REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE

UNIVERSITE MOHAMED BOUDIAF - M'SILA

FACULTE DES MATHEMATIQUES ET DE L'INFORMATIQUE

DEPARTEMENT D'INFORMATIQUE

N° :……..

DOMAINE : MATHEMATIQUES ET INFORMATIQUE

FILIERE : INFORMATIQUE

OPTION : GENIE LOGICIEL

Mémoire présenté pour l'obtention
Du diplôme de Master Académique

Par: Bentoumi Ismail

Intitulé

Extraction de motifs séquentiels

Soutenu devant le jury composé de :

Mr .Mohamed Bounif	Université de M'sila	Président
Mr. Guesmia Salah	Université de M'sila	Rapporteur
Mr.Lakehal Aiat	Université de M'sila	Examinateur

Année universitaire : 2016 /2017

Remerciements

Nous tenons à remercier. En premier lieu le bon Dieu de nous avoir donné la force et le courage pour réaliser à terme ce travail. Nous remercierons notre promoteur pour nous avoir

Proposé ce thème, nous lui sont très reconnaissants pour ses remarques et conseils.

En fin nos remerciements s'adressent aussi aux membres du jury pour nous avoir fait l'honneur d'examiner notre travail.

TABLE DES MATIERES

LISTE DES TABLEAUX

LISTE DES FIGURES

INTRODUCTION GENERALE

Le volume des données numériques collectées et conservées au niveau des systèmes d'information d'aujourd'hui ne cesse de croître et le besoin d'extraire des informations utiles à partir de ces données ne cesse de se transformer en une stratégie. Les entreprises comprennent, maintenant, que leurs données ne sont plus utiles pour uniquement l'utilisation fonctionnelle classique connue, mais qu'elles peuvent leur trouver des utilisations encore plus avancées. Les énormes masses de données conservées souvent dans de gigantesques bases de données dormantes, peuvent sans aucun doute contenir des connaissances de très grande valeur commerciale ou scientifique n'attendant qu'à être exploitées.

La question qui se pose est alors : Comment peut-on extraire les informations cachées au sein de grandes bases de données ? La puissance de calcul des ordinateurs actuels et la baisse des coûts de stockage laissent prédire que nous disposons des moyens physiques pour le faire. Le problème réside alors au niveau logiciel. En effet, les bases de données classiques ne sont plus de taille à faire face à l'analyse de telles informations et c'est grâce à ce besoin pressant que sont apparues les techniques d'extraction de connaissances à partir des données communément connues sous le nom de «Data Mining ». Citons parmi ces techniques : les règles d'association, l'analyse de liens, la détection de clusters, les algorithmes génétiques, les arbres de décision et les règles d'association séquentielles.

Dans la technique de règles d'association séquentielles, l'extraction des connaissances se fait en recherchant des relations d'ordre, sous forme d'enchaînements appelés séquences entre items (objets, attributs, …) ou ensembles d'items d'une base de donnée. L'objectif est alors de trouver dans cette base de données, toutes les séquences d'items apparaissant avec une certaine certitude selon une mesure d'intérêt choisie par l'utilisateur, par exemple, la fréquence. Il s'agit donc, de construire l'ensemble de tous ces motifs intéressants appelés motifs séquentiels [33].

Notre travail consiste dans un premier temps à étudier et à comprendre le fonctionnement des algorithmes d'extraction de motifs séquentiels, dans un deuxième temps, à évaluer et à comparer les performances de ces algorithmes en fonction de différents paramètres.

La structure de ce mémoire va comme suit :

- Le premier chapitre présente les fondamentaux du data Mining
- Dans le deuxième chapitre, nous expliquerons les problématiques de la recherche de règles d'association et de l'extraction de motifs séquentiels.

- Le troisième chapitre focalise sur les approches utilisées par les algorithmes d'extraction de motifs séquentiels.
- Le quatrième chapitre est consacré à la présentation des outils utilisés pour l'implémentation de noter application. Il présente également les expérimentations, les résultats et une discussion sur ces résultats.

Et pour finir, nous conclurons tout en proposant des perspectives.

CHAPITRE 1

FOUILLE DE DONNEES : DATA MINING

1.1 Introduction

L'exploration de données connue aussi sous l'expression de fouille de données, forage de données, prospection de données, data mining, ou encore extraction de connaissances à partir de données, a pour objet l'extraction d'un savoir ou d'une connaissance à partir de grandes quantités de données, par des méthodes automatiques ou semi-automatiques.

Elle se propose d'utiliser un ensemble d'algorithmes issus de disciplines scientifiques diverses telles que les statistiques, l'intelligence artificielle ou l'informatique, pour construire des modèles à partir des données, c'est-à-dire trouver des structures intéressantes ou des motifs selon des critères fixés au préalable, et d'en extraire un maximum de connaissances.

L'utilisation industrielle ou opérationnelle de ce savoir dans le monde professionnel permet de résoudre des problèmes très divers, allant de la gestion de la relation client à la maintenance préventive, en passant par la détection de fraudes ou encore l'optimisation de sites web. C'est aussi le mode de travail du journalisme de données l'exploration de données fait suite, dans l'escalade de l'exploitation des données de l'entreprise, à l'informatique décisionnelle. Celle-ci permet de constater un fait, tel que le chiffre d'affaires, et de l'expliquer comme le chiffre d'affaires décliné par produits, tandis que l'exploration de données permet de classer les faits et de les prévoir dans une certaine mesure notes 2 ou encore de les éclairer en révélant par exemple les variables ou paramètres qui pourraient faire comprendre pourquoi le chiffre d'affaires de tel point de vente est supérieur à celui de tel autre [14].

1.2 Petits historiques

Les premiers termes pour désigner la fouille de données sont apparus dans les années 1960.les statisticiens utilisent des termes comme « Pêche de données » pour désigner ce qu'ils considéraient comme une mauvaise pratique de l'analyse de données sans hypothèse. Le mot « datamining » est apparu dans les années 1990. Gregory Piatesky-Shapiro a inventé le

« Knowledge Discovery in Data bases », ce terme est devenu populaire en apprentissage communautaire. Quant au terme « Datamining », il apparait en 1991 et plus utilisé dans les milieux d'affaires et de presse. Aujourd'hui, les termes datamining et Knowledge Discovery sont tous deux utilisés. De nos jours, le datamining se présente comme un outil incontournable dans un service marketing, et au sein des processus décisionnels d'une entreprise. Il rassemble un faisceau de techniques statistiques qu'il convient d'utiliser au gré des problématiques descriptives ou décisionnelles. Le datamining s'accompagne le plus souvent d'une méthode de travail, afin d'ordonner aux mieux les hypothèses, les modélisations et les actions. De plus, les capacités de stockage et de calcul offertes sont de plus en plus performantes au fil des années [31].

1.3 Définition du DataMining

Data Mining signifie littéralement « fouille de données » ou « forage de données». Ce procédé, basé sur une série d'algorithmes ou modèles de data Mining permet d'extraire des informations à partir de données, informations qui, grâce à l'analyse, se convertissent en connaissances. Le data mining est l'analyse d'un ensemble d'observations qui a pour but de trouver des relations insoupçonnées et résumer les données d'une nouvelle manière, de façon qu'elles soient plus compréhensibles et utiles pour leurs détenteurs ». (David Hand, 2001).Autrement dit, il consiste à analyser des informations collectées dans des entrepôts de données afin d'y détecter des relations qu'il serait a priori impossible d'identifier sans cet outil. C'est un élément essentiel dans la relation client et de système d'aide à la décision. Le Data Mining est un processus inductif, itératif et interactif dont l'objectif est la découverte de modèles de données valides, nouveaux, utiles et compréhensibles dans de larges bases de données [20].

1.4 Extraction de connaissance à partir de données

Le processus d'extraction de l'information consiste à parcourir les données volumineuses contenues dans une base à la recherche de connaissance.

Ce processus est décrit dans le schéma suivant. (Figure1.1). Ce processus comprend des étapes de définition du problème (définition du domaine, but de l'utilisateur final), de préparation des données (sélection, Préparation, transformation), de fouille de données

(sélection, des outils de data mining appropriés, recherche des patrons) et d'évaluation des résultats pour aboutir aux nouvelles connaissances. Le processus présenté est itératif et plusieurs retours en arrière dans les différentes étapes peuvent être nécessaires pour affiner les résultats [20].

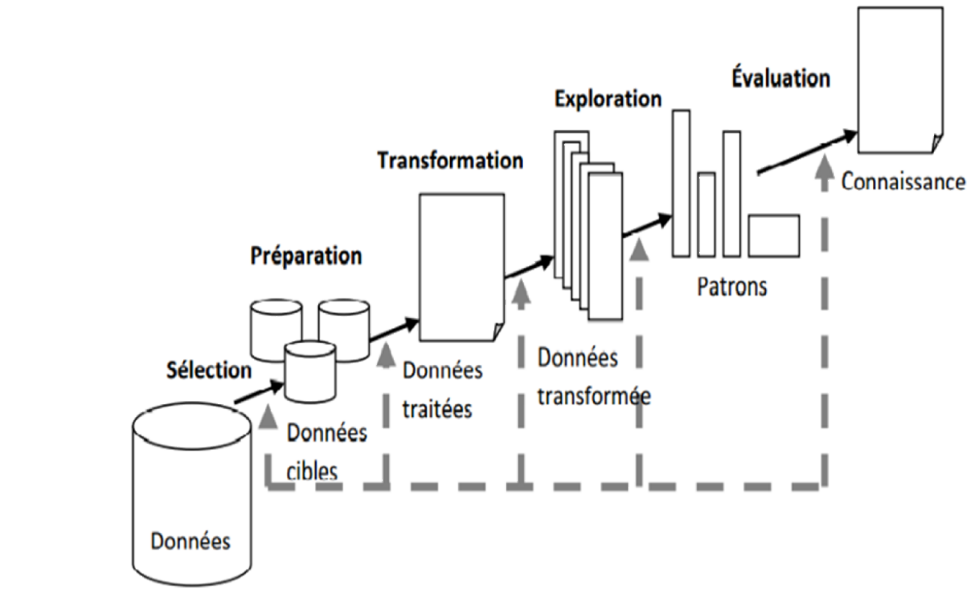

Figure1.1 Processus de découverte de connaissances à partir de données [20].

1.5 Domaines d'application du DataMining

Les domaines d'application du Data Mining sont très nombreux. Parmi lesquels on peut citer :

Assurances et santé :

- Découverte d'associations des demandes de remboursements.
- Identification de clients potentiels de nouvelles polices d'assurances.
- Détection d'association de comportements pour la découverte clients à risque.
- Détection de comportement frauduleux [31].

Banques / Finances :

- Détection d'usage frauduleux de cartes bancaires.
- Gestion du risque lié à l'attribution de prêts bancaires par le scoring.
- Découverte de relations cachées entre les indicateurs financiers.
- Détection de règles de comportement boursier par l'analyse des données du Marché [31].

Vente, distribution / Marketing :

- La gestion de la relation client (GRC ou CRM) consiste en l'ensemble des activités visant à cibler, attirer et conserver les "bons" clients.
- Détection d'associations de comportements d'achat.
- Découverte de caractéristiques de clientèle.
  Prédiction de probabilité de réponse aux campagnes de mailing [31].

Ressources Humaines :

Le DataMining est également utilisé dans les ressources humaines (RH) pour identifier les caractéristiques de leurs employés les plus performants l'information obtenue (comme les universités fréquentées par des employés potentiels) peut contribuer aux efforts de recrutement des ressources humaines. Ces dernières années, l'exploration de données a été largement utilisé dans les domaines de la science et de l'ingénierie, tels que la bioinformatique, la génétique, la médecine, l'éducation et l'énergie électrique [31].

Médical / Pharmaceutique :

- Diagnostic assisté par ordinateur (CAD) par l'apprentissage de systèmes experts.
- Explication ou prédiction de la réponse d'un patient à un traitement.
- Identification des thérapies à succès (combinaison de prescriptions).
- Étude des corrélations entre le dosage dans un traitement et l'apparition d'effets secondaire [31].

La génétique :

Dans l'étude de la génétique humaine, le datamining permet de répondre aux maladies. En effet, il vise à savoir comment les changements dans la séquence d'ADN d'un individu affectent les risques de développer des maladies courantes telles que le cancer, qui est d'une grande importance à l'amélioration des méthodes de diagnostic, la prévention et le traitement de ces maladies. Le data mining peut contribuer de manière significative et avec succès à l'explication ou la prédiction de phénomènes complexes dans les domaines médical et pharmaceutique [31].

Ingénierie électrique

Dans le domaine de l'ingénierie électrique, le Data Mining a été largement utilisés pour la surveillance de l'état du matériel électrique à haute tension. Le but de surveillance de l'état est d'obtenir de précieuses informations par exemple, sur l'état de l'isolation (ou d'autres importantes des paramètres de sécurité) [31].

Aérospatiale

Le Data mining est également intégré aux données spatiales. L'objectif final est de trouver des modèles dans les données relatives à la géographie. Jusqu'à présent, l'exploration de données et de systèmes d'information géographiques ont existé en tant que deux technologies distinctes, chacune avec ses propres méthodes. L'immense explosion de données géo-référencées occasionnée par l'évolution de l'informatique, la cartographie numérique, la télédétection et la diffusion mondiale des systèmes d'information géographiques mettent l'accent sur l'importance de développer une analyse et une modélisation géographique plus fines[31].

1.6 Tâches du DataMining

Le DataMining est utilisé pour accomplir plusieurs tâches :

1.6.1 Association

Cette fonction du datamining permet de découvrir quelles variables vont ensemble, quelles sont les règles qui vont permettre de quantifier les relations entre deux ou plusieurs variables. Par exemple, si l'on s'intéresse à 500 clients qui viennent faire leurs courses au supermarché le vendredi soir et que l'on constate que sur ces 500 clients, 100 achètent des fruits et que sur ce nombre, 30 achètent du lait, ainsi, la règle d'association est « Si L'on achète des fruits, alors on achète du lait », avec une mesure de support de 100/500 = 20% et un seuil de confiance de 30/100 = 33% [22].

1.6.2 Prédiction

La prédiction est la même que la classification et l'estimation, à part que dans la prédiction les enregistrements sont classés suivant des critères (ou des valeurs) prédites (Estimées). La principale raison qui différencie la prédiction de la classification et l'estimation est que dans la création du modèle prédictif on pendent charge la relation temporelle entre les variables d'entrée et les variables de sortie quelques exemples de l'utilisation des tâches de prédiction dans les domaines de Recherche et commerce sont les suivants :

- Prévoir le prix des actions dans les trois prochains mois
- Prévoir le champion de la coupe du monde en football en se basant sur la Comparaison des statistiques des équipes[22].

