الجـــــــــــــــــــمـهـوريـــــة الجـــــــــــــزائـريـة الديمـــــــقـراطيـة

الـشـــــــــعبيـة

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

وزارة التـــــــــــــــــــــــــعـليـم العـــــــالي و البـــــــــــحث العـــــلـمـــي

Ministère de l'Enseignement Supérieur et de la Recherche Scientifique

جـــــــــــــــامعة غـردايـــــة

Université de Ghardaia

كـليـة العـلـوم والتـكـنولـوجيـا

Faculté des Sciences et de Technologie

قسـم الرياضيات و الاعلام الآلـي

Département des Mathématiques et Informatique

MEMOIRE

Présenté pour l'obtention du **diplôme** de **MASTER**

En : (**Informatique**)

Spécialité : (Les systèmes intelligents pour L'Extraction des connaissances(**SIEC**))

Par : (KERFOUH KAMLA et TERBAGOU NAIMA)

Sujet

La Prédiction des Routes avec les Règles Séquentielles dans les Réseaux VANET

Soutenu publiquement, le 14 / 06 / 2016 , devant le jury composé de :

M.Ziadi Djeloul	Professeur	Univ. Ghardaia	Président
Melle Amirat Hanane	Maitre assistant A	Univ.Ouargla	encadreur
M. Ouled Nouaui Slimane	Maitre assistant	Univ .Ghardaia	Examinateur 1

Année Universitaire 2015/2016

Table des matières

Abstract

Predicting future movement of moving objects has attracted a great attention these days. It has emerged as an important technology topic in many applications related to intelligent transportation systems (ITS) and Location based services (LBS). Many prediction models have been applied in this problem basing on data mining techniques (i.e. neural network) or probabilistic models (i.e. markov modeles). However, mining sequential rules from sequence databases is active research topic and broadly applied for many real-world scenarios. Recently, several extensions of the problem of sequential rule mining have been proposed to address specific needs. In this thesis, we propose to apply sequential rules for route prediction problem. We aim to further compare the prediction performances of a new kind of sequential rules called partially order sequential rules with standard sequential rules. Our proposal is evaluated using synthetic dataset. The experiments show promising results that outperform standard sequential rules.

Key works: future movement, route, prediction, sequentially rule, partially ordered sequential rules, standard sequential rules, ITS, LBS.

Résumé

De nos jours, la prédiction de futur mouvement d'un objet mobile a attiré une attention croissante des chercheurs. Elle est une tâche ayant de multiples applications surtout pour les systèmes de transportation intelligents (STI) et services basés sur la localisation (SBL). Plusieurs modèles de prédiction ont été proposés dans la littérature basés sur la fouille de données (ex: les réseaux de neurones, les règles séquentielles) ou des modèles statistiques (ex. modèle de Markov). La génération des règles séquentielles est un sujet de recherche actif et largement appliqué pour nombreuse applications du monde réel. Récemment, plusieurs extensions du problème de génération des règles séquentielles ont été proposées. Dans ce mémoire, nous proposons d'utiliser les règles séquentielles pour la prédiction des routes. Nous suggérons ainsi à appliquer un nouveau type des règles appelé règles séquentielles partiellement ordonnées (POSR) dont le but est de étudier et comparer les performances de ce nouveau type avec les règles séquentielles standard. Notre proposition a été évaluée en utilisant une large trace synthétique de mouvement des véhicules. Les expériences ont montré des résultats prometteuses avec l'utilisation des règles séquentielles dans le problème de prédiction des routes.

Mots clés: future mouvement, route, prédiction, règles séquentielles, règles séquentielles partialement ordonnées, STI, SBL.

الملخص

أخذ موضوع تنبؤ الحركة المستقبلية للأجسام للمتحركة اهتمام كبير من طرف الباحثين هذه الأيام , فقد ظهرت العديد من التطبيقات والتكنولوجيات المتعلقة بأنظمة التنقل الذكية (ITS) وخدمات تحديد الموقع (LBS) وقد تم تطبيق العديد من النماذج لحل هذه المشكلة مستندة على تطبيقات استخراج البيانات (الشبكة العصبية)و النماذج الإحصائية الاحتمالية (نموذج ماركوف).

ان استخراج واستنتاج القواعد التسلسلية من قواعد البيانات التسلسلية , موضوع بحث مهم ونشط وتطبيقه على نطاق واسع وفي العديد من المجالات في العالم حاليا حيث تم مؤخرا تقديم العديد من الاقتراحات والتمديدات لمشكلة استخراج القواعد التسلسلية .في هذه المذكرة نهتم بتطبيق القواعد التسلسلية لحل مشكلة تبؤ الطريق كما نهدف إلى مقارنة أداء التنبؤ باستعمال نوع جديد من القواعد التسلسلية جزئيا مع القواعد التسلسلية القياسية وهذا باعتماد على بيانات اصطناعية حيث تبين التجارب نتائج واعدة يتفوق فيها النوع الجديد على النوع الآخر .

الكلمات المفتاحية : الحركة المستقبلية, التنبؤ, الطريق , القواعد التسلسلية , القواعد التسلسلية القياسية , القواعد التسلسلية جزئيا , بأنظمة التنقل الذكية (ITS) ,وخدمات تحديد الموقع(LBS).

