Rapport de Projet

Détection de patterns dans des séries de voyages

Membres du groupe :

OULMAHDI Riad
KACED Louheb
DJEHA Wassim
KHEYAR Aya
HALIT Rafik
AZOU Ali

Encadrants :

Abdelkader OUALI
Jacques EVERWYN

Entreprise :

Sinay SAS

Date : 26 janvier 2025

Année universitaire : 2024-2025

Table des matières

1 Introduction

1.1 Contexte

Le commerce maritime, essentiel à l'économie mondiale, représente plus de 80 % du commerce global en volume. Des milliers de navires traversent les océans chaque jour, reliant les continents et soutenant les chaînes d'approvisionnement. L'efficacité de ce système dépend de la gestion des itinéraires, des ports et des temps de transit.

L'analyse des données, comme celles de l'AIS (Automatic Identification System), permet d'identifier des motifs récurrents dans les trajectoires des navires. Cela aide à optimiser les flux maritimes, améliorer l'utilisation des ports, réduire les coûts et minimiser l'impact environnemental. Dans ce contexte, l'extraction de motifs séquentiels est donc un outil crucial pour exploiter les données historiques et améliorer la prise de décisions dans ce secteur.

1.2 Entreprise SINAY

SINAY est une entreprise travaillant sur des solutions pour améliorer l'efficacité des entreprises maritimes. L'entreprise collecte et agrège des données provenant de multiples sources telles que la position des navires, la météo, les courants, la faune et la flore et bien d'autres encore. Ces données sont combinées et analysées avec différents algorithmes d'apprentissage automatique pour obtenir les meilleurs indicateurs de performance pour les entreprises maritimes.

1.3 Objectifs

L'objectif principal de cette étude est d'extraire des lignes maritimes régulières (LMR) à partir des historiques de trajets des navires. Ces lignes maritimes se manifestent par des motifs séquentiels récurrents, permettant d'optimiser les itinéraires.

Pour atteindre cet objectif, plusieurs axes d'analyse seront explorés :

— L'extraction de motifs fréquents dans les trajets maritimes, en tenant compte de seuils de fréquence.

— L'identification des ports les plus visités et la formulation de recommandations visant à optimiser l'efficacité des trajets maritimes.

1.4 Approches

Nous utilisons des algorithmes d'extraction de motifs séquentiels, tels que GSP, CLoSPan et PrefixSpan, pour identifier les comportements récurrents dans les trajets maritimes.

2 Fouille de motifs séquentiels

2.1 Notions de base

2.1.1 Base de données séquentielle

Une base de données séquentielle est une structure de données spécifiquement conçue pour stocker et organiser des séquences d'objets, où la dimension temporelle revêt une importance capitale. Les séquences ainsi définies correspondent à des événements ou des transactions qui se déroulent dans un ordre chronologique bien défini. Contrairement aux bases de données transactionnelles classiques, où les transactions sont indépendantes et non ordonnées, les bases de données séquentielles intègrent explicitement l'ordre temporel au sein de leur structure.

Caractéristiques principales :

- Chaque séquence est associée à un identifiant unique, tel qu'un client ou un utilisateur (dans le contexte présent, les **navires**).
- Les éléments (dans notre cas, les **ports**) sont organisés selon leur ordre d'occurrence temporelle.
- La dimension temporelle des événements est cruciale pour l'analyse des séquences.

Exemple de base séquentielle

Nous considérons un tableau illustrant les séquences d'items pour différents clients, avec les dates correspondantes. Les items sont représentés par des lettres et les dates par des valeurs spécifiques :

Client	Date	Items
C_1	01/01/2004	B, F
C_1	02/02/2004	B
C_1	04/02/2004	C
C_1	18/02/2004	H, I
C_2	11/01/2004	A
C_2	12/01/2004	C
C_2	29/01/2004	D, F, G
C_3	05/01/2004	C, E, G
C_3	12/02/2004	A, B
C_4	06/02/2004	B, C
C_4	07/02/2004	D, G
C_4	08/02/2004	I

`https://www.lirmm.fr/~poncelet/publications/papers/motifs_sequentiels.pdf`

Les motifs contenus dans ces séquences jouent un rôle clé dans l'analyse des comportements des utilisateurs et permettent l'identification de tendances ou de motifs récurrents.