1.6.3 Segmentation (analyse des clusters)

Ici, la variable cible n'est pas numérique, mais catégorielle, comme par exemple le revenu, qui peut être divisé en trois catégories : faible revenu, revenu moyen et revenu élevé. « La segmentation consiste à répartir les clients en groupes homogènes, qu'il Convient

ensuite d'aborder par des moyens spécifiques et adaptés aux caractéristiques et attentes de chaque groupe. Les membres d'un même groupe réagissent de la même manière aux stimuli marketing. Ils ont en commun un mode de communication, des Comportements d'achat et/ou des besoins spécifiques» [22].

1.6.4 Classification

La classification est différente de la segmentation en ce sens qu'il n'y a pas de variable cible pour segmenter. La classification va s'intéresser au regroupement de données ou d'observations en groupes d'objets similaires. En d'autres termes, elle va segmenter l'ensemble des données afin de former des sous-groupes homogènes. Ceux-ci s'appellent des clusters, à savoir Des classes, qui sont des groupes dans lesquels les données sont similaires entre elles et, par définition différente des autres groupes [22].

1.6.5 Description

L'importance de cette tâche est de permettre à l'analyste d'interpréter les résultats d'un modèle de data mining, soit d'un algorithme, de manière la plus transparente et efficace possible. « Ainsi, les résultats du modèle de datamining doivent décrire des caractéristiques claires qui puissent amener à une interprétation êta une explication intuitive. certaines méthodes de datamining sont plus adaptées que d'autres pour une Interprétation transparente » [22].

1.7 Techniques du DataMining

Il existe plusieurs techniques de DataMining :

1.7.1 Arbres de décision

Ce sont des outils très puissants principalement utilisés pour La classification, la description ou l'estimation. Il s'agit d'une représentation graphique Sous forme d'arbre de décision représentant un enchaînement hiérarchique de règles logiques qui permettent de diviser la base d'exemples en sous-groupes [23].

- Exemple
  - Climat = ensoleillé)^ (Humidité = normale)) -> (Sortir = OUI).
  - (Climat = ensoleillé)^ (Humidité = forte)) -> (Sortir = NON).
  - (Climat = pluie) ^ (Vent = faible)) -> (Sortir = OUI).
  - (Climat = couvert) -> (Sortir = OUI).

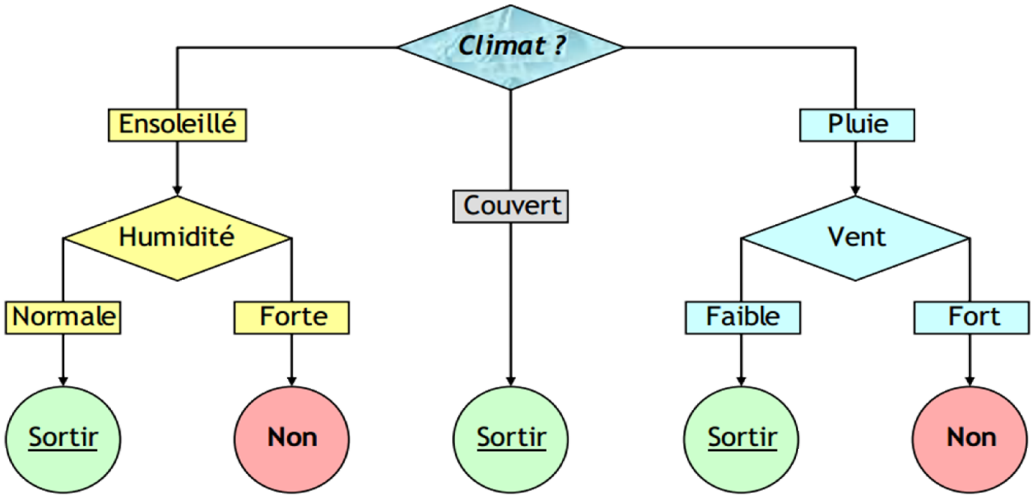

Figure1.2 Exemple d'un arbre de décision [26].

L'existe un certain nombre d'algorithmes de Data mining qui partagent cette qualité de compréhension : les arbres de classification et de régression (CART)et les CHi−squared automatique Interaction Détection (CHAID) qui sont parmi les plus répandus .

1.7.2 Réseaux de neurones

Les réseaux de neurones proposent une simulation du fonctionnement de la cellule nerveuse à l'aide d'un automate le neurone formel.Les réseaux neuronaux sont constitués d'un ensemble de neurones (nœuds) connectés entre eux par des liens qui Permettent de propager les signaux de neurone à neurone[28].

- A apprentissage supervisé :
- lorsque l'on force le réseau à converger vers un état final précis en même temps qu'on lui présente un motif d'apprentissage.
- A apprentissage non supervisé :

Le réseau est laissé libre de converger vers n'importe quel état final lorsqu'on lui présente un motif d'apprentissage. Le principal désavantage de cette technique est qu'un réseau est défini par une architecture et un très grand ensemble de paramètres réels (Coefficients synaptiques, voir figure1.3).

De plus, les modèles qui en découlent sont très sensibles au format des données et donc souvent assez difficiles à comprendre: on parle parfois de boîte noire. [26].

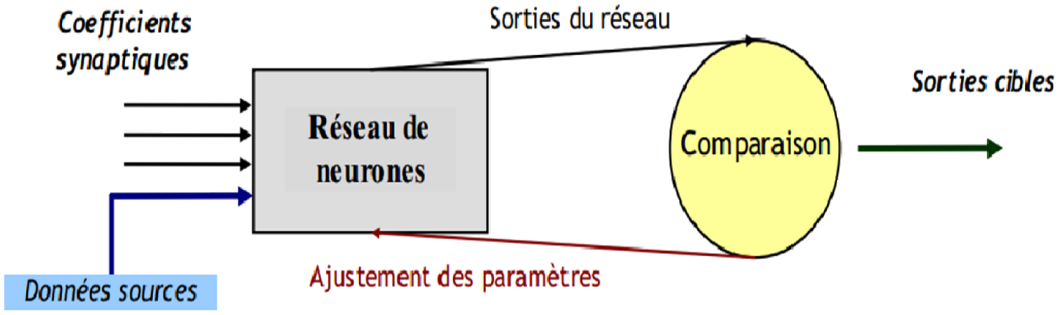

Figure1.3 Schéma de principe d'un réseau de neurones [26].

1.7.3 Algorithmes génétiques

Les algorithmes génétiques sont relativement récents par rapport aux autres concepts d'acquisition de la connaissance. Ils ont été introduits par John Holland en 1975, avec la présentation d'une méthode d'optimisation inspirée de l'observation des Capacités.

1.7.4 Règles d'association

Dans le domaine du DataMining, la recherche des règles d'association est une méthode populaire étudiée d'une manière approfondie dont le but est de découvrir des relations ayant un intérêt pour le statisticien entre deux ou plusieurs variables stockées dans de très importantes bases de données.

Piat et sky-Shapiro présentent des règles d'association extrêmement fortes découvertes dans des bases de données en utilisant différentes mesures d'intérêt. En se basant sur le concept de relations fortes, Rakesh Agrawal et son équipe2 présente des règles d'association dont le but est de découvrir des similitudes entre des produits dans des données saisies sur une grande échelle dans les systèmes informatiques des points de ventes des chaînes de supermarchés. Par exemple, une règle découverte dans les données de ventes dans un supermarché pourrait indiquer qu'un client achetant des oignons et des pommes de terre simultanément, serait susceptible d'acheter un hamburger. Une telle information peut être utilisée comme base pour prendre des décisions marketing telles que par exemple des promotions ou des emplacements bien choisis pour les produits associés. En plus des exemples ci-dessus concernant le panier de la ménagère, les règles d'association sont employées aujourd'hui dans plusieurs domaines incluant celui de la fouille du web, de la détection d'intrusion et de la bio- informatique[31] .

1.7.5 Plus proches voisins

Technique qui repose sur deux concepts, le voisinage d'un individu et la distance entre deux individus. le voisinage d'un individu est déterminé par la fonction de distance d et un seuil minimal € au-dessus duquel On sort du voisinage. La densité du voisinage d'un individu est le nombre de voisins qui l'entourent. Deux points sont dans un même voisinage si la distance qui les sépare est inférieure au seuil E. Un voisinage Est dit dense s'il contient plus qu'un minimum k d'individus.

- Vision (A,B) <:::> distance (A,B) S €
- Dense (A) <:::> nombre (voisins(A)) ~ k

Pour déterminer les groupes, la méthode commence par choisir aléatoirement un point dense. Tous les points qui sont atteignables à partir de ce point, selon le seuil de densité établi, forment un groupe. Les points atteignables sont les voisins directs, les voisins des voisins et ainsi de suite. Le choix de la fonction de distance d, du seuil € et du nombre k de voisins qui déterminera la densité d'un voisinage sont des choix qui détermineront le nombre de groupes et leurs compositions.

Bien que cet Algorithme. Soit simple à implémenter et donne relativement de bons résultats, il est peu utilisé à cause des temps de calcul et son inadaptation aux grandes bases de données.

1.7.6 Motifs séquentiels

Les motifs séquentiels ont initialement été et reposent sur la notion de séquence fréquente maximale .Considérons une base de données DB d'achats pour un ensemble O d'objets chaque enregistrement R correspond à un triplet (id-objet, id-date, itemset) qui caractérise l'objet auquel est rattaché l'enregistrement, ainsi que la date et les items correspondants. Soit I=$\{i1 \dots i2 \dots\dots im\}$ l'ensemble des items de la base. Un itemset est un ensemble non vide et non ordonné d'items, noté $(I_1, I_2 .. I_k)$, où i, j un item . Une séquence s se définit alors comme une liste ordonnée non vide d'itemsets qui sera notée $< S_1 \dots S_j >$ où $S_j$est un itemset. Une n-séquence est une séquence de taille n, c'est-à-dire composée de n item [3]**.**

- Exemples de séquence :
  - $\{Achat\ "Ordinateur"\}\{Achat\ "Imprimante", dans\ 3\ mois\} >$ [70%].
  - $\{(Traitement\ X) < (Traitement\ Y)\}\ \{Effet\ E, dans\ 15\ jours\} >$ [50%].
  - $\{Défaut\ A\}\{Panne\ P1 Dans\ 15\ jours\}\{Panne\ P2\ Dans\ 20\ jours\} >$ [98%].
  - $\{Prêt\ "Livre\ 2"\}\ \{Prêt\ "Livre\ 5"\}\ \{Prêt\ "Livre\ 1"\} >$ [20%].
  - $\{"Google.dz"\}\ \{"Recherche\ Algérie"\}\ \{"Algérie - monde.com\ «\} >$ [30%].

Grâce à la prise en compte de la temporalité, les motifs séquentiels présentent des règles plus Précises et plus utiles pour une diversité d'applications actuelles, allant de la prédiction des chemins de navigation des utilisateurs Web et la détection d'enchaînements de transactions Financières inhabituelles jusqu'à l'analyse des séquences ADN en passant par la maintenance industrielle anticipée. La recherche de motifs séquentiels est une technique non dirigée de Data mining car on ne dispose en entrée que des données d'analyse [26].

1.7.7 Analyse de liens

L'analyse de liens est un outil de data mining qui cherche à découvrir des relations qui peuvent se tisser entre les différents objets d'une population donnée, par exemple des consommateurs, en vue de dégager des catégories qui pourraient être étudiées séparément. Cette technique est généralement basée sur la théorie des graphes. Un graphe est une représentation n intuitive constituée de nœuds et d'arrêtes. Les nœuds symbolisent les objets qui sont en relation et les arrêtes évoquent les relations entre paires d'objets l'analyse de liens est très souvent suivie d'une analyse plus poussée à l'aide d'autres outils comme les arbres de décision ou les réseaux neuronaux artificiels. Il reste à mention er que c'est une technique non dirigée de data mining [26].

1.8 Conclusion

Le data mining est l'extraction d'informations prédictives cachés dans de grandes base de données.C'estune technologie nouvelle et puissante qui donne la possibilité aux entreprises de se concentrer sur les informations les plus importantes dans leurs data workhouses. Les outils du data mining peuvent prédire les futurs tendances et actions permettant de prendre les bonnes décisions. C'est ce qui rend le data mining la technologie la plus importantes. Le chapitre suivant : prendra en détail le technique de motifs séquentiel commençant par les concepts de base des règles d'association car elles en sont à l'origine.

CHAPITRE 2

REGLES D'ASSOCIATION ET MOTIFS SEQUENTIELS

2.1 Introduction

Les motifs séquentiels peuvent être vus comme une extension de la notion de règles d'association intégrant diverses contraintes temporelles. La recherche de tels motifs consiste ainsi à extraire des enchaînements d'ensembles d'items, couramment associés sur une période de temps bien spécifiée. En fait, cette recherche met en évidence des associations Inter-transactions, contrairement à celle des règles d'association qui extrait des combinaisons intra-transactions. Par exemple, des motifs séquentiels peuvent montrer que"60% des gens qui achètent une télévision, achètent un magnétoscope dans les deux ans qui suivent".

Ce problème, posé à l'origine dans un contexte de marketing, intéresse à présent des domaines aussi variés que les télécommunications (détection de Fraudes), la finance, ou encore la médecine (identification des symptômes précédant les maladies). Les relations entre ces items sont qualifiées de règles d'association séquentielles ou motifs séquentiels.

Les motifs séquentiels permettent de définir des règles plus précises et donc plus utiles et trouvent à l'heure actuelle de larges champs d'application allant de l'analyse des séquences ADN.

2.2 Règles d'association

L'extraction des règles d'association a pour but de découvrir des relations significatives entre attributs binaires extraits BDD. Un exemple de règle d'association extraite d'une base de données de ventes de supermarché est : « café sucre →lait (support utilisées avec succès dans de nombreux domaines, parmi lesquels l'aide à la planification commerciale, l'aide au diagnostic et en recherche médicale, l'amélioration des processus de télécommunications, l'organisation et l'accès aux sites Internet, et 7%, confiance 50%) ». Cette règle indique que les clients qui achètent des céréales et du sucre ont également tendance à acheter du lait. La mesure de support définit la portée de la règle, c'est à dire la proportion de clients qui ont acheté les trois articles, et la mesure de confiance définit la précision de la règle, c'est à dire la proportion de clients qui ont acheté du lait parmi ceux qui ont acheté des céréales et du sucre. L'extraction de règles d'association consiste à extraire les règles dont le support et la confiance sont au moins égaux à des seuils minimaux de support et de confiance définis par l'utilisateur.