Introduction générale

1- Contexte et motivations

Prédire les futurs mouvements est d'une grande importance pour nombreuses applications dans plusieurs contextes. Un des contextes le plus important est notamment l'amélioration de la qualité des systèmes de transport intelligents (Intelligent Transportation Systems) en fournissant des données de trafic en temps réel et donc la possibilité de prédire la congestion. Simplement dit, si nous pouvons prédire les futurs mouvements des véhicules, nous serons en mesure d'estimer les futures congestions et les dangers de la circulation. Une autre utilisation de la prédiction des routes est l'optimisation de la consommation de carburant. Les chercheurs de Nissan ont montré qu'il est possible d'économiser le carburant jusqu'à 7,8% si la route est connue à l'avance [1]. Dans le contexte des services basés sur la localisation (Location based services), la prédiction de routes peut être utilisée dans la publicité ciblée pour livrer des messages publicitaires à des clients qui sont susceptibles d'approcher des magasins d'intérêts.
La prédiction de future route consiste à trouver le futur segment de route pour un utilisateur mobile (ex: conducteur). Elle est principalement basée sur l'hypothèse que le comportement des utilisateurs présente une régularité spatiale. Par exemple une personne a toujours la tendance de prendre la même route de la maison vers son lieu de travail. Pour cette raison, un nombre important des trajets (trips) d'une personne sont répétées et donc il est possible de prédire qu'il ou elle prendra cette route.

2- Problématique et objectifs

Actuellement, la prédiction de route a attiré une grande attention de la communauté des chercheurs. La plupart des approches de prédiction proposées consiste à appliquer des techniques d'apprentissage machine, tels que: réseau de neurones, règles séquentielles, etc. ou des modèles statistiques comme les modèles de Markov. Il y a beaucoup de raisons derrière le choix de ces techniques pour problème de prédiction des routes (1) la prédiction est une tâche majeure de data mining cela justifie l'utilisation de ces techniques pour ce problème (2) la prédiction n'est pas exacte tout le temps vue qu'elle est effectuée avec une certaine probabilité, ce qui rend les modèles statistiques un outil approprié pour un tel problème. Par conséquent, ces techniques/modèles ont démontré une grande précision lors de l'application pour la prédiction de route.

Cependant, certaines limitations peuvent être signalées [1] une bonne prédiction dépend de la définition d'un ensemble de paramètres de configuration tels que le poids, une architecture appropriée (ex: réseaux neuronaux) et/ou quelques approches de prédiction proposées supposent que la prédiction de future segment de route ne dépend que de précédent segment (cas de modèle de markov de 1^{ier} ordre) ou exigent la construction d'un modèle qui a une complexité exponentielle, si plus d'un élément est considéré (cas des modèles de markov de K^{ieme}ordre). Ces hypothèses peuvent rendre la prédiction irréaliste ou irréalisable dans certains cas.

Pour répondre à certaines de ces limitations, nous proposons, dans ce travail, une nouvelle approche de prédiction des routes basée sur les règles séquentielles. Prédire à l'aide de règles séquentielles a l'avantage d'être non supervisée, évolutive (scalable), et tolérante au bruit (noise tolèrent).

Dans le contexte de la prédiction des routes, les règles séquentielles ont été appliquées dans les travaux de [2]. Le problème majeur avec cette approche est que l'idée de leur travail a été bien présentée alors qu'aucune étude expérimentale n'a été menée afin de montrer l'efficacité (exactitude) de leur approche.

Dans ce mémoire, nous améliorons sur cette approche en proposant d'utiliser un nouveau type de règles séquentielles nommées règles séquentielles partiellement ordonnée [3] à la place des règles séquentielles standard. Les règles séquentielles partiellement ordonnées ont une propriété intéressante d'être plus général que les règles séquentielles standard. Par conséquent, plusieurs règles séquentielles standard peuvent être représentées par une seule règle séquentielle partiellement ordonnée.

Une autre caractéristique intéressante des règles partiellement ordonnées est la notion de fenêtre de temps (time window). Avec ces règles, l'utilisateur peut générer des règles de la forme X-> Y ou X et Y doivent être proche les uns aux autres en respectant le temps. Par exemple, un utilisateur veux générer les règles apparues dans trois itemsets consécutifs dans les séquences ce qui est très adapté au problème de prédiction de future route ou le prochain segment de route qui doit être prédit et non pas n'importe segment dans le futur dans le cas des règles séquentielle standard. En outre, ce genre des règles séquentielles ont présenté une grande amélioration de l'exactitude de prédiction, comparant avec les règles séquentielles standard, lors de leurs application pour la tâche de prédiction des recommandations web avec réel et large datasets. [3]

3- Contribution

Les principales contributions de ce travail sont résumées comme suit:

1. Nous développons un nouveau modèle prédiction des routes basées sur les règles séquentielles.
2. Introduire et appliquer un nouveau type de règles séquentielles.
3. Nous avons mené une étude expérimentale sur le modèle sur une large dataset.

4- Organisation de mémoire

Le reste de ce mémoire est organisé comme suit. Le chapitre1 présente des définitions et techniques de la fouille de données requises pour la compréhension de ce domaine. Chapitre2 donne quelques définitions liées à notre problème de prédiction et présente l'état de l'art des travaux existants dans la littérature. Nous concentrons le plus sur la fouille de données séquentielles. Le troisième est composé de deux principales parties: présentation de l'approche et étude expérimentale. La première partie est consacrée à la présentation de notre nouvelle approche de prédiction des routes. L'évaluation du modèle proposé ainsi que les résultats expérimentaux sont montrés dans la deuxième partie. À la fin de ce mémoire, nous présentons notre conclusion générale. Nous y dressons le bilan de notre approche et nous présentons les perspectives de recherche.

I.1 Introduction

La fouille de données est un domaine qui est apparu avec l'explosion des quantités d'informations stockées, avec le progrès important des vitesses de traitement et des supports de stockage. La fouille de données (FD) vise à découvrir, dans les grandes quantités de données, les informations précieuses qui peuvent aider à comprendre les données ou à prédire le comportement des données futures. La fouille de données utilise depuis son apparition plusieurs outils de statistiques et d'intelligence artificielle pour atteindre ses objectifs. Dans ce chapitre on s'intéresse au définition globale de fouille de données puis on passant sur les taches utilisés au FD puis on détaille sur les techniques de FD et les algorithmes appropries de chaque technique finalement on va terminer avec quelque d'autre type de FD avancées. On utilise dans cette mémoire les termes suivants pour exprimer la notion de fouille de données **FD** ou **DM** Data mining.