2.1.2 Motif séquentiel

Un motif séquentiel est défini comme une séquence d'items ou d'événements qui apparaissent dans un ordre spécifique au sein des données. La **fréquence** d'un motif séquentiel est déterminée par la proportion de séquences contenant ce motif. Un motif est considéré comme fréquent si sa fréquence dépasse un seuil de support minimal prédéfini par l'utilisateur.

Explication d'un motif

Examinons le cas du client C_2 et analysons ses interactions avec les événements survenus dans ses séquences. Soit un motif particulier de la forme :

$$B \to C$$

- **Motif choisi :** $B \to C$, support = 50%.
- **Interprétation :** Les clients C_1 et C_4 effectuent une action associée à l'item **B**, suivie par une action liée à l'item **C**.
- **Utilité :** Ce motif met en évidence une relation entre les produits **B** et **C**. On peut en déduire que l'achat du produit **B** est souvent suivi de l'achat du produit **C**.

2.1.3 Fréquence dans les motifs séquentiels

Le support d'un motif séquentiel est défini comme la proportion de séquences dans la base de données qui contiennent ce motif particulier. Formellement, si un motif σ apparaît dans n_σ séquences sur un total de N séquences, le support S_σ de ce motif est donné par la relation suivante :

$$S_\sigma = \frac{n_\sigma}{N}.$$

Par exemple, si un motif $\{B, I\}$ apparaît dans les séquences de C_1 et C_4, soit dans 2 séquences sur un total de 4, son support est de :

$$S_{\{B,I\}} = \frac{2}{4} = 50\%.$$

Ainsi, un motif est dit fréquent si son support dépasse un seuil minimal prédéfini, souvent noté θ. Par conséquent, un motif σ est fréquent si :

$$S_\sigma \geq \theta.$$

2.2 Méthodes utilisées

2.2.1 Algorithme GSP

L'algorithme GSP (Generalized Sequential Patterns) est une méthode puissante pour extraire des motifs séquentiels fréquents à partir de bases de données. Il repose sur une approche itérative fondée sur deux étapes principales :

1. Génération de candidats
- GSP commence par identifier les éléments fréquents de taille 1 (1-fréquent) en une seule passe sur la base de données.
- Ensuite, il génère des séquences potentielles (candidats) en combinant les motifs fréquents identifiés lors de l'itération précédente. Cette génération s'appuie sur la jointure d'éléments fréquents et exclut les séquences qui contiennent des sous-séquences infrequentes, grâce à la propriété dite d'anti-monotonie (si une séquence n'est pas fréquente, ses super-séquences ne peuvent pas l'être).

2. Évaluation du support
- Chaque séquence candidate est comparée à la base de données pour calculer son support, c'est-à-dire la proportion de séquences dans la base contenant cette séquence.
- Seules les séquences atteignant le seuil de support minimal sont retenues comme motifs fréquents.

Une explication plus détaillée de cet algorithme est disponible dans le document complet : `https://www.lirmm.fr/~poncelet/publications/papers/motifs_sequentiels.pdf`.

2.2.2 Algorithme PREFIXSPAN

L'algorithme PREFIXSPAN optimise l'extraction de motifs séquentiels fréquents en réduisant le nombre de candidats générés. Il exploite les préfixes communs des séquences pour projeter la base de données en sous-ensembles plus petits, facilitant ainsi la recherche.

1. Extraction des items fréquents
- Une première passe sur la base de données identifie les items fréquents. Ces items servent de préfixes initiaux pour diviser l'espace de recherche.

2. Projection des bases
- Chaque préfixe fréquent définit une base projetée, contenant uniquement les séquences partageant ce préfixe. Seuls les suffixes sont considérés dans ces bases projetées. Cela réduit significativement l'espace de recherche.

