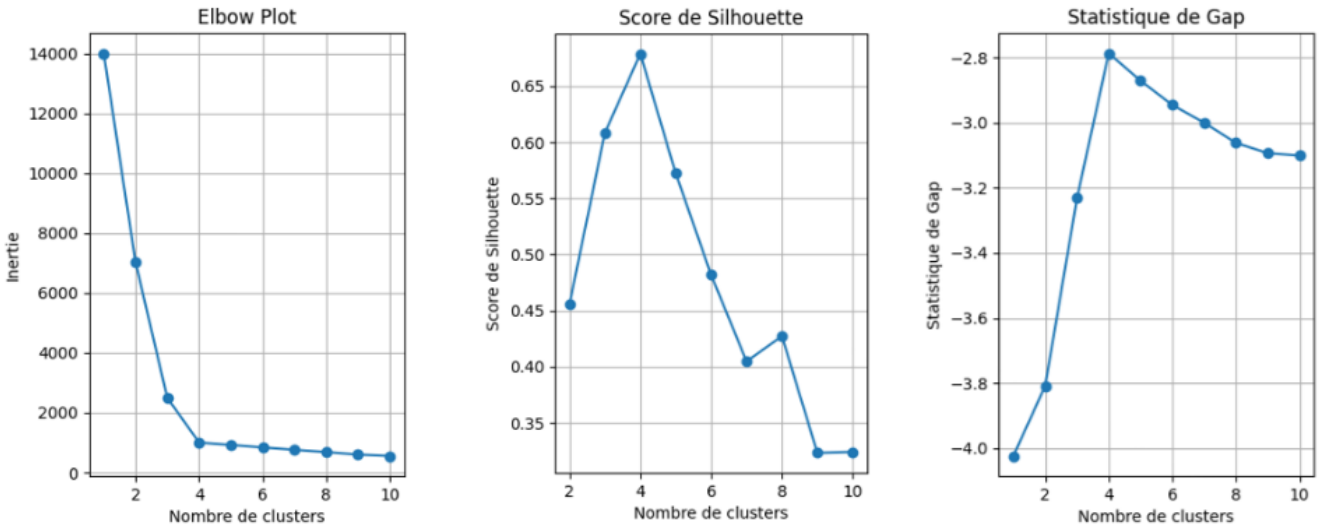

FIGURE 12 : **Représentation des trois métriques calculées sur des données naturellement clusterisables en quatre.** Les trois graphiques montrent les résultats des méthodes d'évaluation des *clusters* : l'*Elbow Plot*, le *Score de Silhouette*, et la *Statistique de Gap*. Chacun de ces graphiques indique que le nombre optimal de *clusters* est $k = 4$.

7.2.3.1 Estimation manuelle de la qualité de la clusterisation. Disponible en **Annexe H**, ce script utilise la même stratégie d'embeddings que pour les plots précédents. Ces clusterisations sont faites en utilisant l'algorithme *K-Means* pour obtenir [500, 1000, 1500, 2000, 2500, 3000, et 3500] regroupements finaux. Pour chacune de ces configurations, les 100 plus grands *clusters* sont sélectionnés et la similarité globale de chaque terme y est calculée. Cette dernière est mesurée en sommant les similarités de ce terme avec tous les autres termes du même *cluster*. Par exemple, la similarité globale du mot `t1` est définie comme suit :

sim_globale(t1) = sim(t1, t2) + sim(t1, t3) + …+ sim(t1, tn)

où `t2`, `t3`, …, `tn` sont les autres termes du même *cluster*, et où la similarité entre deux termes est calculée en calculant le produit scalaire des deux vecteurs.

Les scores obtenus sont ensuite utilisés pour créer une représentation en *WordCloud*, script disponible en **annexe I**, permettant ainsi une évaluation visuelle rapide de la qualité de la clusterisation.

7.2.4 Clustering hiérarchique et *K-Means*

Le clustering hiérarchique utilise des mesures de distance entre données, tels que les liaisons simple, complète, moyenne, centroïde ou de Ward et sont souvent plus ou moins adaptées à des types de structures de données (Grisel & Varoquaux, 2010).

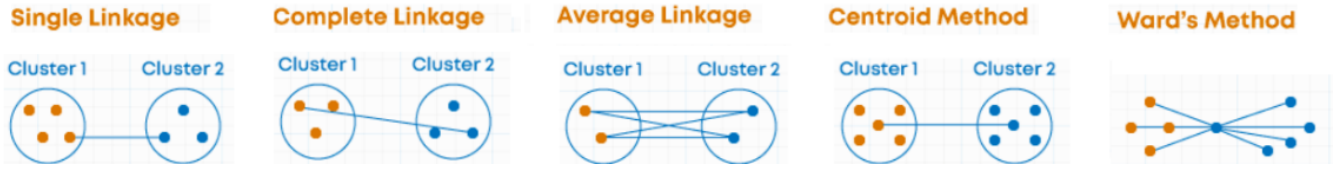

FIGURE 13 : **Types de liaison en clustering hiérarchique. Liaison simple :** distance minimale entre les points de deux clusters, **liaison complète :** distance maximale entre les points de deux clusters, **liaison moyenne :** distance moyenne entre tous les points de deux clusters, **liaison par centroïdes :** distance entre les centroïdes des clusters, et **liaison de Ward :** augmentation de la variance totale au sein des clusters.

7.2.4.1 Similarité et clusterisation

La similarité entre les *word embeddings* a été calculée en utilisant la distance cosinus. Cette mesure, calculée comme le produit scalaire normalisé entre deux vecteurs (AlexPodles, 2024) permet d'évaluer la similarité entre les termes en fonction de l'orientation de leurs vecteurs. Le cosinus a une valeur maximale quand les termes ont leur vecteur orienté dans la même direction (cos(0°)=1), et une valeur minimale quand ils sont opposés (cos(180°)=-1). La similarité entre les vecteurs tend vers 0 lorsque ces derniers sont perpendiculaires, indiquant une faible similitude entre les termes.

Le calcul de la similarité cosinus permet de mesurer dans quelle mesure deux vecteurs sont orientés dans la même direction, et partagent ainsi un degré de ressemblance sémantique.

$$\text{sim}(a, b) = \frac{a \cdot b}{\|a\| \cdot \|b\|}$$

7.2.5 Choix du meilleur représentant de chaque *cluster*

La logique de similarité globale décrite lors de l'estimation manuelle est reprise afin de choisir le meilleur représentant de chaque *cluster*. Les détails sont disponibles en **Annexe J**.

7.3 Choix du modèle de *Zero-shot Classification*

Le modèle de *zero-shot classification* **bart-large-mnli** a été téléchargé le 29 Mars 2024. Il est exécuté sur les deux GPUs, avec la possibilité d'activer le multi-labeling via le paramètre `multi_label=T` ou `F`. Ce modèle de classification provient du checkpoint de bart-large, un LLM créé par facebook, qui a été entraîné sur le dataset Multi-Genre Natural Language Inference (MultiNLI).

7.3.1 Identification des étiquettes non environmentales

L'utilisation de la ZSC permet d'identifier et de supprimer les étiquettes moins pertinentes, et est visible en **Annexe K**. Les étiquettes candidates sont ["nature", "human", "astronomy", "ice"].

Au lieu de tenter d'isoler directement les trois thèmes par rapport à l'étiquette *"nature"*, représentative des termes environmentaux à garder, le script est conçu pour être exécuté plusieurs fois, en

comparant chaque fois une seule étiquette non pertinente face à l'étiquette "nature" [9].

Pour minimiser les faux positifs parmi les termes associés aux étiquettes indésirables, seuls ceux avec un score supérieur à 80 % sont récupérés pour que leur pertinence soit vérifiée manuellement. Certains de ces termes, malgré leur association avec des étiquettes non prioritaires, sont conservés pour garantir une couverture de ce contexte environnemental.

7.3.2 Détermination du seuil de score

Une fois le dictionnaire conçu, le modèle de classification zéro-shot peut être utilisé pour traiter les textes bruts. Pour chaque échantillon, le modèle attribue un score à chaque étiquette, reflétant la probabilité que le texte appartienne à cette catégorie. Comme l'objectif est de récupérer potentiellement plusieurs tags pertinents pour chaque texte, le multi-labeling est activé, ce qui fait que chaque terme peut être associé à plusieurs étiquettes avec des scores distincts. Il est essentiel de déterminer le seuil de score optimal pour équilibrer l'identification de tags pertinents et éviter l'attribution excessive de tags incorrects, en minimisant les faux positifs associés à une probabilité d'appartenance trop faible.

La meilleure manière d'évaluer rapidement la qualité d'un modèle de classification zéro-shot (ZSC) est de recourir à une évaluation effectuée par un observateur humain (Ouyang et al., 2022 ; Z. Zeng et al., 2024). Lorsqu'une étiquette peut être interprétée de plusieurs manières, ou lorsqu'elle n'est pas fondamentalement incorrecte mais difficile à déduire à partir du texte, elle est considérée comme ambiguë. Le code permettant de réaliser la *Zero-Shot Classification* avec différents scores est disponible en **Annexe L**. Il sera lancé avec la fourchette de valeur de score : 0.5, 0.6, 0.7, 0.8, 0.85, 0.9, 0.95, 0.99 sur 20 textes.

Ces textes bruts proviennent des échantillons *SRR13425444, SRR19632696, SRR19742217, SRR20627844, SRR20627848, SRR22012408, SRR22077854, SRR22184787, SRR22198305, SRR10018916, SRR14689465, DRR290133, DRR331387, DRR333369, DRR504407 et ERR3440668.*

Exemple d'étiquetage avec en **gras**, les tags mal assignés.

— neutral condition, non-saline environment, fresh water body, hot spring, sediment, geothermally heated environment, natural environment, microbial mat, chemically enriched sediment, **biogenous sediment**, **haline environment**, underground water, **pier**, interface layer, **concrete**, **pasturable land**, **boreal**, temperate environment
— fresh water body, cryoconite, glacier, alpine, non-saline environment, natural environment, liquid water, ice, aquatic environment, drinking water, depositional process, extreme tem-

9. Le nombre de termes associés aux trois thèmes non pertinents chute drastiquement si le script est exécuté une seule fois avec les quatre étiquettes en même temps : de 90, 42, et 59 résultats, on passe respectivement à 17, 18, et 48

perature environment, **concrete**, basic environment

Le calcul de la précision prend en compte le nombre de tags incorrectement assignés par rapport au nombre total : 100-(7/32)*100=79%

7.3.3 ZSC sur totalité des textes

Le script de détermination du seuil sera modifié pour classifier 1440 textes au lieu des 20 de test. Le modèle ayant des difficultés à interpréter les notions de hauteurs, noms des couches d'eau et températures non extrêmes, celles-ci seront retirées du *tagset*. Le script sera exécuté avec un seuil de 0,85.

8 Matériel et méthodes : Phylogénie

La construction d'arbres phylogéniques est une discipline qui permet de déterminer la position évolutive des organismes et de comprendre leurs relations de parenté. Avant de pouvoir élaborer ces arbres, plusieurs étapes préliminaires sont nécessaires pour collecter et préparer les données.

Il est nécessaire de télécharger les séquences brutes (*reads*), de les assembler en métagénomes et d'identifier les différents organismes présents pour pouvoir les compartimenter. Après avoir récupéré celles apparentées aux cyanobactéries, il faut les filtrer pour n'en garder que les meilleures et rechercher les séquences orthologues. Ce sont ces séquences qui seront ensuite utilisées pour assembler les arbres phylogéniques. Pour mieux comprendre les relations entre les bins, ces métagénomes sont également comparées à des génomes de référence.

8.1 GENERA

GENERA (Cornet et al., 2023) est une suite de workflows Nextflow, soutenus par des conteneurs Singularity, conçus pour la réalisation d'analyses de génomique comparative. Cette boîte à outils sera largement exploitée dans le cadre de l'obtention d'une phylogénie.

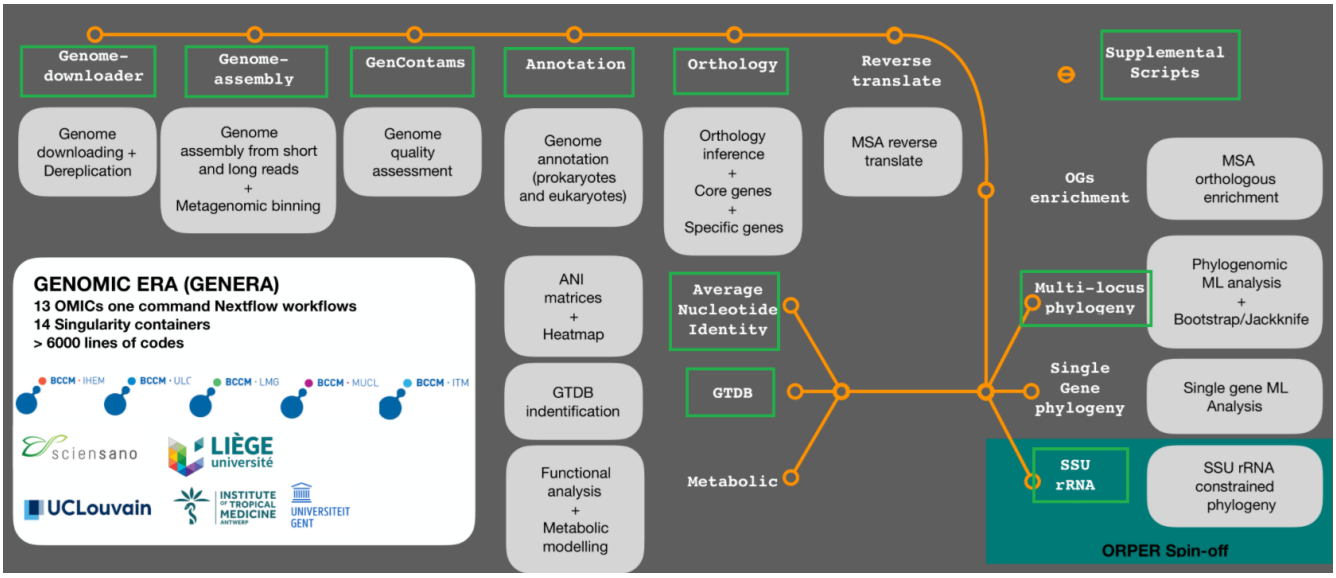

FIGURE 14 : **Architecture de GENERA.** 13 workflows constituent GENERA. En vert, les 9 workflows + scripts supplémentaires utilisés.

Seuls les workflows nécessitant des informations supplémentaires sont présentés plus loin. Les autres sont exécutés avec les paramètres décrits dans le tableau ci-dessous, ou avec les paramètres par défaut si non spécifiés. Ces valeurs sont décrites sur le wiki de GENERA.

Nom du workflow	Version	Programmes principaux et logique d'utilisation	INPUT	OUTPUT	OPTIONS/PARAMETRES
Genome-assembly	2.0.0	fastQC (Andrews, 2018) va télécharger les runs, SPAdes (Bankevich et al., 2012) va les aligner et former des contigs, MetaBAT (Kang et al., 2015) et CONCOCT (Alneberg et al., 2014) vont séparer les organismes, RagTag (Alonge et al., 2022) va créer les scaffolds par organisme.	n° d'accession du run	métagénome, bins, rapports	/
GTDB	1.1.0	GTDBtk (Chaumeil et al., 2020) va nommer les organismes présents dans les bins.	bins	lignée de la bin + fastani sur la lignée la plus proche	/
Genome downloader	3.0.0	Le workflow Genome downloader télécharge les génomes de référence	id taxon et nom du NCBI associé	GCA téléchargé et fichier avec le nom associé	Melainabacteria, Sericytochromatia, Chloroflexia, Vampirovibrionales, Thermostichales, Pseudanabaenales, Gloeomargaritales, Acaryochloridales, Gloeobacter
GenContams	3.0.0	CheckM (Parks et al., 2015) calcule la complétude, la contamination et l'hétérogénéité des bins.	bins + génomes de ref	rapport qualité	contamination <15% complétude>85%
GenContams	3.0.0	GUNC (Orakov et al., 2021) va noter la qualité de la bin avec le tag 'True' (passe le test) ou 'False'.	bins + génomes de ref	rapport qualité	Doit passer le critère avec 'True'
ANI	3.0.0	FastANI (Jain et al., 2018) calcule, pour chaque génome, le pourcentage d'identité avec chaque autre génome.	bins + génomes de ref filtrés	taux d'ANI du premier génome face au second	mode=onetomany
Orthology	2.0.8	Prodigal (Hyatt et al., 2010) traduit les génomes en protéomes, et OrthoFinder (Emms & Kelly, 2019) recherche les gènes orthologues communs aux protéomes de référence et aux bins.	bins + génomes de ref filtrés	Gènes Orthologues (OGs)	presence=60
Orthology_companion	2.0.0	Orthology_companion va filtrer à l'aide de 3 critères les OGs récupérés.	OGs	liste des OG ayant passé le filtre	#1: unwanted=0 presence=60 duplication=5 #2: unwanted=10 presence=60 duplication=1
Multi locus Maximum Likelihood Phylogeny	2.0.0	MUSCLE (Edgar, 2004) va aligner ces OGs, BMGE (Criscuolo, 2010) va les nettoyer, et SCaFoS (Roure et al., 2007) fait une surmatrice, c'est une concaténation des gènes post-BMGE (position conservée) un après l'autre.	OGs filtrés	data-ass.fasta	jackk=no
iqtree	2.3.4	IQ-TREE (Nguyen et al., 2014) va inférer l'arbre le plus probable.	data-ass.fasta	arbre phylogénétique	1 000 réplicas bootstrap 20 threads de calcul
prodigal	2.6.3	Prodigal va traduire les génomes en protéomes.	bins + génomes de ref	protéomes	/
ORPER	2.0.0	Après avoir téléchargé les différents génomes publics avec `wget` et attesté de leur qualité avec CheckM (Parks et al., 2015), RNAmmer (Lagesen et al., 2007) va prédire les gènes codants de l'ARN ribosomal et identifier les ARNr. RAxML (Stamatakis, 2014) va inférer les arbres phylogénétiques.	protéomes, nom et niveau taxo des taxons de référence et outgroup	arbre phylogénétique	reftaxon=Cyanobacteriota reflevel=phylum outtaxon=Candidatus Melainabacteria outlevel=phylum

FIGURE 15 : **Tableau récapitulatif des étapes faites pour l'analyse phylogénétique.**

8.1.1 Download, assemblage, binning et annotation des génomes de cyanobactéries publics

Les processus d'assemblage des génomes et d'annotation taxonomique sont réalisés ensemble dans le script pipeline.job, disponible en **Annexe M**. Il permet d'automatiser l'ensemble des processus liés au download, assemblage, binning et identification de chaque run. La taxonomie utilisée par la GTDB provient du 28 avril 2023 (version 214).

L'assemblage des séquences est actuellement uniquement possible pour les séquençage des reads en *paired-end* et non *single-end*[10].

genome	classification	closest_ani
Mb_bin-37	d___Bacteria ; p___Cyanobacteriota ; c___Cyanobacteriia ; o___Pseudanabaenales ; f___Pseudanabaenaceae ; g___Pseudanabaena ; s___Pseudanabaena sp014696345	98.42
Ct_bin-103	Unclassified Bacteria	Insufficient number of amino acids in MSA (0.8%)
Ct_bin-121	d___Bacteria ; p___Cyanobacteriota ; c___Cyanobacteriia ; o___Pseudanabaenales ; f___Pseudanabaenaceae ; g___Pseudanabaena	N/A

TABLE 2 : **Exemples d'informations fournies par GTDB.** Le nom du génome, la lignée la plus probable, ainsi que le pourcentage de ressemblance face à cette lignée sous forme d'ANI.

Les bins contenant des cyanobactéries ou des bactéries non classifiées sont récupérées des fichiers de sortie de la GTDB à l'aide des commandes :

```
bins_uncl=$(grep 'Unclassified' "$inner_file" | grep -v 'No bacterial'  | awk -F':' '{print $1}'  | awk '{print $1}')
bins_cyano=$(grep 'Cyano' "$inner_file" | awk -F':' '{print $1}'  | awk '{print $1}')
```

8.1.2 ANI

8.1.3 Orthology

Les MAGs et génomes de référence sont traduits en protéomes conceptuels et vont être comparés entre eux sous forme de blast "all-vs-all". Avec une *presence* de 60%, ces blasts vont permettre de générer les OGs.

Le script Python Orthology_companion.py sert à filtrer les groupes orthologues obtenus après que Orthology les ait trouvés. Ces filtres impliquent le pourcentage de présence de la séquence orthologue au sein des génomes[11], le taux de duplication toléré (polygénie) et le nombre d'élé-

10. Single : lecture séquencée d'un seul côté, Paired : des deux côtés, offrant une meilleure qualité et couverture.
11. Exemple : si le nombre total de génomes est de 800, une *presence* de 60% autorise les OGs contenant au minimum 480 génomes de passer au travers du filtre.

ments indésirables tolérés. Ces éléments indésirables (*unwanted*) permettent de minimiser l'effet de l'outgroup, qui pourrait avoir moins de gènes orthologues avec le reste.

8.1.4 Phylogénies

Une fois que ScaFos (Roure et al., 2007) a généré et stocké sous forme d'un fichier fasta l'alignement supermatriciel des séquences, IQ-TREE (Nguyen, 2014) est exécuté.

Le modèle d'évolution moléculaire sélectionné sera déterminé par une méthode heuristique, MFP, qui permet de choisir le modèle offrant le meilleur *fit* par rapport aux données.

Le programme RAXML, utilisé pour la création de phylogénies, est très lent. Pour cette raison, dès que le processus précédant l'utilisation de RAXML est terminé, les fichiers intermédiaires data-ass.fasta générés sont copiés pour être ensuite traités par le programme IQ-TREE, qui est dix fois plus rapide (Nguyen, 2014).

8.1.5 Choix des génomes et MAGs pour la deuxième phylogénie

Plusieurs critères sont utilisés lors de la sélection des bins en fonction des résultats de la première phylogénie :

- Si deux bins appartiennent au même clade d'intérêt et proviennent du même échantillon, donc créés par des programmes de binning différents, la bin à conserver est déterminée en fonction des résultats de l'analyse CheckM.
- Lorsque plusieurs bins apparaissent regroupées sous forme de "râteau" (c'est-à-dire une dizaine de bins présentant des longueurs de branches égales à zéro entre elles), trois à quatre des meilleures bins sont conservées.
- Toutes les *unwanted* sont récupérées.

63 génomes de référence (GCF) tirés de l'article (Cornet et al., 2018) sont sélectionnés, car ils représentent la diversité globale des cyanobactéries. Cependant, cet article datant de 2018, certaines cyanobactéries basales ne sont pas représentées. Pour combler cette lacune, des génomes supplémentaires sont ajoutés : les Acaryochloridales, Pseudanabaenales, Gloeobacter, Gloeomargaritales, Thermostichales, Vampirovibrionales (outgroup), et certaines Melainabactéries.

8.2 ORPER

Dans sa version *lite*, il fonctionne en inférant l'arbre phylogénétique en téléchargeant et extrayant séquences ribosomiques des génomes publics.

8.3 Machines

L'ensemble des programmes et scripts a été exécuté sur des clusters de calcul de l'Université de Liège en raison de leur demande computationnelle élevée :

— L'utilisation de GENERA et de ORPER s'est déroulée sur la partition bio de nic5 : **durandal-2** (2021), composée de nœuds NEC HPC2224Ri-2 avec 128 cœurs AMD Rome, 2 To de RAM, et 18 To de stockage SSD.
— Le reste des analyses, incluant EDirect, le téléchargement d'archives, et l'utilisation de scripts Perl, ont été effectuées sur **durandal** (2013-2019), un cluster IBM/Lenovo Flex comprenant 1 nœud x440 et 1 nœud x240, pour un total de 228 cœurs Intel Xeon, 2,9 To de RAM, et 162 To de stockage.

Les analyses de machine learning ont été réalisées sur **aida** (2023), une *workstation* utilisée comme serveur de calcul. Il s'agit d'un serveur Supermicro SYS-740GP-TNRT équipé de 20 cœurs Intel Xeon, 256 Go de RAM, 2 NVIDIA Tesla A100, et 7 To de stockage SSD.