I.2 Définition de fouille de données

La fouille de données est définie comme étant le processus de découverte des nouvelles connaissances en examinant de larges quantités de données (stockées dans des entrepôts) en utilisant les technologies de reconnaissance de formes de même que les techniques statistiques et mathématiques. Ces connaissances, qu'on ignore au début, peuvent être des corrélations, des patterns ou des tendances générales de ces données. [4]

La fouille de données a aujourd'hui une grande importance économique du fait qu'elle permet d'optimiser la gestion des ressources (humaines et matérielles).

Elle est utilisée par exemple :

– Organisme de crédit : pour décider d'accorder ou non un crédit en fonction du profil du demandeur de crédit, de sa demande, et des expériences passées de prêts.

– Optimisation du nombre de places dans les avions, hôtels, sur réservation. [5]

I.3 Les motivations des fouilles de données

La fouille de données est motivée par les arguments suivants :

- ✓ Les données traitées concernent à la fois des attributs qualitatifs et quantitatifs, ce qui justifie les étapes de discrétisations pour obtenir des contextes booléens ;
- ✓ Les données sont volumineuses : des objets par millions, des attributs par milliers.
- ✓ Ces caractéristiques posent de nombreux problèmes algorithmiques ;
- ✓ La fouille de données poursuit un but d'exhaustivité des connaissances découvertes.

- ✓ À la différence des techniques statistiques, ce ne sont pas seulement les tendances globales des données qui sont recherchées mais également des propriétés locales qui concernent un petit nombre d'objets ;
- ✓ Dans l'optique des méthodes d'exploration qui permettent d'aider l'expert dans sa prise de décision, il est souhaitable que l'aide fournie soit clairement justifiée, expliquée. [6]

1.4 Le processus de l'ECD

Le processus de l'extraction de connaissance à partir de données) ou KDD (Knowledge Discovery in Databases) est un processus interactif et itératif, impliquant de nombreuses étapes avec beaucoup de décisions faites par l'utilisateur. Ce processus découpé en 5 étapes. [7]

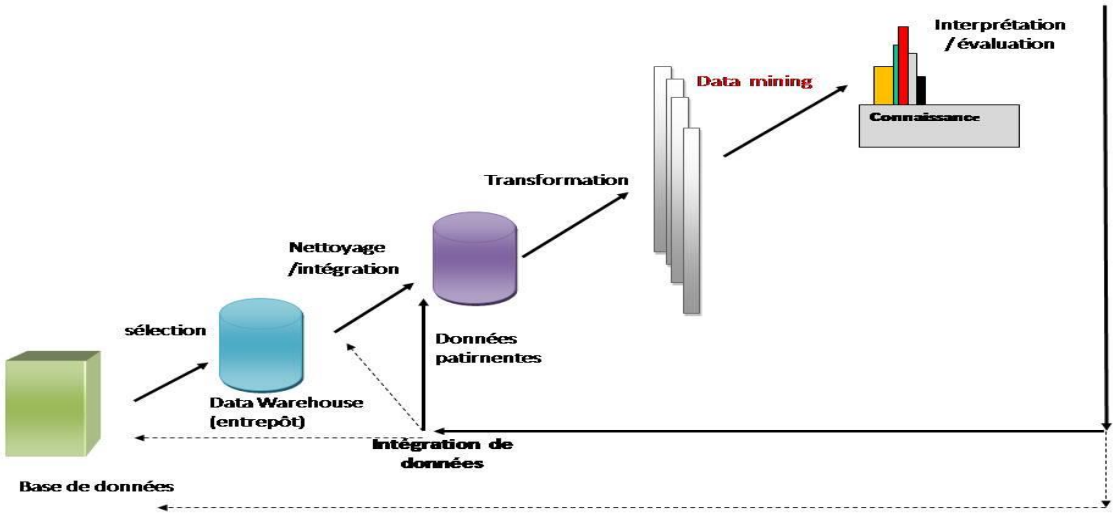

Figure 1-1: les étapes de processus d'ECD

I.4.1 La sélection de données

Dans cette phase les données se filtrent par la réduction de la dimension et de la taille de données. La réduction peut être faite par des techniques statistiques d'échantillonnage . [8]

I. 4.2 Le nettoyage et l'intégration des données

Cette étape consiste à la détection et la suppression des erreurs, du bruit, de l'incohérence de données pour améliorer leur qualité et l'intégration de données de sources multiples. [8]

I.4.3 La transformation de données

Cette étape consiste à préparer les données brutes et les convertir en données appropriées. [8]

I.4.4. Data mining

Le data mining (DM) ou fouille de données est au cœur du processus d'ECD, il se réfère à une série d'activités comme le choix du type de la tâche de DM, la sélection de la technique et d'algorithme de DM et l'extraction des modèles.

D'abord, le type de la tâche de DM doit être choisi en se basant sur l'objectif attendu, une technique de DM appropriée est alors utilisée. Une fois que cette technique est choisie, un algorithme particulier est associe. Le choix d'un algorithme de DM inclut une méthode pour chercher les modèles dans les données. Cette décision doit apparier la technique de DM particulière à l'objectif global de l'ECD. Tout le problème de DM réside dans le choix de la technique adéquate à un problème donné. Il est également possible de combiner plusieurs techniques pour essayer d'obtenir une solution optimale globale. [7]

I.4.5 Interprétation et évaluation

Analyser l'intérêt de la connaissance (résultat) et vérifier sa validité (sur le reste de la base de données). Réitérer le processus si nécessaire, gérer la connaissance découverte par la mettre à la disposition des décideurs et l'échanger avec d'autres applications (système expert, etc).