3. Exploration récursive
- À partir de chaque base projetée, PREFIXSPAN identifie de nouveaux préfixes fréquents et génère des motifs plus longs. Ce processus se répète de manière récursive jusqu'à ce qu'aucune séquence fréquente ne puisse être étendue.

Une explication plus détaillée de cet algorithme est disponible dans le document complet : `https://www.lirmm.fr/~poncelet/publications/papers/motifs_sequentiels.pdf`.

2.2.3 Algorithme CLOSPAN

L'extraction de motifs séquentiels peut générer des résultats redondants, où toutes les sous-séquences fréquentes d'un motif complet partagent le même support. Pour éliminer cette redondance, les motifs séquentiels fermés (Closed Sequential Patterns) ont été introduits : ils regroupent toute l'information sans répétition inutile. CLOSPAN optimise cette extraction en se basant sur PREFIXSPAN.

1. Ordre lexicographique des motifs :
L'exploration structurée permet de limiter l'espace de recherche en triant les motifs de manière lexicographique.

2. Relations systématiques entre items :
CLOSPAN détecte les relations fixes entre les items (par exemple : « A précède toujours B ») pour éviter des calculs superflus.

Fonctionnement général

- **Identification des motifs fréquents :**
  La base de cette étape repose sur l'algorithme PREFIXSPAN.
- **Identification des motifs fermés :**
  Seuls les motifs qui ne possèdent aucun super-motif ayant le même support sont conservés.

Une explication plus détaillée de cet algorithme est disponible dans le document complet : `https://www.lirmm.fr/~poncelet/publications/papers/motifs_sequentiels.pdf`.

3 Préparation des données

3.1 Données de l'entreprise

L'entreprise nous a fourni des données sous la forme d'un fichier JSON contenant des informations sur les voyages des navires. Voici une description des principaux champs du fichier :

- **id** : Identifiant unique attribué à chaque voyage.
- **mmsi** : Numéro d'Identification Mobile Maritime (*Maritime Mobile Service Identity*). Ce numéro est utilisé pour identifier les navires, bien qu'il puisse changer au cours de leur vie.
- **imo** : Numéro attribué par l'Organisation Maritime Internationale (*International Maritime Organization*), servant d'identifiant permanent et unique pour chaque navire.
- **departure_date** : Date et heure de départ du navire de son port d'origine.
- **arrival_date** : Date et heure d'arrivée du navire à son port de destination.
- **departure_port** : Nom du port de départ.
- **arrival_port** : Nom du port d'arrivée.


```
[
  {
    "id": 228283,
    "mmsi": 229338000,
    "imo": 9625906,
    "departure_date": "2022-07-11T21:10:04",
    "arrival_date": "2022-07-16T13:27:32",
    "departure_port": "AUBTB",
    "arrival_port": "AUPOR",
    "berth_date": "2022-07-14T23:34:17",
    "unberth_date": "2022-07-16T13:27:32"

  }
]
```


FIGURE 1 – Echantillon des données sous format JSON.

3.2 Nettoyage de données

L'objectif initial est de construire des séquences de ports visités par chaque navire en utilisant comme identifiant principal le champ **imo**, en cas d'absence de ce dernier on utilise le champ **mmsi**. Pour ce faire :

1. Nous avons sélectionné les champs `departure_port` et `arrival_port` afin de créer des séquences ordonnées par la **date d'arrivée**.
2. Au cours de ce processus, nous avons identifié plusieurs anomalies dans les données, notamment :
   — Des ports non renseignés (`null`).
   — Des voyages avec un port de départ identique au port d'arrivée.
   — Le port d'arrivé dans un voyage `n` est le même que le port de départ d'un voyage `n+1` , ce qui cause une duplication du même port consecutivement dans les séquences.