9 Résultat : Récupération des données brutes

9.1 Récupération d'accessions susceptibles de contenir des cyanobactéries basales

Dans le but de collecter des runs séquencés issus d'expériences (SRR) contenant des cyanobactéries, une interrogation des bases de données du NCBI est effectuée. La démarche se décline en deux approches qui, bien qu'elles puissent présenter un certain chevauchement de résultats, visent chacune à récupérer des échantillons présents dans des parties différentes de la taxonomie des Cyanobacteriota.

— D'une part, une sélection dite **"positive"**, permet d'identifier les échantillons contenant un minimum spécifié de paires de bases (pb) au sein des clades d'intérêts, les taxons basaux.
— D'autre part, une sélection dite **"négative"**, qui consiste à retenir les échantillons uniquement si la différence entre le nombre total de paires de bases associées au phylum des cyanobactéries et celui des clades non pertinents de cyanobactéries (les taxons enfants) atteint un seuil spécifié.

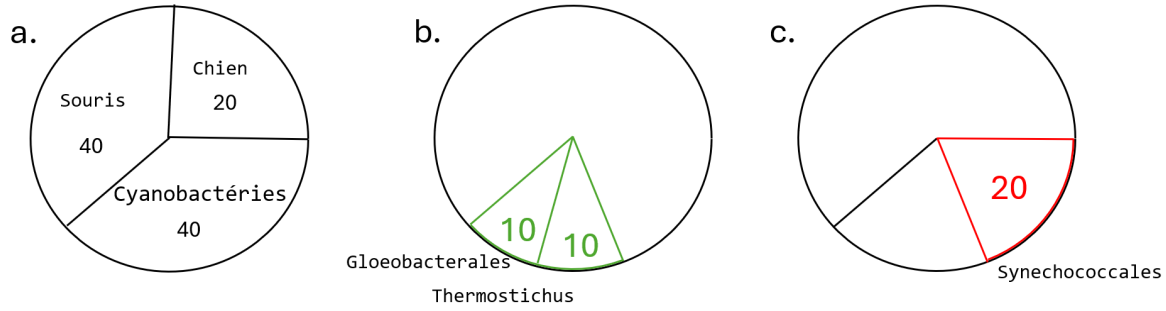

FIGURE 16 : **Principe des deux types de sélection sur un échantillon avec un seuil minimal de 19 pb. a.** Un échantillon de 100 pb contient 40 pb associés aux cyanobactéries. **b. Sélection positive :** les Gloeobacterales et les Thermostichus, des taxons d'intérêt, comptabilisent chacun 10 pb. Leur somme permet à l'échantillon d'être retenu car supérieur à 19 pb. **c. En sélection négative,** les Synechococcales, présentes sur 20 pb et jugées non intéressantes, seront soustraites des 40 pb totale, permettant également à l'échantillon d'être retenu car 20 pb non assignés.

Voici ces logiques traduites en requêtes SQL qui servent à sélectionner des runs dont certains reads ont été associés à la présence de Cyanobacteriota (tax_id=1117) :

```sql
SELECT DISTINCT m.acc
FROM `nih-sra-datastore.sra.metadata` AS m
JOIN `nih-sra-datastore.sra_tax_analysis_tool.tax_analysis` AS tax
      ON m.acc = tax.acc
WHERE tax.tax_id IN (IDS_TAXONS_INTERESSANTS)
      AND tax.total_count * m.avgspotlen > MINIMUM_SPECIFIE;
```

```sql
SELECT m.acc
FROM `nih-sra-datastore.sra.metadata` AS m
JOIN `nih-sra-datastore.sra_tax_analysis_tool.tax_analysis` AS tax
     ON m.acc = tax.acc
WHERE tax.tax_id IN (IDS_TAXONS_NON_INTERESSANTS)
GROUP BY m.acc
HAVING
     MAX(CASE WHEN tax.tax_id = 1117 THEN tax.total_count * m.avgspotlen ELSE 0
          END)   --1117= id des CYANOBACTERIA
      - SUM(CASE WHEN tax.tax_id != 1117 THEN tax.total_count * m.avgspotlen
         ELSE 0 END) > MINIMUM_SPECIFIE;
```

9.1.1 Récupération des identifiants taxonomiques

Afin de minimiser le risque d'exclure des cyanobactéries potentiellement basales lors d'une sélection négative, il est préférable de ne pas retirer un ordre entier présenté comme étant non basal dans la phylogénie de référence. Il est souhaitable et plus prudent de se concentrer sur le retrait des genres ou familles de cet ordre quand ils sont clairement identifiés dans la phylogénie. Cette stratégie permet de conserver des organismes affiliés à cet ordre qui pourraient avoir été mal assignés taxonomiquement et se retrouveraient ainsi évolutivement éloignés de leur position réelle basale, créant de la polyphylie. Cela entraîne une situation où des organismes censés partager une origine évolutive commune (même lignée) se retrouvent en réalité très éloignés d'un point de vue évolutif. En effet, les premières classifications se basaient principalement sur des critères morphologiques, rendus obsolètes par les avancées en génétique (Lahr et al., 2014 ; Taylor, 1980). Ce décalage entre taxonomie traditionnelle et phylogénie génétique moderne crée des problèmes de classification importants, rendant la taxonomie actuelle des cyanobactéries particulièrement complexe (Komárek, 2018).

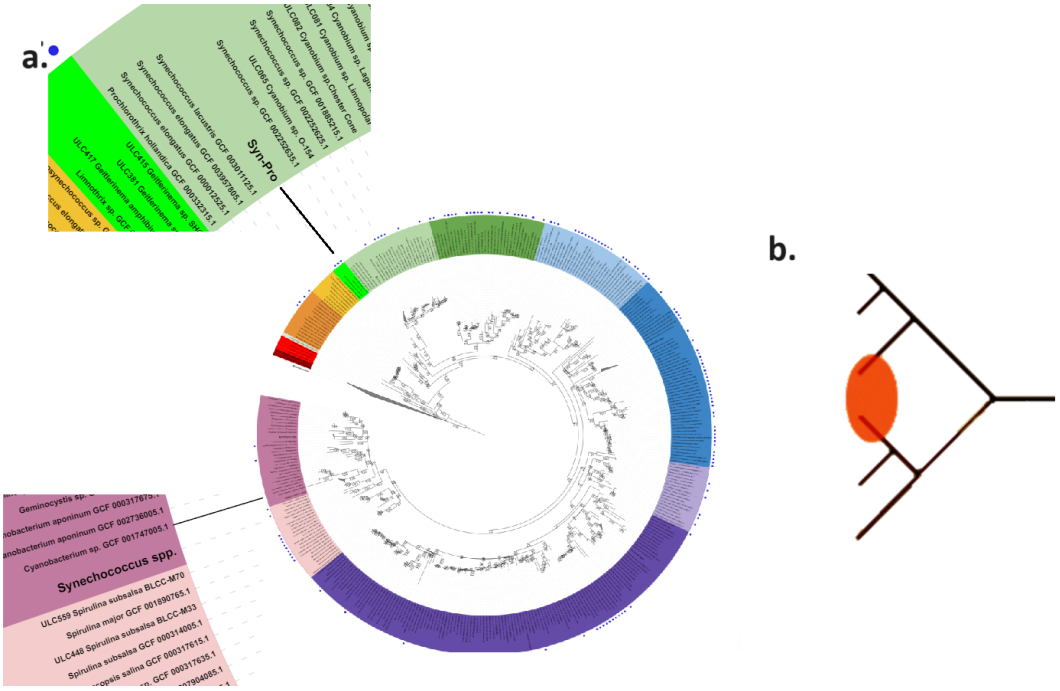

FIGURE 17 : **Mise en évidence de polyphylie dans l'arbre phylogénétique ARNr 16S des cyanobactéries selon L. Cornet (2021). a.** Deux clades de Synechococcus, qui devraient être proches en raison de leur classification fortement apparentée, se trouvent à des endroits très éloignés dans la phylogénie. **b.** Vue schématique de la polyphylie : ensemble de deux clades excluant leur ancêtre commun le plus récent.

Pour la sélection positive, il est préférable cette fois de se baser sur les identifiants des ordres, plus englobants, afin d'éviter de perdre certaines cyanobactéries basales. Certaines souches de Synechococcus seront également prises en compte dans cette sélection, car, bien que l'ordre des Synechococcocales soit globalement beaucoup plus éloigné des positions basales, certaines souches

se trouvent bel et bien à la base de l'arbre phylogénétique et sont donc intéressantes à étudier.

9.1.2 Ajout de conditions à la requête

Tous les échantillons du NCBI ne sont pas éligibles à la création de MAGs (Metagenome-Assembled Genomes) en raison de leur type de librairie (métagénomique, génomique, synthétique, etc) et/ou de leur stratégie expérimentale (amplicon, RNA-seq, WGS, etc). Seuls les échantillons issus de librairies métagénomiques et ayant utilisé la stratégie de WGS (whole genome sequencing) sont retenus, car les expériences métagénomiques peuvent toujours révéler des organismes encore inconnus, ces dernières n'étant pas toujours exploitées au maximum.

Une condition supplémentaire concerne le nombre de paires de bases identifiées comme appartenant à un ou plusieurs des taxons sélectionnés. Les génomes des cyanobactéries présentent une variabilité de tailles, allant de 1,5 à 9 Mb (Larsson et al., 2011 ; Pessi et al., 2023). Pour garantir la qualité de l'assemblage, il est crucial de s'assurer que les données brutes permettent d'obtenir une profondeur de séquençage adéquate. En effet, pour obtenir une couverture fiable, il est nécessaire que chaque nucléotide soit lu un certain nombre de fois, conformément à la distribution de Poisson. Le seuil de profondeur de séquençage permettant de couvrir à 99% du génome est théoriquement 5X mais il est recommander de monter à 10X. Ainsi, les échantillons doivent contenir un minimum de 30 millions de paires de bases afin d'assurer une qualité d'assemblage suffisante pour les génomes de 6 Mb.

```
m.librarysource='METAGENOMIC' AND assay_type='WGS' AND pb_restant >30000000 --
    pb_restant = tax.total_count
* m.avgspotlen
```

9.1.3 Champs supplémentaires environnementaux

La requête SQL, utilisée à la fois dans l'objectif de phylogénie et dans celui d'annotation, devra non seulement récupérer les numéros d'accession des runs, mais également inclure les informations pertinentes sur l'environnement dans lequel l'échantillon a été récolté. Les champs extraits sont : *"acc", "bioproject", "biosample", "continent", "country", "date", "organism"* et *"attributes"*.

Ce dernier champ est particulier car ses valeurs sont stockées sous forme de structure imbriquée (*nested structure*). Il est constitué d'une collection de paires clé-valeur, chaque paire décrivant un attribut spécifique de l'échantillon. La fonction SQL **UNNEST** sera utilisée pour désimbriquer cette structure (Group, 2024).

Les deux logiques de sélection ont été combinées en une seule requête, aboutissant, le 13 février 2024, à une récupération de 1470 accessions. Quand les logiques sont traitées par des requêtes différentes,

elles révèlent que la partie positive récupère 474 accessions, tandis que la partie négative en récupère 1338. L'analyse des résultats révèle un recouvrement significatif attendu entre les deux sélections, avec environ 75 % des accessions identifiées dans la sélection positive se retrouvant également dans la sélection négative. Étant donné que la sélection positive est plus simple à effectuer que la négative, cet important recouvrement des données est également un indicateur de la réussite de la sélection négative.

La requête est disponible en **Annexe AAAA**.

9.2 Suppression d'accessions retrouvées dans les assemblages publics

Ces données publiques ont été utilisées dans le cadre de diverses expériences et, dans certains cas, ont déjà été assemblées et répertoriées dans la base de données du NCBI selon un identifiant GCA ou GCF. Détecter ces runs permet de réduire le nombre d'assemblages à réaliser en priorité, car la probabilité de découvrir une nouvelle espèce, bien que toujours significative, reste moins grande pour les échantillons déjà assemblés, étant donné que d'autres chercheurs ont potentiellement déjà effectué des analyses et décrit leur contenu (Ashrafi et al., 2015 ; Benler et al., 2021).

A l'inverse des assemblages RefSeq (GCFxxx), les assemblages GenBank (GCAxxx) offrent une plus grande diversité d'échantillons (Pruitt et al., 2020). Tandis que les assemblages RefSeq sont réalisés par des experts du NCBI et constituent des génomes de référence soigneusement assemblés et annotés, les critères de qualité pour appartenir à la base de données GenBank sont moins rigoureux. Tout chercheur peut y déposer des assemblages, ce qui en fait une ressource essentielle par sa richesse, malgré une qualité potentiellement inférieure. Les GCA sont associés à des *biosamples* et non à des runs, ce qui a pour effet que la recherche et la suppression des échantillons récupérés lors de la requête SQL doivent également se baser sur eux. Étant donné que les assemblages GCF proviennent initialement des données GCA ayant atteints certains seuils de qualité, il est donc possible de se limiter aux GCA pour rechercher les assemblages publics (Ciufo et al., 2018). Parmi ces derniers, 46 de la requête SQL sont liés à des GCA, ce qui réduit le dataset final à 1 413 accessions.

9.3 Récupération de métadonnées complémentaires

En plus des données environnementales disponibles sous forme de termes dans la base de données *metatada*, il est possible d'extraire des métadonnées supplémentaires de cette base de données à partir du programme EDirect (NCBI, 2014). Ces informations complémentaires, fournies sous forme de phrases complètes, enrichissent le contexte environnemental en établissant des liens entre les différents termes. Les valeurs supplémentaires disponibles incluent le "titre de l'expérience", la "description du design", ainsi que le "titre" et le "résumé de l'étude".

acc: DRR290133

title: Sequel sequencing of SAMD00319324KatS3 sedimentSequel sequencing of SAMD00319324

study_abstract: The goal of this study is to determine metabolic potentials of microbial communities in Japanese hot springs

design_description: /

study_title: Hot spring metagenomics

9.4 Gestion des données

9.4.1 Concaténation et nettoyage des champs

Lorsque les données ne sont pas nettoyées au préalable, certaines lignes peuvent contenir de nombreux attributs non pertinents, tels que des adresses e-mail, des noms d'instituts, des marques de machines, ou de nombreux caractères de séparation ('|'). Ces informations superflues, quand elles sont analysées par le modèle de classification, augmentent non seulement le temps de traitement mais entraînent également une mauvaise interprétation des données :

— Les informations pertinentes peuvent être noyées dans un excès de bruit, ce qui peut conduire à une perte de données (sorte de faux négatif).
— Les attributs non pertinents peuvent également induire des faux positifs, car ils sont souvent mal interprétés par le modèle. Par exemple, le mot "Illumina" (une marque de machine de séquençage) reçoit toujours l'étiquette *"illuminated part of the biosphere"* lors de son traitement.

Afin de remédier à ces problèmes, un nettoyage des données est effectué, en ne gardant que les colonnes pertinentes et en concaténant les champs de même nature, si ces données n'entrent pas en conflit en se chevauchant. La réduction du volume d'informations non pertinentes et l'amélioration de la qualité des données fournies optimisent leur analyse ultérieure. En effet, les modèles de Classification Zéro-Shot comme *"bart-large-mnli"* possèdent une limite maximale de tokens (morceau du texte, souvent sous forme de sous-mot), qu'ils peuvent traiter en une seule fois, fixée pour ce dernier à 1024. Certaines lignes, qui comptaient initialement 604 tokens, n'en comptent désormais plus que 172.

Certaines lignes, qui comptaient initialement 604 tokens, n'en affichent désormais plus que 172.

broad_scale_	biome_sam	environment_	env_broad_sc	hix__proporti	isolation_source_sam	fusion
			freshwater biome			freshwater biome
			freshwater biome			freshwater biome
			freshwater biome			freshwater biome
aquatic biome						aquatic biome
					Cryoconite	Cryoconite
					Cryoconite	Cryoconite

FIGURE 18 : **Exemple de la fusion des colonnes.**

Les coordonnées GPS sont également extraites de la colonne *lat_lon_sam_s_dpl34*. Si cette colonne est vide, les coordonnées sont générées à partir des colonnes où sont reportées les latitudes et longitudes [12].

Après cette étape, le nombre de colonnes passe de 440 à 20.

9.4.2 Standardisation des valeurs numériques

Certaines colonnes récupérées contiennent des données numériques continues, telles que la température, l'altitude ou la profondeur. Étant donné que les modèles de ZSC ne sont pas particulièrement performants pour interpréter ces types de données (Fangwei et al., 2024), des tags pré-définis les remplacent, permettant d'enrichir le contexte et standardiser les données.

	date	altitude	depth	temperature_C
Avant	25/07/2012	0.05m	-2cm	22.5
Après	2012	0-0.1 m high	0-1 m deep	20-25°C

FIGURE 19 : **Transformation des 4 champs.**

Le dataset final comprend 20 colonnes et 1413 lignes d'échantillons. Voici un exemple avant/après la gestion des données :

SRR22198305|PRJNA781406|SAMN31614025|Asia|Nepal|2021-03-17|sediment metagenome|135|0.773333333|0.606|0.073084333||||||1.385041333|||||||46969594920|0.785408333|||||||0.760147667||30.751685||||14288412715|||||||0.044590333||||||0.023713667|0.011139|0.014811333|0.007102333|||2021-03-17|||||86.2|||||Sentinel-2|20201101|ASTER V003|20000301-20131130|||||||||||||||||||||8.75|97.5|115|||1.778333333|1.995666667||4067|4062|||||||||||||||||||||||NA|||||||||||||1.238366|||||NA|NA|NA|||||||||16.301251|0.34|Lirung|5.52|G085556E28239N||||3.498558|||||NA|3.005|1.2435|1.481|11.729|0.3685|18.436|||||||||||||||||sediment|||||||||2.165494333||28.13 N 85.33 E|28.2307|28.2298|||||||||85.5628|85.56236||||NA|||||7205||NA|NA|5340|||||||||||NA|NA|0.64|13.7|255.6|4.03||1.137289333|||||||||||||||||NA|||||||7.92|||||||||||||||781406|||||||1|||||||||replicate6||NA||RGI60NA15.04045|2022-11-05T04:08:00.000Z|||0.006|0.003|0.002|||||||||||||||||||||||||||||sediment|||||||||||||||||||1235549.117|||||||||||||||GPS|20210317|432|3||||NA|NA|2.917|||||||||||0.292666667|||||||||11:16|||||||||||||||NA|NA|NA|NA|NA|NA|NA|NA|NA||8|||NA|||||0.2|NA||NA|SRR22198305|NOMIS_Vanishing_Glaciers_Project|Investigating the microbiomes associated to glacier-fed streams around the world|Sequencing

12. Neuf colonnes expriment la latitude ou la longitude.

```
was performed at the Functional Genomics Centre Zurich on a NovaSeq (Illumina)
using a S4 flowcell|Vanishing Glaciers Project

-->
SRR22198305|PRJNA781406|SAMN31614025|Asia|Nepal|2021|SRR22198305|NOMIS_Vanishi-
ng_Glaciers_Project|Investigating the microbiomes associated to glacier-fed st-
reams around the world|Sequencing was performed at the Functional Genomics Cen-
tre Zurich on a NovaSeq (Illumina) using a S4 flowcell|Vanishing Glaciers Proj-
ect|0-1 °C|sediment metagenome|4000m high and higher|NA|2.165494333|28.13 N 85-
.33 E
```

10 Résultat : Classification automatique

10.1 Choix du dictionnaire initial

La Zero-shot Classification requiert l'emploi d'un dictionnaire, également appelé *tagset*, afin d'assurer l'annotation des données textuelles. Celui-ci peut avoir été acquis, élaboré ou adapté selon les exigences spécifiques du projet.

C'est selon ce dernier cas de figure que l'ontologie [13] ENVO va servir de fondation pour la création d'un dictionnaire capable d'étiqueter des échantillons contenant des informations environnementales. Après suppression des 457 termes obsolètes, qui ajoutent de la redondance, et des 2 878 termes provenant d'autres ontologies intégrées dans ENVO (FOODON, UBERON, BFO, NCBITaxon), considérés comme hors sujets [14], le dictionnaire fait état de 3877 termes exploitables, soit presque la moitié du volume initial de 7 212 termes.

Voici des exemples de termes non retenus :

[Term]
id: ENVO:00000018
name: obsolete dry river
def: "A river that has either permanently or temporally lost its water." [MA:ma]
is_obsolete: true
replaced_by: false

[Term]

13. Ontologie : représentation formelle et structurée de concepts et de relations entre concepts.
14. Sujets des différentes ontologies présentes au sein de ENVO. FOODON : culture et aliment, UBERON : anatomie, BFO : mots liens , NCBITaxon : tax_id du NCBI.

id: FOODON:00001001

name: orange juice (liquid)

is_a: FOODON:03301103 ! orange juice

is_a: FOODON:03430109 ! food (liquid, low viscosity)

intersection_of: FOODON:03301103 ! orange juice

intersection_of: FOODON:03430109 ! food (liquid, low viscosity)

Le dictionnaire, bien que déjà affiné, conserve une similitude importante entre certains termes. Malgré des définitions légèrement différentes, ils décrivent des concepts si proches qu'ils seront systématiquement assignés ensemble lors de la classification automatique.

Exemple : *"industrial wastewater treatment plant", "wastewater treatment process", "waste treatment plant", et "wastewater treatment plant".*

L'objectif de la classification est de découvrir des liens entre différents *tags* environnementaux, comme "lac", "volcan", ou "neige", associés à des échantillons contenant la même espèce. Ces étiquettes ne sont pas nécessairement présentes à chaque occurence de l'espèce, mais suffisamment fréquentes pour indiquer une corrélation significative qui pourrait être informative sur une vision plus globale du biome de l'espèce. Cependant, si des termes très similaires dans le vocabulaire, tels que *"wastewater", "water waste", etc.*, sont tous conservés, ils seront constamment associés, introduisant une corrélation très haute mais vaine. Ce bruit réduit l'interprétabilité des résultats en masquant les liens potentiels entre les tags véritablement distincts et informatifs.

De plus, le modèle doit, pour chaque échantillon, évaluer l'ensemble des catégories, rendant le traitement inutilement lent. Pour réduire la taille du *tagset* tout en préservant la précision de l'annotation, il est nécessaire d'appliquer des techniques de *clustering* afin de regrouper les termes très similaires, sans compromettre la diversité des concepts abordés.

10.2 Prérequis afin d'obtenir un bon *clustering*

10.2.1 Recherche du nombre optimal de *clusters*

10.2.1.1 Réduction de la dimensionalité. Le regroupement de termes sous forme de *word embedding*, des vecteurs de 384 dimensions, peut être problématique pour les algorithmes tels que *K-Means* en raison de la "malédiction de la dimensionnalité", un phénomène décrit par Richard Bellman (Bellman, 1984). À mesure que le nombre de dimensions augmente, les techniques d'analyse de données comme le *clustering* rencontrent des difficultés en raison de la dispersion croissante des données : le volume de l'espace augmente si fortement qu'il contient de nombreuses valeurs nulles, ce qui complique le calcul des distances euclidiennes (Peng et al., 2023 ; Steinbach et al., 2004). La mesure de la variance expliquée par chaque composante principale, après une analyse en composantes principales (PCA), aide à atténuer cet effet en déterminant le nombre minimal de

dimensions à conserver. Cela permet de maintenir une représentation efficace des données tout en réduisant au minimum la perte d'information.

La représentation de la proportion de la variance expliquée par chaque composante principale permet de sélectionner un nombre approprié de composantes à conserver.

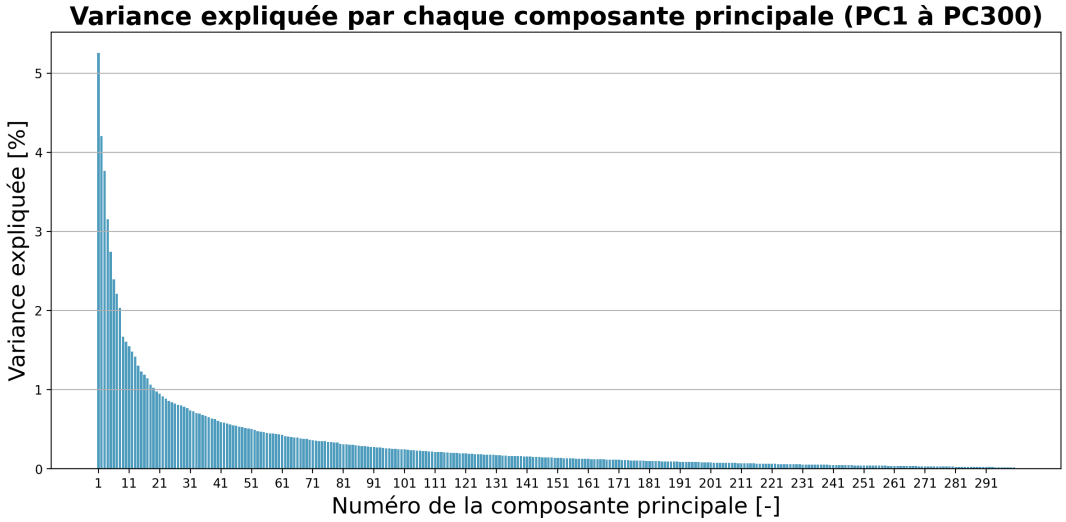

FIGURE 20 : **Variance expliquée par les 300 premières composantes principales.** Les 31, 69, 102 composantes principales expliquent respectivement 50, 70 et 80% de la variance.