Les règles d'association ont été L'analyse d'images de données spatiales, de données géographiques et de données statistiques**.**

- Exemples de règles :
  - $\{Achat\ "Télévision"\} -> \{Achat\ "Décodeur"\}$ [69%].
  - $\{(Traitement\ X)\ ^\wedge\ (Traitement\ Y)\} -> \{Effet\ E\}$ [50 %].
  - $\{(Maladie\ M)\ ^\wedge (Traitement\ X)\} -> \{Guérison\}$ [97%].
  - $\{(Défaut\ A)^\wedge(Défaut\ B)\} -> \{Panne\ P\}$ [65%].
  - $\{Panne\ P\}^\wedge\{(Panne\ P1)\ ^\wedge(Panne\ P2) -> (Panne\ P3)\}$ [90%].

Ces exemples expliquent le grand avantage des règles d'association : leur pouvoir explicatif. En effet, et contrairement à d'autres méthodes d'analyse, elles fournissent des réponses simples et sont facilement interprétables quel que soit le degré de complexité de la règle. C'est pourquoi, elles représentent un sujet de recherche très attractif. L'objectif principal de cette technique est descriptif, toutefois, et dans la mesure où les résultats peuvent être utilisés dans le futur, il pourra être considéré comme prédictif.

La recherche de règles d'association est une technique non dirigée car on ne dispose en entrée que des données d'analyse [27].

2.2.1 Domaines d'application

Les règles d'associations sont appliquées dans différents domaines. On peut citer par exemple :

- Recherche médicale
- Internet.
- Applications spatiales**.**
- Réseaux de télécommunication
- Maintenance industrielle
- Planification commerciale

2.2.2 Concepts généraux

Une règle d'association peut être vue comme une relation d'occurrence simultanée qui lie deux ou plusieurs items particuliers. Les concepts ou termes souvent employés dans ce contexte peuvent être définis comme suit.

Item :

On appelle un item est un champ et chaque instance de bd est un item un ensemble fini d'éléments distinct $I = \{i1, i2, \dots, i_n\}$.

- Exemple :

Dans une application de vents les articles des vents sont des items

Itemset :

On appelle un itemset est un sous-ensemble des item Un item set constitué de k items est un k −itemset.

- Exemple :

Itemset $\{A, B, C, D\}$ est un 4− itemset représentant les articles : Café, Sucre, pain et lait. Ces quater articles ont pu ou non être achetés ensemble lors d'une même transaction.

Transaction :

Est un la relation représentant l'ensemble des items de la transaction.et composer par $(tID, itemset)$.

Base de transactions :

On Apple une base de transactions D peut être vue comme un ensemble de transactions (TID, Itemset).

$D = \{T : (TID, itemset)/item\ set = \{a\ appertain\ I\}\}$

- Exemple:

TID	Itemset
1	{A, C,D}
2	{B, C, E}
3	{A, B, C,E}
4	{B, E}
5	{A, B, C,E}
6	{B,C, E}

Table2.1 : Base de données à six transactions

Ce tableau contient un BDD représente une base de transactions constituée de 6 transactions. Chaque transaction peut être vue comme un itemset identifié parle numéro de la transaction, regroupant l'ensemble des items de cette transaction.

Support d'un Itemset :

Le support d'un item est égal au nombre de transactions dans lesquelles il apparaît.

Par exemple, le support de $\{S1\}$ est égal à 5. Le support peut être exprimé également en termes relatifs. Dans ce cas, nous division le support (absolu) par le nombre total de transactions. Pour S1, nous avons $SUP\ (\{S1\}) = 5/10 = 20\%$.[5]

Itemset fréquent

On applet un itemset fréquent dont le support est supérius fixé seuil minsup si support n'est pas atteint on dit que l'itemset est infréquenté.

- Exemple :

Considérons la base de transactions représentée par le Tableau. Pour un seuil de support minimum $Minsup = 2$, on aura l'itemset $\{A, C, E\}$ fréquent dans cette base. En effet, son support Sup $(\{A, C, E\}) = 2 \geq Minsup$. Ceci n'est pas le cas pour $\{C, D\}$ qui est de support 1

Une règle d'association :

On dit une Règle d'association compose ce la forme $X -> Y$ $[Support, Confiance]$ avec x et y deux itemset .Tel que $x \cap y = 0$.

- Support : le support d'un Règle d'association contenant tous les items de l'itemset xUy ce la forme Sup $(Y -> X) = Sup\ (xUy)$.
- Confiance : La confiance d'une règle est une mesure dite de précision, c'est la probabilité qu'on achète un certain nombre d'articles A sachant qu'on a déjà acheté B soit la probabilité conditionnelle $p(A/B)$ $conf(ae -> \{/bc\}) = sup(\{abce\})/sup(\{ae\})$.

On voit immédiatement que la confiance se traduit par un rapport de support. La notion de confiance serra bien détaillée dans la partie mise en œuvre.

Il existe d'autres critères d'évaluation utilisant différentes formules comme, L'Intérêt, la conviction, et le Cosinus.[25].

- Exemple **:**

Reprenons BDD à six transactions du Tableau et considérons la règle d'association $Café -> Sucre$. Son support est donné par la proportion dans cette base, de clients qui ont acheté les deux articles Café et Sucre simultanément (c'est- à- dire, dans la même transaction), ce qui est égal à $2/6$. La mesure de support définit donc, la portée de la règle sa confiance est donnée par la proportion dans cette base, de clients qui ont acheté l'article

sucre parmi ceux qui ont acheté l'article Café, ce qui est égal à 1/15. La mesure de confiance définit donc, la précision de la règle.

Règle d'association valide :

On Apple dit un règle d'association est dite « valide » si son support et sa confiance sont supérieurs ou égaux à deux seuils, minsup et minconf respectivement définir par :

$X -> Y$ *règle valide* $< -> Sup\ (X -> Y) \geq\ minsup\text{^}conf\ (X -> Y\ )\ \geq minconf$[32].

2.2.3 Recherche de règles d'association

La méthode de recherche des règles d'association est née pour analyser les articles fréquemment achetés ensemble dans les supermarchés. Chaque sortie de caisse correspond à une transaction où plusieurs items ont été achetés simultanément. Une règle d'association est une implication $A \rightarrow C$où l'antécédent A et le conséquent C sont des ensembles d'items, où $A \cap C\ = \theta$ Une règle repose sur les notions de support et de confiance. Ilya un plusieurs algorithmes de recherche de règles d'association. Ces algorithmes procèdent en deux phases :

- La phase de recherche et extraction de l'ensemble des itemsets fréquents.
- La phase de génération des règles d'association à partir de cet ensemble [24].

Extraction des itemset fréquents :

Le problème de recherche des itemset fréquents est un problème non trivial par ce que le nombre d'itemset potentiellement fréquents est exponentiel par rapport au nombre d'items considérés dans la base de données. La phase de recherche de ces itemset fréquents est la phase la plus coûteuse de l'extraction de règles d'association du fait de la taille exponentielle de l'espace de recherche et du nombre élevé nécessaire de balayages complets du jeu de données[22].

Génération des règles d'association :

A partir de l'ensemble des itemsets fréquents pour un seuil minimal de support minsup, la génération des règles d'association est un problème qui dépend exponentiellement de la taille de l'ensemble des itemsets fréquents.et le cou de temps de cette phase est très faible par rapport au coût d'extraction des itemsets fréquents.

2.2.4 Algorithmes d'extraction de règles d'association

Dans Algorithmes d'extraction de règles d'association cita dire le temps de répéter des algorithmes d'extraction des règles d'association dépendent essentiellement des temps d'extraction des itemsets fréquents. Plusieurs balayages de la base de données doivent être

réalisés en comptant pour chaque itemset candidat le nombre de transactions de la base dans lesquelles il est contenu. Le nombre d'itemsets candidats et la taille des jeux de données étant généralement importants, de nombreuses approches visant à minimiser le nombre d'itemsets candidats et le nombre de ces balayages sont proposées [26].

2.2.4.1 Algorithmes d'extraction des itemsets fréquents

Les algorithmes d'extraction des itemsets fréquents par niveaux considèrent un ensemble d'itemsets d'une taille donnée lors de chaque itération, c'est à dire un ensemble d'itemsets d'un« niveau » du treillis des itemsets. Ces algorithmes se basent sur les propriétés suivantes afin de limiter le nombre d'itemsets candidats considérés, en les générant à partir des itemsets fréquents de l'itération précédente : tous les sur-ensembles d'un itemset in fréquents sont in fréquents et tous les sous-ensembles d'un itemset fréquent sont fréquents.

Parmi ceux-ci nous pouvons citer les algorithmes Apriori qui réalisent un nombre de balayages du contexte égal à la taille des plus longs itemsets fréquents, l'algorithme Partition qui autorise la parallélisassions du processus d'extraction, et l'algorithme DIC qui réduit le nombre de balayages du contexte en considérant les itemsets de plusieurs tailles différentes lors de chaque itération. Les algorithmes Partition et DIC entraînent un coût supplémentaire en temps CPU par rapport aux algorithmes Apriori et OCD dû à l'augmentation du nombre d'itemsets candidats testés.[29]

2.2.4.2 Algorithmes d'extraction des itemsets fréquents Maximaux

Ces algorithmes sont basés sur la propriété que les itemsets fréquents maximaux, c'est à dire les itemsets dont tous les sur-ensembles sont infréquentés, forment une bordure au-dessous de laquelle tous les itemsets sont fréquents. L'extraction des itemsets fréquents maximaux est réalisée par une exploration itérative du treillis des itemsets fréquents, en « avançant » de un niveau du bas vers le haut et de un ou plusieurs niveaux du haut vers le bas lors de chaque itération. À partir des itemsets fréquents maximaux, tous les itemsets fréquents sont dérivés et leurs supports sont déterminés en réalisant un balayage du contexte. Quatre algorithmes basés sur cette approche ont été proposés, ce sont les algorithmes Pincer-Search Max Clique et Max Eclat et Max-Miner Ces algorithmes permettent de réduire le nombre d'itérations, et donc de diminuer le nombre de balayages du contexte et d'opérations CPU réalisés [29].

2.3 Motifs séquentiels

La problématique de l'extraction de motifs séquentiels peut être perçue comme une extension de celle de l'extraction de règles d'association.

Effet la prise en compte de la temporalité dans les enregistrements à étudier permet une plus grande précision dans les résultats, mais implique aussi un plus grand nombre de calculs et de contraintes elles représentent une e extension des règles d'association à deux niveaux les BDD manipulées par les motifs séquentiels présentent la particularité de comporter pour chaque transaction, en plus de l'identifiant unique TID de la transaction et de l'ensemble des item s de cette transaction, un Identifiant Temporel Cet identifiant permet à la base de données d'enregistrer le temps valide pour chaque nouvelle transaction Insérée. Par exemple, dans une base de données relative aux opérations de vente d'un supermarché chaque transaction inclut un attribut supplémentaire qui indique quand la transaction s'est produite (par exemple, le mois, la date, l'heure, …etc.).

Les motifs séquentiels permettent d'analyser l'ordre d'apparition d'items ou de groupes d'items dans une base de données selon cet identifiant temporel. Ainsi, une règle d'association séquentielle permet par exemple, d'identifier une relation de La forme : 80% clients achètent du Sucre après avoir acheté du Café".

Ce type de connaissances n'était pas envisageable de découvrir sans la considération de cette nouvelle composante temporelle, Absente dans la technique des règles d'association classiques [8].

2.3.1 Motifs séquentiels vs règles d'association

Elaborés en premiers lieux, les algorithmes de recherche de règles d'association connaissent de grandes difficultés d'adaptation aux problèmes d'extraction de motifs séquentiels. En effet, si le problème de la recherche de règles d'association est proche de celui des motifs séquentiels, les études dans ce sens montrent que, lorsque l'adaptation est possible, c'est au prix de temps de réponse inacceptables. Depuis la définition du problème dans de nombreuses approches destinées à résoudre la problématique de l'extraction de motifs séquentiels ont été proposées [8].

2.3.2 Champs d'application plus étendus

Les motifs séquentiels utiliser dans plus champs et attractifs pour une diversité d'applications actuelles :

- Planification commerciale :
- . Recherche génomique
- Banques et Assurances
- Recherche médicale et Réseaux de télécommunication
- Maintenance industrielle
- Applications Internet et Analyse de données spatiales

2.3.3 Notions fondamentaux

2.3.3.1 Transaction

On appelle une transaction pour un client C est un triple (CID, Date, Itemset) formé de l'identificateur unique du client, de la valeur de l'identifiant temporel pour cette transaction et de l'ensemble des items de la transaction, représentant tous les items achetés par C à cette même date.

2.3.3.2 Base de données temporelle

On appelle une base de données temporelle D est un ensemble de transactions $(CID, Date, Itemset)$.Avec $D = \{T : (TID, Date, Itemset)\ tellque\ \ Itemset = \{a \in I\}\}$.

- Exemple:

CID	DATE	ITEMSET
C1	01/01/2008	{B, F}
	02/01/2008	{B}
	04/01/2008	{C}
	18/01/2008	{H, I}
C2	11/01/2008	{A}
	12/01/2008	{C}
	29/01/2008	{D, F, G}
C3	05/01/2008	{C, E, G}
	12/02/2008	{A, B}
	18/02/2008	{I}
C4	06/02/2008	{B, C}
	07/02/2008	{D, G}

Table 2.2 : Base de données temporelle ordonnée Par CID et par date de transaction.

Ce tableau représente une base de données temporelle ordonnée selon L'identifiant unique du client CID et l'identifiant temporel Date de cette base. La transaction effectuée par le client C1à la date du $18/01/2008$ est vue comme un triplet $(C1\ 18/01/2008, \{H, I\})$.Les deux items H et I étant été achetés lors de cette même transaction.

2.3.3.3 Séquence de données

On appelle une séquence de données est une liste ; ordonne non vide d'itemsets Si, appelé $S = < S_1; S_2 \dots ; S_n > avec, i \in [1 \dots n]$ Indique l'ordre d'apparition de si dans S.

- Exemple **:**

Reprenons la BDD du Table 2.2. La liste ordonnée des trois transactions effectuées par le client $C2$ est donnée par la $séquence < \{A\}\{C\}\{D, F, G\} > avec : S_1 = \{A\}, = \{C\}$ et $s3 = \{D, F, G\}$. Cette séquence se lit de la manière suivante : le client C2 a acheté l'article A, puis l'article C, puis simultanément les trois articles D, F et G.

2.3.3.4 Longueur d'une séquence

La longueur d'une séquence S est le nombre d'un d'items dans cette séquence. Une séquence de longueur k est une $k - séquence$.

- Exemple :

$Séquence < \{A\}\{C\}\{D\}\{C, E\} >$ est une $5 - séquence$, même si cette dernière contient seulement 4 $itemsets$. L'item C est contenu dans deux transactions et est donc compté 2 fois.