I.5 Communautés impliquées

Le DM s'implique dans tous les domaines Voici un liste non exhaustive des applications possibles du DM par secteur d'activités.

- Grande distribution et VPC: Analyse des comportements des consommateurs, recherche des similarités des consommateurs en fonction de critères géographiques ou socio-démographiques, prédiction des taux de réponse en marketing direct, vente croisée et activation sélective dans le domaine des cartes de fidélité, optimisation des réapprovisionnements.
- Laboratoires pharmaceutiques: Modélisation comportementale et prédiction de médications ou de visites, optimisation des plans d'action des visiteurs médicaux pour le lancement de nouvelles molécules, identification des meilleures thérapies pour différentes maladies.

- Banques: Modélisation prédictive des clients partants, détermination de pré-autorisations de crédit.
- Assurance: Modèles de sélection et de tarification, analyse des sinistres, recherche des critères explicatifs du risque ou de la fraude, prévision d'appel sur les plates-formes d'assurance directe.
- Aéronautique, automobile et industries: Contrôle qualité et anticipation des défauts, prévision des ventes, dépouillement d'enquêtes de satisfaction.
- Transport et voyagistes: Optimisation des tournées, prédiction de carnets de commande, marketing relationnel dans le cadre de programmes de fidélité.
- Télécommunications, eau, énergie: Simulation de tarifs, détection de formes de consommation frauduleuses, classification des clients selon la forme de l'utilisation des services, prévision de ventes.

I.6 Les tâches de fouille de données

L'utilisation du DM dans différents domaines a peu résoudre une multitude de problèmes d'ordre intellectuel, économique ou commercial. Ces problèmes peuvent être exprimés, dans leur formalisation, dans l'une des six tâches suivantes : la classification, l'estimation, la prédiction, le groupement par similitude, l'analyse des clusters et la description.

Les trois premières tâches sont des exemples de datamining supervisé dont le but est d'utiliser les données disponibles pour créer un modèle décrivant une variable particulière prise comme but en termes de ces données. Le groupement par similitude et l'analyse des clusters sont des tâches non-supervisées où le but est d'établir un certain rapport entre toutes les variable

I.6.1 La classification

La classification consiste à examiner des caractéristiques d'un élément nouvellement présenté afin de l'affecter à une classe d'un ensemble prédéfini. [9]

Elle permet de créer des classes d'individus .Celles-ci sont discrètes:

Homme / femme, oui / non, rouge / vert / bleu, ...etc. [9]

Un exemple des techniques utilisées pour accomplir cette tâche sont les arbres de décisions.

I.6.2 L'estimation

L'estimation est similaire à la classification à part que la variable de sortie est numérique plutôt que catégorique. En fonction des autres champs de l'enregistrement, l'estimation

consiste à compléter une valeur manquante dans un champ particulier. La technique la plus appropriée à l'estimation est: le réseau de neurone.

I.6.3 La prédiction

La prédiction ressemble à la classification et à l'estimation mais dans une échelle temporelle différente. Tout comme les tâches précédentes, elle s'appuie sur le passé et le présent mais son résultat se situe dans un futur généralement précisé. La seule méthode pour mesurer la qualité de la prédiction est d'attendre. [8]

Les techniques les plus appropriées à la prédiction sont : les règles d'association, les arbres de décision, les réseaux de neurones.

I.6.4 Le groupement par similitude

Le groupement par similitude consiste à déterminer quels attributs "vont ensemble". Elle est considérée comme la tâche la plus répandue dans le monde du business, où elle est appelée l'analyse du panier du marché. Elle présente l'association des recherches pour mesurer la relation entre deux et plusieurs attributs. Les règles d'associations sont de la forme "Si antécédent, alors conséquent". [8]

I.6.5 L'analyse des clusters

Le clustering (ou la segmentation) est le regroupement d'enregistrements ou des observations en classes d'objets similaires. Un cluster est une collection d'enregistrements similaires l'un à l'autre, et différents à ceux existants sur les autres clusters. La différence entre le clustering et la classification est que dans le clustering il n'y a pas de variables sortantes. La tâche de clustering ne classifie pas, n'estime pas, ne prévoit pas la valeur d'une variable sortantes. [7]

I.6.6 La description

C'est souvent l'une des premières tâches demandées à un outil de DM. On lui demande de décrire les données d'une base complexe. Cela engendre souvent une exploitation supplémentaire en vue de fournir des explications. [8]

I.7 Les techniques de fouille de données FD

Les techniques de FD ou DM peuvent être classifiés selon l'information à priori les techniques supervisées et celles dites non supervisées.

I.7.1 Les techniques supervisées

Les techniques supervisées proposent une classification des objets en s'appuyant sur un modèle préétabli d'exemples ou d'échantillons sélectionnés au hasard.

Ces techniques ont pour objectif d'assurer les tâches supervisées : classification et régression. On peut citer dans une liste non exhaustive : les arbres de décision, les réseaux de neurones, les classificateurs bayésiens, les machines à vecteurs de support, les K plus proches voisins et la régression.

I .7.1.1. Arbres de décision

Un arbre de décision permet de représenter les objets étudiés sous une forme arborescente, selon une hiérarchie des attributs déterminée par un calcul d'entropie. Ces méthodes sont populaires pour la présentation synthétique des données qu'elles fournissent, ainsi que pour la clarté des explications concernant la décision rendue.