Ainsi, les 100 premières composantes principales sont conservées pour les étapes suivantes du *clustering*, ce qui réduit la taille du jeu de données par un facteur de 3,8.

10.2.1.2 Graphes et détermination de *k*. La recherche du meilleur nombre de clusters k se fait vis-à-vis de trois techniques de visualisation : l'*elbow plot*, le graphe de silhouette et la statistique de *gap*. Ci-dessous, les résultats et graphes obtenus suite à l'imprémentation de ces techniques.

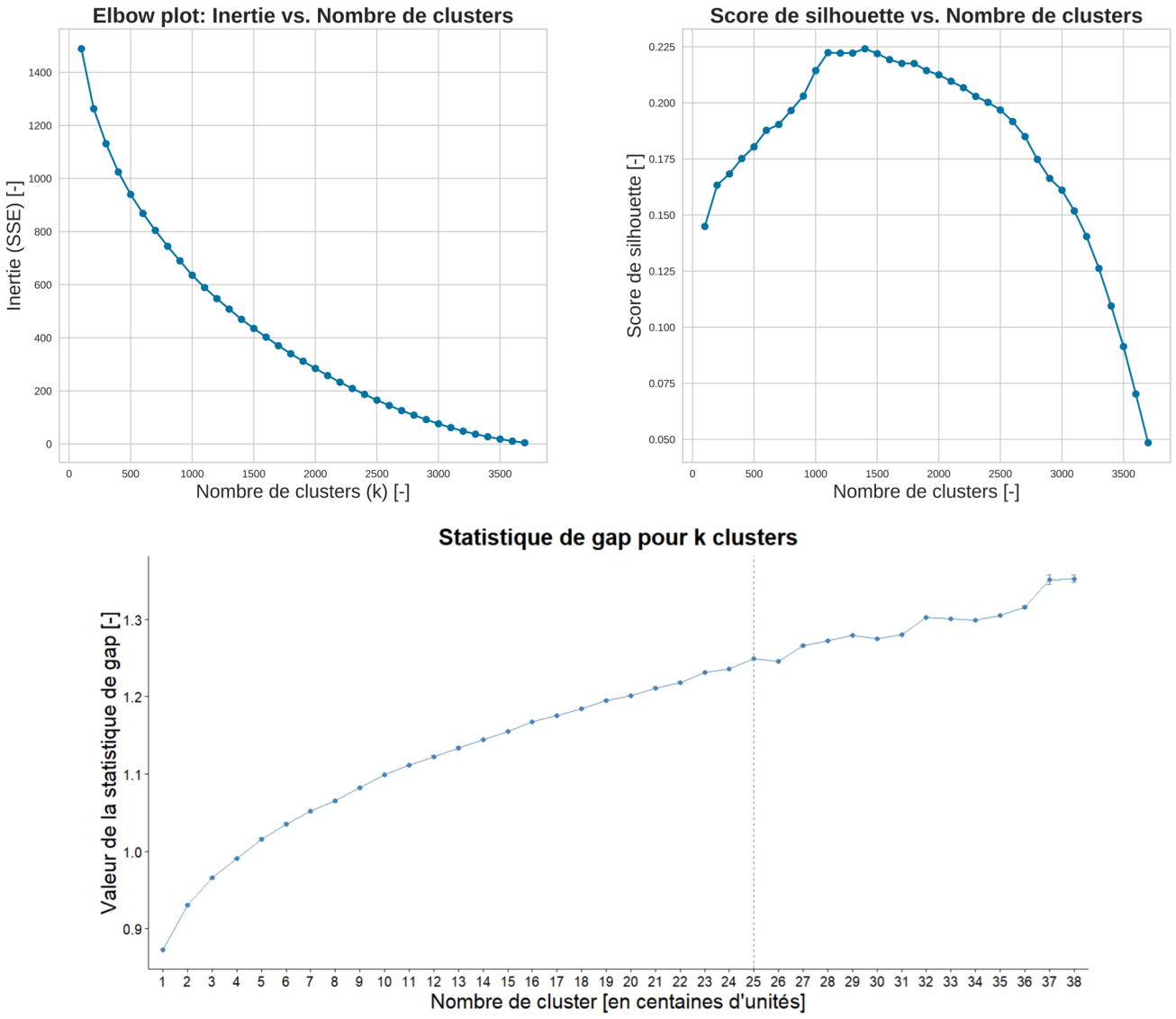

FIGURE 21 : **Graphes déterminant le nombre optimal de clusters après réduction à 100 dimensions. a.** Graphe en coude : il n'y a pas de véritable coude visible. **b.** Score de silhouette : un maximum est observé au niveau de 1500 *clusters*. **c.** Statistique de *gap* : le meilleur k est observé au niveau de 2500.

Elbow Plot : Le coude recherché dans cette technique n'est pas bien défini, avec une courbure progressive et non un creux anguleux. Cette technique est souvent critiquée car le k choisi dans ce type de cas laisse place à l'interprétation de l'utilisateur, la plage de valeur de k étant très large (Shi et al., 2021). Même dans un cas où l'ensemble des donnéesest homogène, sans clusters naturels, la courbe de l'inertie suit généralement une tendance similaire, se comportant selon un ratio de 1/k (Schubert, 2023). Dans ce contexte, il est difficile de tirer des conclusions précises à partir du graphique en coude.

Silhouette Plot : Le score de silhouette atteint son maximum pour des nombres de clusters situés autour de 1500 et 1800, avec un score de 0.225.

Gap Statistic : Bien que la statistique de *gap* soit maximale pour 3800 groupes, ce qui suggère théoriquement un *cluster* par terme, cette solution est inappropriée. En conséquence, il est dès lors plus pertinent d'utiliser le calcul de la "1-erreur standard" (Tibshirani, 2000), qui identifie le point où le taux d'augmentation de la statistique de *gap* commence à ralentir à la place du maximum. Le nombre optimal de *clusters* est alors ramené à 2500.

Ces scores, bien que suggérant une tendance vers une clusterisation autour de 1500-2500, révèlent une robustesse limitée.

10.2.1.3 Estimation manuelle de la qualité de la clusterisation La fourchette de valeur de k déterminée par les techniques d'estimation étant trop vaste, il est nécessaire de procéder à une vérification supplémentaire afin d'éviter des gains ou pertes de termes excessifs lors de la clusterisation. Cette évaluation est manuelle et permet de rendre compte de la qualité de la clusterisation pour plusieurs valeurs de nombre total de *clusters* (500, 1000, ..., 3500).

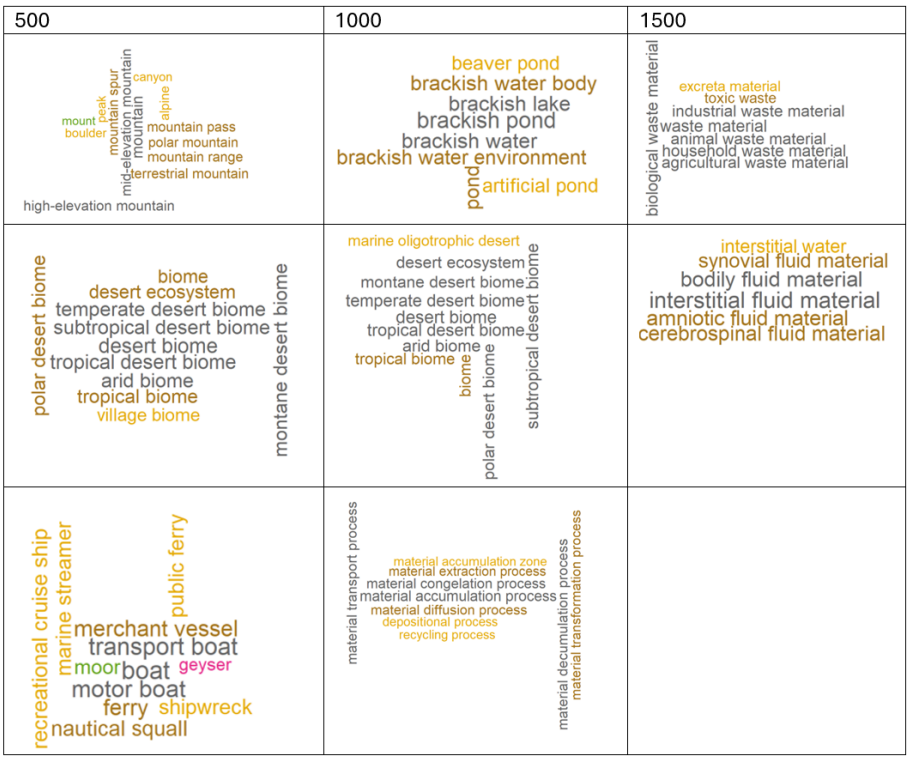

FIGURE 22 : **Analyse des nuages de mots pour différents nombres de clusters.** **500 groupes.** Parmi les 100 plus grands clusters, plusieurs montrent une homogénéité limitée, combinant des concepts variés comme canyon avec pic, divers biomes ou des moyens de transport aquatique avec des geysers. **1000 groupes.** Les clusters continuent de regrouper des concepts hétérogènes, incluant des mares avec de l'eau saumâtre, des déserts chauds ou froids ainsi qu'un avec des notions de "procédés". **1500 groupes.** Parmi les 100 plus grands clusters, seuls deux montrent une homogénéité discutable.

À partir de 2 000 clusters, les représentations en WordCloud montrent des résultats satisfaisants, sans *overlapping* de notions.

10.2.2 Choix du type de *clustering*

Une fois le nombre de groupes déterminé, il est nécessaire de choisir la méthode de *clustering* la plus adaptée aux données. Quatre techniques de *clustering* hiérarchique, en plus du *"K-Means"*, sont appliquées afin de comparer leurs résultats. Les dendrogrammes des 100 plus grands *clusters* sont analysés pour chaque méthode hiérarchiques, car ce sont ces grands clusters qui vont connaître le plus de changements, et donc avoir un impact sur l'exhaustivité du vocabulaire.

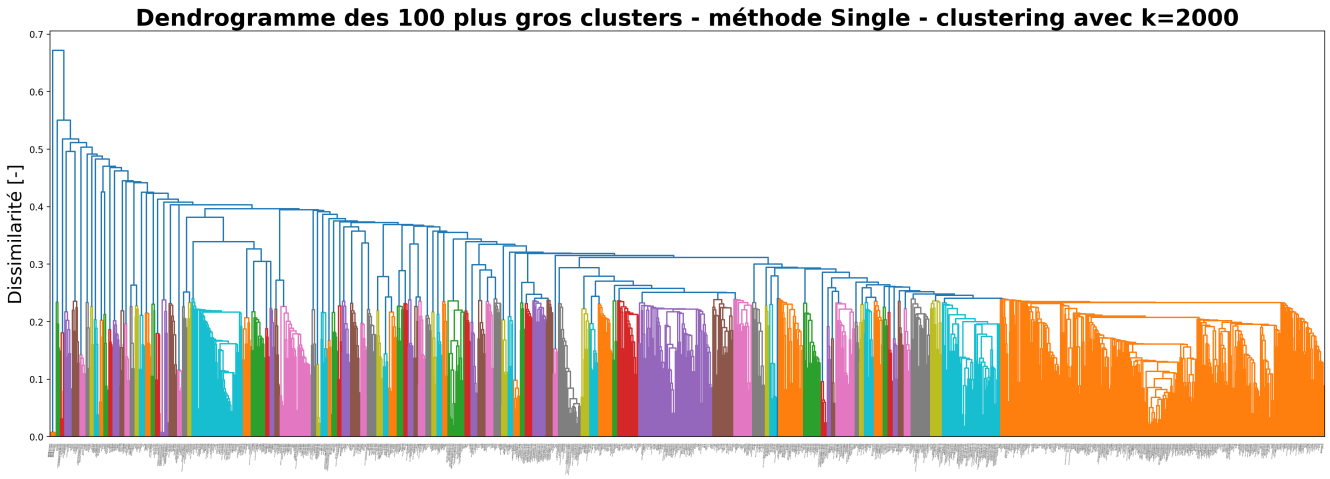

FIGURE 23 : **Single.**

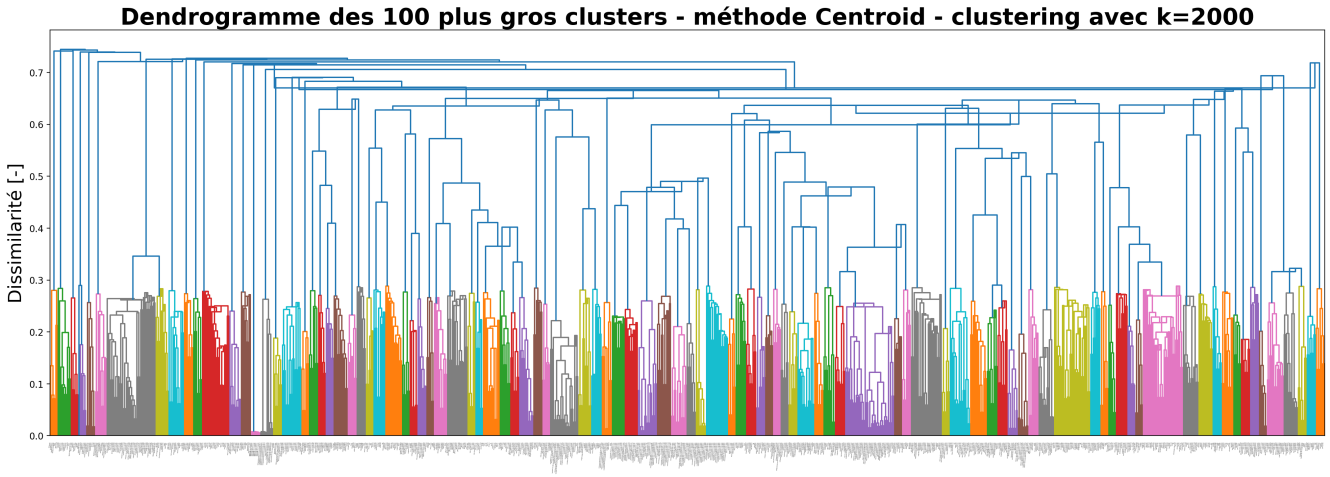

FIGURE 24 : **Centroïde.**

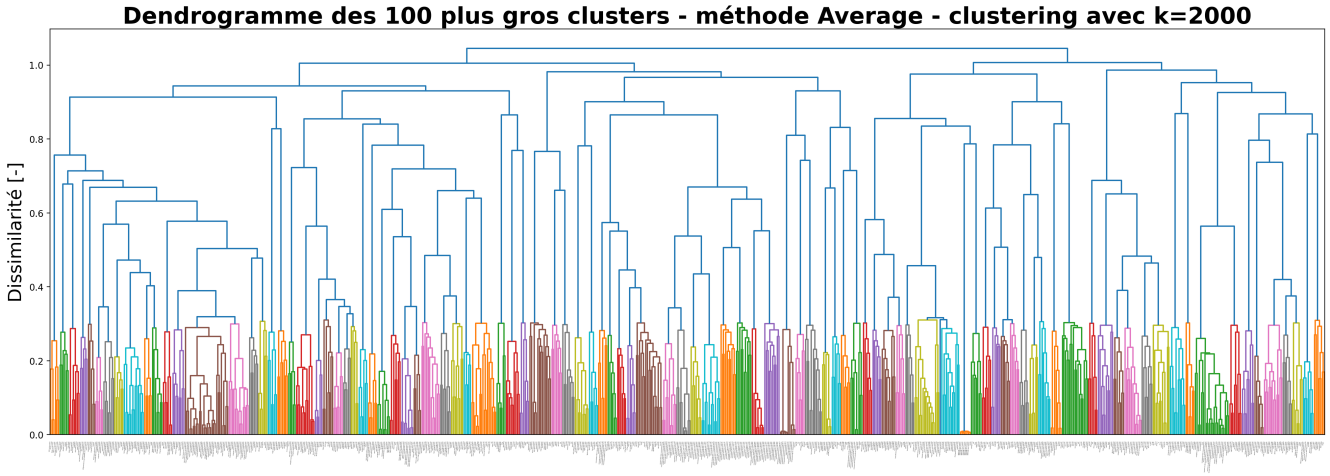

FIGURE 25 : **Average.**

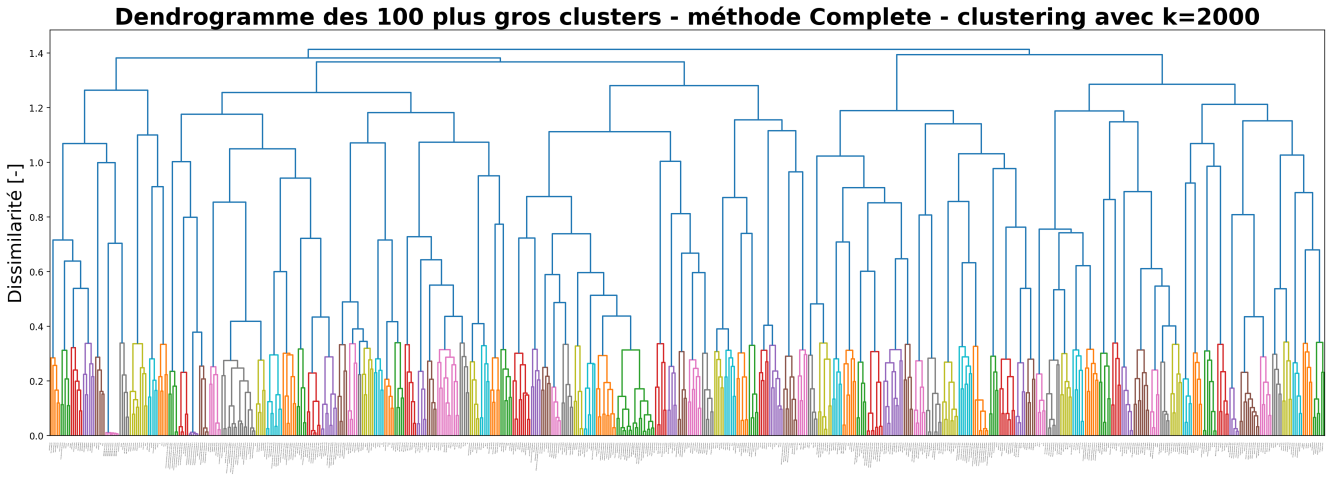

FIGURE 26 : **Complete.**

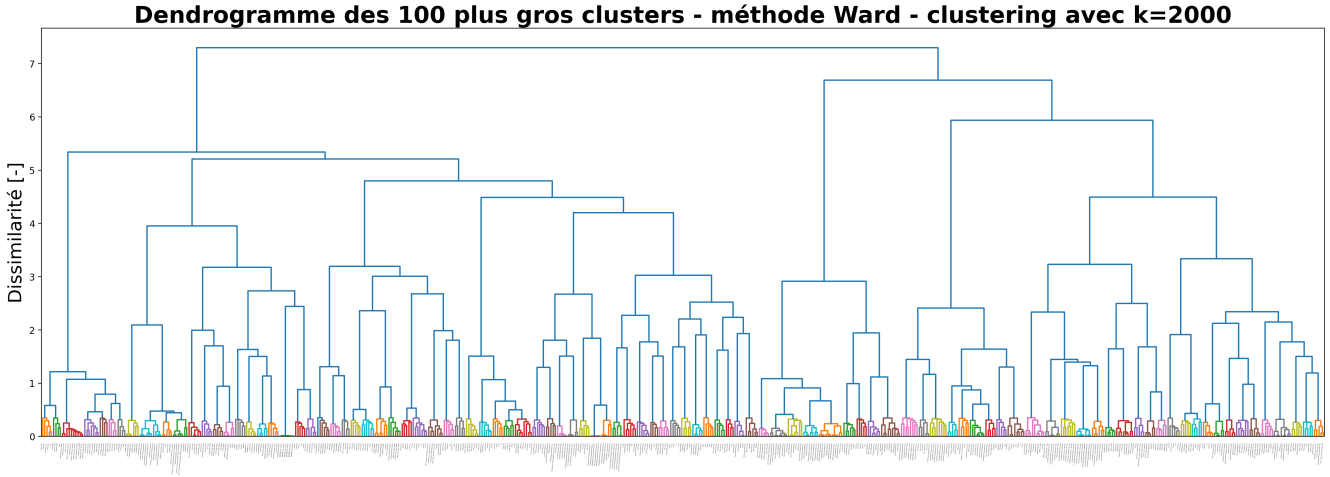

FIGURE 27 : **Ward.**

Afin de déterminer quel algorithme a le mieux fonctionné sur les données, sans avoir à examiner chaque terme individuellement, plusieurs mesures sont calculées : le nombre maximal de termes par cluster, la moyenne des termes par cluster, ainsi qu'une vue d'ensemble approximative des dissimilarités intracluster et intercluster.

Méthode	Max termes	$\overline{m}$ termes	dissim. intra	dissim. inter
Single	329	14.91	0.05-0.2	0.25-0.4
Centroid	38	9.84	0.15-0.3	0.4-0.65
Average	28	8	0.25	0.35-0.6
Complete	20	6.42	0.3	0.4-0.6
Ward	11	5.69	0.3	1
K-Means	20	6.77	pas calculé	pas calculé

Les méthodes de liaison simple et de centroïde ont été écartées car elles sont sujettes au phénomène de *chaining* (Trieschnigg & Kraaij, 2004 ; ttnphns, 2016), bien que ce phénomène soit moins prononcé avec la méthode de centroïde. Cette situation se produit lorsque les données rassemblées au sein d'un cluster forment une chaîne qui s'étend sur de longues distances, ce qui entraîne la formation de gros *clusters* sans véritable cohérence interne. Par exemple, dans le cas de la méthode Single, le premier cluster contient 687 termes très variés, allant de *"bank"* à *"planet"*, en passant par *"crater"* et *"ice"*. La méthode centroïde présente également des résultats contre-intuitifs aux premiers abords, où certaines barres horizontales liant deux clusters sont plus basses que celles liant deux membres d'un même cluster, suggérant ainsi une plus grande similarité entre les nouveaux centroïdes qu'entre les membres de leurs propres *clusters*.

Complete et *average* ont une dissimilarité intracluster d'en moyenne 0.25 à 0.3 qui indique une similarité importante. *A contrario*, la dissimilarité intercluster atteint en moyenne 0.35 à 0.6, ce qui est bien mais pas parfait (certains clusters sont donc similaires à 65%).

La similarité de Ward est favorisée car, en plus d'une forte similarité intragroupe, se trouve également une dissimilarité intergroupe de 1 à 2.5.

Par définition, Ward et *K-Means* sont des techniques de *clustering* qui visent toutes deux à minimiser la variance intracluster. Cependant, *K-Means* est fortement influencé par l'initialisation des centroïdes et suppose que la variance des distributions des attributs est sphérique. En revanche, en utilisant la méthode de *linkage* basée sur la minimisation de la somme des carrés, Ward ne fait pas ces hypothèses et tend à offrir une meilleure performance, en particulier pour des données bruitées, comme dans le cas présent (Ferreira & Hitchcock, 2009 ; Grisel & Varoquaux, 2010).

En conséquent, la méthode de Ward sera utilisée comme technique privilégiée de *clustering*.

10.2.2.1 Choix du meilleur représentant pour chaque cluster. Une fois les groupes formés, le meilleur représentant de chaque *cluster* est sélectionné de manière à ce qu'il maximise la proximité sémantique avec les autres termes du groupe.

Pour *"temperate mixed forest biome", "temperate mixed forest"* et *"temperate forest"*, le terme rencontrant la plus forte similarité, car englobant les autres, est *"temperate mixed forest"*.