   Exemple :

$$\text{Séquence 1 :}\langle \text{AUBTB} \rightarrow \text{AUPOR} \rightarrow \text{AUPOR} \rightarrow \text{VNSGN} \rightarrow \ldots \rangle$$

Pour garantir la qualité des données, nous avons exclu ces cas afin d'obtenir des séquences cohérentes. Voici un exemple de séquence obtenue après nettoyage :

$$\text{Séquence 1 :}\langle \text{AUBTB} \rightarrow \text{AUPOR} \rightarrow \text{VNSGN} \rightarrow \ldots \rangle$$

Visualisation de la séquence :
Pour mieux comprendre la séquence de voyages, nous avons créé un graphique représentant les ports de départ et d'arrivée. Chaque flèche représente un voyage d'un port à un autre.

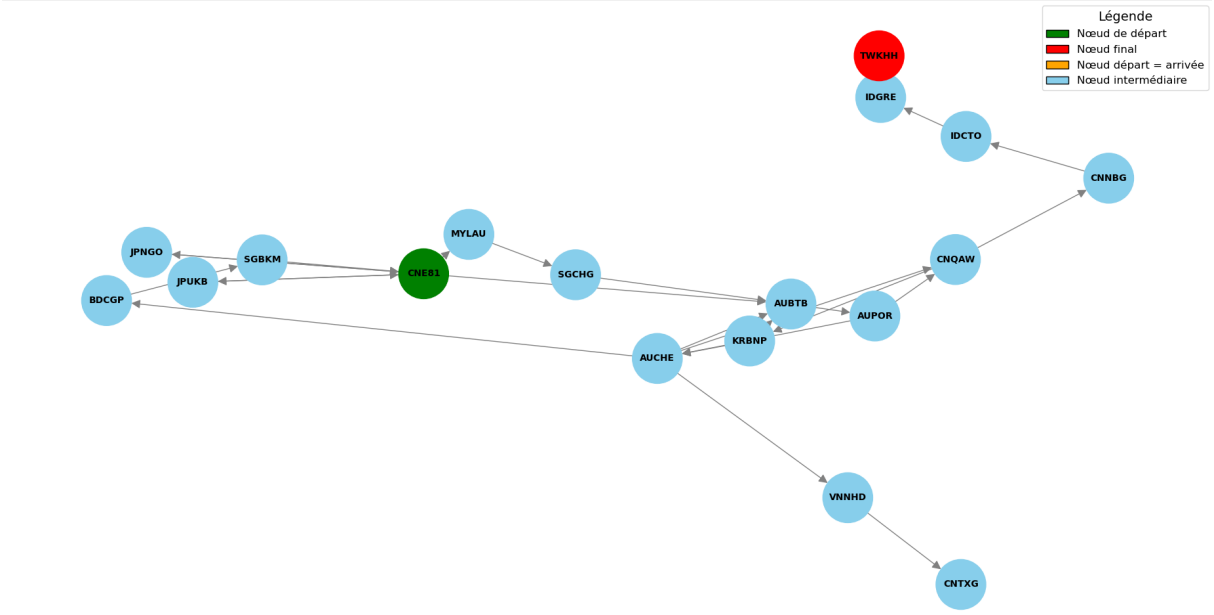

FIGURE 2 – Graphique de la séquence de voyages

3.3 Adaptation au format de données SPMF

Lors de l'analyse des séquences sous leur forme actuelle (chaînes de caractères représentant les ports), nous avons observé une perte de performance dans les traitements. Cela s'explique par le coût élevé de manipulation des chaînes de caractères, en particulier lorsque le volume de données est important.

3.4 Solution proposée

Pour optimiser le traitement des séquences, nous avons adopté une approche consistant à convertir les chaînes de caractères en nombres entiers. Voici les étapes mises en œuvre :

1. **Création d'un dictionnaire de conversion** : Chaque port est associé à un entier unique. Cela permet de représenter les ports de manière plus compacte et rapide à traiter.
2. **Séparation des éléments dans les séquences** :
   — Les différents itemsets (ports) sont séparés par la valeur **-1**.
   — La fin d'une séquence est indiquée par la valeur **-2**.