2.4 Algorithme général d'extraction de motifs séquentiels

L'algorithme APRIORI et ses diverses alternatives les chercheurs dans le domaine du data mining se sont vite orientés vers l'application du même principe à l'extraction de motifs séquentiels. Toutefois, il s'est avéré que le nouveau problème est beaucoup plus complexe, car l'ordre des items, comme nous l'avons évoqué tout au long de ce chapitre, est dans ce cas une propriété fondamentale l'inverse des ensembles fréquents et par conséquent, plusieurs optimisations identifiées pour traiter ces ensembles ne se transposent pas aux séquences d'items. le but de L'algorithme APRIORI est minimiser les candidats.

Le Principe de cet algorithme se divise en deux étapes : une étape de jointure et une autre d'élagage [25].

2.4.1 L'algorithme APRIORI

$Input : Ck: itemsets\ candidats\ de\ taille\ k.$

$Output : Lk: itemsets\ fréquents\ de\ taille\ k$

$L1 = \{items\ fréquents\};$

$for\ (k = 1;\ Lk! = \emptyset;\ k++)\ do$

$Ck + 1 =\ candidats\ générés\ à\ partir\ de\ Lk\ ;$

$Pour\ chaque\ transaction\ t\ de\ la\ base\ de\ données, incrémenter\ lecompteur\ de\ tous\ les\ candidat$

$+ 1 qui\ sont\ contenus\ dans\ t.$

$Lk + 1 =\ candidats\ dans\ Ck + 1 avec\ MinSupp\ .$

$Return \cup k\ Lk.$

2.4.2 Détails de l'algorithme APRIORII

- Générer les candidats :
  - Etape : auto-jointure sur Lk
  - Etape 2: élagage
- compter le support des candidats.
- Les items de $Lk - 1$ sont ordonnés par ordre lexicographique.
  - Etape 1:

$Auto - jointure\ sur\ Lk - 1$

$INSERT\ INTO\ Ck$

$SELECT\ p.item1, p.item2 \ldots\ p.itemk - 1, q.itemk - 1$

$FROM\ Lk - 1\ p, Lk - 1\ q$

$WHERE\ p.item1 = q.item1 P_i temk - 2 = q.itemk - 2, p.itemk - 1 < q.itemk - 1$

  - Etape 2: élagage

$For\ each\ itemset\ c\ in\ \ C_I\ do$

$For\ each\ (k - 1) - subsets\ s\ of\ c\ do\ if (s\ is\ not\ in\ Lk - 1)\ then\ delete\ c\ from\ Ck.$

- Exemple de Génération des candidats :

$L3 = \{abc, abd, acd, ace, bcd\}$

- auto-jointure : L3*L3
  - $abcd$ à partir de $abc\ et\ abd$
  - $acde$ à partir de $acd\ et\ ace$
- Élagage :
  - $acde$ est supprimé car ade n'est pas dans L3

- $C4 = \{abcd\}$[27] .

- Exemple application

Un grand marché dépasse les données sur les ventes par l'unité de conservation des stocks (SKU) pour chaque article, et est donc capable de savoir quels articles sont habituellement achetés ensemble. Apriori est un moyen modérément efficace d'établir une liste de paires fréquentes d'articles achetés à partir de ces données. Laissez la base de données des transactions se composer des ensembles {1,2,3,4},{1,2,3,4,5}, {2,3,4}, {2,3,5}, {1,2,4}, {1,3,4}, {2,3,4 , 5}, {1,3,4,5},{3,4,5},{1,2,3,5}. Chaque numéro Correspond à un produit tel que "beurre" ou "eau". La première étape d'Apriori consiste à compter les fréquences, Appelé les supports, de chaque élément membre séparément [9].

Item	Support
1	**6**
2	**7**
3	**9**
4	**8**
5	**6**

Table 2.3 Base de séquences de données

Item	Support
{1,2}	**4**
{1,3}	**5**
{1,4}	**5**
{1,5}	**3**
{2,3}	**6**
{2,4}	**5**
{2,5}	**4**

Table 2.4: représenter Itération 2

Nous pouvons définir un niveau minimum de support pour être qualifié de "fréquent", qui dépend du contexte. Pour ce cas, laissez le support min. = 4.Par conséquent, tous sont fréquents. L'étape suivante consiste à générer une liste de toutes les 2 paires des articles fréquents. Si l'un des éléments ci-dessus n'avait pas été fréquent, ils n'auraient pas été inclus en tant que Membre possible des paires possibles de 2 éléments. De cette façon, Apriori érable l'arbre de tous les ensembles possibles. À la prochaine étape, nous ne sélectionnons plus que ces éléments (maintenant les 2 paires sont des éléments) fréquentes (les paires écrites en gras texte): [9]

Nous générons la liste des 3 triples des éléments fréquents (en connectant une paire fréquente avec fréquents Un seul article).:[9]

Item	Support
{1, 3,4}	**4**
{2, 3,4}	**4**
{2, 3,5}	**4**
{3, 4,5}	**4**

Table 2.5 représenter Itération 3

L'algorithme se terminera ici parce que la paire {2, 3,4,5} générée à l'étape suivante n'a pas la soutien souhaité[9].

2.4.3 Discussion

L'algorithmes apriori est une algorithme complexe qui peut être très coûteuse en temps UC et en espace mémoire dans le cas d'une base de données très dense ou d'un nombre d'items très élevé. Car parcours et jointer plusieurs fois la base de donnée.

2.5 Conclusion

Nous avons au chapitre ouvre la voie aux règles de l'Association et nous l'avons mentionné les règles les plus importantes pour eux-mêmes et ensuite nous avons parlé à des modèles les plus importants motifs séquentiels de a nous l'avons mentionné dans ce chapitre, l'algorithme le plus important dans l'exploration de données et apparoir gen, que ce soit dans le chapitre suivant, nous allons parler de la plus importante spécialisée dans ce domaine l'algorithme et donner un résumé de chacun d'entre eux et Ces facteurs seront sans doute, les éléments clés que nous considérerons dans le notamment, lorsque nous savons maintenant que notre problème est fortement lié à un aspect combinatoire très complexe pouvant réduire considérablement , non seulement l'efficacité mais aussi l'utilité de la tâche d'extraction.

CHAPITRE 3

ALGORITHMES D'EXTRACTION DE MOTIFS SEQUENTIELS

3.1 Introduction:

Dans ce chapitre, nous présentons les différentes approches condensées. Tout d'abord, nous nous focaliserons sur les représentations condensées dans le cadre de l'extraction d'itemsets puis nous aborderons les représentations dans le cadre d'extraction des motifs séquentiels.

3.2 Classification les algorithmes

Nous présentons dans cette section une classification des algorithmes en essayant de mettre en avant leurs particularités communes de développant pour chacune d'elles un algorithme choisi comme représentant de cette classification. Nous développons ainsi plus en détails les algorithmes GSP, Spade, préfixant, et spam.

3.2.1 Algorithmes basées sur Apriori

3.2.1.1 Algorithme GSP :

GSP (Generalized Sequential Patterns)est un algorithme basé sur la méthode générer-élaguer mise en place depuis Apriori et destinée à effectuer un nombre de passes raisonnable sur la base de données. La technique généralement utilisée par les algorithmes de la recherche de séquences est basée sur une création de candidats, suivie du test de ces candidats pour confirmer leur fréquence dans la base, en Bénéficiant de propriétés relatives aux séquences et à leur fréquence d'apparition, ces techniques sont tout de même contraintes "d'essayer" des séquences avant de les déterminer fréquentes (ou non**) [**8] .

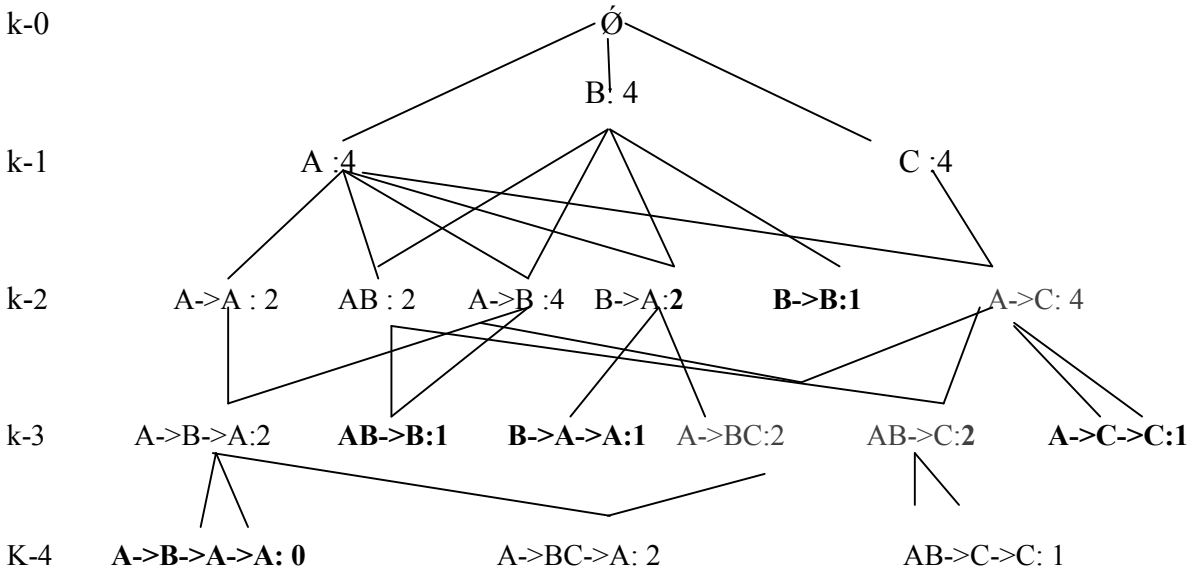

Figure 3.1 Généré lors différents itération de l'algorithme GSP parcours en largeur [21].

3.2.1.1.1Génération de séquences candidatent :

GSP procédé à la génération des séquences candidates en deux phases : Phase d'légale et phase de jointure. Lors de la phase d'légale. GSP aboutit l'ensemble de1-séquence fréquentes L_1qui servira comme pont de départ pour générer les 2- séquences candidates C_2qui constitue l'amorce de la phase de jointure .GSP effectue un balayage de la base de donne et généré les 2-séquencecandidates potentielles ou fréquentes L_2,tout en calculant leurs support L_2 servira d'amorce pour le prochain balayage ,et ainsi de suite. Plus généralement CK ou les K-séquences candidates sont générées en joignant les k-séquence fréquentes (lk+1 avec lui-même).Soient S_2est exactement la mémé séquences fréquentes S_1peut être jointe a'S_2a' laquelle le dernier item de S_2aura été rajouté. L'exemple suivant (tré de Agrawal et la 1993) explique la phase jointure de deux séquences Une séquenceS_1joindra une séquence S2si la sous séquence obtenue en retirant le premier item de S_1est la même que celle obtenue en retirant le dernier item de S_2.La séquence ? Générée notée S = S_1xS_2 n'est autre que S1 à laquelle a été ajouté à la fin le dernier item de S_2qui devient séparé dans S s'il l'était dansS_2, sinon, il est ajouté au dernier itemset de S_1 [26].

- Exemple de jointer

$$\frac{\{A,B\}\{C\} \quad < \{A,B\}\{C\} >}{< \{A,B\}\{C,D\} > \quad < \{A,B\}\{C\}\{E\} >}$$

Figure 3.2 Jointure de séquences dans GSP [10].

3.2.1.1.2 AlgorithmeGSP (D,p_s,I)

1. $Algorithme\ GSP(D, p_s, I)$
2. $C :-\{< \{i\} > | i \in I\}$
3. $Scan\ to\ get\ support\ of\ very\ sequence\ in\ C_1$
4. $L := \{s\ \varepsilon\ \in C_1, sup(s) => p_s\}$
5. $I = 1;$
6. $I := 1; while(L \neq \emptyset)$
7. $C_{i+1} := GSP\_Gen\ (L_i)$
8. $Scan\ to\ get\ support\ of\ very\ sequence\ in\ C_{i+1}$
9. $L_{i+1} :=\ \{s\ \varepsilon\ \in C_{i+1}, sup(s) => p_s\}$
10. $i = i + 1$
11. $ReturnL_1, \cup\ L_2\ \cup, \ldots L_i$

Figure 3.3 Algorithme GSP [1] .

3.2.1.1.3 Calcul des supports:

L'algorithme GSP augmente de 1 le support de la séquence s pour chaque séquence Client contenant au moins une fois la séquence s en question. Dans le cas où aucune fenêtre d'événement ni max-gap min-gap ont été définis, l'algorithme ne fait que calculer le nombre de séquences clients qui contiennent les séquences candidates dont les supports doivent être déterminés. Par contre si un ou plusieurs de ces paramètres sont définis, l'algorithme doit s'assurer que les séquences respectent les contraintes imposées par les paramètres avant d'en augmenter le support [25].

- Exemple :

Candidats initiaux: toutes les séquences singleton $< a >, < b >, < c >, < d >, < e >$, <f>, <g>,<h>Scan base de données une fois, comptez le soutien pour les candidats [28].

Seq.id	Séquence
10	<(bd)cb(ac)>
20	<(bf)(ce)b(fg)>
30	<(ah)(bf)abf>
40	<(be)(ce)d>
50	<a (bd)bcb(ade)>

Min_sup=2

Cande	Sup
a	3
b	5
c	4
d	3
e	3
f	2
g	1
h	1

Figure 3.4 Représentation des données horizontalement

	<a>	<b>	<c>	<d>	<e>	<f>
A	<aa>	<ab>	<ac>	<ad>	<ae>	<af>
B	<ba>	<bb>	<bc>	<bd>	<be>	<bf>
C	<ca>	<cb>	<cc>	<cd>	<ce>	<cf>
D	<da>	<db>	<dc>	<dd>	<de>	<df>
E	<ea>	<eb>	<ec>	<ed>	<ee>	<ef>
F	<fa>	<fb>	<fc>	<fd>	<fe>	<ff>

Table 3.1 représentation des données horizontalement

	<a>	<b>	<c>	<d>	<e>	<f>
<a>		<ab>	<ac>	<ad>	<ae>	<af>
<b>			<bc>	<bd>	<be>	<bf>
<c>				<cd>	<ce>	<cf>
<d>					<de>	<df>
<e>						<ef>
<f>						

Table 3.2 : Calculer les candidats

Calculer les candidats pour cet exemple : 8*8+8*7/2=92 candidates avec confi= 47 ,57% candidats.

3.2.1.1.4 Les limites de GSP **:**

L'algorithme GSP content trois limites :

- un nombre très considérable de candidates peut être généré dans les grand es basse de donnes .en effet .cette génération produit : $n^2+n(n-1)/2$)candidats ou n est le nombre de 1-séquenece fréquentes plusieurs de cette candidates ne font même pas partie d la basse de donnes.
- Vu que la longueur de chaque séquence candidate augmente d'un item chaque balayage, le nombre de ce dernier va augmenter en parallèle, ce qui constitue <<Un cout non trivial>>.