10.3 Relecture du dictionnaire : Curation des termes non pertinents

10.3.1 Identification des termes non environnementaux par ZSC

Les thèmes non environnementaux incluent les types de constructions humaines, les éléments astronomiques, ainsi que les trop nombreux termes liés à la neige et à la glace.

Etiquette comparée	**Nbr de termes**	**Termes conservés**
Human	90	*city, anthropogenic environment, human construction, food waste*
Astronomy	42	*abyssal feature, andesite, hypolimnion*
Ice	59	*ice-covered lake, iceberg, ice, slush ice, sewage*

Le vocabulaire passe de 2 000 à 1 809 termes restants.

10.3.2 Curation manuelle

Pour finaliser la création d'un dictionnaire de qualité, une curation manuelle est nécessaire. Certains termes présents ne sont pas indispensables, voire non pertinents. Par ailleurs, des termes très généraux comme *"ecosystem process"* ou *"environmental system process"* apparaissent fréquemment sans apporter de valeur ajoutée. D'autres, tels que *"illuminated part of the biosphere", "well"*, ou *"pier"*, sont souvent mal interprétés par le modèle et assignés de manière non spécifique aux textes, rendant la catégorie non informative. En appliquant la classification *zero-shot* sur des échantillons de textes, il est possible de repérer et d'éliminer ces termes parasites.

Le dictionnaire final compte au total 401 termes.

10.4 Classification automatique sur l'entièreté des données brutes

10.4.1 Détermination du score minimum qu'un tag doit obtenir pour être retenu dans la classification

Tableau récapitulatif produit à partir de l'évaluation de la pertinence des tags, permettant de déterminer à quel moment le *trade-off* est le plus avantageux.

Seuil	Incorrect	Ambigu	Total	Précision
0.5	156	17	332	53.02
0.6	64	14	213	69.95
0.7	34	8	159	78.62
0.8	11	4	102	89.22
0.85	6	2	80	92.5
0.9	3	0	64	95.31
0.95	3	0	64	95.31

Le seuil retenu pour traiter les 1440 textes bruts est de 0.85. À ce niveau, parmi 80 classes attribuées, seulement 6 se révèlent incorrectes, indiquant que 92,5 % des tags sont correctement assignés.

10.4.2 Résultats de l'étiquetage automatique

Une fois le vocabulaire formé et le score minimal déterminé, il est enfin possible de lancer le job sur l'entièreté des textes. Ce dernier dure 1h30, soit 5 fois moins qu'avec le vocabulaire clusterisé et 10 fois moins qu'avec le vocabulaire complet. Ainsi que le montrent les deux exemples repris ci-dessous, les résultats sont également bien meilleurs.

```
Complet:
 - neutral hot spring, neutral spring, hydrothermally-influenced sediment, non-
    saline environment, sedimentation in a water body, hot spring, thermophilic
     sediment, non-saline sediment environment, non-saline aerosol environment,
     neutral occluded front, temperature of environmental material,
    hyperthermophilic sediment, sediment permeated by freshwater, mass of
    biological material, organic object formed through microbial activity,
    water body, freshwater ecosystem, fresh water body, depth of water, well,
    sedimentary stratum, non-saline planetary subsurface environment, microbial
     mat material, freshwater environment, container of an intermittent water
    body, illuminated biosphere part, natural environment, sediment,
```

altitudinal condition, area of low atmospheric pressure, underground water body, flattened elevation, thermal energy, environmental system determined by an organism

- organic object formed through microbial activity, glacial ice, elevated landform, glacial process, glacier, glacial ice accumulation zone, glaciation, cryoconite deposit, high-elevation mountain, area of ice cover, ice surface layer, glacial ice loss, glacier ice field, glacial surface layer, altitudinal condition, mass of compounded environmental materials, geographic sill, well, alpine glacier, temperature of environmental material, distributary, illuminated biosphere part, mass of ice and snow, erosionally enriched glacial ice, cold environment, high temperature environment, two-dimensional fiat ice surface, ice, ice accumulation process, window, snow and ice accumulation process, biological product, results in expansion of, fresh water body, glacial ice gain, national geopolitical entity, yard

Clusterisé :

- neutral spring, hydrothermally-influenced sediment, sedimentation in a water body, thermophilic sediment, temperature of environmental material, mass of biological material, organic object formed through microbial activity, water body, fresh water body, well, sedimentary stratum, container of an intermittent water body, illuminated biosphere part, sediment, altitudinal condition, area of low atmospheric pressure, flattened elevation, thermal energy

- organic object formed through microbial activity, glacial ice, elevated landform, glacial process, glacier, glaciation, glacial surface layer, altitudinal condition, geographic sill, well, temperature of environmental material, distributary, illuminated biosphere part, mass of ice and snow, erosionally enriched glacial ice, cold environment, high temperature environment, ice, window, snow and ice accumulation process, results in expansion of, fresh water body

Personnalisé :

```
- neutral condition, non-saline environment, hot spring, geothermally heated
    environment, fresh water body, natural environment, sediment

- cryoconite, glacier, ice, fresh water body
```

Après être passés par cet étiquetage, les échantillons récupèrent les *tags* pré-définis (température, altitude).

```
- neutral condition, non-saline environment, hot spring, geothermally heated
    environment, fresh water body, natural environment, sediment, 0-1m high,
    0-1m deep, 34.804 N 139.064 E

- cryoconite, glacier, ice, fresh water body, 1000-2000m high, 63.2547 N
    145.4271 W
```

11 Résultat : assemblage et phylogénie

11.1 Assemblage et identification GTDB

L'assemblage des métagénomes n'a pas couvert l'ensemble des 1440 accessions obtenues par requête SQL. Le processus étant extrêmement long, l'assemblage a été réalisé sur une période de 53 jours, afin de garantir suffisamment de temps pour la génération des arbres phylogénétiques. Durant ce laps de temps, 235 métagénomes ont été correctement assemblés.

Certains échecs d'assemblage peuvent être attribués à certains facteurs connus. Premièrement, 6.47% des échantillons ont échoué en raison de l'incompatibilité du pipeline avec les données single-end. De plus, d'autres assemblages ont échoué parce que la taille de certains métagénomes nécessitait plus de mémoire que celle qui pouvait être allouée [attendre luc pour info]. Afin de maintenir le parallélisme de trois tâches d'assemblage simultanées, il n'était pas possible d'allouer plus de RAM par job, rendant certains métagénomes impossible à completer.

Une fois que les programmes de séparation des génomes issus des expériences de métagénomique ont été organisés dans des répertoires distincts appelés *"bins"* (corbeilles), la GTDB indique la lignée la plus probante associée à chacune d'elles.

En récupérant les *bins* que la GTDB a identifié comme contenant des cyanobactéries ou des bac-

téries non classifiées à partir des fichiers de résumé[15], un total de 945 *bins* cyanos et 2825 *"unclassified"* sont collectées, formant ainsi un recueil de 3 770 génomes.

11.2 Génomes de référence

Afin de comparer les *bins* entre elles et d'interpreter leur position dans l'arbre, des génomes de référence plus ou moins proches des cyanobacteries basales vont être téléchargés. Ces neuf taxons, représentant une partie de la diversité des cyanobactéries, vont permettre de récupérer 1588 génomes de référence.

11.3 Contrôle qualité

Une fois les 3 770 génomes néo-assemblés identifiés ainsi que les génomes de référence téléchargés, la qualité de ces génomes va être estimée sous forme de 2 valeurs : la complétude du génome et son degré de contamination[16]. Seules les *bins* avec une complétude suffisante et une contamination mineure sont conservées afin de continuer les manipulations sur des génomes de qualité. 731 *bins* sur les 945 cyano ainsi que 10 des 2 825 unclassified forment les 741 MAGS d'intérêts. Ceux-ci seront complétés par 81 génomes de référence.

Le faible taux de récupération des *bins "unclassified"* est lié au fonctionnement des outils de *binning* et de CheckM. Les *binners* placent souvent les fragments génétiques de faible qualité ou mal assemblés dans des *bins*, entraînant des 'génomes' de faible qualité. De plus, si certains génomes sont extrèmement différents de ceux déjà connus et utilisés par CheckM, les métriques calculées sont nécessairement mauvaises en raison de l'écart significatif par rapport aux références disponibles.

11.4 ANI

Dans sa base de données, la GTDB ne comprend que des lignées issues de collections de cultures, c'est-à-dire des génomes de haute qualité. Les souches récemment découvertes ne sont donc pas incluses, ce qui peut fausser l'assignation des lignées des MAGs. Par exemple, des MAGs proches de ces nouvelles espèces basales de Gloeobacter pourraient ne pas être correctement classifiés. Selon l'hypothèse proposée, le calcul du pourcentage d'identité nucléotidique (ANI) entre les différentes *bins* et les génomes de référence sur l'espèce *Gloeobacter violaceus*, la cyanobactérie la plus basale et présente dans GTDB, pourrait permettre de déterminer si certains MAGs sont proches des

15. Les alignements de séquences multiples bactériens et archéens (MSAs) sont formés à partir de la concaténation de 120 marqueurs informatifs sur le plan phylogénétique pour les bactéries (**gtdbtk.bac120.summary.tsv**) ou de 53 pour les archées (**gtdbtk.ar53.summary.tsv**) (Chaumeil et al., 2024).

16. **Complétude** : proportion de gènes prédits dans l'assemblage présents dans le HMM de la région de l'arbre phylogénétique. **Contamination** : nombre de fois où des HMM de gènes marqueurs sont retrouvés dans l'assemblage. Leur détection répétée peut indiquer la présence de séquences étrangères ou de contaminants.

Gloeobacter en général, en partageant un certain degré d'identité avec une espèce de ce genre.

En réalité, la majorité des valeurs d'ANI, tant pour les taxons que pour les MAGs, varient de 75 à 85 %, aucun MAG ne dépassant 84 %. Par ailleurs, aucune distinction n'est observée entre les taxons en fonction de leur "ordre évolutif" : ils sont tous situés à peu près au même endroit. La valeur anormalement faible de l'ANI (78,9 %) pour *Gloeobacter kilaueensis JS1* par rapport à *Gloeobacter violaceus*, malgré leur appartenance au même genre, remet en question l'hypothèse de rechercher des MAGs intéressants de cette manière. Si au sein d'un même genre, l'ANI descend à 78 %, soit presque autant que pour Pseudanabaena, Melainabacteria et Vampirovibrionales, la recherche ne semble pas justifiée. Cette forte différence pourrait indiquer un transfert horizontal massif de gènes dans l'une des deux espèces, ce qui pourrait expliquer leur forte divergence.

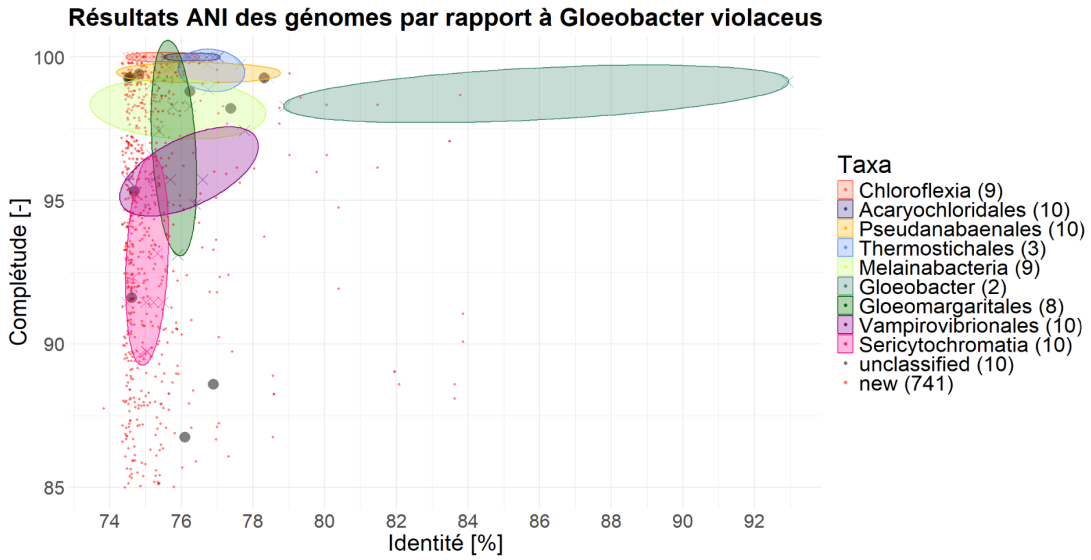

FIGURE 28 : **Graphique représentant les valeurs d'ANI par rapport à *G. violaceus* sur l'axe des x et le degré de complétude sur l'axe des y.** Aucun regroupement distinct n'est observé pour les taxons de référence à l'exception de *Gloeobacter* (les croix représentent les données et les ellipses de couleur soulignent leur couverture dans le graphique).

11.5 Orthologie

La dernière étape avant la construction des arbres phylogénétiques consiste à identifier les séquences orthologues, c'est-à-dire des séquences homologues retrouvées chez des espèces ayant subi une spéciation à partir d'un ancêtre commun.

Les 822 protéomes conceptuels générés, provenant des MAGs et génomes de référence, vont aboutir à la détermination de 116 072 groupes d'orthologie (OGs).

En appliquant des critères de présence [17], de duplication et d'exclusion d'éléments indésirables avec orthology_companion.py, 80 OGs sont conservés.

17. Une présence de 60% signifie que, pour garder un groupe d'orthologie (OG), il doit contenir au moins 494 des 822 génomes.

11.6 Phylogénie

La notion d'orthologie précédemment utilisée est la clé pour la construction des arbres phylogénétiques : plus deux organismes partagent un ancêtre commun récent, plus ils partageront des séquences homologues. Le premier arbre, disponible en **annexe N**, permet de visualiser le placement des MAGs par rapport aux génomes de référence et le groupe de sortie (*"outgroup"*) des Chloroflexi.

Les valeurs suivant le symbole '@' des identifiants représentent la taille de la super-matrice obtenue après concaténation de multiples séquences alignées. Ces valeurs, relativement faibles, avoisinent 2200 pb [18] et peuvent parfois descendre jusqu'à seulement 500 pb. Ces scores, bien en deçà des standards actuels (Dessimoz & Gil, 2010), impliquent un faible recouvrement des séquences. Cette limitation remet en question la fiabilité et la robustesse de l'arbre phylogénétique ainsi que des potentielles hypothèses évolutives qui pourraient être formulées à partir des organismes étudiés.

De plus, certains groupes bien définis, comme les Thermosticales, apparaissent ici comme polyphylétiques.

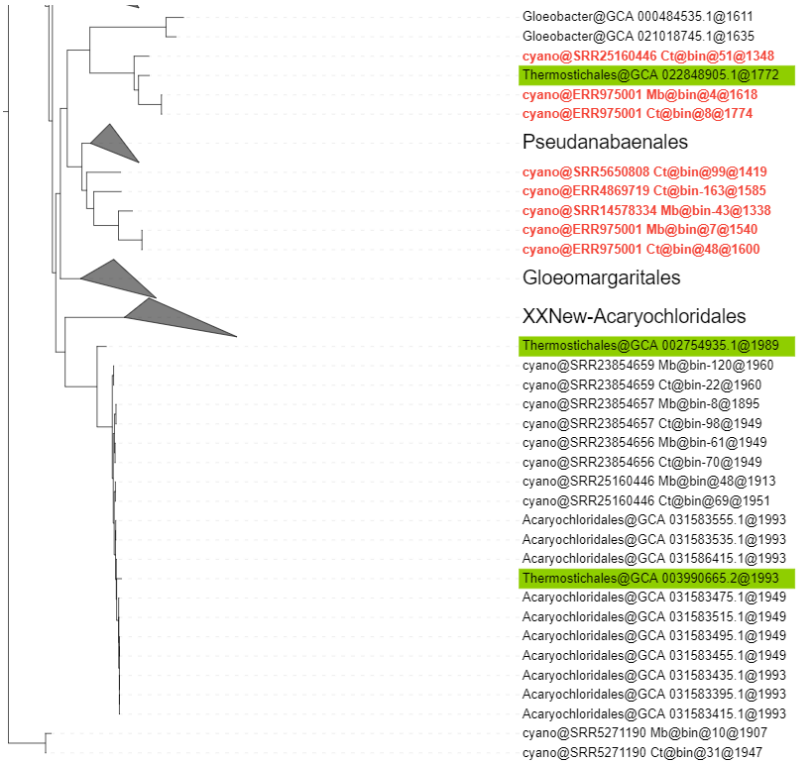

FIGURE 29 : **Phylogénie des Thermosticales.** Un génome se trouve proche des Gloeobacter, tandis que les deux autres sont plus proches des Acaryochloridales.

De nombreuses *bins* se regroupent avec des distances entre elles proches de zéro, indiquant qu'il s'agit probablement d'une même espèce. Une grande partie de ces doublons résulte de la duplication

18. Info disponible sur première ligne du fichier intermédiaire généré par SCaFoS : data-ass.phy.

des MAGs lorsque CONCOCT et MetaBAT ont été utilisés pour générer les *bins*.

FIGURE 30 : **Visualisation de la position de certains MAGs.** Quatre *bins* se retrouvent alignées sur une même branche. Deux d'entre elles proviennent du run SRR27453501, ce qui indique qu'elles proviennent forcément de CONCOCT et MetaBAT. Une autre *bin* provient d'un run différent, SRR27453500 et contient la même espèce.

11.7 Seconde phylogénie

Afin d'obtenir un arbre plus robuste, il est possible de réduire le nombre de *bins*, en ne conservant que quelques *bins* par clade identifié (XXNew), passant ainsi de 741 *bins* de départ à seulement 83.

Un autre ajustement concerne la composition des génomes de référence. Plutôt que d'utiliser les différents taxons téléchargés et filtrés précédemment, de nouveaux génomes vont être téléchargés, et après un filtrage à l'aide de CheckM et GUNC, un total de 207 génomes de référence a été retenu (144 basaux + 63 GCF).

Le processus d'orthologie a été réappliqué sur ces 290 génomes, identifiant 74 965 groupes d'orthologies, ce nombre se réduisant à 125 avec les critères de filtrage n°2.

Une augmentation de la longueur de l'alignement total, passant de 2 242 à 4 091 positions, est obsevée, mais malgré cela, l'arbre n'est toujours pas assez robuste et montre toujours certaines faiblesses.

11.7.1 ORPER

Le workflow ORPER permet de déterminer la position phylogénétique d'un ensemble d'organismes en utilisant un arbre ribosomique (Cornet et al., 2021). Il va utiliser les 290 génomes sélectionnés précédemment et va les comparer aux assemblages publics, aussi bien GCF que GCA, créant une certaine forme de redondance.

ORPER va créer un arbre de 1800 branches. Les espèces, une fois correctement collapsées, offrent une meilleure lisibilité de l'arbre, disponible en **annexe P**.

En premier lieu, le nombre moyen de nucléotide alignés est de 5 000. Bien que ce nombre se rapproche du résultat du deuxième arbre, il reste bien meilleur en raison de la nature des alignements. Comme ce sont des alignements d'ARN ribosomique, ces régions sont naturellement plus petites,

limitant la taille de l'alignement. De plus, ce genre d'alignement est plus robuste car les séquences sont mieux conservées.

11.7.1.1 Analyse des Alignements Nucléotidiques. Le nombre moyen de nucléotides alignés est d'environ 5 000. Bien que ce chiffre soit proche de celui obtenu pour le deuxième arbre phylogénétique, il s'avère plus fiable en raison de la nature des données. En effet, cet arbre repose sur des séquences ribosomiques, des régions naturellement plus petites, ce qui limite la taille de l'alignement. Cependant, ces séquences sont aussi plus conservées, offrant ainsi une robustesse accrue.

11.7.1.2 Ordre des Clades des Cyanobactéries. Les premiers clades des cyanobactéries identifiés dans cette phylogénie suivent l'ordre établi par les phylogénies actuelles. Ces quatre articles illustrent à la fois les similitudes et les divergences observées après l'ordre des Gloeobacterales : (Bettina et al., 2015 ; Chen et al., 2021 ; Cornet et al., 2021 ; Jiri et al., 2014).

- Cornet : Souches de Synechococcus thermophiles, Gloeomargarita, Pseudanabaena, Acaryochloris, Cyanothece, Thermosynechococcus
- Chen : Souches de Synechococcus thermophiles, Gloeomargarita, Pseudanabaena, Cyanothece, Thermosynechococcus, Acaryochloris
- Bettina : Souches de Synechococcus thermophiles, Pseudanabaena, Synechocystis, Thermosynechococcus, Acaryochloris
- Komárek : Souches de Synechococcus thermophiles, Pseudanabaena, Thermosynechococcus, Synechocystis, Cyanothece, Acaryochloris

Bien que certaines zones de l'arbre présentent une variabilité plus marquée, ce phénomène est bien documenté et s'explique par les problèmes de taxonomie évoqués en introduction.

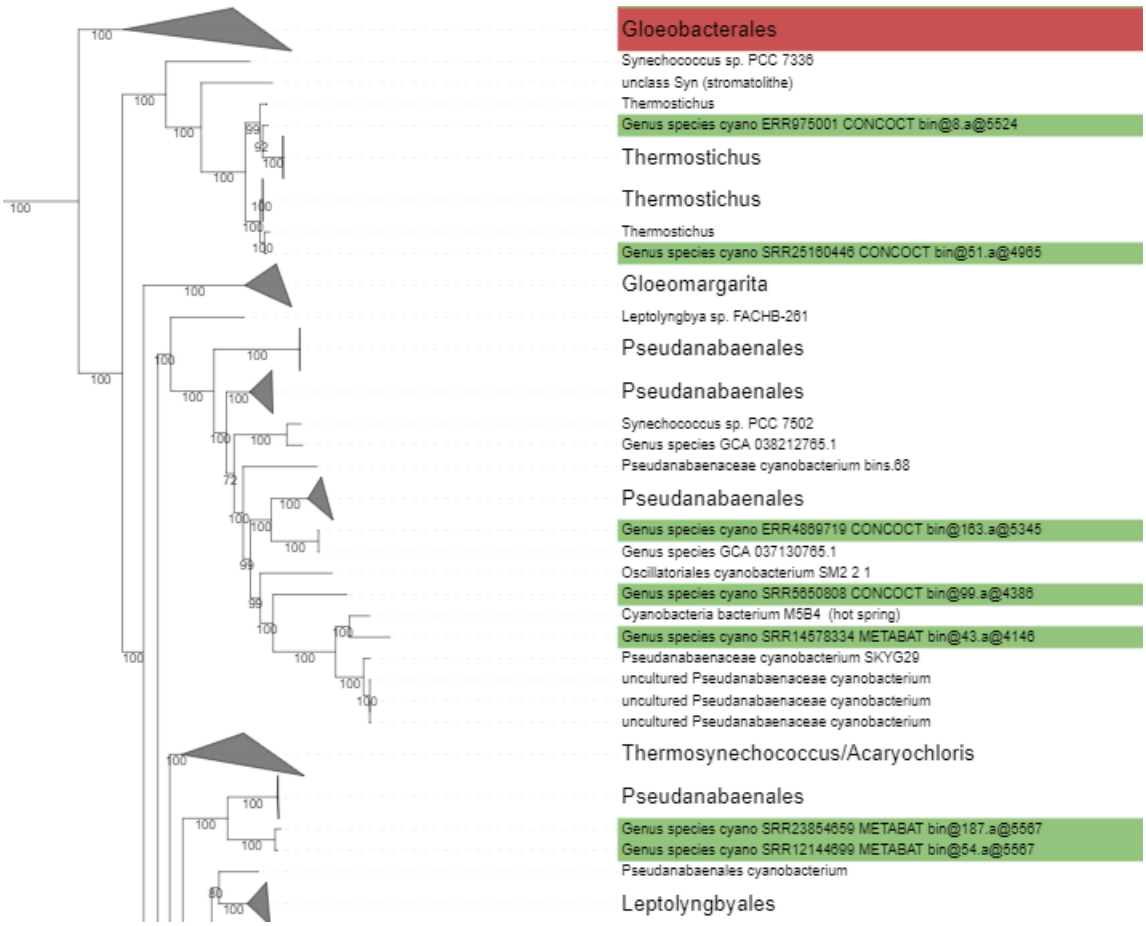

FIGURE 31 : **Phylogénie des Gloeobacterales et autres clades basaux.**

11.7.1.3 Clade intermédiaire Aucun MAG n'a été identifié dans le clade des Gloeobacterales. Toutefois, entre les Melainabacteria (Vampirovibrianales) et les Gloeobacter, on trouve six bins et certains GCA, bien que ces derniers soient situés sur une branche relativement longue et où les noms des GCA semblent hétérogènes. Étant donné que le bootstrap est de 100, cette branche pourrait représenter un aspect particulièrement intéressant et prometteur. Des analyses supplémentaires des gènes photosynthétiques pourraient fournir des informations plus détaillées sur les caractéristiques de ce groupe.