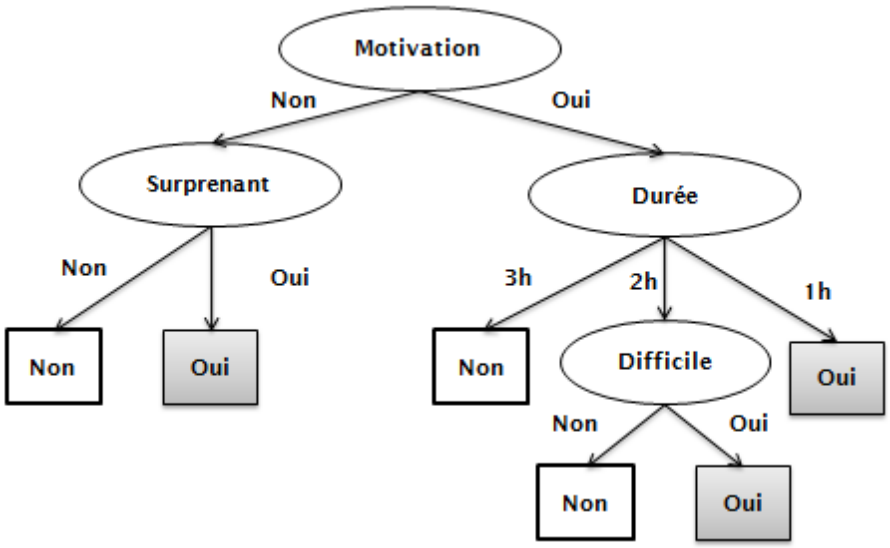

Figure 1.2: Un arbre de décision simple

Pour construire un tel arbre, plusieurs algorithmes existent : *ID3, CART, C4.5,* etc.

I.7.1.2 Réseau de neurones

Les réseaux neurones (ou réseaux connexionnistes) utilisent l'analogie avec l'architecture physiologique du cerveau humain.

Les neurones sont des entités élémentaires qui reçoivent des signaux en entrée et transmettent à d'autres neurones des signaux de sortie qui résultent d'une combinaison des signaux d'entrée. Les premiers neurones d'entrée sont reliés aux valeurs des attributs d'un objet. Par exemple pour la reconnaissance d'images, ce sont les pixels allumés ou éteints. Les neurones

de sortie indiquent la valeur finale de la décision, c'est-à-dire la classe de l'objet. Des neurones intermédiaires sont organisés en couches et l'ensemble constitue un réseau.

Pendant sa phase d'apprentissage, des objets sont présentés au réseau et lorsque la réponse diffère de la classe supervisée, un algorithme de rétro-propagation modifie les comportements des neurones intermédiaires. Techniquement, un tel réseau calcule des équations d'hyper-plans séparateurs des classes, selon un algorithme de descente de gradient. Les réseaux de neurones ont connu un rapide succès, particulièrement pour le traitement des images. Cependant, il est très difficile d'expliquer comment la décision est rendue par ces réseaux, du fait de la grande complexité de leur architecture. [6]

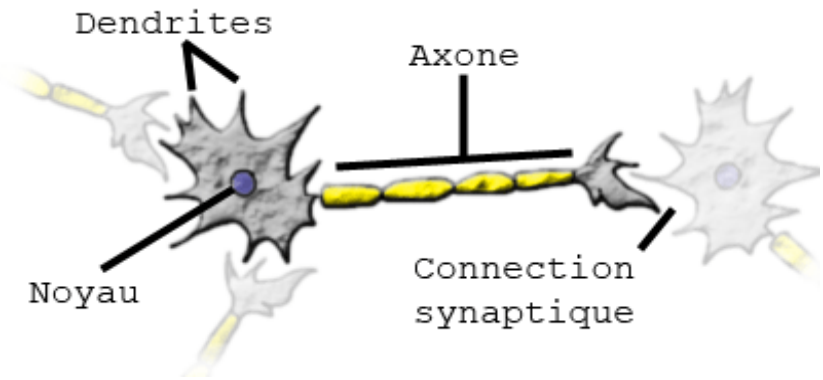

Figure 1-3: Neurones biologique

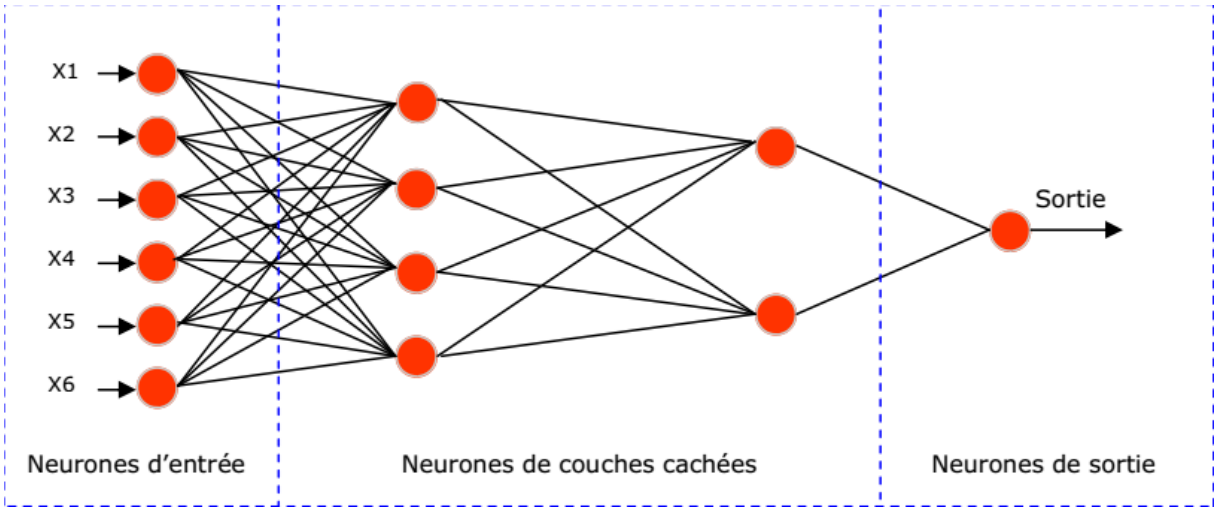

Figure 1-4: Réseau de neurone artificiel

I .7.1.3 Les classificateurs bayésiens

Les classificateurs bayésiens sont des classificateurs statistiques. Ils peuvent prévoir des probabilités d'appartenance aux différentes classes, telles que la probabilité qu'un échantillon donné appartient à une classe particulière.