- les méthodes basées sur Apriori rencontrent souvent des difficultés pour la découverte de longues séquence .Ceci est du fait que le nombre des séquences candidates est Fonction exponentielle de longueur séquences plus coutes qui les forme [10].

3.2.1.2 Algorithme SPADE**:**

SPADE est un nouvel algorithme pour la découverte rapide de modèles séquentiels. Spade, présenté dans (ZAKI M., 2001), se classe dans la catégorie des algorithmes qui cherchent à réduire l'espace des solutions en regroupant les motifs séquentiels par catégorie. Pour Spade, les motifs fréquents présentent des préfixes communs qui permettent de décomposer le problème en sous-problèmes qui seront traités en mémoire. Le calcul de F2 (les fréquents de taille 2) par Spade, passe par une inversion de la base qui la transforme d'un format vertical vers un format horizontal. Il gère les candidats et les séquences fréquentes à l'aide de classes d'équivalence comme suit : deux k-séquences appartiennent à la même classe si elles présentent un suffixe commun de taille (k-1). Plus formellement, soit $Pk\text{-}1(\alpha)$ la séquence de taille $k\text{-}1$ qui préfixe la séquence α.Comme α est fréquente, Pk-1(α) Fk-1 les fréquents de taille k-1. Une classe d'équivalence est définie de la manière suivante: $[\rho\ Fk\text{-}1] = \{\alpha\ Fk | Pk\text{-}1(\alpha) = \rho\}$. Chacune de ces classes d'équivalence contient alors deux types d'éléments, $[\rho.l1] =< \rho\ (x) >$ ou bien $[\rho.l2] =< \rho\ x >$ selon que l'item x appartient ou pas à la même transaction que le dernier item de ρ. Les candidats sont ensuite générés selon trois critères : Auto jointure ([ρ.l1] x [ρ.l1]), Auto jointure ([ρ.l2] x [ρ.l2]) et jointure ([ρ.l1] x [ρ.l2]). Le reste de l'algorithme, à savoir le comptage du support pour les candidats générés, repose sur la réécriture préalable de la base de données. En effet, la transformation consiste associer à chaque k-séquence l''ensemble des couples (Client, itemset) qui lui correspondent dans la base [25].

2.1.2.1 Principe de base:

Le principe de SPADE consiste à n'effectuer, qu'une unique lecture de la base de séquences de données afin de représenter celle- ci en mémoire central e sous la forme de listes d'occurrences de séquences, en vue que tous ses traitement s ultérieurs s'effectueront sur ces listes.[26].

3.2.1.2.2 Occurrence d'une séquence :

Occurrence d'une séquence S' dans une séquence de données S est définie par un triplet (CID, Début, Fin), où CID représente l'identifiant unique de la séquence de données S,

alors que " Début" et " Fin" représentent l'information de localisation temporelle de S' dans [26].

- Exemple :

Cet exemple à expliquer algorithme du algorithme SPADE et appliqué à la Base de séquences de données D' de la figure 3.5, avec un seuil de support minimum de 2/5.

D'	**1**	**2**	**3**	**4**
C1	{A}	{B}	{B}	{C}
C2	{C}	{A}	{B}	
C3	{A}	{C}	{B}	
C4	{C}	{A}	{B}	{A}
C5	{B}	{A}	{D}	{A}

L'unique parcours de D'

ID−LIST = A		
CID	Début	Fin
C1	1	1
C2	2	2
C3	1	1
C4	2	2
C4	4	4
C5	2	2
C5	4	4

D−List = B		
CID	Début	Fin
C1	2	2
C1	3	3
C2	3	3
C3	3	3
C4	3	3
C5	1	1

ID−LIST = C		
CID	Début	Fin
C1	4	4
C2	1	1
C3	2	2
C4	1	1

ID−LIST = D		
CID	Début	Fin
C5	3	3

Figure 3.5 Représentation SPADE en listes d'occurrences D'items d'une base de séquences de données

ID−LIST = AA		
CID	Début	Fin
C4	2	4
C5	2	4

ID−LIST = BA		
CID	Début	Fin
C4	3	2
C5	1	4

ID−LIST =CA		
CID	Début	Fin
C2	1	2
C4	1	2
C4	1	4

ID –LIST=AB		
CID	Début	Fin
C1	1	2
C1	1	3
C2	2	3
C3	1	3
C4	2	3

ID−LIST=BB		
CID	Début	Fin
C1	2	3

ID−LIST = AC		
CID	Début	Fin
C1	1	4
C3	1	2

ID−LIST=CC		
CID	Début	Fin
VIDE		

ID−LIST=CB		
CID	Début	Fin
C2	1	3
C3	1	3
C4	1	3

D –LIST=BC		
CID	Début	Fin
C1	2	4
C1	3	4

Figure 3.6 Jointures temporelles SPADE poules 1−préfixes

$< \{A\} >, < \{B\} > \ et \ < \{C\} >$

ID−LIST =CBA		
CID	**Début**	**Fin**
C2	1	1
C4	1	3

Table 3.3 Liste d'occurrences SPADE pour la 3−séquence fréquente

$< \{C\}\{A\}\{B\} >$

L'Algorithme SPADE contient deux phases suivant :La fusion de deux k−séquences de la même classe d'équivalence, construit une (k+1) −séquence candidate. Formellement, si deux séquencesS_1,=<P;X>etS_2=<P;Y>partagent même préfixe P, alors Fusion (S_1,S_2) = <P;X;Y>La jointure temporelle des ID−LIST de deux séquences S1et S2, notée Joint (S_1,S_2) est une nouvelle ID−LIST, tel que Joint (S_1,S_2) ID −LIST (Fusion (S_1,S_2) avec : suit Z =Fusion(S_1,S_2). Pour chaque (CID_1, d1, f1) dans ID−LIST (S1) et (CID_2, d_2, f_2) dans ID−LIST (S_2), satisfaisant $CID_1 = CID_2$ et f1<d, ajouter (CID_1, d1, f2) à ID−LIST (Z) [26] .

3.2.1.2.3 Algorithme : SPADE (D', MinSup) :

Extraction de motifs séquentiels dans une base de séquences de données D'transformée d'une base de données temporelles, avec un seuil de support minimum MinSup.

Entrée : Base de séquences de données D'; Liste d'attributs I; Support minimum MinSup;

Sortie : Ensemble des séquences fréquentes dans D';

1. $Pour\ chaque\ item\ a \in I\ faire\ \ créer\ ID-LIST\ (< a >) = \emptyset;$
2. $Pour\ chaque\ séquence\ de\ données\ S \in D'\ faire$
3. Pour chaque ID-LIST (< a>) faire // Construction des listes d'occurrences d'items
4. $Pour\ chaque\ itemset\ T\ dans\ S\ faire$
5. $si\ \ a\ T\ alors\ Insérer\ (CID(S), Date(S,T), Date(S,T))\ dans\ ID-LIST(< a >);$
6. $Pour\ chaque\ ID-LIST\ (< a >)\ faire$
7. $Si\ nombre\ CID\ distincts\ dans\ ID-LIST\ (< a >) < MinSup\ alors$
8. Supprimer ID-LIST (>); // Purge des listes d'occurrences d'items non fréquents
9. $F1 = \emptyset;$
10. Pour chaque ID-LIST (<a>) faire F_1= F_1∪ a>}; // 1-séquences fréquentes
11. $Pour\ (k = 2;\ F_K - 1 \neq \emptyset;\ k++)\ faire$
12. $Début.$
13. $Pour\ chaque\ \ deux\ (k-1) - séquences S_1\ \ et\ S_2 \cup F_k - 1\ faire$
14. SiPréfixe (S_1, k -2) = Préfixe (S_2, k -2) alors// même classe d'équivalence
15. $Début$
16. $Générer\ la\ séquence\ \ candidate\ Z = S_1[k-2]; S_1[k-1]; S_2[k-1]; // Fusion$
17. $ID - LIST\ (Z) = Jointure - temporelle\ (ID - LIST\ (S_1), ID - LIST\ (S_2));$
18. $Si\ nombre\ CID\ distincts\ dans\ ID-LIST\ (< a >) < MinSup\ alors$
19. Supprimer ID-LIST (< a>)
20. Sinon$F_k = F_k \cup Z\}$;// Séquence fréquente trouvée
21. $Fin;$
22. $Fin;$
23. $Retourner\ F = F_1 \cup F_2 \ldots \cup F_k$ // Ensemble des séquences fréquentes dans D'.

Figure 3.7 Algorithme : SPADE [26]

3.2.1.2.4Limites de L'algorithme SPADE :

L'algorithme SPADE est un algorithme utilisé de base de donnes de type transactions et Représentation des données Verticalement avec Localisation des données pendant l'analyse de Mémoire centrale et Structuré de Donnée de Listes d'occurrences Cette solution présente toutefois un inconvénient parce que réside dans les coûts UC des opérations de jointure temporelle entre listes à chaque génération de séquences Candidates, sans compter l'a contrainte de limitation mémoire relative.[19]

3.2.2 Les Algorithmes basées sur parcours en profondeur :

3.2 .2.1AlgorithmePREFIX SPAN:

L'algorithme FREESPAN (Frequent pattern projected Sequential pattern mining).

L'idée générale est de proposer des projections récursives de la base de données en fonction des items fréquents. La base est alors projetée en plusieurs bases plus petites et les séquences fréquentes grandissent avec le nombre de projections. Les temps de réponses sont alors a méthode PREFIXSPAN se base sur une étude du nombre de candidats qu'un algorithme de recherche de motifs séquentiels peut avoir à produire afin de déterminer les séquences fréquentes. En effet, selon les auteurs, pour envisager D'utiliser un algorithme comme GSP il faut s'attendre à devoir générer, uniquement pour la seconde passe, pas moins de n2+n×(n−1)2candidats de taille 2 à partir des n items trouvés fréquents lors de la première passe. L'objectif des auteurs est alors, de réduire le nombre de candidats générés.

Pour parvenir à cet objectif, prefixspan propose (à l'instar de PSP avec les candidats) d'analyser les préfixes communs que Présentent les séquences de données de la base à traiter. à partir de cette analyse, l'algorithme construit des bases de données intermédiaires qui sont des projections de la base d'origine déduites à partir des préfixes identifiés. Ensuite, dans

Chaque base obtenue, prefixspan cherche à faire croître la taille des motifs séquentiels découverts en appliquant la même méthode de manière récursive. Deux sortes de projections sont alors mises en place pour réaliser cette méthode : la projection dite niveau par niveau" et la "bi-projection ". au final, les auteurs proposent une méthode d'indexation permettant de considérer plusieurs bases virtuelles partir d'une seule, dans le cas où les bases générées ne pourraient être maintenues en mémoire en raison de leurs tailles [8].

Client	Séquence
10	< (a) (a b c) (a c) (d) (c f) >
20	< (a d) (c) (b c) (a e) >
30	< (e f) (a b) (d f) (c) (b) >
30	< (e) (g) (a f) (c) (b) (c) >

Table 3.4base de données exemple pour préfixant

Préfixe	base projetée (suffixes)	motifs séquentiels
<a>	<(abc)(ac)(d)(cf)>, (_d)(c)(bc)(ae)>, <(_b)(df)(c)(b)>, <(_f)(c)(b)(c)>	a>, <(a)(a)>, <(a)(b)>, <(a)(bc)>, <(a)(bc)(a)>, <(a)(b)(a)>, <(a)(b)(c)>, <(ab)>, <(ab)(c)>, <(ab)(d)>, <(ab)(f)>, <(ab)(d)(c)>, <(a)(c)(a)>, <(a)(c)(b)>, <(a)(c)(c)>, <(a)(d)>, <(a)(d)(c)>, <(a)(f)>
<a>	<(_c)(ac)(d)(cf)>, <(_c)(ae)>, <(df)(c)(b)>, <c>	<b>, <(b)(a)>, <(b)(c)>, <(bc)>, <(bc)(a)>, <(b)(d)>, <(b)(d)(c)>, <(b)(f)>

Table 3.5. Résultat de prefixspan sur la base de données

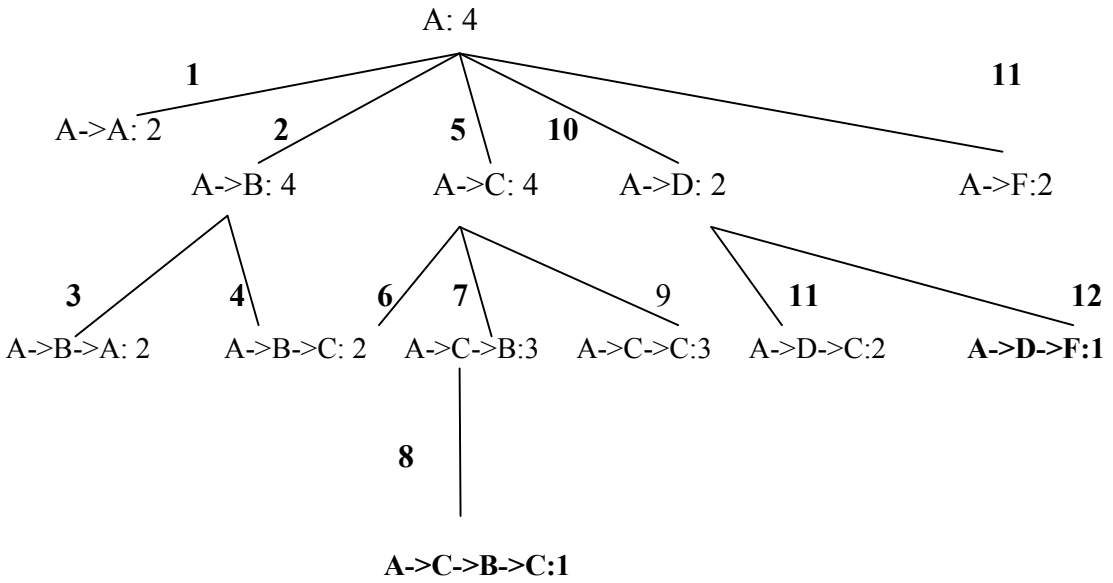

Figure 3 .8 Exemple un arbre de parcoure en profondeur [21]

3.2 .2.1.1 Principe de base

La méthode de projection préfixée va permettre de procéder à une extraction des Motifs séquentiels avec un support minimum de deux clients, en appliquant les étapes suivantes :

- Trouver les items fréquents. Pour cela, une passe sur la base de données

Va permettre de collecter le nombre de séquences supportant chaque item rencontré et Donc d'évaluer le support des items de la base. Les items trouvés sont (sous la forme <item> :support) : $< a >: 4 , < b >: 4 , < c >: 4 , < d >: 3 , < e >: 3 , < f >: 3$.

- Diviser l'espace de recherche**.** L'espace de recherche complet peut être

Divisé en six sous-ensembles, puisqu'il y a six préfixes de taille 1 dans la base ($i.e.$ les Six items fréquents). Ces sous-ensembles seront : (1) les motifs séquentiels ayant pour Préfixe <a>, (2) ceux ayant pour préfixe <b>, ... et (6) ceux ayant pour préfixe $< f >$.