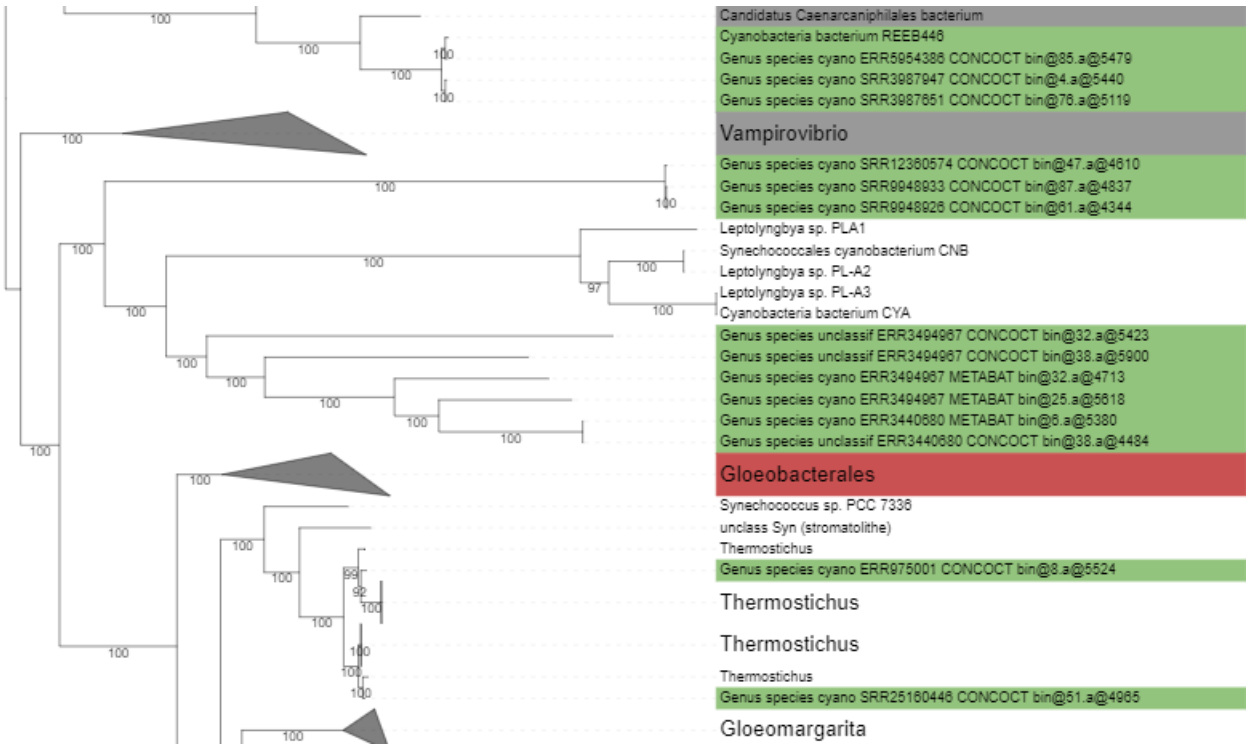

FIGURE 32 : **Mise en évidence du clade se trouvant entre les Melainabacteria et les Gloeobacterales.**

12 Discussion

12.1 Assemblages publics

En recherchant lesquelles de nos expériences avaient déjà été assemblées dans le but de ne pas refaire le travail inutilement, très peu d'assemblages publics avaient *matchés*.

Cette faible présence de biosamples (2.2%) dans les assemblages publics contraste fortement avec la fréquence beaucoup plus élevée observée lors de l'analyse de l'ensemble des assemblages WGS métagénomiques, qui atteint 20,7 %, sans inclure la composante 'présence de cyanobactéries'. A ce stade, aucune explication n'a été trouvée.

12.2 Clusterisation des termes environnementaux

Les calculs pour déterminer le k optimal de clusters n'ont pas fourni de valeurs claires et précises. Cette limitation était attendue, car le dictionnaire, issu d'une ontologie, est conçu pour être aussi exhaustif que possible. Il ne se contente pas de définitions thématiques, mais constitue plutôt un réseau de termes interconnectés, ce qui complique l'identification de clusters distincts.

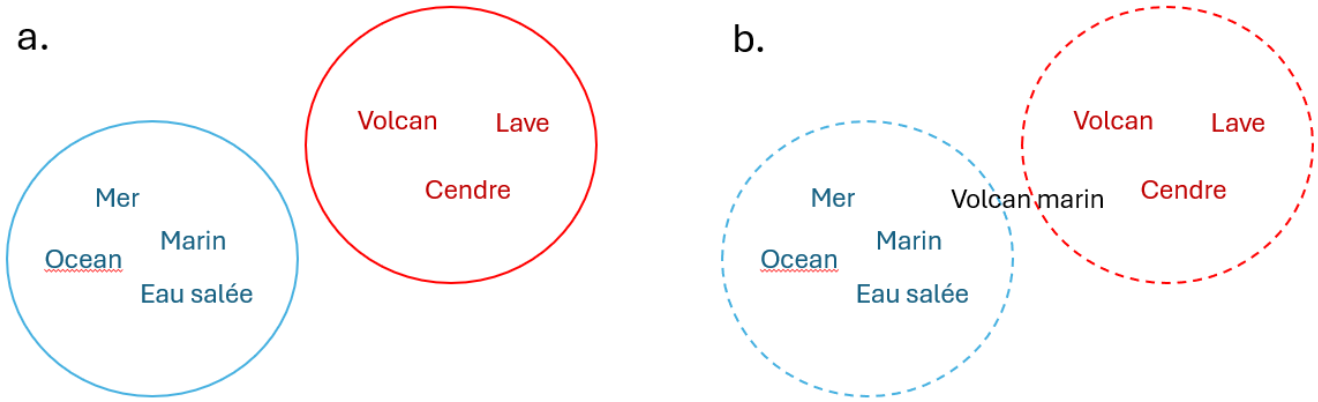

FIGURE 33 : **Illustrations de deux cas pour le calcul du nombre optimal de clusters. a.** Les deux clusters sont bien distincts lorsque que la similarité intracluster est forte tandis que la similarité intercluster est faible. **b.** Un terme intermédiaire complique la clusterisation en étant similaire à la fois à un cluster et à l'autre.

12.3 Filtrage des bins

Le nombre de bins "unclassified" diminue drastiquement après l'analyse de qualité, passant de 2825 à 10. Ces bins, caractérisés par une très mauvaise qualité ou une faible complétude, sont principalement le résultat du fait que les binners, lorsqu'ils traitent des génomes microbiens, génèrent de nombreux MAGs de piètre qualité (Bowers et al., 2017). Ces bins contiennent souvent des contigs fragmentés, couvrant rarement l'ensemble du génome (Arikawa et al., 2021).

GTDBtk les étiquette ensuite comme *"unclassified"* en raison de leur qualité insuffisante ou qu'ils

se trouvent en trop petite quantité dans l'échantillon.

12.4 ANI

Suite aux calculs d'Average Nucleotide Identity par rapport à une espèce de Gloeobacter, le but était de visualiser si les bins allaient se placer plus ou moins proches des différentes Gloeobacterales.

Comme le suggèrent certaines études, le calcul de l'ANI (Average Nucleotide Identity) n'est pas particulièrement utile pour différencier plusieurs taxons, mais il est principalement utilisé pour déterminer si deux génomes appartiennent à la même espèce ou souche, avec des seuils respectifs de >95% et 99,99% (Jain et al., 2018). Au lieu d'une différenciation nette entre les taxons, on observe une distribution bimodale (Rodriguez-R et al., 2021) où la majorité des génomes, à l'exception d'une espèce de Gloeobacter, se regroupent dans une plage de valeurs supérieures à 85% .

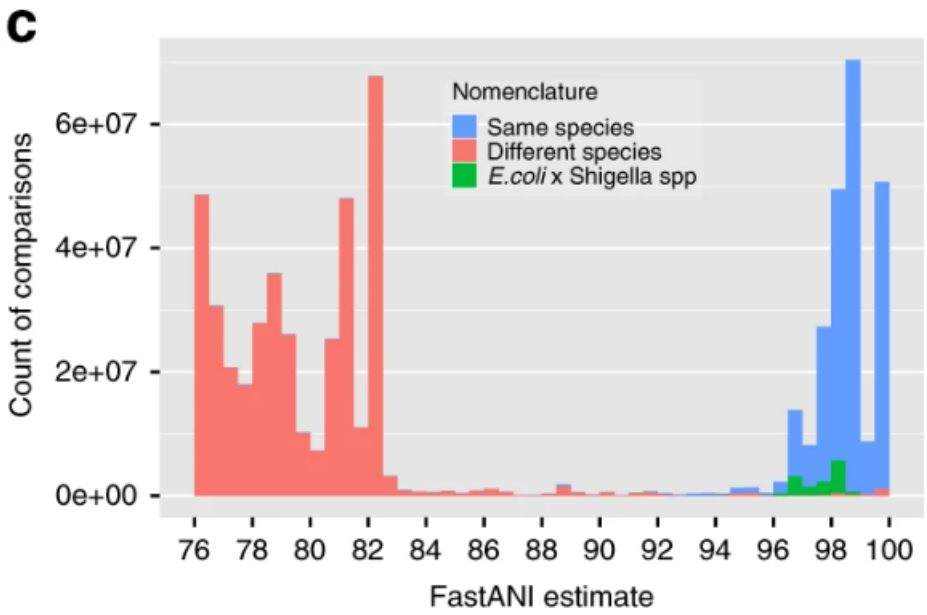

FIGURE 34 : **Exemple reprit du papier (Rodriguez-R et al., 2021) montrant la distribution bimodale des valeurs d'ANI. Rouge :** plage de valeurs s'étendant de 76 à 83% représentant les ANI quand la comparaison est interespèce. **Bleu :** petite plage de valeurs s'étalant de 96 à 100% représentant les ANI intraespèce. Les distributions ne se chevauchent pas, laissant un trou entre elles.

12.5 Arbres phylogénétiques

La récupération d'arbres phylogénétiques aussi peu robustes est plutôt surprenante.

Les faibles niveaux d'alignement entre les organismes peuvent être dus à deux facteurs :

— Les génomes sont incomplets.
— La distance évolutive entre les branches est très importante.

Dans notre cas, la possibilité que les MAG soient incomplets est peu probable. CheckM a permis de filtrer les mauvaises complétudes et si la complétude indiquée par CheckM est suffisante, l'alignement final devrait être correct car BGME, le programme capable de nettoyer l'alignement final, ne forme pas de trou.

Dans notre cas, la possibilité que les MAG soient incomplets est peu probable. CheckM a permis de filtrer les génomes avec une mauvaise complétude. Si la complétude indiquée par CheckM est suffisante, la supermatrice générée par BMGE (Criscuolo & Gribaldo, 2010) devrait être correcte, représentant les alignements multiples. En effet, le programme ne forme pas de trou supplémentaire dans l'alignement multiple.

La distance évolutive explique pas contre mieux la situation. Entre les Melainabacteria et les Cyanobacteriota, il y a eu un minimum de 2,4 milliards d'années pour leur ancetre commun, les cyanobactéries ayant acquis leurs capacités photosynthétiques à ce moment. A cause de ce temps très long, le nombre de gènes est réduit (moins d'OGs), ce qui entraîne des lacunes dans la matrice d'alignement final.

La distance évolutive explique mieux la situation. Entre les Melainabacteria et les Cyanobacteriota, il y a eu au moins 2,4 milliards d'années depuis leur ancêtre commun, période pendant laquelle les cyanobactéries ont acquis leurs capacités photosynthétiques. En raison de ce laps de temps très long, les clades ont eu le temps d'évoluer différemment, le nombre de gènes partagés (OGs) a diminué, ce qui entraîne des lacunes dans la supermatrice d'alignement.

Une solution pourrait être de remplacer l'inférence de l'arbre basée sur les coregenes par une approche utilisant une sous-catégorie, les gènes de *housekeeping*, qui, en général, ne subissent pas de transfert horizontal de gènes (HGT) et pourraient offrir de meilleurs résultats. Former un arbre phylogénétique à partir des séquences ribosomiques est également une option, et c'est d'ailleurs celle qui a été choisie.

13 Perspectives

Compte tenu des résultats encourageants obtenus avec l'arbre phylogénétique basé sur les 235 runs, il est interéssant de poursuivre l'assemblage et l'analyse des runs d'intérêt sélectionnés. Cette expansion pourrait révéler de nouveaux organismes qui enrichiraient les parties basales de la phylogénie. Depuis la fin de l'assemblage durant le mémoire, le workflow a été réactivé à plusieurs reprises, atteignant actuellement plus de 500 assemblages.

L'accent initialement mis sur les clades basaux en partant de calcul de paires de base total basé sur les cyano, au détriment des Melainabacteria, peut nécessiter une réévaluation. Explorer les accessions obtenues en partant des paires de bases totale de Melainabacteria, et non les cyanobactéries, moins les taxons non intéressants permettrait d'élargir le spectre des accessions, avec la possibilité

de découvrir également des lignées jusque-là inconnues.

Le groupe d'organismes identifié entre les clades Melainabacteria et Gloeobacter requiert une investigation approfondie. Une première étape consisterait à confirmer leur positionnement phylogénétique afin d'éliminer toute possibilité d'erreurs liées aux workflows d'analyse. Si ces organismes se révèlent être un clade bien défini, une analyse approfondie de la présence de gènes impliqués dans la photosynthèse permettrait de déterminer leur affinité avec des clades connus. Cette analyse fonctionnelle aiderait à mieux comprendre les caractéristiques biologiques de ces organismes et à évaluer leur potentiel rôle dans l'évolution de la photosynthèse. De plus, ce travail pourrait mettre en lumière des adaptations évolutives uniques, enrichissant ainsi notre compréhension de l'évolution des cyanobactéries et de leurs clades apparentés.

Actuellement, le nombre d'échantillons assemblés et étiquetés est encore trop limité pour permettre une analyse approfondie des descriptions de biomes. Toutefois, à mesure que le nombre d'échantillons augmentera, il deviendra possible d'effectuer plusieurs types d'analyses.

En comparant les environnements typiquement associés aux différents clades à l'aide d'analyses bibliographiques, il sera possible d'identifier de nouveaux biomes associés à ces clades en regardant les étiquettes des MAGs présents.

L'application de l'algorithme Apriori (Xie, 2021) pourrait offrir des perspectives supplémentaires en révélant des corrélations inédites entre les étiquettes environnementales et les groupes taxonomiques. Contrairement aux analyses basées sur des étiquettes isolées, Apriori examine des ensembles d'étiquettes pour identifier des motifs de co-occurrence au sein des échantillons appartenant à un clade spécifique. Cette approche pourrait dévoiler des liens plus subtils et complexes entre les caractéristiques environnementales et les taxons.

Cette méthodologie pourrait également faciliter l'identification de cyanobactéries basales qui n'ont pas été détectées lors des requêtes SQL, en raison des limites de l'outil STAT dans leur identification. Un étiquetage à grande échelle des SRRs métagénomiques permettrait de détecter des runs présentant des niches environnementales similaires à celles associées aux cyanobactéries basales déjà référencées.

La méthode de classification utilisée, fondée sur l'apprentissage automatique, peut être optimisée par un entraînement supplémentaire, plus axée sur l'environnement, apellé *finetuning* (*Fine-Tuned "Small" LLMs (Still) Significantly Outperform Zero-Shot Generative AI Models in Text Classification*, n.d.). Bien que cette approche n'ait pas été appliquée dans le cadre de ce mémoire en raison de la quantité de données nécessaires, elle constitue une voie prometteuse pour améliorer les performances des classifications futures. En augmentant la précision et la fiabilité des résultats, cet entraînement pourrait rendre l'outil encore plus performant.

Un *preprint* récemment disponible (Chikhi et al., 2024) annonce la mise à disposition gratuite d'un

ensemble de données d'assemblage. Cet ensemble couvre 88 % des 27 millions de fichiers de reads disponibles sur la Sequence Read Archive (SRA). Le projet a nécessité 30 millions d'heures de calcul CPU pour aboutir à une base de données dont la taille dépasse de deux ordres de grandeur celle de GenBank.

Une comparaison avec les données issues de ce travail pourrait permettre d'évaluer l'efficacité et l'utilité de cet outil, en examinant si les assemblages réalisés de notre côté sont similaires à ceux obtenus par cet outil. Si LOGAN se révèle robuste, il pourrait offrir une solution aux limitations de temps de calcul associées à la phase d'assemblage dans notre pipeline.

14 Références

15 Annexes

15.1 Annexe AAAA : Requête SQL

```
SELECT DISTINCT
    m.acc,
    m.bioproject,
    m.biosample,
    m.geo_loc_name_country_continent_calc,
    m.geo_loc_name_country_calc,
    m.collection_date_sam,
    m.organism,
     attr.k AS attribute_key,
     attr.v AS attribute_value
FROM
    `nih-sra-datastore.sra.metadata` AS m
JOIN
    `nih-sra-datastore.sra_tax_analysis_tool.tax_analysis` AS tax
    ON m.acc = tax.acc

JOIN
    UNNEST(m.attributes) AS attr
WHERE
    m.bioproject IS NOT NULL
    AND m.librarysource = 'METAGENOMIC'
    AND m.assay_type = 'WGS'
    AND tax.tax_id IN (
        307595, 1955042, 1152, 3079744, 1892249, 132605, 3079751,
        195253, 1173263, 195250, 321332, 1353266, 321327, 1917166
    ) -- Intéressants
    AND tax.total_count * m.avgspotlen > 30000000
    AND m.mbases <> 0

UNION ALL

SELECT DISTINCT
    m.acc,
    m.bioproject,
    m.biosample,
```

```
    m.geo_loc_name_country_continent_calc,
    m.geo_loc_name_country_calc,
    m.collection_date_sam,
    m.organism,
    attr.k AS attribute_key,
    attr.v AS attribute_value
FROM
    `nih-sra-datastore.sra.metadata` AS m
JOIN
    `nih-sra-datastore.sra_tax_analysis_tool.tax_analysis` AS tax
    ON m.acc = tax.acc
JOIN
    UNNEST(m.attributes) AS attr
WHERE
    m.bioproject IS NOT NULL
    AND m.librarysource = 'METAGENOMIC'
    AND m.assay_type = 'WGS'
    AND tax.tax_id IN (
        102231, 1121, 1150, 1161, 1890424, 1890443,
        1890449, 1890505, 2055830, 217161, 2303508, 2303528, 241421, 2546365,
        262068, 263510, 268175, 2784135, 28070, 2815910, 3039985, 3039989,
        3079745, 3079753, 3079756, 44474, 52605, 52607, 582491, 688241, 693222,
        945733, 1117, 3079750
    ) -- Non intéressants
GROUP BY
    m.acc,
    m.bioproject,
    m.biosample,
    m.geo_loc_name_country_continent_calc,
    m.geo_loc_name_country_calc,
    m.collection_date_sam,
    m.organism,
    attr.k,
    attr.v
HAVING
    MAX(CASE WHEN tax.tax_id = 1117 THEN tax.total_count * m.avgspotlen ELSE 0 END)
    - SUM(CASE WHEN tax.tax_id != 1117 THEN tax.total_count * m.avgspotlen ELSE 0 END) >
        3000000
```

15.2 Annexe BBBB : infos sur identifiant, lignée et clade

Tableau récaptitulatif des ids utilisés durant la requête SQL. La deuxième colonne indique la lignée des différents identifiants à partir de la lignée commune commançant par : *cellular organisms ; Bacteria ; Terrabacteria group ; Cyanobacteriota/Melainabacteria group ; Cyanobacteriota ; Cyanophyceae*. La dernière colonne représente le nom associé à l'identifiant par le NCBI.

ID	Lineage	Clade
102231	Oscillatoriophycideae ; Chroococcales ; Chroococcaceae	Gloeocapsa
1117	Cyanobacteriota/Melainabacteria group	Cyanobacteriota
1121	Oscillatoriophycideae ; Chroococcales ; Aphanothecaceae	Aphanothece
1150	Oscillatoriophycideae	Oscillatoriales
1152	Pseudanabaenales ; Pseudanabaenaceae	Pseudanabaena
1161	Cyanobacteriota ; Cyanophyceae	Nostocales
1173263	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. PCC 7502
132605	Pseudanabaenales ; Pseudanabaenaceae	Limnothrix
1353266	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. 63AY4M2
1890424	Cyanobacteriota ; Cyanophyceae	Synechococcales
1890443	Cyanobacteriota ; Cyanophyceae	Spirulinales
1890449	Oscillatoriophycideae ; Chroococcales	Microcystaceae
1890505	Cyanobacteriota ; Cyanophyceae	Chroococcidiopsidales
1892249	Gomontiellales	Cyanothecaceae
195250	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. PCC 7336
195253	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. PCC 6312
1955042	Cyanobacteriota ; Cyanophyceae	Gloeomargaritales
2055830	Oculatellales ; Oculatellaceae	Timaviella
217161	Gomontiellales ; Chamaesiphonaceae	Chamaesiphon
2303508	Oculatellales ; Oculatellaceae	Pegethrix
2303528	Oculatellales ; Oculatellaceae	Thermoleptolyngbya
241421	Gomontiellales ; Gomontiellaceae	Crinalium
2546365	Oscillatoriophycideae ; Chroococcales ; Aphanothecaceae	Rippkaea
262068	Oscillatoriophycideae ; Chroococcales ; Cyanothrichaceae	Johannesbaptistia
263510	Oscillatoriophycideae ; Chroococcales ; Aphanothecaceae	Crocosphaera
268175	Oscillatoriophycideae ; Chroococcales ; Chroococcaceae	Chondrocystis
2784135	Oculatellales ; Oculatellaceae	Shackletoniella
28070	Oscillatoriophycideae ; Chroococcales ; Aphanothecaceae	Gloeothece
2815910	Oscillatoriophycideae ; Chroococcales	Geminocystaceae
3039985	Cyanobacteriota ; Cyanophyceae	Nodosilineales
3039989	Cyanobacteriota ; Cyanophyceae	Desertifilales
3079744	Cyanobacteriota ; Cyanophyceae	Acaryochloridales

ID	Lineage	Clade
3079745	Cyanobacteriota ; Cyanophyceae	Prochlorotrichales
3079750	Leptolyngbyales	Neosynechococcaceae
3079751	Cyanobacteriota ; Cyanophyceae	Geitlerinematales
3079753	Cyanobacteriota ; Cyanophyceae	Coleofasciculales
3079756	Oscillatoriophycideae ; Chroococcales	Halothecacae
307595	Cyanobacteriota ; Cyanophyceae	Gloeobacterales
321327	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. JA-3-3Ab
321332	Synechococcales ; Synechococcaceae ; Synechococcus ; unclass. S.	Synechococcus sp. JA-2-3B'a(2-13)
44474	Pleurocapsales ; Hyellaceae	Pleurocapsa
52605	Pleurocapsales ; Hyellaceae	Myxosarcina
52607	Pleurocapsales ; Xenococcaceae	Xenococcus
582491	Oscillatoriophycideae ; Chroococcales ; Aphanothecaceae	Rubidibacter
688241	Oscillatoriophycideae ; Chroococcales ; Entophysalidaceae	Chlorogloea
693222	Oscillatoriophycideae ; Chroococcales ; Chroococcaceae	Gloeocapsopsis
945733	Pleurocapsales ; Hyellaceae	Hyella

15.3 Annexe A : download_make_tree_Cyanobacteriota_NCBI_taxo.py

```
#ete3 is a package for manipulating trees and provides utilities to work with the NCBI
#taxonomy tree.
from ete3 import NCBITaxa, Tree

#initialize the NCBITaxa instance to use it after
ncbi = NCBITaxa()

#retrieve the descendant tree for the group 'Cyanobacteriota/Melainabacteria group'
#collapses_subspecies=true: simplified tree structure because the subspecies are not showed,
#don't need it
#return_tree=True: returns the data as an ete3.Tree object for easy manipulation
tree = ncbi.get_descendant_taxa('Cyanobacteriota/Melainabacteria group', collapse_subspecies=
    True, return_tree=True)

def filter_and_rename_tree(tree, ncbi):
    nodes_to_remove = []
    for node in tree.traverse():
```

```
        #get the scientific name from the ID
        name_dict = ncbi.get_taxid_translator([node.name])
        node_name = name_dict.get(int(node.name), "")

        #filter out undesired nodes
        if node.rank in ['species', 'genus'] or 'environmental sample' in node_name or '
            unclassified' in node_name:
            nodes_to_remove.append(node)
        else:
            node.name = node_name  # replace ID with scientific name

    for node in nodes_to_remove:
        node.detach()

#filter and rename the nodes in the tree
filter_and_rename_tree(tree, ncbi)

#convert the tree to Newick format
newick_str = tree.write(format=1)

with open('filtered_tree_with_names.newick', 'w') as f:
    f.write(newick_str)
print(newick_str)
```