La classification bayésienne est basée sur le *théorème de Bayes*. Les études comparant des algorithmes de classification ont trouvé un classificateur bayésien simple connu sous le nom de *classificateur bayésien naïf* aussi performant que les autres classificateurs : d'arbre de décision, réseau de neurones, etc. .

Ces classificateurs bayésiens ont également montré de grande précision une fois appliqués même dans des grandes bases de données. Les classificateurs bayésiens naïfs supposent que l'effet d'une valeur d'attribut sur une classe donnée est indépendant des valeurs des autres attributs. Cette prétention s'appelle l'*indépendance conditionnelle de classe*. Elle est faite pour simplifier les calculs impliqués et, dans ce sens, il est considéré «naïf».

Les réseaux bayésiens de croyance (belief) sont des modèles graphiques, qui à la différence des classificateurs bayésiens naïfs, permettent la représentation des dépendances parmi des sous-ensembles d'attributs. Des réseaux bayésiens belief peuvent également être employés pour la classification. [2]

- **Le théorème de Bayes**

  Soient *A*, *B* et *C* trois événements. Le théorème (ou règle) de Bayes démontre que :

$$Pr[A|B,C] = \frac{Pr[B|A,C]Pr[A|C]}{Pr[B|C]} \quad \text{ou}$$

  – *Pr*[*B*|*A*, *C*] est la vraisemblance de l'´evénement *B* si *A* et *C* sont vérifiés ;

  – *Pr*[*A*|*C*] est la probabilité *a priori* de l'´evénement *A* sachant *C* ;

  – *Pr*[*B*|*C*] est la probabilité marginale de l'´evénement *B* sachant *C* ;

  – *Pr*[*A*|*B*, *C*] est la probabilité *a posteriori* de *A* si *B* et *C*.

  Dans cette formulation de la règle de Bayes, *C* joue le rôle de la connaissance que l'on a. [1]

1.7.1.4. Les K plus proches Voisins (K-Nearest Neighbors)

C'est une méthode de classification qui propose une analyse de similitude entre des données en utilisant la distance entre elles .L'algorithme fait un calcul de distance entre tous les individus et chaque objet est classé dans le groupe où se trouvent ses *K* plus proches voisins,

Quand on parle de voisin cela implique la notion de distance ou de dissimilarité. La distance la plus populaire est la distance euclidienne:

$$D((x_1,x_2,\ldots,x_p),(u_1,u_2,\ldots,u_p) = \sqrt{q(x_1 - u_1))2 + ((x_2 - u_2))2 + \ldots + ((x_p - u_p))2}$$

Contrairement aux autres méthodes de classification (arbres de décision, réseaux de neurones, etc.) l'algorithme KNN ne construit pas de modèle à partir d'un échantillon d'apprentissage, mais c'est l'échantillon d'apprentissage, la fonction de distance et la fonction de choix de la classe en fonction des classes des voisins les plus proches, qui constituent le modèle.[2]

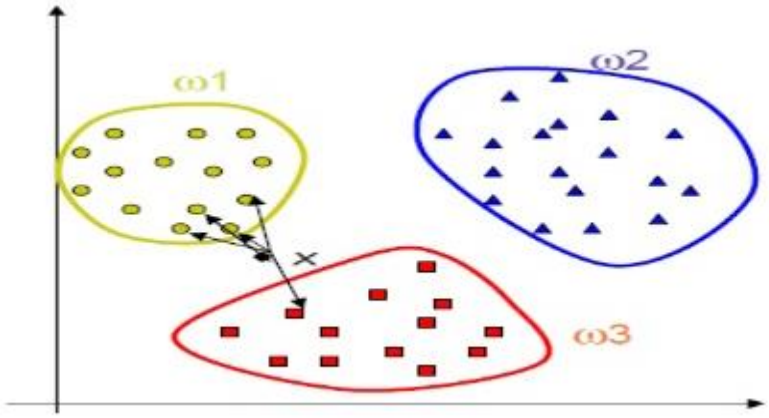

Figure 1-5: l'algorithme KNN

I.7.1.5 La régression

L'analyse de régression comme la corrélation ont leur origine dans le travail de célèbre généticien *Francis Galton* (1822-1911). Dans les statistiques, l'analyse de régression signifie le modèle mathématique qui établit (concrètement, par l'équation de régression) le raccordement entre les valeurs d'une variable donnée (variable dépendante) et les valeurs d'autres variables (facteur prédictif/variables indépendantes). L'exemple le plus connu de la régression est peut- être l'identification du rapport entre la taille d'une personne et le poids, évaluant de ce fait un poids idéal pour une taille spécifique. L'analyse de régression se rapport en principe :

– À la détermination d'un rapport quantitatif corrélation (liaison) entre des variables multiples;

– Aux prévisions des valeurs d'une variable selon les valeurs d'autres variables (déterminant l'effet des «variables de facteur prédictif » sur la «variable dépendante») [7].

I.7.2 Les techniques non supervisées

Les méthodes non supervisées construisent des modèles sans information à priori. Ces méthodes ont pour objectif d'assurer les tâches non supervisées: clustering, règles d'association, etc.