- Trouver les sous-ensembles de motifs séquentiels**.** Les sous-ensembles de Motifs séquentiels peuvent être trouvés en construisant les projections préfixées des Bases obtenues et en réappliquant l'algorithme de fouille de manière récursive [8] .

3.2.2.1.2 Algorithme PRIFIXSPAN (D ,Min_sup)

Input: A Base de données de séquence et le support minimum Seuil min_sup

Output: L'ensemble complet de motifs séquentiels

Méthode: Apple Préfix$Span$ $(<> ,0, S)$

Subroutine Prefixspan (α, l, S|α)

Paramètres: a: paternelle séquentielle: la longueur d'un? S | a: la base de données une projection,

$if \alpha \neq <>$;autrement; La base de données Séquence S.

Fonction

1. Scan $S|\alpha$une fois que; Trouver l'ensemble des objets fréquents b

Such that:

a) b $Peut$ $\hat{e}tre$ $assembl\acute{e}$ au $dernier$ $\acute{e}lément$ de αto $form$ a

Motifs séquentielle; or

b) $< b >$Peut être ajouté à α pour former un motif séquentiel.

2. For each frequent item b, append it to αto form a Sequential pattern α' and output α;

3. For each α', construit α'-Base de données projected S|α', and call Prefixspan (α')

Figure 3 .9 AlgorithmePrefixspan

3.2.2.1.3 Limites de L'algorithme PREFIX−SPAN

Algorithme PREFIXSPAN :est un algorithme utilisé de base de type de séquences et représentation des données dans Projections horizontales avec localisation des données pendant l'analyse dans mémoire centrale et structuré de donnée dans bases projetées delà base de données en fon cation des préfixes que présentent souvent les séquences, ce qui permet de réaliser des gains considérables en dirigeant l'effort d'extraction vers les parties utile s de cette base. Un autre avantage de cette idée réside dans le fait que l'extraction peut se Faire en deux parcours seulement. De plus, la méthode ne génère aucune séquence candidate. Le prix à payer est une grande contrainte de limitation mémoire ,toutes les projections doivent tenir en mémoire, ce qui ne peut être garanti dans tous les cas [18].

3.2.2.2L'algorithme SPAM

L'algorithme SPAM gère la présence ou l'absence d'un item dans une séquence par l'intermédiaire de vecteurs de bits. Ainsi, un arbre de représentation est créé respectivement Par S-extension (ajout d'un item dans une autre transaction) ou par I extension (Ajout d'un item dans la même transaction) et les résultats sont stockés sous la forme de vecteurs de bits. La vérification des candidats est immédiate car il suffit de compter les bits positionnés à 1 dans la structure et de les comparer avec le support
Minimum [8].

3.2.2.2 .1 Principe de base :

SPAM considère l'existence d'un ordre lexicographique noté " $\geq$ " , sur les items de la base de données : si l'item j apparaît après l'item i alors $j \geq i$. Cette relation d'ordre est étendue aux séquences en définissant $Sb \geq Sa$ si Sbest sur−séquence $de\ Sa$. En se basant sur ce principe, l'algorithme parcourt l'espace de recherche en profondeur à l'aide d'un arbre lexicographique de séquences respectant cette relation qu'il construit récursivement à fur et à mesure de l'avancement de son exploration de cet espace [26].

3.2.2.2.2 Représentation en vecteurs de bits verticaux des séquences candidates :

De manière à minimiser les temps nécessaires aux opérations de génération de séquences candidates par S−extension et I−extension ainsi que les temps de calcul de leurs supports, SPAM apporte une représentation en vecteurs de bits verticaux de ces

Séquences. En sa première étape, l'algorithme crée en mémoire pour chaque item (1−séquence candidate) de la base de données, ce qui constitue l'unique parcours de cette dernière, un vecteur de bits vertical contenant un bit correspondant à chaque transaction de la base. Si un item i apparaît dans une transaction j d'un CID donné pour une Date donnée, alors le bit correspondant est mis à 1, autrement il a pour valeur 0. Le résultat est une représentation bitmap verticale de cette base, comme le montre la Figure 3.10 de l'exemple suivant [26].

- Exemple

D'	**01**	**02**	**03**	**04**	**05**	**06**	**07**
C1	{A, B, D}		{B, C, D}			{B, C,D}	
C2		{B}		{A, B, C}			
C3					{A, B}		{B, C,D}

Table 3.6 Base de transactions.

CID	**DATE**
C1	01 02 03
C2	02 04 -
C3	05 07 -

< {B}>
1 1 1
1 1 0
1 1 0

<{C}>
1 1 1
1 1 0
1 1 0

< {D}>
1 1 1
1 1 0
1 1 0

< {A}>
1 0 0
0 1 0
1 0 0

Figure 3.10 Représentation SPAM en vecteurs de bits verticaux d'une base de séquences de donnée

La Figure 3.10 illustre la manière par laquelle l'algorithme SPAM représente la base de séquences de données D' par des vecteurs de bits verticaux correspondant aux ite ms de cette base. Le parcours de D' lors de cette étape permet de connaître le nombre de valeurs de l'identifiant temporel (Dates) pour chaque CID[26].

3.2.2.2.3 Algorithme SPAM (D', MinSup)

Extraction de motifs séquentiels dans une base de séquences de données D', transformée d'une base de données temporelle, avec un seuil de support minimum MinSup.

Entrée : Base de séquences de données *D*'; Liste d'attributs I; Support minimum MinSup;

Sortie : Ensemble des séquences fréquentes dans D'.

1. Pour chaque item a ϵ I faire // Initialisation des vecteurs de bits des 1−séquences
2. Début
3. $Créer\ Vecteur\ (<\{a\}>)\ =\ (0,0,\dots,0);$
4. Nbre− dates− différentes$<\{a\}>=\ 0$;// Servira pour la partition des vecteurs de bits
5. Fin;
6. Pour chaque séquence de données $S\ \in\ D$' faire
7. Pour chaque Vecteur (<{a}>) faire // Construction des vecteurs de bits des 1−séquences
8. Pour chaque itemset T dans S faire
9. si a ϵ T alors Début
10. $Vecteur\ (<\{a\}>)[CID(S), Date(\ S,T)]\ =\ 1;$
11. $Incrémenter\ \ Nbre-\ \ dates-\ différentes_{<\{a\}>;}$
12. Fin;
13. $Nbre-bits-par-section\ =\ Max\ \{Nbre-\ \ dates-\ \ différentes_{<\{a\}>;}a\ \epsilon\ I\};$
14. Pour chaque Vecteur $(<\{\ a\}>)$ faire
15. Début // Partition des vecteurs de bits en section par CID
16. Etendre Vecteur (<{a}>) à $(Nbre-bits-par-section*Nbre-CID\ )$ bits;
17. Partitionner Vecteur (<{ a}>) en $Nbre-CID$ sections de Nbre−bits−par−section;
18. Fin;
19. Pour chaque $Vecteur\ (<\{\ a\}>)$ faire
20. Si Support−séquence $(Vecteur\ (<\{\ a\}>))<$ MinSup alors
21. Supprimer Vecteur (< {a}>)// Purge des vecteurs des 1−séquences non fréquentes
22. Sinon Début

23. $Retourner < \{a\} >$;// Initialisation de l'arbre lexicographique de séquences : branche < {a}>

24. $Créer\ nœud\ N \quad < \{a\} > fils\ S - extension\ de\ la\ racine; //\ Ensemble\ S < \{a\} > des\ items\ candidats\ pour\ le\ S - Step\ (les\ S - extensions)\ de\ N$

25. $SN = \{b \in I / \ni Vecteur(< \{b\} >)\}$; // Ensemble $I < \{a\} >$ des items candidatspour le $I - Step$ (les I-extensions) N

26. $IN = \{b \in I / \ni Vecteur\ (< \{b\} >) \quad b > a\}$;

27. $SPAM - Traverse - nœud(N, \quad SN, IN)$;

28. Fin;

***Procedure*: $SPAM - Traverse - Noeud\ (N = < S_1; S_2 .. S_n >, S_N, I_N)$**

Construction récursive de l'arbre lexicographique de séquences à partir du nœud N représentant la séquence fréquente $<S_1;S_2 \ldots S_n>$, en l'étendant par S_Nl'ensemble des items candidats pour les S−extensions et I_Nl'ensemble des items candidats pour les I−extensions. // Traitement des S−extensions de la séquence $<S_1;S_2 \ldots;S_n>$,par tous les items dans S_N

1. Pour chaque item $a \in S\_N$ faire

2.$Si\ Nombre - sections - non - nulles(Transformée(Vecteur(< S_1; S_2 \ldots; S_n >) \wedge Vecteur(\{a\})) \geq$MinSup.

3. alors Début.

4. $Return < S_1; S_2 \ldots S_n >; \{a\} >$;

5. Créer nœud $N' \quad < S_1; S_2 \ldots; S_n >; \{a\} >$ fils S−extension de $N = < S_1; S_2 \ldots; S_n >$.

6. s_N'= {b ∈ I /∃Vecteur (< {b}>)}; // Uniquement les fréquents

7.$I_{N'}$= {b ∈ I / ∃Vecteur (< {b}>) ∧b >a} ; // Fréquents plus dans l'ordre ≥

8. SPAM−Traverse−Nœud (N', $s_N I_N$); //Appel récursif sur le nouveau nœud créé

9. Fin;// Traitement des I− extensions de la séquence $< s1; s2; \ldots ; sn >$ par tous les items dans IN

10. Pour chaque item a ∈ $I_N$faire

11. Si$Nombre - sections - non - nulles(Vecteur(< S_1; S_2 \ldots; S_n >;) \wedge Vecteur(\{a\}) \geq MinSup\ alors$

12. Début

13. Retourner $< S_1; S_2 \ldots; S_n >; \cup \{a\} >$;

14. Créer nœud $N' \quad < S_1; S_2 \ldots; S_n > U$ {a}> fils I−extension de $N = < S_1; S_2 \ldots; S_n >$

15.s_N'= {b∈I /∃$Vecteur$ ($< \{ b\} >$)}; // Uniquement les fréquents.

16.I_N' = {b∈I /∃Vecteur (<{b}>)∧b > Max(s_n{a})}; // Fréquents + dans l'ordre ≥

17. SPAM−Traverse−Nœud (N', S_N', I_N'); //Appel récursif sur le nouveau nœud créé

18. Fin;

Figure 3.11 algorithme spam [26] .

3.2.2.2.4 Calcul des supports :

Grâce à leur représentation en vecteur de bits partitionnés en section s correspondant chacune à $un\ CID$ de la base, le calcul du support de n'importe quelle séquence candidate générée par un nœud fréquent dans l'arbre lexicographique de séquences (Soit par S−extension ou par I−extension), est immédiat dans SPAM. En effet, ceci revient à simplement compter, le nombre de sections non nulles dans le vecteur de bits lui correspondant**.[**26]

3.2.2.2.5 Limites de L'algorithme SPAM

L'algorithme SPAM :est un algorithme utilisé de base de type de transactions et représentation des données dans Bitmap verticalement avec Localisation des données pendant l'analyse dans Mémoire centrale et Structuré de Donnée dans Bases Vecteurs de bits, Ensuite, il effectue l'extraction de Motifs séquentiels en exploitant un arbre lexicographique de séquences définissant son espace De recherche. L'avantage principal est que l'algorithme n'effectue qu'un unique parcours de cette base, sans compter l'avantage des opérations binaires sur les vecteurs de bits, qui Permettent une e meilleure génération de séquences candidates ainsi que le comptage rapide de Leurs supports. Toutefois, le prix à payer est le même que pour les deux algorithmes PREFIX−SPAN et SPADE : une grande contrainte de limitation mémoire [17].

3.3. Conclusion :

Dans ce chapitre, nous avons présenté certaines techniques d'extraction de motifs séquentielle dans base des séquences. Une étude comparative de ces approches ainsi que des différents algorithmes dans ce domaine ont été réalisés nous avons montré que l'application de ces algorithmes sur une même base de données avec les mêmes seuils de support donne les mêmes motifs. Il en découle que la comparaison doit concerner principalement les performances, en d'autres termes, les temps de réponse de ces algorithmes ainsi que l'espace mémoire requis pour la tâche d'extraction.

CHAPITRE 4

IMPLEMENTATION

4.1 Introduction

Dans ce dernier chapitre après l'aperçu théorique général des chapitres précédents, nous offrons un côté pratique pour appliquer notre objectif. Il est un système de de comparaison entre les deux algorithmes prefixspan et spam en termes de mieux la vitesse de mise en œuvre et consommation moins de mémoire spatiale note que la comparaison est entre deux nouveaux algorithmes dans l'extraction de motifs séquentielle Et représentent les destinations du spectre résultant. Grâce à des courbes graphiques.

4.2 Présentation des outils de développement

4.2.1 NetBeans

Est un environnement développement intégré (EDI), placé en open source par Sun en juin 2000 sous licence CDDL (Common Développent and Distribution License) et GPLv2. En plus de Java, NetBeans permet la prise en charge native de divers langages tels le C, le C++, le JavaScript, le XML, le Groovy, le PHP et le HTML, ou d'autres (dont Python et Ruby) par l'ajout de greffons. Il offre toutes les facilités d'un IDE moderne (éditeur en couleurs, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

Compilé en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86 et SPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Développent Kit JDK est requis pour les développements en Java. NetBeans constitue par ailleurs une plate-forme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plate-forme [13].

Figure 4.2 NetBeans.

4.2.1.1 Plate-forme NetBeans

NetBeans est aussi une plate-forme générique pour le développement d'applications pour stations de travail (bibliothèque Swing (Java)).Elle fournit des ressources pour développer les éléments structurants de ces applications: gestion des menus, des fenêtres, configuration, gestion des fichiers, gestion des mises àjou Des présentations détaillées sont fournies par le centre de documentation de NetBeans [13].

L'IDE NetBeans comprend toutes les ressources utiles mais il est aussi possible d'installer la plate-forme séparément.

Le développement d'applications sur la base de la plate-forme NetBeans consiste en la réalisation de « modules » qui s'insèrent dans la plate-forme et en étendent dynamiquement les fonctions.

Un module est un groupe de classes Java, de portée variée : elle peut consister en une simple classe Java réalisant des fonctions simples (exemple : ajouter une action dans un menu pour éditer le contenu du presse papier) comme elle peut intégrer une application externe complète (exemple : Java profiling suite). Un module peut s'appliquer à l'IDE NetBeans lui-même. La réalisation des modules s'appuie sur une API normalisée [13].