15.4 Annexe B : parse_nested_structure.pl

```
#!/usr/bin/env perl
use Modern::Perl '2011';
use Text::CSV;

my $infile = shift;

#initialization of the CSV object that handles binary data properly (binary => 1) and enables
#automatic diagnostics (handle error)
my $csv = Text::CSV->new({ binary => 1, auto_diag => 1 });

#opening the CSV object with utf8 encoding
```

```
open my $in_fh, '<:encoding(utf8)', $infile ;

#recuperation of the header
my $header = $csv->getline($in_fh);


#INITIALIZATION of the data structures, attributes and accessions
my %accession_attributes;
my %attributes;

#loop to go one every line and retrieve the last 2 columns
while (my $row = $csv->getline($in_fh)) {
    my ($acc, undef, undef, undef, undef, undef, undef, $key, $value) = @$row;
    next unless defined $acc && defined $key;  #ignore accession-free or without key-value
    # pair lines.
    $accession_attributes{$acc}{$key} = $value;
    $attributes{$key} = 1;  #register every uniq keys
}
close $in_fh;

#writing the output file
my $outfile = defined($infile) ? $infile =~ s/\.csv$/_summary.tsv/r : "output_summary.tsv";
open my $out_fh, '>', $outfile;

print $out_fh "acc", join("|", sort keys %attributes), "\n";
for my $acc (sort keys %accession_attributes) {
    print $out_fh "$acc";
    for my $key (sort keys %attributes) {
        my $value = $accession_attributes{$acc}{$key};
        print $out_fh "|", defined $value ? $value : '';
    }
    print $out_fh "\n";
}
close $out_fh;
print "Summary table has been written to $outfile\n";
```

15.5 Annexe C : EDirect_metadata_retrieval.pl

```
#!/usr/bin/env perl
use Modern::Perl '2011';
use autodie;
use Smart::Comments;

if (@ARGV != 2) {
    die "Usage: $0 <input_file> <output_file>\n";
}

my $input_file  = shift ;
my $output_file = shift ;
open my $input_fh, '<', $input_file ;
open my $output_fh, '>', $output_file;

#header
say $output_fh "acc|title|study_abstract|design_description|study_title";

#retrieve the title, study and design metadata
while (my $accession = <$input_fh>) {
    chomp $accession;

    my $title              = qx(efetch -db sra -id $accession -format xml | xtract -pattern
        EXPERIMENT_PACKAGE_SET -element TITLE);
    my $study_abstract     = qx(efetch -db sra -id $accession -format xml | xtract -pattern
        EXPERIMENT_PACKAGE_SET -element STUDY_ABSTRACT);
    my $design_description = qx(efetch -db sra -id $accession -format xml | xtract -pattern
        EXPERIMENT_PACKAGE_SET -element DESIGN_DESCRIPTION);
    my $study_title        = qx(efetch -db sra -id $accession -format xml | xtract -pattern
        EXPERIMENT_PACKAGE_SET -element STUDY_TITLE);

    chomp($title, $study_abstract, $design_description, $study_title);

    my $result = join "|", ($title, $study_abstract, $design_description, $study_title);
    say $output_fh "$accession|$result";
}
close $input_fh;
```

```
close $output_fh;

say "Finished.";
```

15.6 Annexe D : fuse_columns.pl

```
#!/usr/bin/env perl
use Modern::Perl '2011';
use autodie;
use Smart::Comments;

unless (@ARGV > 3) {
    die <<"EOT";
Usage: $0 <input_file> <output_file> <column_1> [<column_2>]...
<input_file>: Path to the input CSV file containing the data.
<output_file>: Path to the output CSV file to be created.
<column>: A space-separated list of column names.

Example: ./merge_columns.pl metadata.csv merged_environment
    broad_scale_environmental_context_sam biome_sam environment__biome__sam env_broad_scale_sam
     hix__proportion__sam isolation_source_sam
EOT
}

my ($input_file, $output_file, @columns) = @ARGV;
open my $input_fh, '<', $input_file;
open my $output_fh, '>', $output_file;

#obtain the name of the merged_columns
(my $output_file_header= $output_file) =~ s/\..*$//;
print $output_fh "$output_file_header\n";

#reading of the fh to obtain indexes of columns
my $header = <$input_fh>;
chomp $header;
my @header_columns = split /\|/, $header;
my %column_indices;
```

```
for my $i (0..$#header_columns) {
    $column_indices{$header_columns[$i]} = $i;
}

foreach my $col (@columns) {
    die "the col '$col' does not exist.\n" unless exists $column_indices{$col};
}

while (my $line = <$input_fh>) {
    chomp $line;
    my @fields = split /\|/, $line;
#.   my @fields = split /\t/, $line;
    my $fusion_value = '';

    foreach my $col (@columns) {
        if (defined $fields[$column_indices{$col}] && $fields[$column_indices{$col}] ne '') {
            $fusion_value = $fields[$column_indices{$col}];
            last;
        }
    }
    say $output_fh $fusion_value;
}
close $input_fh;
close $output_fh;
print "Columns correctly written in $output_file.\n";
```

15.7 Annexe E : predined_tags.pl

```
#!/usr/bin/env perl
use Smart::Comments '####';
use Modern::Perl '2011';
use autodie;

my $file_in = shift;
my $file_out = shift;
open(my $fh_in, '<', $file_in);
open(my $fh_out, '>', $file_out);
```

```
#header
my $header = <$fh_in>;
print $fh_out $header;

#values to be removed
my @remove_values = qw(Unclassified missing miss uncalculated not applicable not collected
    unspecified NA);
my %remove_hash = map { $_ => 1 } @remove_values;

#definition of the column that need to be modify
while (my $ligne = <$fh_in>) {
    chomp $ligne;
    my @colonnes = split /\|/, $ligne;
    my $altitude = $colonnes[1];
    my $depth = $colonnes[2];
    my $date = $colonnes[0];
    my $temperature = $colonnes[3];

    #feet to cm conversion
    if ($altitude =~ /([\d.]+)\s*ft/i) {
        $altitude = $1 * 0.3048;
    } elsif ($altitude =~ /(\d+\.?\d*)\s*cm/) {
        $altitude = $1 * 0.01;
    }
    if ($depth =~ /([\d.]+)\s*ft/i) {
        $depth = $1 * 0.3048;
    } elsif ($depth =~ /(\d+\.?\d*)\s*cm/) {
        $depth = $1 * 0.01;
    }
#COULD USE hashe structures instead
#Value's range
    # altitude modification
    if ($altitude ne '') {
        if ($altitude >= 0 && $altitude <= 0.1) {
            $altitude = "0-0.1m high";
        } elsif ($altitude >= 0.1 && $altitude <= 1) {
            $altitude = "0.1-1m high";
```

```
    } elsif ($altitude >= 1.1 && $altitude <= 10) {
        $altitude = "1-10m high";
    } elsif ($altitude >= 10.1 && $altitude <= 50) {
        $altitude = "10-50m high";
    } elsif ($altitude >= 50.1 && $altitude <= 200) {
        $altitude = "50-200m high";
    } elsif ($altitude >= 200.1 && $altitude <= 1000) {
        $altitude = "200-1000m high";
    } elsif ($altitude >= 1000.1 && $altitude <= 2000) {
        $altitude = "1000-2000m high";
    } elsif ($altitude >= 2000.1 && $altitude <= 4000) {
        $altitude = "2000-4000m high";
    } elsif ($altitude >= 4000.1 && $altitude <= 100000) {
        $altitude = "4000m high and higher";
    } else {
        $altitude = "";
    }
} else {
    $altitude = "";
}

# depth modification
if ($depth ne '') {
    if ($depth =~ /-/) {
        my ($min, $max) = split /-/, $depth;
        my $average = ($min + $max) / 2;
        $depth = $average;
    }

    if ($depth >= -1 && $depth < 1) {
        $depth = "0-1m deep";
    } elsif ($depth >= 1.1 && $depth <= 5) {
        $depth = "1-5m deep";
    } elsif ($depth >= 5.1 && $depth <= 10) {
        $depth = "5-10m deep";
    } elsif ($depth >= 10.1 && $depth <= 50) {
        $depth = "10-50m deep";
    } elsif ($depth >= 50.1 && $depth <= 20000) {
```

```
        $depth = "deeper than 50m";
    } else {
        $depth = "";
    }
} else {
    $depth = ""; # if empty
}

# date modification
if ($date =~ /(\d{2})\/(\d{2})\/(\d{4})/) {
    $date = $3; # keep only the year
}

# temperature modification
if ($temperature ne '') {
    if ($temperature < 0 && $temperature > -5) {
        $temperature = "(-5) - 0°C";
    } elsif ($temperature >= 0 && $temperature <= 1) {
        $temperature = "0-1°C";
    } elsif ($temperature > 1 && $temperature <= 5) {
        $temperature = "1-5°C";
    } elsif ($temperature > 5 && $temperature <= 10) {
        $temperature = "5-10°C";
    } elsif ($temperature > 10 && $temperature <= 15) {
        $temperature = "10-15°C";
    } elsif ($temperature > 15 && $temperature <= 20) {
        $temperature = "15-20°C";
    } elsif ($temperature > 20 && $temperature <= 25) {
        $temperature = "20-25°C";
    } elsif ($temperature > 25 && $temperature <= 30) {
        $temperature = "25-30°C";
    } elsif ($temperature > 30 && $temperature <= 40) {
        $temperature = "30-40°C";
    }
}

#remove unwanted values
@colonnes = map { $remove_hash{$_} ? '' : $_ } @colonnes;
```

```
    $colonnes[1] = $altitude;
    $colonnes[2] = $depth;
    $colonnes[0] = $date;
    $colonnes[3] = $temperature;

    print $fh_out join("|", @colonnes), "\n";
}
close($fh_in);
close($fh_out);
```

15.8 Annexe F : elbow_silhouette_plot.py

```
from sentence_transformers import SentenceTransformer
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score
from gap_statistic import OptimalK
import matplotlib.pyplot as plt
import os
os.environ["TOKENIZERS_PARALLELISM"] = "false"
from sklearn.decomposition import PCA

#load and read vocabulary
vocabulary_file = "../../vocabulary/vocabulary.txt"
with open(vocabulary_file, "r", encoding="utf-8") as file:
    vocabulary = [line.strip() for line in file.readlines()]


#encode and reduction of dim
model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2')
sentence_embeddings = model.encode(vocabulary)
pca = PCA(n_components=100)
embeddings_30d=pca.fit_transform(sentence_embeddings)


range_n_clusters = range(100, 3800, 100)
```

```
#elbow and silhouette initialization
inertias = []
silhouette_scores = []

for n_clusters in range_n_clusters:
    print(f"Calcul pour les {n_clusters} clusters")
    kmeans = KMeans(n_clusters=n_clusters, random_state=0)
    cluster_labels = kmeans.fit_predict(embeddings_30d)
    #keep the inertia values and store them
    inertia = kmeans.inertia_
    inertias.append(inertia)
    #silhouette score computing
    silhouette_avg = silhouette_score(embeddings_30d, cluster_labels, metric='cosine')
    silhouette_scores.append(silhouette_avg)
    print(f"Inertie: {inertia}, Silhouette: {silhouette_avg}")

#gap statistic
optimalK = OptimalK(parallel_backend='joblib')
n_clusters_optimal = optimalK(embeddings_30d, cluster_array=range_n_clusters)
print(f"Le nombre optimal de clusters selon la statistique de gap est : {n_clusters_optimal}")



fig = plt.figure(figsize=(15, 7))
gs = fig.add_gridspec(1, 2, width_ratios=[1, 1])
plt.rcParams.update({'font.size': 16})
plt.rcParams.update({'axes.titlesize': 20})
plt.rcParams.update({'axes.labelsize': 18})



#elbow plot
ax1 = fig.add_subplot(gs[0, 0])
ax1.plot(range_n_clusters, inertias, marker='o', color='b')
ax1.set_xlabel('Nombre de clusters (k)', fontsize=14)
ax1.set_ylabel('Inertie (SSE)', fontsize=14)
ax1.set_title('Graphe en coude (Elbow plot) pour K-means')
ax1.grid(True)
```

```
#silhouette
ax2 = fig.add_subplot(gs[0, 1])
ax2.plot(range_n_clusters, silhouette_scores, marker='o', linestyle='-', color='b')
ax2.set_title('Score de silhouette vs. Nombre de clusters')
ax2.set_xlabel('Nombre de clusters', fontsize=14)
ax2.set_ylabel('Score de silhouette', fontsize=14)
ax2.grid(True)

plt.tight_layout()
plt.savefig('combined_plot_pca_2_100.png', bbox_inches='tight', dpi=1000)
plt.close()
```

15.9 Annexe G : gap_standard_error.R

```
library(cluster)
library(factoextra)
library(ggplot2)

#read the embedding's values
df <- read.csv("embeddings_100d.csv")
set.seed(42)
#compute the SE
gap_stat<-clusGap(df, FUN=kmeans, nstart=2,K.max =3800, B=2 )
print(gap_stat, method = "firstmax")
saveRDS(gap_stat, file = "gap_stat.rds")

pdf("gap_stat_plot_100.pdf", width = 8, height = 6)
fviz_gap_stat(gap_stat)
dev.off()
```

15.10 Annexe H : cluster_dendro_best.py

```
from sentence_transformers import SentenceTransformer
from scipy.cluster.hierarchy import linkage, fcluster, dendrogram
import matplotlib.pyplot as plt
import numpy as np
from scipy.spatial.distance import pdist
from sklearn.cluster import KMeans
import os
import time

#timer
start_time = time.time()

#directory setup
def create_directories(base_dir, k_values, methods):
    for k in k_values:
        for method in methods:
            os.makedirs(os.path.join(base_dir, f'{k}/{method}'), exist_ok=True)
        os.makedirs(os.path.join(base_dir, f'{k}/kmeans'), exist_ok=True)
    os.makedirs('somme', exist_ok=True)


#data loading and embeddings
vocabulary_file = "../../../vocabulary/vocabulary.txt"
with open(vocabulary_file, "r", encoding="utf-8") as file:
    vocabulary = [line.strip() for line in file.readlines()]
model = SentenceTransformer("sentence-transformers/all-MiniLM-L6-v2")

sentence_embeddings = model.encode(vocabulary)
condensed_distances = pdist(sentence_embeddings, metric='cosine')

#Parameters
methods = ['single', 'centroid', 'complete', 'average', 'ward']
k_values = [500, 1000, 1500, 2000, 2500, 3000, 3500]


create_directories('clusters', k_values, methods) #creation of directories
```

```
#clustering and saving results
for k in k_values:
    for method in methods:
        print(f"Processing {method} method with k={k}")
        linkage_matrix = linkage(condensed_distances, method=method)
        cluster_assignments = fcluster(linkage_matrix, k, criterion='maxclust')

        cluster_contents = {i: [] for i in range(1, k + 1)}
        for i, term in enumerate(vocabulary):
            cluster = cluster_assignments[i]
            cluster_contents[cluster].append(term)

        output_dir = os.path.join('clusters', f'{k}/{method}')
        output_file = os.path.join(output_dir, 'cluster_output.txt')
        with open(output_file, 'w') as f:
            for cluster, terms in cluster_contents.items():
                f.write(','.join(terms) + '\n')

        #sorting the 100 largest clusters
        comma_counts = {line.strip(): line.count(',') for line in open(output_file, 'r')}
        sorted_lines = sorted(comma_counts.items(), key=lambda x: x[1], reverse=True)[:100]
        output_file_100_best = os.path.join(output_dir, 'cluster_100_best.txt')
        with open(output_file_100_best, 'w') as file:
            for line, count in sorted_lines:
                file.write(f"{line}\n")

        terms_100_best = [line.split(',') for line, _ in sorted_lines]
        terms_100_best_flat = [term for sublist in terms_100_best for term in sublist]
        indices_100_best = [vocabulary.index(term) for term in terms_100_best_flat]
        sentence_embeddings_100_best = np.array([sentence_embeddings[i] for i in
            indices_100_best])
        condensed_distances_100_best = pdist(sentence_embeddings_100_best, metric='cosine')

        #plot dendrogram for hierarchical methods
        if method != 'kmeans':
            linkage_matrix_100_best = linkage(condensed_distances_100_best, method=method)
            plt.figure(figsize=(15, 8))
```

```
            dendrogram(linkage_matrix_100_best, labels=terms_100_best_flat)
            plt.xticks(rotation=85, ha='right', fontsize=8)
            plt.title(f'Dendrogram of the 100 biggest clusters - method {method.capitalize()}
                - clustering with k={k}')
            plt.savefig(os.path.join(output_dir, 'dendrogram_100_best.png'), bbox_inches='
                tight', dpi=1500)
            plt.close()


#K-means clustering
for k in k_values:
    print(f"K-means clustering with k={k}")
    kmeans = KMeans(n_clusters=k, random_state=0).fit(sentence_embeddings)
    cluster_assignments = kmeans.labels_

    cluster_contents = {i: [] for i in range(k)}
    for i, term in enumerate(vocabulary):
        cluster = cluster_assignments[i]
        cluster_contents[cluster].append(term)

    output_dir = os.path.join('clusters', f'{k}/kmeans')
    output_file = os.path.join(output_dir, 'cluster_output.txt')
    with open(output_file, 'w') as f:
        for cluster, terms in cluster_contents.items():
            f.write(','.join(terms) + '\n')

    #sorting the 100 largest clusters
    comma_counts = {line.strip(): line.count(',') for line in open(output_file, 'r')}
    sorted_lines = sorted(comma_counts.items(), key=lambda x: x[1], reverse=True)[:100]
    output_file_100_best = os.path.join(output_dir, 'cluster_100_best.txt')
    with open(output_file_100_best, 'w') as file:
        for line, count in sorted_lines:
            file.write(f"{line}\n")

    terms_100_best = [line.split(',') for line, _ in sorted_lines]
    terms_100_best_flat = [term for sublist in terms_100_best for term in sublist]
    terms_100_best_flat = [term for term in terms_100_best_flat if term]
```

```
    indices_100_best = [vocabulary.index(term) for term in terms_100_best_flat if term in
        vocabulary]
    sentence_embeddings_100_best = np.array([sentence_embeddings[i] for i in indices_100_best
        ])
    condensed_distances_100_best = pdist(sentence_embeddings_100_best, metric='cosine')

    plt.figure(figsize=(15, 8))
    dendrogram(linkage(condensed_distances_100_best, method='ward'), labels=
        terms_100_best_flat)
    plt.xticks(rotation=85, ha='right', fontsize=0.2)
    plt.title(f'Dendrogram of the 100 biggest clusters - method K-means - clusterization with
        k={k}')
    plt.savefig(os.path.join(output_dir, 'dendrogram_100_best_kmeans.png'), bbox_inches='tight
        ', dpi=1500)
    plt.close()

#compute similarity sums
def compute_similarity_sums(file_path):
    with open(file_path, 'r') as file:
        lines = file.readlines()
    terms = [line.strip().split(',') for line in lines]
    sum_similarity = []
    for line_terms in terms:
        term_embeddings = model.encode(line_terms, convert_to_tensor=True)
        term_similarities = []
        for i in range(len(line_terms)):
            similarity_sum = 0
            for j in range(len(line_terms)):
                if i != j:
                    similarity_sum += np.inner(term_embeddings[i].cpu().detach().numpy(),
                        term_embeddings[j].cpu().detach().numpy()).item()
            term_similarities.append(similarity_sum)
        num_terms = len(line_terms)
        normalized_similarities = [score / num_terms for score in term_similarities]
        term_sums = {term: similarity_sum for term, similarity_sum in zip(line_terms,
            normalized_similarities)}
        sorted_terms = sorted(term_sums.items(), key=lambda x: x[1], reverse=True)
        sum_similarity.append({term: score for term, score in sorted_terms})
```

```
    return sum_similarity


for k in k_values:
    method = 'kmeans'
    file_path = os.path.join('clusters', f'{k}/{method}/cluster_100_best.txt')
    sum_similarity = compute_similarity_sums(file_path)
    for i, rep in enumerate(sum_similarity):
        output_file = os.path.join('somme', f'somme_{k}_{method}_{i}.txt')
        with open(output_file, 'w') as file:
            for term, score in rep.items():
                file.write(f"{term}:{score}\n")

end_time = time.time()
execution_time = end_time - start_time
print(f"Execution time: {execution_time} seconds")
```

15.11 Annexe I : Word_cloud.R

```
library(wordcloud)
library(RColorBrewer) #more colors

#where "sums" directory is located. create if needed the new directory
base_directory <- "~/tfe_r/somme_verif"

if (!dir.exists(file.path(base_directory, "pictures"))) {
  dir.create(file.path(base_directory, "pictures"))
}

#the range of clusterisation processes
for (size in c(500, 1000, 1500, 2000, 2500, 3000, 3500)) {
  #listing of the files located on the current size loop value
  files <- list.files(base_directory, pattern = paste0("^somme_", size, "_.*\\.txt$"), full.
      names = TRUE)

  if (length(files) == 0) {
    message(paste("no file found for this size", size))
```

```
    next
  }

  #extraction of file's number and sort numeric
  file_numbers <- as.numeric(sub(".*_(\\d+)\\.txt$", "\\1", basename(files)))
  files <- files[order(file_numbers)]
  print(files)

  #picture with 16 wordcloud on them for speed
  for (start_index in seq(1, length(files), by = 16)) {
    png_name <- paste0(start_index, "-", min(start_index + 15, length(files)), "
        _clusters_from_", size, ".png")
    png(file.path(base_directory, "pictures", png_name), width = 1800, height = 1000)

    par(mfrow = c(4, 4))
    #wordcloud creation
    for (i in start_index:(min(start_index + 15, length(files)))) {
      data <- read.csv(files[i], header = FALSE, sep = ":")
      colnames(data) <- c("word", "frequence")

      #max 50 words, 30% vertical words
      wordcloud(words = data$word, freq = data$frequence, min.freq = 1,
                max.words = 50, random.order = FALSE, rot.per = 0.3,
                colors = brewer.pal(8, "Dark2"), scale = c(1.2, 1))
    }
    dev.off()
  }
}
```

15.12 Annexe J : choose_best_representive.py

```
#numpy will be used to compute scalar product between embedding of each term's pair (np.inner)
from sentence_transformers import SentenceTransformer
import numpy as np

model = SentenceTransformer("sentence-transformers/all-MiniLM-L6-v2")
```

```
#read the file and splitting into terms
with open('./voc_clustered_ward_2000.txt', 'r') as file:

    lines = file.readlines()
terms = [line.strip().split(',') for line in lines]
print (terms)
best_representatives = []

for line_terms in terms:
    #no computation when there is only one term
    if len(line_terms) == 1:
        best_representatives.append(line_terms[0])
        continue

    #when there is multiple terms
    #embedding of the terms, convertion of tensor pytorch (more efficient)
    term_embeddings = model.encode(line_terms, convert_to_tensor=True)
    term_similarities = []
    #loop for each term in the line
    for i in range(len(line_terms)):
        similarity = []
        #j loop  (i != j)
        for j in range(len(line_terms)):
            if i != j:
                #use of the scalar product to compute the similarity of embedding[i] and [j]
                #cpu to transfer tensor fro GPU on the CPU
                #detach  the gradients ensures that the tensor is disconnected from its
                    computation history, preventing gradients from being tracked for this
                    tensor.
                #This is useful when we only want to work with the tensor's values without
                    affecting any future gradient computations.
                #numpyto convert the tensor to a NumPy array allows us to work with the tensor
                    's data using the NumPy library
                #np.inner: compute scalar product
                #itemextract a single scalar value from a tensor containing a single element
                similarity.append(np.inner(term_embeddings[i].cpu().detach().numpy(),
                    term_embeddings[j].cpu().detach().numpy()).item())
        #similarities of the current term with every other terms
```

```
        term_similarities.append(similarity)

    #retrieve the best
    term_sums = {term: sum(similarities) for term, similarities in zip(line_terms,
        term_similarities)}
    print(term_sums,"\n")
    best_representative = max(term_sums, key=term_sums.get)
    best_representatives.append(best_representative)

#output
output_file = './best_representatives_500.txt'
with open(output_file, 'w') as file:
    for rep in best_representatives:
        file.write(rep + '\n')
print(f"best representative terms have been written in '{output_file}'.")
```