Voici un exemple de séquence transformée avec cette méthode :

$$1\ \mathbf{-1}\ 3\ \mathbf{-1}\ 5\ \mathbf{-1}\ 6\ \mathbf{-1}\ 1\ \mathbf{-1}\ 2\ \mathbf{-1}\ \ldots\ \mathbf{-2}$$

Avantages de cette méthode

— **Efficacité** : La manipulation de nombres entiers est bien plus rapide que celle de chaînes de caractères, réduisant ainsi le temps de traitement.
— **Simplicité** : Le format compact des séquences facilite leur analyse et leur exploitation dans des algorithmes.
— **Scalabilité** : Cette méthode est mieux adaptée à un grand volume de données.

3.5 Filtrage des motifs non pertinents

Après l'éxecution de l'algorithme PrefixSpan sur les séquences, un processus de post-traitement est appliqué pour garantir que seuls les motifs pertinents sont retenus. Ce processus a pour but d'exclure les motifs qui ne reflètent pas notre objectif d'identifier des lignes maritimes régulières. Les étapes suivantes sont effectuées :

1. **Exclusion des motifs de taille 1** : Les motifs contenant un seul port ne sont pas pertinents, car ils ne représentent pas des itinéraires maritimes significatifs.
2. **Suppression des motifs avec des ports consécutifs identiques** : Si un motif contient deux ports identiques consécutifs, il existe forcément un motif équivalent sans cette duplication, ayant le même support. Par conséquent, ces motifs sont éliminés pour éviter les redondances.
3. **Exclusion des motifs avec un unique port répété plusieurs fois** : Les motifs composés d'un seul port répété (par exemple : $\langle AUPOR \rightarrow AUPOR \rightarrow AUPOR \rangle$) sont exclus, car ils ne représentent pas des lignes maritimes, mais plutôt des mouvements locaux ou des anomalies.

4 Expérimentations et analyse

Dans cette section, nous présentons le protocole expérimental ainsi que les résultats obtenus à partir de notre base de données séquentielle.

4.1 Protocole expérimental

Les expérimentations ont été réalisées sur une base de données séquentielle contenant 1552 navires. Nous avons fait varier le support minimal de 1% à 20%, avec un pas de 1. Les métriques observées incluent :

— le temps d'exécution (en milli-secondes),
— le nombre de motifs trouvés,
— la taille moyenne des motifs,
— l'écart-type des tailles des motifs.

4.2 Résultats expérimentaux

Support minimum	**Temps pris (ms)**	**Nombre de motifs trouvés**	**Taille moyenne des motifs**	**Écart type**
0.01	131600.55	6462944	10.19	2.02
0.02	2021.33	25696	5.66	2.00
0.03	768.38	4345	4.92	2.14
0.04	569.71	1539	4.47	2.10
0.05	504.18	741	4.29	2.04
0.06	420.92	411	3.91	1.77
0.07	437.76	225	3.29	1.39
0.08	363.63	133	3.06	1.19
0.09	357.14	74	2.59	0.80
0.10	347.67	51	2.37	0.62
0.11	341.53	29	2.28	0.52
0.12	345.66	17	2.18	0.38
0.13	337.82	8	2.00	0.00
0.14	339.09	5	2.00	0.00
0.15	329.06	1	2.00	0.00
0.16	334.06	1	2.00	0.00
0.17	335.52	0	nan	nan
0.18	333.35	0	nan	nan
0.19	332.50	0	nan	nan
0.20	339.00	0	nan	nan