I.7.2.1 Techniques de classification non supervisées

Beaucoup de techniques de classification non supervisées ou clustering ont été proposées dans la littérature. Les algorithmes de clustering peuvent être classifiés selon la méthode adoptée pour définir des clusters en :

– Clustering par partitionnement (Partitional clustering)
– Clustering hiérarchique (Hierarchical clustering)
– Clustering basé sur la densité (Density-based clustering)
– Clustering basé sur les grilles (Grid-based clustering)

I.7.2.2 Clustering par partitionnement

Les techniques par partitionnement créent un partitionnement des points de données, d'un seul niveau. Si *k* est le nombre désiré de clusters, alors les approches par partitionnement trouvent typiquement tous les *k* clusters immédiatement.

Les techniques par partitionnement sont divisées en deux sous-catégories principales: les algorithmes basés sur les médoïdes et les algorithmes basés sur les centroïdes. Pour cette technique il ya deux algorithmes les plus connus: *K-medoid* et *K-means*.

I.7.2.3 Clustering hiérarchique

Le clustering hiérarchique construit une hiérarchie de clusters, ou, en d'autres termes, un arbre de clusters ou dendrogramme. Ce dendrogramme décrit l'ordre dans lequel les points sont fusionnés (vue de bas en haut) ou les clusters sont fractionnés (vue de haut en bas)

Il y a deux approches de base pour générer un clustering hiérarchique :

– Agglomérative: commence par des clusters d'un seul point (singleton) et fusionne récursivement deux ou plusieurs clusters les plus appropriées.

– Divisive: commence par un cluster de tous les points de données et fractionne récursivement le cluster le plus approprié.

I.7.2.4 Clustering basé sur la densité

Cette méthode de clustering est basée sur la notion de la densité. Son idée générale est d'accroitre le cluster tant que la densité (nombre d'objets ou de points de repères) dans le

voisinage dépasse un certain seuil ce qui signifie que pour chaque point de repères dans un cluster donné, le voisinage d'un rayon donné doit contenir au moins un nombre minimum de points. Un des algorithmes les plus bien connus de cette catégorie est le *DBSCAN*

I.7.2.5 Clustering basé sur les grilles

Le clustering basé sur les grilles divise l'espace en un nombre fini de cellules qui forment une structure de grille sur laquelle toutes les opérations pour groupement sont effectuées. L'avantage principal de l'approche est sa durée de transformation rapide, qui est typiquement indépendante de la quantité des données.

L'algorithme *STRING* est un exemple typique des algorithmes basés sur les grilles, les algorithmes *CLIQUE* et *WaveCluster* sont deux algorithmes de groupement qui se basent sur les grilles et la densité en même temps.

I.7.3 Les règles d'association

L'extraction des règles d'association est sans doute une tâche «phare» de DM qui a attiré le plus l'attention des chercheurs et pour laquelle beaucoup de travaux ont été effectués. L'analyse du panier de la ménagère est l'une des applications typiques de l'extraction des règles d'association.

Une règle d'association de la forme $T_1 \rightarrow T_2$ où T_1 et T_2 sont des motifs. T_1 est appelé la prémisse ou l'antécédent de la règle, et T_2 est la conclusion ou le conséquent de la règle où $T_1 \cap T_2 = \varphi$. Cette technique permet la découverte de règles intelligibles et exploitables dans un ensemble de données volumineux, règles exprimant des associations et corrélations entre motifs ou attributs dans une base de données.

L'extraction des règles d'association se fait en général en deux étapes. La première consiste à extraire l'ensemble des itemsets fréquents (motifs fréquents), la deuxième génère des règles à partir de ces motifs. [2]

Définition 1(Motif fréquent): Soient $I = i_1, ..., i_m$ un ensemble de m items et $B = t_1, ..., t_n$ une base de données de n transactions. Chaque transaction est composée d'un sous ensemble d'items $I_0 \in I$. Le sous-ensemble I' de taille k est appelé un k-motif. Une transaction t_i contient un motif I' si et seulement si $I_0 \in t_2$. Le support d'un motif I' est la proportion de transactions de B qui contiennent I'. Le support est donné par la formule suivante. [2]

$$Support(I') = \frac{|t \in B, I' \subseteq t|}{|t \in B|}$$

Un motif dont le support est supérieur ou égal au seuil minimal du support *minsup*, défini par l'utilisateur, est appelé un motif fréquent. [7]

I.7.3.1 Extraction des motifs fréquents

Cette section a pour objectif de présenter brièvement les principes de fonctionnement de quelques algorithmes d'extraction des motifs fréquents. La littérature fait état d'un nombre de plus en plus important.

. Cependant, on se retrouve face à un très grand nombre des motifs fréquents, ce qui réduit considérablement, non seulement l'efficacité mais aussi l'utilité de la tâche. En effet, un grand nombre de motifs fréquents conduit à beaucoup de règles d'association, car rappelant qu' à partir d'un seul *k-itemset* fréquent on peut générer 2 *k* règles. Ceci impose à l'utilisateur de fouiller dans les règles pour trouver les règles les plus intéressantes. C'est pourquoi, d'autres alternatives ont été proposées, notamment dans l'extraction des représentations condensées des motifs fréquents.

I.7.3.2 Génération des motifs fréquents

Plusieurs algorithmes traitent le problème de la recherche des motifs fréquents. Nous citons, à titre d'exemple : Apriori, AprioriTID, Partition, etc.