15.13 Annexe K : identify_non_environmental_tag.py

```
from transformers import pipeline

def read_file(file_path):
    with open(file_path, "r", encoding="utf-8") as file:
        return file.readlines()

sequence_to_classify_file = "best_representatives_ward.txt"
output_file = "env_or_not_human.txt"

sequences_to_classify = read_file(sequence_to_classify_file)
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli", device=0)


#launch the script several times, with one of these lines un-commented
candidate_labels = ["nature", "human"]
#candidate_labels =["nature", "astronomy"]
#candidate_labels =["nature", "ice"]

with open(output_file, "w", encoding="utf-8") as file:
```

```
    for sequence_to_classify in sequences_to_classify:
        result = classifier(sequence_to_classify, candidate_labels, multi_label=False)
        best_score = 0.0
        best_label = 'N/A'
        for label, score in zip(result['labels'], result['scores']):
            if score > best_score:
                best_label = label
                best_score = score


#also change the tags here
        if best_label == "nature":
            file.write(f"{sequence_to_classify.strip()} (env {best_score})\n")
        elif best_label == "human":
            file.write(f"{sequence_to_classify.strip()} (non env {best_score})\n")
print(f"Terms with best tags and scores written in: {output_file}")
```

15.14 Annexe L : generate_dynamic_tags_score.py

```
from transformers import pipeline
import time
import argparse
start_time = time.time()


def read_file(file_path):
    with open(file_path, "r", encoding="utf-8") as file:
        return file.readlines()

#setup command-line argument parsing
parser = argparse.ArgumentParser(description="Zero-shot classification script")
parser.add_argument("--score", type=float, default=0.85, help="Minimum score threshold for
    labels")
parser.add_argument("--output_file", type=str, default="tags_output.txt", help="Output file
    name")
args = parser.parse_args()
```

```
#read input files
sequence_to_classify_file = "texts.txt"
candidate_labels_file = "vocabulary_clustered.txt"
output_file = args.output_file
score_threshold = args.score

sequences_to_classify = read_file(sequence_to_classify_file)
candidate_labels = [label.strip() for label in read_file(candidate_labels_file)]

#initialize classifier
classifier = pipeline("zero-shot-classification", model="facebook/bart-large-mnli", device=0)
#classifier = pipeline("zero-shot-classification", model="cross-encoder/nli-roberta-base",
    device=0)
selected_labels_all = []

#classify sequences and select labels
for sequence_to_classify in sequences_to_classify:
    result = classifier(sequence_to_classify, candidate_labels, multi_label=True)
    selected_labels = [label for label, score in zip(result['labels'], result['scores']) if
        len(label) > 2 and score >= score_threshold]
    selected_labels_all.append(selected_labels)

with open(output_file, "w", encoding="utf-8") as file:
    for labels in selected_labels_all:
        file.write(", ".join(labels) + "\n")

print(f"Tags written in: {output_file}")
end_time = time.time()
elapsed_time = end_time - start_time
print(f"Temps écoulé: {elapsed_time} secondes")
```

15.15 Annexe M : download_assembly_gtdb.sh

```
#!/bin/bash
#SBATCH --time=30-01:00:00 # jours-hh:mm:ss
#SBATCH --mail-user=marie.harmel@student.uliege.be
#SBATCH --mail-type=END
```

```
#SBATCH --ntasks=1
#SBATCH --cpus-per-task=20
#SBATCH --mem-per-cpu=30000 # mégaoctets
#SBATCH --partition=bio

export OMP_NUM_THREADS=20
export MKL_NUM_THREADS=20
module --ignore-cache load Nextflow/21.08.0

#path where the processes will begin
cd /data/GENERA/MARIES/

#download of SRA accessions present in the SRR_a list
for f in $(cat SRR_a); do
    echo -n "$f";
    mkdir -p "$f"

    singularity exec --bind /data/GENERA/MARIES/:/temp /scratch/ulg/GENERA/SRA.sif fasterq-
        dump $f \\
         --outdir /temp/ --temp /temp/LOG/
#move fastq fle in the correct directory
    mv "$f"*fastq "$f"/

#Genome assembly
    cd "$f"/
    cp -f ../Assembly.config nextflow.config
    cp ../Assembly.nf .
    nextflow run Assembly.nf --shortreadsR1=$f\_1.fastq --shortreadsR2=$f\_2.fastq --
        metagenome=yes \\
         --binner=all --cpu=20;
#when process finished, change extension, make it pretty
    mkdir -p genome_"$f"
    mv GENERA-assembly/* genome_"$f"/

    cd genome_"$f"
    mv Genome.fasta metagenome_"$f"
    /scratch/ulg/GENERA/Supplemental-scripts/change-ext.py --currentext fasta --newext fna
```

```
    rm -f *.fasta
cd ..

#GTDB classification: need to change the working directory, in $GLOBALSCRATCH
    cp -r genome_"$f"/ /scratch/ulg/GENERA/mharmel/ASSEMBLY/
    cd /scratch/ulg/GENERA/mharmel/ASSEMBLY/genome_"$f"
    cp -f /home/users/m/h/mharmel/tfe/scripts/GTDB.config nextflow.config
    sed -i "s|singularity.runOptions = '-B /scratch/ulg/GENERA/mharmel/ASSEMBLY/genome_ \\
        -B /scratch/users/m/h/mharmel/ASSEMBLY/genome_ -B /scratch/ulg/GENERA/Databases/GTDB
            '\\
             |singularity.runOptions = '-B /scratch/ulg/GENERA/mharmel/ASSEMBLY/genome_"$f" \\
             -B /scratch/users/m/h/mharmel/ASSEMBLY/genome_"$f" \\
             -B /scratch/ulg/GENERA/Databases/GTDB'|" nextflow.config
    cp -f /home/users/m/h/mharmel/tfe/scripts/GTDB.nf .
    nextflow run GTDB.nf --genome=/scratch/ulg/GENERA/mharmel/ASSEMBLY/genome_"$f" --cpu=20
    rm -Rf  nextflow.config GTDB.nf work/ .nextflow* GENERA_GTDB/GENERA-gtdb.log
    cd /data/GENERA/MARIES/
    rm -fR "$f"
done
echo DONE
```

15.16 Annexe N : premier arbre phylogénétique

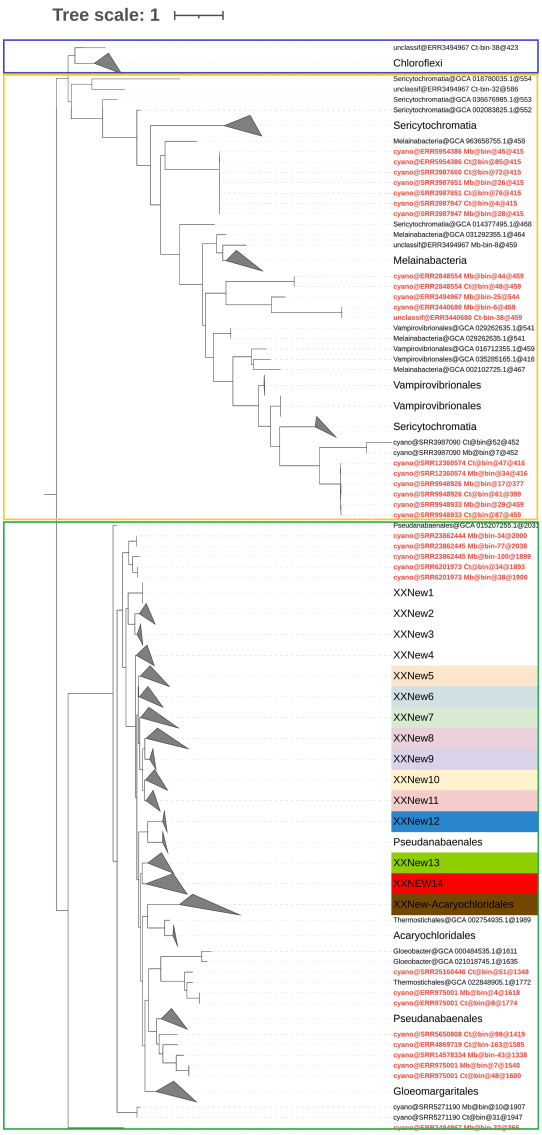

FIGURE 35 : **Phylogénie des 741 bins d'intérêt et mise en évidence des trois clades taxonomiques. Bleu** : Chloroflexia, un groupe légèrement plus ... que les cyanobactéries. **Orange** : Mélainabactéria, Vampirovibrionales et Sericytochromatia. **Vert** : Cyanobactéries

15.17 Annexe P : arbre ORPER

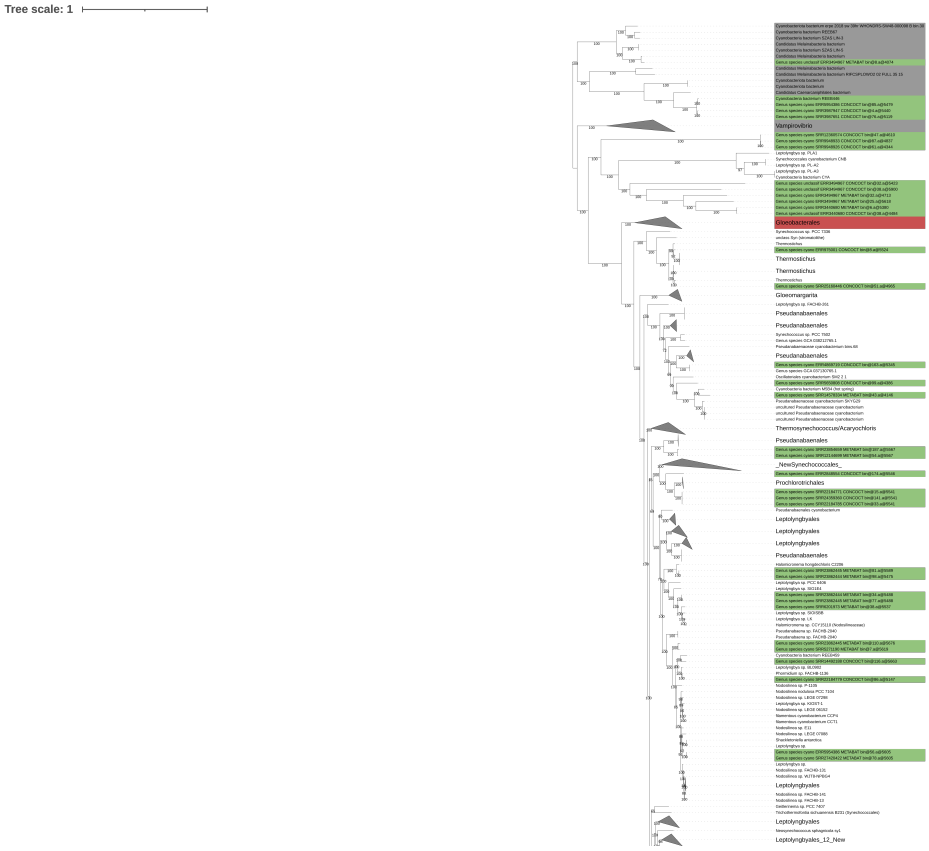

FIGURE 36 : Première partie de ORPER.

FIGURE 37 : Seconde partie de ORPER.

15.18 Annexe Q : librairies utilisées

librairie python :

- ete3
- gap_statistic
- sklearn
- transformers
- scipy
- sentence_transformers
- matplotlib,

R :

- cluster
- factoextra
- wordcloud
- RColorBrewer

perl :

- use Smart ::Comments '####' ;

— use Modern ::Perl '2011';
— use autodie;
— use Text ::CSV;

Références

AlexPodles. (2024, April 2). *To Cosine or Not to Cosine, That Is the Question : Understanding Similarity Metrics*. Medium. https://medium.com/@alexpodles/to-cosine-or-not-to-cosine-that-is-the-question-understanding-similarity-metrics-67ac0eb2e586

Álvarez, C., Jiménez-Ríos, L., Iniesta-Pallarés, M., Jurado-Flores, A., Molina-Heredia, F. P., Ng, C. K. Y., & Mariscal, V. (2023). Symbiosis between cyanobacteria and plants : From molecular studies to agronomic applications. *Journal of Experimental Botany*, *74*(19), 6145–6157. https://doi.org/10.1093/jxb/erad261

Arikawa, K., Ide, K., Kogawa, M., Saeki, T., Yoda, T., Endoh, T., Matsuhashi, A., Takeyama, H., & Hosokawa, M. (2021, March 12). *Recovery of high-quality assembled genomes via metagenome binning guided with single-cell amplified genomes*. https://doi.org/10.1101/2021.01.11.425816

Ashrafi, H., Hulse-Kemp, A. M., Wang, F., Yang, S. S., Guan, X., Jones, D. C., Matvienko, M., Mockaitis, K., Chen, Z. J., Stelly, D. M., & Van Deynze, A. (2015). A Long-Read Transcriptome Assembly of Cotton (Gossypium hirsutum L.) and Intraspecific Single Nucleotide Polymorphism Discovery. *The Plant Genome*, *8*(2), plantgenome2014.10.0068. https://doi.org/10.3835/plantgenome2014.10.0068

Bahdanau, D., Cho, K., & Bengio, Y. (2016, May 19). *Neural Machine Translation by Jointly Learning to Align and Translate*. http://arxiv.org/abs/1409.0473

Barrett, T., Clark, K., Gevorgyan, R., Gorelenkov, V., Gribov, E., Karsch-Mizrachi, I., Kimelman, M., Pruitt, K. D., Resenchuk, S., Tatusova, T., Yaschenko, E., & Ostell, J. (2012). BioProject and BioSample databases at NCBI : Facilitating capture and organization of metadata. *Nucleic Acids Research*, *40*, D57–D63. https://doi.org/10.1093/nar/gkr1163

Barsanti, L., Coltelli, P., Evangelista, V., Frassanito, A. M., Passarelli, V., Vesentini, N., & Gualtieri, P. (2008). Oddities and Curiosities in the Algal World. In V. Evangelista, L. Barsanti, A. M. Frassanito, V. Passarelli, & P. Gualtieri (Eds.), *Algal Toxins : Nature, Occurrence, Effect and Detection* (pp. 353–391). Springer Netherlands. https://doi.org/10.1007/978-1-4020-8480-5_17

Bellman, R. (1984). *Dynamic programming*. Princeton Univ. Pr.

Benler, S., Yutin, N., Antipov, D., Rayko, M., Shmakov, S., Gussow, A. B., Pevzner, P., & Koonin, E. V. (2021). Thousands of previously unknown phages discovered in whole-community human gut metagenomes. *Microbiome*, *9*, 78. https://doi.org/10.1186/s40168-021-01017-w

Bettina, S., Gugger, M., & Donoghue, P. (2015). Cyanobacteria and the Great Oxidation Event : Evidence from genes and fossils. *Palaeontology*, *58*. https://doi.org/10.1111/pala.12178

BioProject Frequently Asked Questions : (n.d.). Retrieved August 19, 2024, from https://www.ncbi.nlm.nih.gov/bioproject/docs/faq/#what-is-a-bioproject

Blurock, E. (2021, March 8). *String Similarity Metrics : Sequence Based | Baeldung on Computer Science*. https://www.baeldung.com/cs/string-similarity-sequence-based

Bowers, R. M., Kyrpides, N. C., Stepanauskas, R., Harmon-Smith, M., Doud, D., Reddy, T. B. K., Schulz, F., Jarett, J., Rivers, A. R., Eloe-Fadrosh, E. A., Tringe, S. G., Ivanova, N. N., Copeland, A., Clum, A., Becraft, E. D., Malmstrom, R. R., Birren, B., Podar, M., Bork, P., … Woyke, T. (2017). Minimum information about a single amplified genome (MISAG) and a metagenome-assembled genome (MIMAG) of bacteria and archaea. *Nature Biotechnology*, *35*(8), 725–731. https://doi.org/10.1038/nbt.3893

Brinkmann, H., Göker, M., Koblížek, M., Wagner-Döbler, I., & Petersen, J. (2018). Horizontal operon transfer, plasmids, and the evolution of photosynthesis in *Rhodobacteraceae*. *The ISME Journal*, *12*(8), 1994–2010. https://doi.org/10.1038/s41396-018-0150-9

Broder, A. Z. (2000). Identifying and Filtering Near-Duplicate Documents. In R. Giancarlo & D. Sankoff (Eds.), *Combinatorial Pattern Matching* (pp. 1–10). Springer. https://doi.org/10.1007/3-540-45123-4_1

Brown, P. F., Della Pietra, V. J., deSouza, P. V., Lai, J. C., & Mercer, R. L. (1992). Class-Based *n*-gram Models of Natural Language. *Computational Linguistics*, *18*(4), 467–480. https://aclanthology.org/J92-4003

Bruin, J. (2006). *Principal Components (PCA) and Exploratory Factor Analysis (EFA) with SPSS*. https://stats.oarc.ucla.edu/spss/seminars/efa-spss/#top

Bryant, D. A., Costas, A. M. G., Maresca, J. A., Chew, A. G. M., Klatt, C. G., Bateson, M. M., Tallon, L. J., Hostetler, J., Nelson, W. C., Heidelberg, J. F., & Ward, D. M. (2007). Candidatus Chloracidobacterium thermophilum : An Aerobic Phototrophic Acidobacterium. *Science*, *317*(5837), 523–526. https://doi.org/10.1126/science.1143236

Bryant, D. A., & Frigaard, N.-U. (2006). Prokaryotic photosynthesis and phototrophy illuminated. *Trends in Microbiology*, *14*(11), 488–496. https://doi.org/10.1016/j.tim.2006.09.001

Bubeck, S., Chandrasekaran, V., Eldan, R., Gehrke, J., Horvitz, E., Kamar, E., Lee, P., Lee, Y. T., Li, Y., Lundberg, S., Nori, H., Palangi, H., Ribeiro, M. T., & Zhang, Y. (2023, April 13). *Sparks of Artificial General Intelligence : Early experiments with GPT-4*. https://doi.org/10.48550/arXiv.2303.12712

Burja, A. M., Abou-Mansour, E., Banaigs, B., Payri, C., Burgess, J. G., & Wright, P. C. (2002). Culture of the marine cyanobacterium, Lyngbya majuscula (Oscillatoriaceae), for bioprocess intensified production of cyclic and linear lipopeptides. *Journal of Microbiological Methods*, *48*(2-3), 207–219. https://doi.org/10.1016/s0167-7012(01)00324-4

Buttigieg, P. L., Morrison, N., Smith, B., Mungall, C. J., Lewis, S. E., & the ENVO Consortium. (2013). The environment ontology : Contextualising biological and biomedical entities. *Journal of Biomedical Semantics*, *4*(1), 43. https://doi.org/10.1186/2041-1480-4-43

Capone, D. G., Burns, J. A., Montoya, J. P., Subramaniam, A., Mahaffey, C., Gunderson, T., Michaels, A. F., & Carpenter, E. J. (2005). Nitrogen fixation by Trichodesmium spp. : An important source of new nitrogen to the tropical and subtropical North Atlantic Ocean. *Global Biogeochemical Cycles*, *19*(2). https://doi.org/10.1029/2004GB002331

Cardona, T. (2015). A fresh look at the evolution and diversification of photochemical reaction centers. *Photosynthesis Research*, *126*(1), 111–134. https://doi.org/10.1007/s11120-014-0065-x

Cardona, T., Murray, J. W., & Rutherford, A. W. (2015). Origin and Evolution of Water Oxidation before the Last Common Ancestor of the Cyanobacteria. *Molecular Biology and Evolution*, *32*(5), 1310–1328. https://doi.org/10.1093/molbev/msv024

Chaudhary, A. (2020, May 30). *Zero Shot Learning for Text Classification*. Amit Chaudhary. https://amitness.com/posts/zero-shot-text-classification

Chaumeil, P.-A., Mussig, A. J., & Parks, D. H. (2024). *Summary.tsv — GTDB-Tk 2.4.0 documentation*. https://ecogenomics.github.io/GTDBTk/files/summary.tsv.html

Chen, C., Zhou, Y., Fu, H., Xiong, X., Fang, S., Jiang, H., Wu, J., Yang, H., Gao, J., & Huang, L. (2021). Expanded catalog of microbial genes and metagenome-assembled genomes from the pig gut microbiome. *Nature Communications*, *12*(1), 1106. https://doi.org/10.1038/s41467-021-21295-0

Cheong, X. C. (2010). *The Origin of Plastids | Learn Science at Scitable*. https://www.nature.com/scitable/topicpage/the-origin-of-plastids-14125758/

Chikhi, R., Raffestin, B., Korobeynikov, A., Edgar, R., & Babaian, A. (2024, July 31). *Logan : Planetary-Scale Genome Assembly Surveys Life's Diversity*. https://doi.org/10.1101/2024.07.30.605881

Cho, K., van Merrienboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2014, September 2). *Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation*. http://arxiv.org/abs/1406.1078

Ciufo, S., Kannan, S., Sharma, S., Badretdin, A., Clark, K., Turner, S., Brover, S., Schoch, C. L., Kimchi, A., & DiCuccio, M. (2018). Using average nucleotide identity to improve taxonomic assignments in prokaryotic genomes at the NCBI. *International Journal of Systematic and Evolutionary Microbiology*, *68*(7), 2386–2392. https://doi.org/10.1099/ijsem.0.002809

Cornet, L., Ahn, A.-C., Wilmotte, A., & Baurain, D. (2021). ORPER : A Workflow for Constrained SSU rRNA Phylogenies. *Genes*, *12*(11), 1741. https://doi.org/10.3390/genes12111741

Cornet, L., Bertrand, A. R., Hanikenne, M., Javaux, E. J., Wilmotte, A., & Baurain, D. (2018). Metagenomic assembly of new (sub)polar Cyanobacteria and their associated microbiome from non-axenic cultures. *Microbial Genomics*, *4*(9), e000212. https://doi.org/10.1099/mgen.0.000212

Cornet, L., Durieu, B., Baert, F., D'hooge, E., Colignon, D., Meunier, L., Lupo, V., Cleenwerck, I., Daniel, H.-M., Rigouts, L., Sirjacobs, D., Declerck, S., Vandamme, P., Wilmotte, A., Baurain, D., & Becker, P. (2023). The GEN-ERA toolbox : Unified and reproducible workflows for research in microbial genomics. *GigaScience*, *12*, giad022. https://doi.org/10.1093/gigascience/giad022

Criscuolo, A., & Gribaldo, S. (2010). BMGE (Block Mapping and Gathering with Entropy) : A new software for selection of phylogenetic informative regions from multiple sequence alignments. *BMC Evolutionary Biology*, *10*, 210. https://doi.org/10.1186/1471-2148-10-210

De Philippis, R., Margheri, M. C., Materassi, R., & Vincenzini, M. (1998). Potential of Unicellular Cyanobacteria from Saline Environments as Exopolysaccharide Producers. *Applied and Environmental Microbiology*, *64*(3), 1130–1132. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC106378/

Dessimoz, C., & Gil, M. (2010). Phylogenetic assessment of alignments reveals neglected tree signal in gaps. *Genome Biology*, *11*(4), R37. https://doi.org/10.1186/gb-2010-11-4-r37

DRA Update. (n.d.). Retrieved August 19, 2024, from https://www.ddbj.nig.ac.jp/dra/update-e.html

Editor, C. (2024, March 1). *What is Weight Initialization : LLMs Explained*. chatgptguide.ai. https://www.chatgptguide.ai/2024/03/01/what-is-weight-initialization-llms-explained/

Fangwei, Z., Damai, D., & Zhifang, S. (2024). *Language Models Know the Value of Numbers*. https://arxiv.org/ht

ml/2401.03735v3

Ferreira, L., & Hitchcock, D. B. (2009). A Comparison of Hierarchical Methods for Clustering Functional Data. *Communications in Statistics - Simulation and Computation*, *38*(9), 1925–1949. https://doi.org/10.1080/03610910903168603