4.2.2 Le langage Java

Java est un langage de programmation et une plate-forme informatique créée par Sun Microsystems en 1995. Il s'agit de la technologie sous-jacente qui permet l'exécution de programmes modernes et performants, notamment dans la construction des utilitaires, des jeux et des applications professionnelles. Java est utilisée sur plus de 850 millions d'ordinateurs de bureau et plus d'un milliard de périphériques dans le monde, dont des périphériques mobiles et des systèmes de diffusion télévisuelle. [13]

4.2.2 2 Les caractéristiques du Java

- Java est interprétée.
- Java est portable.
- Java est orienté objet.
- Java est simple.
- Java est fortement typé.
- Java assure la gestion de la mémoire.
- Java est sûre.
- Java est économique.
- Java est multitâche. [13]

4.2.3 Packagejfreechart :

JFreeChart est une bibliothèque graphique gratuite de 100% Java qui permet aux développeurs d'afficher des graphiques de qualité professionnels dans leurs applications. Le vaste ensemble de fonctionnalités de JFreeChart comprend une API cohérente et bien documentée, prenant en charge une large gamme de types de diagramme. Un design flexible qui est facile à étendre et cible les applications côté serveur et côté client; Prise en charge de nombreux types de sortie, y compris les composants Swing et Java FX, les fichiers image (y compris PNG et JPEG) et les formats de fichiers graphiques vectoriels (y compris PDF, EPS et SVG); JFreeChart est un logiciel gratuit ou, plus précisément, un logiciel gratuit. Il est distribué selon les termes de la GNU Lesser General Public License (LGPL), qui permet l'utilisation dans des applications propriétaires. Pour un examen plus approfondi de JFreeChart, essayez la vaste application de démonstration incluse dans le téléchargement ou parcourez la page Samplers. [12]

4.2.4 Une bibliothèque de données extra-source :

Spamf est une bibliothèque minière exploration de données open-source écrit en Java, spécialisé dans le secteur minier de modèle. Il est distribué sous licence GPL. Il propose des implémentations de 129 algorithmes d'extraction des données il code source de chaque algorithme peut être facilement intégré dans d'autres logiciels Java. De plus, FMSP peut être utilisé en tant que programme autonome avec une interface utilisateur simple ou à partir de la

ligne de commande.SPMF est rapide et léger (pas de dépendance à d'autres bibliothèques).La version actuelle est v2.13 et a été libéré le 16 Mars 2017. [14].

4.3 Description de la base de Données :

Une base de données de séquence est un ensemble de séquences où chaque séquence est une liste d'itemsets. Un jeu d'éléments est un ensemble non ordonné d'éléments distincts. Par exemple, le tableau ci-dessous contient quatre séquences. La première séquence, nommée S1, contient 5 itemsets. Cela signifie que l'élément 1 a été suivi des éléments 1 2 et 3 en même temps, suivis de 1 et 3, suivis de 4 et suivis de 3 et 6.On suppose que les éléments dans un ensemble d'éléments sont triés en ordre lexicographique. Cette base de données est fournie dans le fichier "contextPrefixSpan.txt" de la distribution SPMF. Notez qu'il est supposé qu'aucun élément n'apparaît deux fois dans le même jeu d'éléments et que les éléments dans un jeu d'éléments sont commandés lexicalement.

ID	Séquences
S1	(1), (1 2 3), (1 3), (4), (3 6)
S2	(1 4), (3), (2 3), (1 5)
S3	(5 6), (1 2), (4 6), (3), (2)
S4	(5), (7), (1 6), (3), (2), (3)

Table 4.1 Base de données de séquence.

Dans notre travail, nous avons utilisé beaucoup de fichier texte d'entrée de données de base et l'importation seuil base de données spécifiée par la séquence minsup d'utilisateur nommé (valeur dans [0,1] en pourcentage).En outre, une longueur maximale d'un motif séquentiel en termes d'éléments. Séquence de base de données est un ensemble de séquences dans lequel chaque séquence est une liste de jeux d'éléments. Un ensemble de choses est un groupe d'Eléments est disposé. Par exemple, dans le fichier contextPrefixSpan.txt Ces données proviennent du SPMF du site, des données réelles, et il est significatif en termes développés par des études antérieures spéciales.

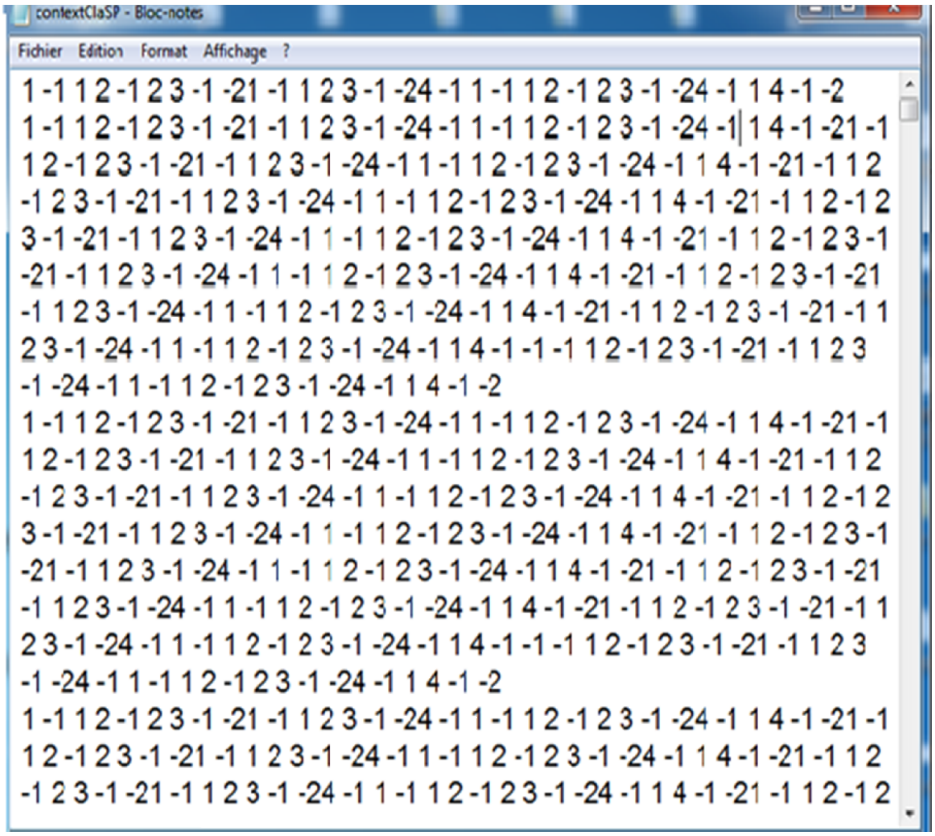

figure 4 .3 Présenter le ficher de entrée de base de donnée.

4.4 Description de l'application

La Conception dès l'interfaces applications est une étape important dans cette partie, nous allons passer en revue quelques-unes des formes et des situations rendues par l'utilisation de cette application, Lorsque la demande contient une interface principale contient toutes les fonctions de l'application.

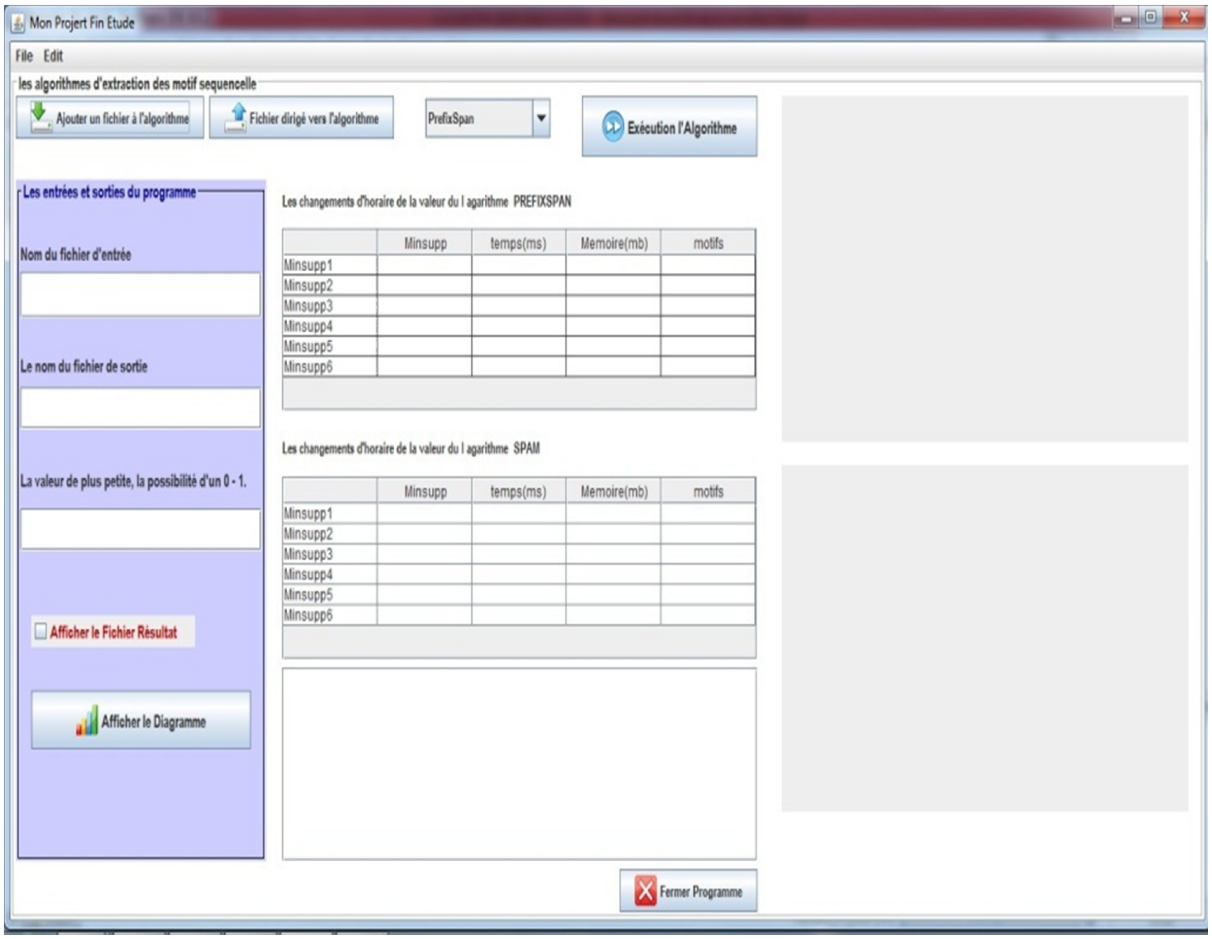

Figure 4 .4 Interface principale d'extraction des motifs séquentielle.

L'interface principale contient les différent algorithmes qui exécuter par l'utilisateur, En Passent les paramètres parmi ces paramètres en cite.

- Le fichier d'entrée : c'est un fichier texte où chaque ligne représente une séquence à partir d'une base de données de séquence. Chaque élément d'une séquence est un entier positif et les éléments du même ensemble d'éléments dans une séquence sont séparés par un espace unique. Notez qu'il est supposé que les éléments d'un même ensemble d'éléments sont triés selon un ordre total et qu'aucun élément ne peut apparaître deux fois dans le même jeu d'éléments. La valeur "-1" indique la fin d'un jeu d'éléments. La valeur "-2" indique la fin d'une séquence (elle apparaît à la fin de chaque ligne).
- Le fichier sorti : C'est un fichier texte. Chaque ligne est un motif séquentiel fréquent. chaque élément d'un motif séquentiel est un entier positif et les éléments du même

ensemble d'éléments dans une séquence sont séparés par des espaces uniques. La valeur "-1" indique la fin d'un jeu d'éléments. Sur chaque ligne, le modèle séquentiel est indiqué pour la première fois. Ensuite, le mot-clé "#SUP:" apparaît suivi d'un entier indiquant le support du motif en tant que nombre de séquences. Par exemple, quelques lignes du fichier de sortie.

- Min support : c'est le support minimal saisi par l'utilisateur.

4.4 .1 Les scénario des algorithmes

4.4.1.1 Le scénario d'algorithme prefixspan

- choisir l'algorithme prefixspan.
- Sélectionner le fichier d'entrée «contextPrefixSpan.txt» .définir le nom du fichier de sortie (par exemple « sortie.txt »).
- définir minsup
- cliquer sur le bouton exécuter l'algorithme.
- cliquer afficher les diagrammes.

4.4.1.2 Le scénario d'algorithme Spam

- choisir l'algorithme Spam.
- Sélectionner le fichier d'entrée «contextPrefixSpan».
- définir le nom du fichier de sortie (par exemple « sortie.txt ») .
- définir minsup.
- cliquer sur le bouton exécuter l'algorithme.
- cliquer afficher les diagrammes.

4.5 Exécution les algorithmes :

Nous allons mettre en exécuter des algorithmes utilisant les variables suivantes :

Min support =0.2, Min support=0.3, Min support=0.4 .Les résultats sont présentés dans l'interface suivante :

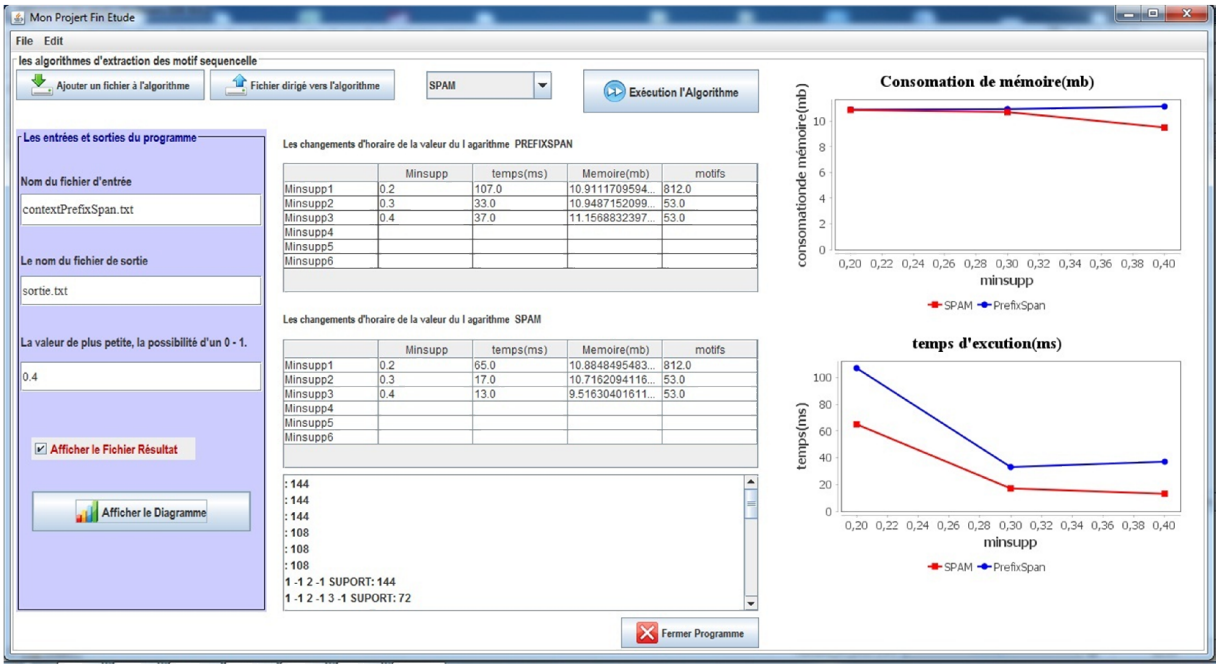

Figure 4.5 résultats de l'exécution.