TABLE 1 – Résultats d'expérimentation

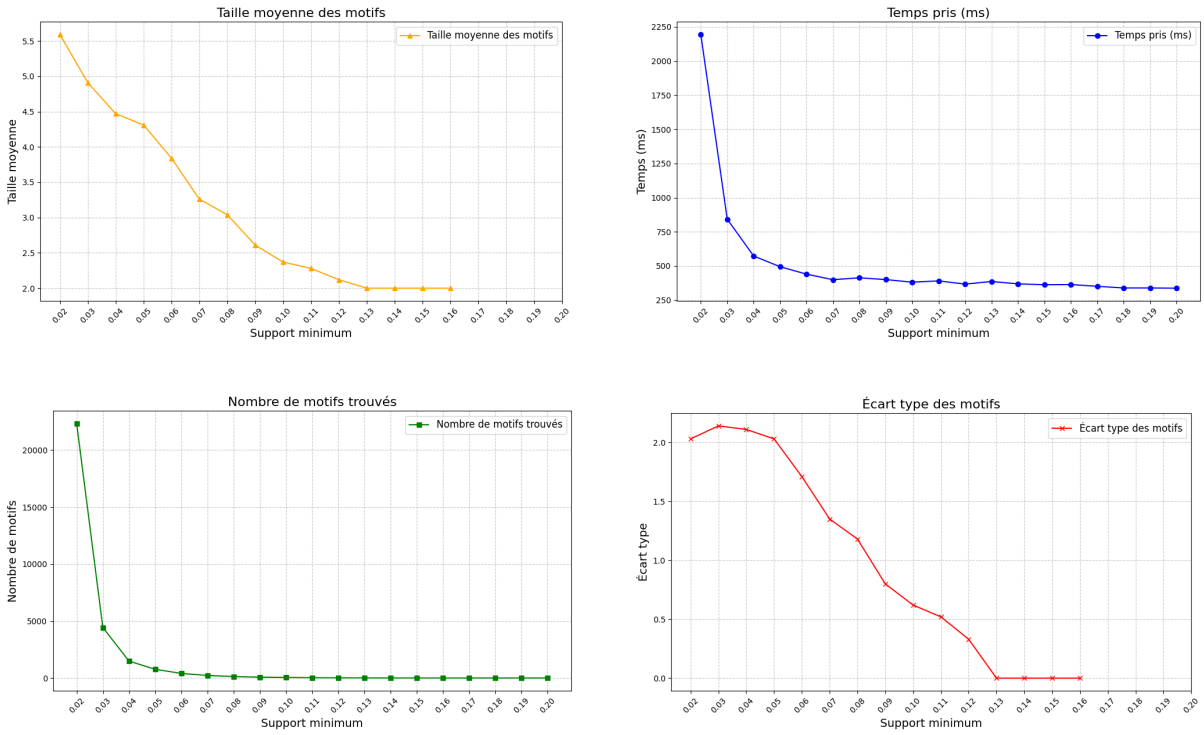

FIGURE 3 – Visualisations des résultats expérimentaux.

4.3 Interprétation des résultats

Les résultats expérimentaux présentés dans le tableau 4.2 mettent en évidence plusieurs tendances importantes :

— **Impact du support minimum :**
En augmentant le seuil du support minimum, le nombre de motifs trouvés diminue considérablement. Par exemple, pour un support de 0.01, plus de 6 millions de motifs sont identifiés, tandis qu'à partir d'un support de 0.17, aucun motif n'est découvert. Cela reflète qu'il n'existe pas de motifs (séquences de ports) parcourus par au moins 264 navires dans cet ordre.

— **Temps d'exécution :**
Le temps d'exécution est fortement corrélé au nombre de motifs trouvés. Lorsque le support minimum est bas (par exemple, 0.01), le temps d'exécution dépasse 125 secondes. À l'inverse, pour des supports élevés (supérieurs à 0.10), le temps d'exécution est réduit à environ 350 millisecondes, traduisant la diminution de la complexité de recherche. En effet, avec un support plus élevé, le nombre de motifs 1-fréquents diminue, ce qui réduit également le nombre de motifs k-fréquents à générer.

— **Taille moyenne des motifs :**
La taille moyenne des motifs tend également à diminuer avec l'augmentation du support minimum. Cela indique que les séquences plus longues deviennent moins fréquentes et sont progressivement éliminées des résultats lorsque le seuil de support augmente.

— **Écart-type des tailles des motifs :**
L'écart-type diminue avec l'augmentation du support minimum, reflétant une homogénéité croissante des tailles des motifs. À faible support (0.01), la variabilité est notable (2.03), tandis qu'à partir de 0.13, elle devient nulle, indiquant que les motifs restants ont tous la même taille.