Fine-Tuned "Small" LLMs (Still) Significantly Outperform Zero-Shot Generative AI Models in Text Classification. (n.d.). Retrieved August 27, 2024, from https://arxiv.org/html/2406.08660v1

Gonçalves, R. S., & Musen, M. A. (2019). The variable quality of metadata about biological samples used in biomedical experiments. *Scientific Data*, *6*, 190021. https://doi.org/10.1038/sdata.2019.21

Grisel, O., & Varoquaux, G. (2010). *Comparing different hierarchical linkage methods on toy datasets*. scikit-learn. https://scikit-learn/stable/auto_examples/cluster/plot_linkage_comparison.html

Group, G. (2024). *Use nested and repeated fields | BigQuery*. Google Cloud. https://cloud.google.com/bigquery/docs/best-practices-performance-nested

Guljamow, A., Kreische, M., Ishida, K., Liaimer, A., Altermark, B., Bähr, L., Hertweck, C., Ehwald, R., & Dittmann, E. (2017). High-Density Cultivation of Terrestrial Nostoc Strains Leads to Reprogramming of Secondary Metabolome. *Applied and Environmental Microbiology*, *83*(23), e01510–17. https://doi.org/10.1128/AEM.01510-17

Heinz, S., Rast, A., Shao, L., Gutu, A., Gügel, I. L., Heyno, E., Labs, M., Rengstl, B., Viola, S., Nowaczyk, M. M., Leister, D., & Nickelsen, J. (2016). Thylakoid Membrane Architecture in Synechocystis Depends on CurT, a Homolog of the Granal CURVATURE THYLAKOID1 Proteins. *The Plant Cell*, *28*(9), 2238–2260. https://doi.org/10.1105/tpc.16.00491

Hidalgo-Arias, A., Muñoz-Hisado, V., Valles, P., Geyer, A., Garcia-Lopez, E., & Cid, C. (2023). Adaptation of the Endolithic Biome in Antarctic Volcanic Rocks. *International Journal of Molecular Sciences*, *24*(18), 13824. https://doi.org/10.3390/ijms241813824

Huisman, J., Codd, G. A., Paerl, H. W., Ibelings, B. W., Verspagen, J. M. H., & Visser, P. M. (2018). Cyanobacterial blooms. *Nature Reviews. Microbiology*, *16*(8), 471–483. https://doi.org/10.1038/s41579-018-0040-1

Hull, R. (1997). Managing semantic heterogeneity in databases : A theoretical prospective. *Proceedings of the Sixteenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems*, 51–61. https://doi.org/10.1145/263661.263668

Husen, A. (2021). Big Data Analysis using BigQuery on Cloud Computing Platform. *Australian Journal of Engineering and Innovative Technology*, 1–9. https://doi.org/10.34104/ajeit.021.0109

Jain, C., Rodriguez-R, L. M., Phillippy, A. M., Konstantinidis, K. T., & Aluru, S. (2018). High throughput ANI analysis of 90K prokaryotic genomes reveals clear species boundaries. *Nature Communications*, *9*(1), 5114. https://doi.org/10.1038/s41467-018-07641-9

Jiri, K., Jan, K., & R, M. J. &. J. J. (2014). *Taxonomic classification of cyanoprokaryotes (cyanobacterial genera) 2014, using a polyphasic approach*. *86*(4). https://www.preslia.cz/article/103

Kans, J. (2013). *Entrez Direct : E-utilities on the Unix Command Line*. ([Updated 2024 Jul 17]) [Computer software]. National Center for Biotechnology Information (US). https://www.ncbi.nlm.nih.gov/books/NBK179288/

Kans, J. (2024). Entrez Direct : E-utilities on the Unix Command Line. In *Entrez Programming Utilities Help [Internet]*. National Center for Biotechnology Information (US). https://www.ncbi.nlm.nih.gov/books/NBK179288/

Katz. (2021). *SRA Taxonomy Analysis Tool*. https://www.ncbi.nlm.nih.gov/sra/docs/sra-taxonomy-analysis-tool/

Katz, K. S., Shutov, O., Lapoint, R., Kimelman, M., Brister, J. R., & O'Sullivan, C. (2021). STAT : A fast, scalable, MinHash-based k-mer tool to assess Sequence Read Archive next-generation sequence submissions. *Genome Biology*, *22*(1), 270. https://doi.org/10.1186/s13059-021-02490-0

Katz, K., Shutov, O., Lapoint, R., Kimelman, M., Brister, J. R., & O'Sullivan, C. (2022). The Sequence Read Archive : A decade more of explosive growth. *Nucleic Acids Research*, *50*(D1), D387–D390. https://doi.org/10.1093/nar/gkab1053

Keeling, P. J. (2010). The endosymbiotic origin, diversification and fate of plastids. *Philosophical Transactions of the Royal Society B : Biological Sciences*, *365*(1541), 729–748. https://doi.org/10.1098/rstb.2009.0103

Keshari, N., Zhao, Y., Das, S. K., Zhu, T., & Lu, X. (2022). Cyanobacterial Community Structure and Isolates From Representative Hot Springs of Yunnan Province, China Using an Integrative Approach. *Frontiers in Microbiology*, *13*, 872598. https://doi.org/10.3389/fmicb.2022.872598

Koblížek, M., Dachev, M., Bína, D., Nupur, Piwosz, K., & Kaftan, D. (2020). Utilization of light energy in phototrophic Gemmatimonadetes. *Journal of Photochemistry and Photobiology B : Biology*, *213*, 112085. https://doi.org/10.1016/j.jphotobiol.2020.112085

Komárek, J. (2018). Delimitation of the family Oscillatoriaceae (Cyanobacteria) according to the modern polyphasic approach (introductory review). *Brazilian Journal of Botany*, *41*(2), 449–456. https://doi.org/10.1007/s40415-017-0415-y

Lahr, D. J. G., Laughinghouse, H. D., Oliverio, A., Gao, F., & Katz, L. A. (2014). How discordant morphological and molecular evolution among microorganisms can revise our notions of biodiversity on earth. *BioEssays : News and Reviews in Molecular, Cellular and Developmental Biology*, *36*(10), 950–959. https://doi.org/10.1002/bies.201400056

Larsson, J., Nylander, J. A., & Bergman, B. (2011). Genome fluctuations in cyanobacteria reflect evolutionary, developmental and adaptive traits. *BMC Evolutionary Biology*, *11*(1), 187. https://doi.org/10.1186/1471-2148-11-187

Li, L., Huang, D., Hu, Y., Rudling, N. M., Canniffe, D. P., Wang, F., & Wang, Y. (2023). Globally distributed Myxococcota with photosynthesis gene clusters illuminate the origin and evolution of a potentially chimeric lifestyle. *Nature Communications*, *14*, 6450. https://doi.org/10.1038/s41467-023-42193-7

Lindgren, I. (2020). *Dealing with Highly Dimensional Data using Principal Component Analysis (PCA) | by Isabella Lindgren | Towards Data Science*. https://towardsdatascience.com/dealing-with-highly-dimensional-data-using-principal-component-analysis-pca-fea1ca817fe6

Mareš, J. (2013). *The Primitive Thylakoid-Less Cyanobacterium Gloeobacter Is a Common Rock-Dwelling Organism - PMC*. https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3688883/

McFadden, G. I. (2014). Origin and Evolution of Plastids and Photosynthesis in Eukaryotes. *Cold Spring Harbor Perspectives in Biology*, *6*(4), a016105. https://doi.org/10.1101/cshperspect.a016105

Moreno, J., Vargas, M. A., Olivares, H., Rivas, J., & Guerrero, M. G. (1998). Exopolysaccharide production by the cyanobacterium *Anabaena* sp. ATCC 33047 in batch and continuous culture. *Journal of Biotechnology*, *60*(3), 175–182. https://doi.org/10.1016/S0168-1656(98)00003-0

Morone, J., Alfeus, A., Vasconcelos, V., & Martins, R. (2019). Revealing the potential of cyanobacteria in cosmetics

and cosmeceuticals — A new bioactive approach. *Algal Research*, *41*, 101541. https://doi.org/10.1016/j.algal.2019.101541

Mungall, C. (2024). *EnvironmentOntology/envo* [Computer software]. The Environment Ontology. https://github.com/EnvironmentOntology/envo (Original work published 2015)

Murakami, N., Morimoto, T., Imamura, H., Ueda, T., Nagai, S., Sakakibara, J., & Yamada, N. (1991). Studies on glycolipids. III. Glyceroglycolipids from an axenically cultured cyanobacterium, Phormidium tenue. *Chemical & Pharmaceutical Bulletin*, *39*(9), 2277–2281. https://doi.org/10.1248/cpb.39.2277

Murphy, D., & Cardona, T. (2022). *Photosynthetic Life, Origin, Evolution, and Future from Summerfield Books*. Summerfield Books. https://www.summerfieldbooks.com/product/photosynthetic-life-origin-evolution-and-future/

Mutalipassi, M., Riccio, G., Mazzella, V., Galasso, C., Somma, E., Chiarore, A., de Pascale, D., & Zupo, V. (2021). Symbioses of Cyanobacteria in Marine Environments : Ecological Insights and Biotechnological Perspectives. *Marine Drugs*, *19*(4), 227. https://doi.org/10.3390/md19040227

Naveed, H., Khan, A. U., Qiu, S., Saqib, M., Anwar, S., Usman, M., Akhtar, N., Barnes, N., & Mian, A. (2024, April 9). *A Comprehensive Overview of Large Language Models*. http://arxiv.org/abs/2307.06435

NCBI. (2014). *EDirect Documentation - The Insiders Guide to Accessing NLM Data - National Library of Medicine* [Training Material and Manuals]. U.S. National Library of Medicine. https://www.nlm.nih.gov/dataguide/edirect/documentation.html#

Nguyen, L.-T. (2014). *IQ-TREE : A Fast and Effective Stochastic Algorithm for Estimating Maximum-Likelihood Phylogenies | Molecular Biology and Evolution | Oxford Academic*. https://academic.oup.com/mbe/article/32/1/268/2925592

Nishihara, A., Tsukatani, Y., Azai, C., & Nobu, M. K. (2024). Illuminating the coevolution of photosynthesis and Bacteria. *Proceedings of the National Academy of Sciences*, *121*(25), e2322120121. https://doi.org/10.1073/pnas.2322120121

O'Leary, N. A., Cox, E., Holmes, J. B., Anderson, W. R., Falk, R., Hem, V., Tsuchiya, M. T. N., Schuler, G. D., Zhang, X., Torcivia, J., Ketter, A., Breen, L., Cothran, J., Bajwa, H., Tinne, J., Meric, P. A., Hlavina, W., & Schneider, V. A. (2024). Exploring and retrieving sequence and metadata for species across the tree of life with NCBI Datasets. *Scientific Data*, *11*, 732. https://doi.org/10.1038/s41597-024-03571-y

Oliver, T., Sánchez-Baracaldo, P., Larkum, A. W., Rutherford, A. W., & Cardona, T. (2021). Time-resolved comparative molecular evolution of oxygenic photosynthesis. *Biochimica Et Biophysica Acta. Bioenergetics*, *1862*(6), 148400. https://doi.org/10.1016/j.bbabio.2021.148400

OpenAI. (2021). *Create Art or Modify Images with AI*. OpenArt. https://openart.ai/home

OpenAI. (2022). *ChatGPT | OpenAI*. https://openai.com/chatgpt/

Oren, A., Mareš, J., & Rippka†, R. (2022). Validation of the names Cyanobacterium and Cyanobacterium stanieri, and proposal of Cyanobacteriota phyl. nov. *International Journal of Systematic and Evolutionary Microbiology*, *72*(10), 005528. https://doi.org/10.1099/ijsem.0.005528

Ouyang, L., Wu, J., Jiang, X., Almeida, D., Wainwright, C. L., Mishkin, P., Zhang, C., Agarwal, S., Slama, K., Ray, A., Schulman, J., Hilton, J., Kelton, F., Miller, L., Simens, M., Askell, A., Welinder, P., Christiano, P., Leike, J., & Lowe, R. (2022, March 4). *Training language models to follow instructions with human feedback*.

http://arxiv.org/abs/2203.02155

Parks, D. H., Chuvochina, M., Waite, D. W., Rinke, C., Skarshewski, A., Chaumeil, P.-A., & Hugenholtz, P. (2018). A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. *Nature Biotechnology*, *36*(10), 996–1004. https://doi.org/10.1038/nbt.4229

Peng, D., Gui, Z., & Wu, H. (2023). *Interpreting the Curse of Dimensionality from Distance Concentration and Manifold Effect*.

Pessi, I. S., Popin, R. V., Durieu, B., Lara, Y., Tytgat, B., Savaglia, V., Roncero-Ramos, B., Hultman, J., Verleyen, E., Vyverman, W., & Wilmotte, A. (2023). Novel diversity of polar Cyanobacteria revealed by genome-resolved metagenomics. *Microbial Genomics*, *9*(7), mgen001056. https://doi.org/10.1099/mgen.0.001056

Pfanzagl, B., Zenker, A., Pittenauer, E., Allmaier, G., Martinez-Torrecuadrada, J., Schmid, E. R., De Pedro, M. A., & Löffelhardt, W. (1996). Primary structure of cyanelle peptidoglycan of Cyanophora paradoxa : A prokaryotic cell wall as part of an organelle envelope. *Journal of Bacteriology*, *178*(2), 332–339. https://doi.org/10.1128/jb.178.2.332-339.1996

Piontek, M., Czyżewska, W., & Mazur-Marzec, H. (2023). Effects of Harmful Cyanobacteria on Drinking Water Source Quality and Ecosystems. *Toxins*, *15*(12), 703. https://doi.org/10.3390/toxins15120703

Pruitt, K., Murphy, T., Brown, G., & Murphy, M. (2020). RefSeq Frequently Asked Questions (FAQ). In *RefSeq Help [Internet]*. National Center for Biotechnology Information (US). https://www.ncbi.nlm.nih.gov/books/NBK50679/

Ratclif, J. W. (1988). *Pattern Matching : The Gestalt Approach*. Dr. Dobb's. http://www.drdobbs.com/database/pattern-matching-the-gestalt-approach/184407970

Rau, L. F. (1991). *Extracting company names from text*. 29, 30, 31, 32–29, 30, 31, 32. https://doi.org/10.1109/CAIA.1991.120841

Reimers, N., & Gurevych, I. (2021). *Sentence-transformers/all-MiniLM-L6-v2 · Hugging Face*. https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

Rodriguez-R, L. M., Jain, C., Conrad, R. E., Aluru, S., & Konstantinidis, K. T. (2021). Reply to : "Re-evaluating the evidence for a universal genetic boundary among microbial species." *Nature Communications*, *12*(1), 4060. https://doi.org/10.1038/s41467-021-24129-1

Roure, B., Rodriguez-Ezpeleta, N., & Philippe, H. (2007). SCaFoS : A tool for Selection, Concatenation and Fusion of Sequences for phylogenomics. *BMC Evolutionary Biology*, *7*(1), S2. https://doi.org/10.1186/1471-2148-7-S1-S2

Rousseeuw, P. J. (1987). Silhouettes : A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics*, *20*, 53–65. https://doi.org/10.1016/0377-0427(87)90125-7

Saini, M. K., Sebastian, A., Shirotori, Y., Soulier, N. T., Garcia Costas, A. M., Drautz-Moses, D. I., Schuster, S. C., Albert, I., Haruta, S., Hanada, S., Thiel, V., Tank, M., & Bryant, D. A. (2021). Genomic and Phenotypic Characterization of Chloracidobacterium Isolates Provides Evidence for Multiple Species. *Frontiers in Microbiology*, *12*, 704168. https://doi.org/10.3389/fmicb.2021.704168

Sánchez-Baracaldo, P., Bianchini, G., Wilson, J. D., & Knoll, A. H. (2022). Cyanobacteria and biogeochemical cycles through Earth history. *Trends in Microbiology*, *30*(2), 143–157. https://doi.org/10.1016/j.tim.2021.05.008

Sanino, A. (2020, October 15). *Gestalt-pattern-matcher*. npm. https://www.npmjs.com/package/gestalt-pattern-

matcher

Schubert, E. (2023). Stop using the elbow criterion for k-means and how to choose the number of clusters instead. *ACM SIGKDD Explorations Newsletter*, *25*(1), 36–42. https://doi.org/10.1145/3606274.3606278

Sessions, A. L., Doughty, D. M., Welander, P. V., Summons, R. E., & Newman, D. K. (2009). The continuing puzzle of the great oxidation event. *Current Biology : CB*, *19*(14), R567–574. https://doi.org/10.1016/j.cub.2009.05.054

Setubal, J. C. (2021). Metagenome-assembled genomes : Concepts, analogies, and challenges. *Biophysical Reviews*, *13*(6), 905–909. https://doi.org/10.1007/s12551-021-00865-y

Sheffield, N. C., LeRoy, N. J., & Khoroshevskyi, O. (2023). Challenges to sharing sample metadata in computational genomics. *Frontiers in Genetics*, *14*, 1154198. https://doi.org/10.3389/fgene.2023.1154198

Shi, C., Wei, B., Wei, S., Wang, W., Liu, H., & Liu, J. (2021). A quantitative discriminant method of elbow point for the optimal number of clusters in clustering algorithm. *EURASIP Journal on Wireless Communications and Networking*, *2021*(1), 31. https://doi.org/10.1186/s13638-021-01910-w

Sibbald, S. J., & Archibald, J. M. (2020). Genomic Insights into Plastid Evolution. *Genome Biology and Evolution*, *12*(7), 978–990. https://doi.org/10.1093/gbe/evaa096

Soo, R. M., Skennerton, C. T., Sekiguchi, Y., Imelfort, M., Paech, S. J., Dennis, P. G., Steen, J. A., Parks, D. H., Tyson, G. W., & Hugenholtz, P. (2014). An Expanded Genomic Representation of the Phylum Cyanobacteria. *Genome Biology and Evolution*, *6*(5), 1031–1045. https://doi.org/10.1093/gbe/evu073

Sousa, F. L., Shavit-Grievink, L., Allen, J. F., & Martin, W. F. (2013). Chlorophyll Biosynthesis Gene Evolution Indicates Photosystem Gene Duplication, Not Photosystem Merger, at the Origin of Oxygenic Photosynthesis. *Genome Biology and Evolution*, *5*(1), 200–216. https://doi.org/10.1093/gbe/evs127

SRA Metadata and Submission Overview. (n.d.). Retrieved August 19, 2024, from https://www.ncbi.nlm.nih.gov/sra/docs/submitmeta/

Stanier, R. Y., Deruelles, J., Rippka, R., Herdman, M., & Waterbury, J. B. (1979). Generic Assignments, Strain Histories and Properties of Pure Cultures of Cyanobacteria. *Microbiology*, *111*(1), 1–61. https://doi.org/10.1099/00221287-111-1-1

StatQuest with Josh Starmer (Director). (2023, March 13). *Word Embedding and Word2Vec, Clearly Explained !!!* https://www.youtube.com/watch?v=viZrOnJclY0

Steinbach, M., Ertöz, L., & Kumar, V. (2004). The Challenges of Clustering High Dimensional Data. In L. T. Wille (Ed.), *New Directions in Statistical Physics* (pp. 273–309). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-662-08968-2_16

Stirbet, A., Lazár, D., Guo, Y., & Govindjee, G. (2019). Photosynthesis : Basics, history and modelling. *Annals of Botany*, *126*(4), 511–537. https://doi.org/10.1093/aob/mcz171

Stunda-Zujeva, A., Berele, M., Lece, A., & Šķesters, A. (2023). Comparison of antioxidant activity in various spirulina containing products and factors affecting it. *Scientific Reports*, *13*, 4529. https://doi.org/10.1038/s41598-023-31732-3

Sudo, Y., Hata, K., & Nakadai, K. (2023, May 28). *Retraining-free Customized ASR for Enharmonic Words Based on a Named-Entity-Aware Model and Phoneme Similarity Estimation*. http://arxiv.org/abs/2305.17846

Sutskever, I., Vinyals, O., & Le, Q. V. (2014, December 14). *Sequence to Sequence Learning with Neural Networks*. http://arxiv.org/abs/1409.3215

Tae, J. (2021, February 10). *NLI Models as Zero-Shot Classifiers*. Jake Tae. https://jaketae.github.io/study/zero-shot-classification/

Talamadupula, K. (2024, May 31). *A Guide to Building an LLM from Scratch*. Symbl.ai. https://symbl.ai/developers/blog/a-guide-to-building-an-llm-from-scratch/

Taylor, F. J. (1980). On dinoflagellate evolution. *Bio Systems*, *13*(1-2), 65–108. https://doi.org/10.1016/0303-2647(80)90006-4

Thallam, R. (2020). *An overview of BigQuery's architecture and how to quickly get started*. Google Cloud Blog. https://cloud.google.com/blog/products/data-analytics/new-blog-series-bigquery-explained-overview

Thorndike, R. L. (1953). Who belongs in the family ? *Psychometrika*, *18*(4), 267–276. https://doi.org/10.1007/BF02289263

Tibshirani, R. (2000). Estimating the number of clusters ina data set via the gap statistic. *Royal Statistical Society*.

Trieschnigg, D., & Kraaij, W. (2004). *TNO Hierarchical topic detection report*.

ttnphns. (2016, June 7). *Answer to "Choosing the right linkage method for hierarchical clustering"*. Cross Validated. https://stats.stackexchange.com/a/217742

Verlinden, R. a. J., Hill, D. J., Kenward, M. A., Williams, C. D., & Radecka, I. (2007). Bacterial synthesis of biodegradable polyhydroxyalkanoates. *Journal of Applied Microbiology*, *102*(6), 1437–1449. https://doi.org/10.1111/j.1365-2672.2007.03335.x

Vidal, L., & Ballot, A. (2021). *Introduction to cyanobacteria*.

Wang, Z., Pang, Y., & Lin, Y. (2023, December 2). *Large Language Models Are Zero-Shot Text Classifiers*. https://doi.org/10.48550/arXiv.2312.01044

Ward, L. M., Hemp, J., Shih, P. M., McGlynn, S. E., & Fischer, W. W. (2018). Evolution of Phototrophy in the Chloroflexi Phylum Driven by Horizontal Gene Transfer. *Frontiers in Microbiology*, *9*, 260. https://doi.org/10.3389/fmicb.2018.00260

Ward, L. M., & Shih, P. M. (2022). Phototrophy and carbon fixation in Chlorobi postdate the rise of oxygen. *PLoS ONE*, *17*(8), e0270187. https://doi.org/10.1371/journal.pone.0270187

Warren, C. (2008). *Genomic Data Resources : Curation, Databasing, and Browsers | Learn Science at Scitable*. https://www.nature.com/scitable/topicpage/genomic-data-resources-challenges-and-promises-743721/

Whitton, B. A. (2012). *Ecology of Cyanobacteria II : Their Diversity in Space and Time*. Springer Science & Business Media. https://books.google.com?id=4oJ_vi27s18C

Xie, H. (2021). Research and Case Analysis of Apriori Algorithm Based on Mining Frequent Item-Sets. *Open Journal of Social Sciences*, *09*(04), 458–468. https://doi.org/10.4236/jss.2021.94034

Yin, J., Li, H., & Xiao, K. (2023). Origin of Banded Iron Formations : Links with Paleoclimate, Paleoenvironment, and Major Geological Processes. *Minerals*, *13*(4, 4), 547. https://doi.org/10.3390/min13040547

Zeng, Y., Feng, F., Medová, H., Dean, J., & Koblížek, M. (2014). Functional type 2 photosynthetic reaction centers found in the rare bacterial phylum Gemmatimonadetes. *Proceedings of the National Academy of Sciences of the United States of America*, *111*(21), 7795–7800. https://doi.org/10.1073/pnas.1400295111

Zeng, Z., Yu, J., Gao, T., Meng, Y., Goyal, T., & Chen, D. (2024, April 16). *Evaluating Large Language Models at Evaluating Instruction Following*. http://arxiv.org/abs/2310.07641

Zhang, W., Liu, J., Xiao, Y., Zhang, Y., Yu, Y., Zheng, Z., Liu, Y., & Li, Q. (2022). The Impact of Cyanobacteria

Blooms on the Aquatic Environment and Human Health. *Toxins*, *14*(10), 658. https://doi.org/10.3390/toxins14100658

Zielezinski, A., Vinga, S., Almeida, J., & Karlowski, W. M. (2017). Alignment-free sequence comparison : Benefits, applications, and tools. *Genome Biology*, *18*, 186. https://doi.org/10.1186/s13059-017-1319-7