4.5.1 Résulta de l'exclusion

Il représentera les résultats obtenus à l'ordre du jour des éclaircissements plus.

Min support	Algorithme	Temps (ms)	Mémoire (mb)	Les motifs
0.2	**Prefixspan**	107	10.9111	812
	Spam	65	10.8848	812
0. 3	**Prefixspan**	33	10.9487	53
	Spam	17	10.17620	53
0.4	**Prefixspan**	37	11.15688	53
	Spam	17	9.51630	53

Table 4.2. Résultat obtenus après l'exécution

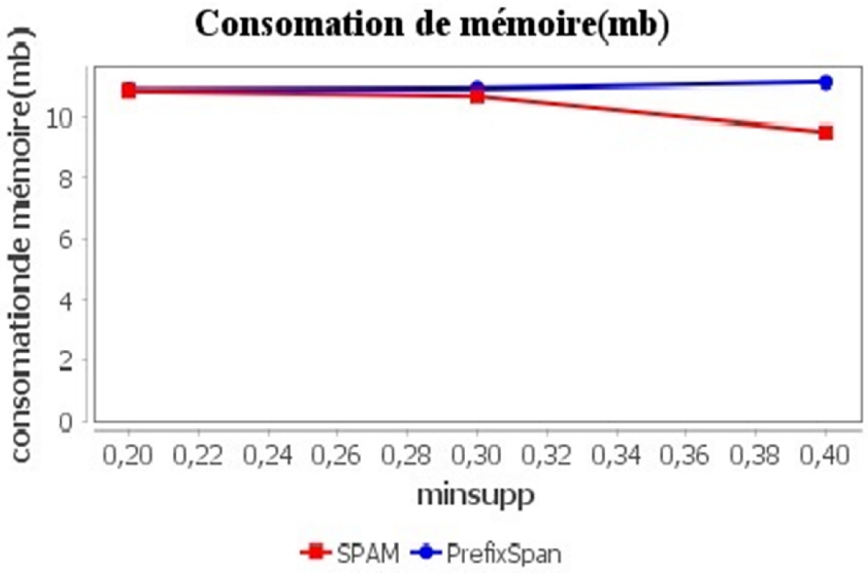

Figure 4 .6 Résultat obtenus de consommation de mémoire pour minsup1 = 0.2, minsup2 = 0.3, minsup3 = 0.4.

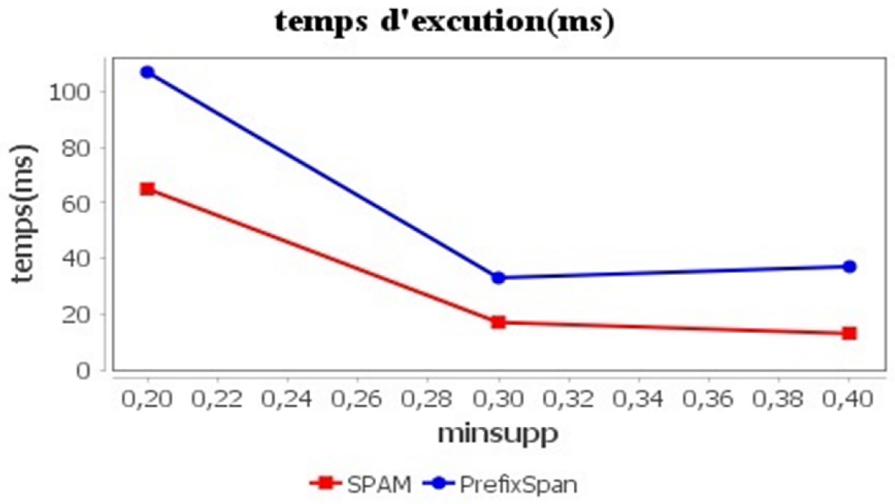

Figure 4 .7. Résultat obtenus de temps d'exécution (ms) pour minsup1 = 0.2, minsup2 = 0.3, minsup3 = 0.4.

À travers les diagrammes notez que plus minsup nous constatons qu'il est moins coûteux en termes de temps d'exécution et de la consommation de mémoire pour cela ou mois génères les candidats.

4.6 Comparaison

En basant sur le résultat obtenus précédent, on peut être établir une comparaison les entre deux algorithmes prefixspan et spam avec les paramètres 0.2, 0.3, 0.4.

- L'Algorithme prefixspan est trop coûteux en terme de temps d'exécution est espace mémoire par rapport à L'algorithme spam car l'algorithme spam représentation de la base de données en mémoire centrale dans une structure en vecteurs de bits tandis que l'algorithme prefixspan proposer des projections récursives de la base de données.

4.7 Conclusion

Dans ce chapitre, nous avons vu les différents outils utilisés pour implémenter noter application ainsi les interfaces de l'application, ensuite nous introduisons tout le résultat obtenus après l'exécution de plusieurs fois de chaque algorithme et comparer enter deux algorithmes.

CONCLUSION GENERLE

Dans le cadre de ce travail de fin d'étude, nous avons découvert des domaines complètement nouveaux telle la fouille de données et le monde de la recherche. Cette expérience nous a amenés à comprendre les diverses problématiques liées aux règles d'association et à l'extraction de motifs séquentiels.

Lors de ce travail, nous avons essayé d'étudier et d'évaluer les performances de quelques algorithmes d'extraction de motifs séquentiels en termes de temps d'exécution et de consommation de l'espace mémoire. Les deux algorithmes prefixspan et spam sont étudiés et les résultats obtenus font l'objet d'une étude comparative.

Nous pouvons conclure que l'extraction de motifs séquentiels est une tâche difficile et qu'il n'est pas possible de dire quel algorithme est meilleur que les autres, dans la mesure où leurs performances sont étroitement liées aux types de données manipulées et aux paramètres choisis. Et nous sommes désolés que la recherche est, compléter le spectre des contraintes de temps.

Et comme perspective, nous proposons d'étendre ce travail à d'autres algorithmes et de faire des études comparatives en fonction d'autres types de séquences de données.

Et vu que la plupart des données réelles intéressantes dans ce contexte sont numériques (quantitatives), il serait donc intéressant d'introduire la notion de flou à ce problème d'extraction de motifs séquentiels.

BIBLIOGRAPHIE

[1] Agrawal, R. Srikant Mining sequential patterns :Generalizations and performance improvements,IBM Research Division Almaden Research C enter 650 Harry road san jose,CA 95120-6099.

[2] B.Kao, C.Lap Yip, A_GSP-based_Efficient_Algorithm_for_Mining_Frequent_Sequences, The_University_of_Hong_Kong August2002

[3] C. Fiot, Motifs séquentiels et approximation des valeurs manquantes, LIRMM (CNRS - UMII) Campus Saint-Priest 161 rue Ada .

[4] C.Fiot , Extraction de séquences fréquentes des données numériques aux valeurs manquantes, Spécialité Doctorale : Informatique, II Université Montpellier Sciences et Techniques

du Languedoc , le 28 septembre 2007.

[5] Didacticiel - Études de cas/ Extraction des itemsets à l'aide du composant FREQUENT ITEMSETS/ 3 octobre 2011.

[6] Etudes des principaux algorithmes de data Mining. Kremlin-Bicêtre- France, Spécialisation Sciences Cognitives et Informatique Avancée2009

[7] F. Masseglia,F.Cathala, P.Poncelet The PSP approach for mining sequential patterns 2nd European Symposium on Principles of Data mining and Knowledge Discovery (PKDD'98), pp. 176−184, Nantes, September 1998.

[8] F. Masseglia,M. Teisseire ,P. Poncelet/ Extraction de motifs séquentiels Problèmes

et méthode,ingenieree de système d'information, Université de MontpellierII , August2004.

[9] http://software.ucv.ro/~cmihaescu/ro/teaching/AIR/docs/Lab8-Apriori ,15/5/2017

[11] H.M'zail les règles d'association séquentielles, 17/4 /2006.

[12] http://docplayer.fr/2357529-Temporal-pattern-mining-beyond-simple-sequential-patterns-

fouille-de- motifs-temporels-au-dela-des-motifs-sequentiels-simples.html 15/5/2017.

[13] http://www.jfree.org/jfreechart/,16/05/2017

[14] https://fr.wikipedia.org/wiki/NetBeans, 17/05/2017

[15] http://www.philippe-fournier-viger.com/spmf/index.php ,18/05/2017.

[16] https://fr.wikipedia.org/wiki/Exploration_de_donn%C3%A9es,19 /05/2017.

[17] http://www.wikiwand.com/fr/Cycle_de_d%C3%A9veloppement_(logiciel)#/ ,18/05/2017.

[18]: J.Ayres,J.Gehrke,T.Yiu,J.Flannick Sequential pattern mining using a bitmap representation

International Conference on Knowledge Discovery and Data mining (SIGKDD'02),

pp. 429−435, Edmonton, July 2002.

[19]: J. Pei, J. Han, B. Mortazavi, H. Pinto, Q. Chen, U. Dayal, M. C. Hsu PREFIX−SPAN: Mining sequential patterns efficiently by prefix projected pattern- growth In th e 17Th

International Conference on Data Engineering (ICDE'01), Heidelberg, April 2001.

.[20] J. Zaki ,SPADE : An efficient algorithm for mining frequent sequences

Machine Learning Journal, Vol. 42(1−2), pp. 31−60, January 2001

[21] M. Betouati Fatiha, data mining distribue,magister,universite des sciences et de

la technologie d'oran mohamed boudiaf,

[27] M.LELEU, extraction de motifs séquentiels sous contraintes dan des donnés contenant des répétition consécutives, ingénieur, l'institut national des sciences appliquées de Lyon ,2004 .

[22] : M. Mohammed Nassim./ Extraction des connaissances dans un environnement distribué :

[23] Mr. MOUNA,Azzeddine ,datamining distribue dans les grilles approche
regles d'association,Magister en Informatique,universite des sciences et de la technologie d'oranMohamed Boudiaf,2012/2013.

[24] M. Plasse ,N.Gilbert Saporta ,A.Villeminot, L.Leblond,

Méthodes de classification pour l'extraction de règles/ CNAM Laboratoire CEDRIC 292 Rue St Martin Case 441/ 75141 Paris Cedex 03, France.

[25] Mr. ALLIA Mohamed Rachid Mlle. BOUADI Tassadit Mr. El MOUTAOUKILSami

Mr. KEIRA Mamadou,Fouille de donnéesRègles séquentielles ,

UNIVERSITE MONTPELLIER II

[26] M.Sadek Reguieg Yssaad , extraction de motifs séquentiels application à

la maintenance industrielleAval / sonatrach, magister, université des Sciences et de la

Technologie d'Oran Mohamed Boudiaf, 2010

[27] Nicolas Pasquier, Extraction de Bases pour les Règles d'Association partir des Itemsets Fermés Fréquents, Laboratoire d'Informatique (LIMOS) - Université Clermont-Ferrand II

Complexe scientifique des Cézeaux, 24 avenue des Landais, 63177 Aubière cedex France

[28] N. NAFFAKHI, Apprentissage supervisé pour la classification des images à l'aide de l'algèbre P-tree Université de Tunis Institut Supérieur de Gestion de Tunis, Février 2004

[29] N. Pasquier . Extraction de Bases pour les Règles d'Association, à partir des Itemsets

Fermés Fréquents. Laboratoire d'Informatique (LIMOS) - Université Clermont-Ferrand II ,Complexe scientifique des Céteaux, 24 avenue des Landais, 63177 Aubière cedex France. Catégorie :Jeune Chercheur.

[30] Préparation agrégation : Algorithmique Schémas algorithmiques et graphes Benjamin Monème Année 2010/2011

[31] Wikipédia, www.wikipedia.com consulté le : 12 / 2 /2017.

[32] Y . Bastide –R. Taouil — N .Pasquier-G. Stumme ,L . Lakhal. Pascal : un algorithme D'extraction des motifs fréquents. Technique et science informatiques. Volume 21 – n 1/2002.

ملخص

في مجال التنقيب عن البيانات، أصبح استخراج الانماط المتسلسلة، منذ إطلاقها، تقنية أساسية لها العديد من التطبيقات (تحليل سلوك المستهلك، المعلوماتية الحيوية، الأمن، وما إلى ذلك).

هنالك العديد من الخوارزميات لاستخراج هذه الأنماط تستخدم أساسا طريقتين، تستند الأولى على مبدأ البحث الأفقي لقاعدة البيانات وهي مستوحاة من الطريقة التقليدية Apriori. وتستخدم الثانية بحث عمودي جعلها تتكيف بشكل جيد مع هذه الإشكالية.

في هذا المشروع قمنا بتطبيق خوارزميات استخراج الأنماط المتسلسلة على بيانات حقيقية لتقييم أداءها.

كلمات مفتاحية : التنقيب عن البيانات، أنماط متسلسلة، قواعد الارتباط.

Abstract

In the field of data mining, sequential pattern mining is a key technique of data mining with broad applications (user behavior analysis, bioinformatics, security, etc.).

There are many algorithms for mining such patterns. These proposals use essentially two methods; the first one is based on the principle of a horizontal search in the database and which is inspired by the traditional method Apriori. And the second uses a vertical search very well adapted to this problem.

In this project, we apply sequential pattern mining algorithms on real data to evaluate their performance.

Keywords: data mining, sequential patterns, association rules.

Résumé

Dans le domaine de la fouille de données, l'extraction de motifs séquentiels est devenue, depuis son introduction, une technique majeure avec de nombreuses applications (analyse du comportement des consommateurs, bioinformatique, sécurité, etc.).

Il existe de nombreux algorithmes permettant l'extraction de tels motifs. Ces propositions utilisent essentiellement deux méthodes, la première est basée sur le principe d'une recherche horizontale dans la base de données et qui s'inspire de la méthode traditionnelle Apriori. Et la deuxième utilise une recherche verticale très bien adaptée à cette problématique.

Dans ce projet, nous appliquons des algorithmes d'extraction de motifs séquentiels sur des données réelles afin d'évaluer leurs performances.

Mots-clés: fouille de données, motifs séquentiels, règles d'association.