4.4 Analyse des résultats de PrefixSpan

Voici un échantillon de motifs fréquents extraits à l'aide de l'algorithme PrefixSpan :

— {PATBG} → {PACTB} → {PAROD} → {PACTB} → {NLMSV} → {PAROD} → {PACTB} → {BEANR} → {PACTB} → {PAROD} → {PATBG} → {PAROD} → {PACTB} #SUP : 16

— {CNE76} → {KRBNP} → {CNE76} → {KRBNP} → {CNE76} → {KRBNP} #SUP : 43

— **{GBLGP} → {BEANR} → {NLMSV} #SUP : 63**

— {PAONX} → {PAROD} → {PATBG} → {PAROD} → {PACTB} → {PAROD} → {PATBG} → {PAROD} → {PACTB} #SUP : 40

— {USSAV} → {PATBG} #SUP : 62

— {COBUN} → {PATBG} → {PAROD} → {PATBG} → {COBUN} #SUP : 33

Chaque motif correspond à une séquence consécutive de ports fréquents dans la base de données, où SUP indique le support, c'est-à-dire le nombre de séquences dans lesquelles motif apparaît.

Analyse :
Si nous prenons, par exemple, le motif **GBLGP → BEANR → NLMSV** avec SUP : **63**, cela signifie que 63 navires ont suivi ce trajet, c'est-à-dire qu'ils sont passés par ces ports dans cet ordre. Cependant, entre deux ports, comme **GBLGP → BEANR**, il est possible qu'ils aient effectué d'autres trajets intermédiaires, tout comme il est possible qu'ils aient effectué ce trajet directement.

5 Conclusion

Ce projet a permis d'explorer l'extraction de motifs séquentiels dans les trajectoires maritimes, un domaine clé pour optimiser l'efficacité du commerce maritime. En utilisant des algorithmes d'extraction tels que GSP, PrefixSpan, et CLoSPan, nous avons pu identifier des motifs récurrents dans les trajets des navires, permettant ainsi de proposer des solutions pour améliorer l'utilisation des ports et optimiser les itinéraires maritimes.

L'analyse des données AIS a démontré l'importance d'une approche basée sur les motifs séquentiels pour identifier des tendances et des comportements récurrents dans les mouvements des navires. Grâce à l'application de ces algorithmes, nous avons non seulement extrait des lignes maritimes régulières (LMR), mais aussi optimisé les itinéraires en fonction de la fréquence des ports et des trajectoires les plus empruntées. Ces résultats peuvent, à terme, contribuer à la réduction des coûts et de l'impact environnemental du secteur maritime.

L'approche méthodologique adoptée a permis de tirer parti des données historiques pour prendre des décisions éclairées concernant la gestion des flux maritimes. Cependant, certains défis demeurent, notamment l'amélioration des performances des algorithmes face à de grandes quantités de données, ainsi que la prise en compte de facteurs externes, tels que les plages de dates durant lesquelles les navires passent par les ports, ou encore l'intégration des compagnies afin de cibler les motifs réguliers d'une compagnie donnée.

En conclusion, ce travail ouvre la voie à de futures améliorations dans la gestion du trafic maritime, avec des applications possibles dans l'optimisation des trajets. Des recherches supplémentaires sont nécessaires pour affiner les modèles proposés et intégrer des données complémentaires, offrant ainsi des solutions toujours plus robustes et adaptées aux enjeux du secteur maritime.

Références

[1] *Wikipedia GSP Algorithm, The Free Encyclopedia.* https://en.wikipedia.org/wiki/GSP_algorithm

[2] *Motifs séquentiels.* https://www.lirmm.fr/~poncelet/publications/papers/motifs_sequentiels.pdf

[3] NUMPY. *Fundamental package for scientific computing with Python.* https://www.numpy.org.

[4] PANDAS. *Python data analysis library.* https://www.pandas.pydata.org.

[5] Matplotlib. https://matplotlib.org/.

[6] *SPMF*, https://www.philippe-fournier-viger.com/spmf/