I.7.3.3 la génération des règles d'association

Pour générer les règles d'association, on considère l'ensemble *F* des motifs fréquents trouvés dans la phase de génération des motifs fréquents. Pour chaque motif fréquents *l*, on considère tous ses sous-ensembles. À partir de ces sous-ensembles fréquents, on génère toutes les règles solides. La génération de règles d'association est beaucoup moins coûteuse que la recherche des motifs fréquents, car il n'est plus nécessaire de faire des parcours coûteux de la base de transactions. [7]

I.8 La fouille de données séquentielles (extraction des motifs séquentiels)

L'extraction de motifs séquentiels est un sujet d'exploration de données concerné à trouver des modèles statistiquement pertinentes entre les exemples de données où les valeurs sont livrées dans une séquence. Il est généralement présumée que les valeurs sont discrètes, et donc l'exploitation minière des séries chronologiques est étroitement liée, mais généralement considéré comme une activité différente. L'extraction de motifs séquentiels est un cas particulier de l'extraction de données structurées**.**

Définition 1: Soit A = {a_1, a_2, ..., a_n} un ensemble de n attributs. Chaque attribut a_i de A est également appelé item. Un itemset I est un ensemble d'items non vide noté I = (i_1, i_2, ..., i_k) ou ij ∈ A et ∀i, j ∈ I, i ≠ j. Si l'itemset I contient k éléments, il est appelé k − itemset.

Définition 2: Soit C un ensemble de clients et D un ensemble de dates. Soit C' ∈ C un client et d ∈ D une date. Une transaction constitue, pour C, l'ensemble des items I associés à (achetés par) C à la date D et s'écrit sous la forme d'un triplet : < C', D', I >.

Définition 3: Un motif séquentiel (également appelé séquence) est une liste ordonnée non vide d'itemsets notée < s_1 s_2 ...s_k >, où s_j est un itemset.

Une séquence de données DS représente les achats d'un client : soit T_1, T_2, ..., T_n les transactions d'un client, ordonnées par id − date croissants et soit itemset (Ti) l'ensemble des items correspondant à Ti, alors la séquence de données de ce client est D =< itemset (T_1) itemset (T_2) ... itemset (T_n) >. Exemple 1 Soit C un client et S =< (a) (b c)(f) >, la séquence de données représentant les achats de ce client. S peut être interprétée par C a acheté l'item a, puis en même temps les items b et c et en n l'item f. [10]

Plusieurs algorithmes ont été proposés pour trouver tous les motifs séquentiels dans une base de données tels que CM-SPADE, PrefixSpan et GSP. Ces algorithmes prennent en entrée une base de données de séquence et un seuil de support minimal (minsup). Ensuite, ils vont sortir tous les motifs séquentiels ayant un support au moins minsup. Ces modèles sont censés être les motifs séquentiels fréquents.

Définition 4: Une base de données de clients BD est l'ensemble des séquences de données des clients. Soit C l'ensemble des clients, BD est donnée par : BD = {DSc|c ∈ C)}, [12]

ID	Séquence
S1	{(a)(a.b)(d)}
S2	{(a.b)(a.b.d)(b.c)(f)}
S3	{(b.c)(b.d)(f)}

Tableau 1.1: représente une base de donne de séquence

Définition 5: Le support d'un itemset I dans BD, noté supp (I, BD) ou supp (I) quand le contexte est clair, est le pourcentage de tous les clients dans BD dont l'union des transactions contient I : supp(I, BD) = |{DS ∈ BD | I ⊆ S ∀d ∈D itemset (DSd)}|/|{DS ∈ BD}|.[12]

1.8.2 Découvrir les règles séquentielles

La découverte de la règle séquentielle a été proposée comme une alternative à l'extraction de motifs séquentiels de prendre en compte la probabilité qu'un modèle sera suivi.

Une règle séquentielle est une règle de la forme X => Y où X et Y sont des ensembles d'éléments (itemsets). Une règle X => Y est interprétée comme si les objets dans X se produit (dans un ordre quelconque), alors il sera suivi par les éléments de Y (dans l'ordre). Par exemple, considérons la règle {a} => {e, f}. Cela signifie que si un client article d'achat "a", le client sera ensuite acheter les articles "e" et "f". Mais l'ordre parmi les objets dans {e, f} est pas important. Cela signifie qu'un client peut acheter "e" avant "f" ou "f" avant "e".

Pour trouver des règles séquentielles, deux mesures sont généralement utilisées: le support et la confiance. Le support d'une règle X => Y est le nombre de séquences contient les articles de X suivis par les articles de Y. Par exemple, le support de la règle {a} => {e, f} est de 3 séquences car {a} apparaît avant les éléments de {e, f} en trois séquences (s1, s2 et S3).

La confiance d'une règle X => Y est le support de la règle divisée par le nombre de séquences contenant les articles de X. Il peut être compris comme la probabilité conditionnelle P (Y | X). Par exemple, la confiance de la règle {a} => {e, f} est 1 (ou 100% si écrit comme un pourcentage), parce que chaque fois qu'un article d'achat du client "a", il a ensuite acheter "e" et "f" dans la base de données par exemple. Un autre exemple est la règle {a} => {b}. Cette règle a un support de 2 séquences et une confiance de 0,66 (soit 66%). [11]

I.8.3 Les méthodes et les algorithmes des motifs séquentiels

Plusieurs algorithmes ont été propose pour recherche des motifs séquentiels ,nous citons par exp : GSP (Generalized Sequential Patterns) , SPADE et VPSP (Vertical Prefix-Tree for Sequential Pattern).

I.8.4 Les règles séquentielles standard et partialement ordonnées

Dans les règles séquentielles il existe deux types: les règles standard et les règles partiellement ordonnée.

I.8.4.1 Les règles séquentielles standard

Une règle séquentielle est une règle d'association à laquelle on rajoute le facteur temps. La recherche de règles séquentielles est un processus complexe et passe par différentes étapes notamment, la recherche de motifs séquentiels, etc.

D'autre part est une règle séquentielle SR (X⇒ Y) peut être défini comme une relation entre les deux séquences en X, Y ⊆ I tel que X∩Y = O, X, Y ≠ Ø. X est appelé antécédent tandis que Y est le conséquent de SR.

(Support): Le support d'un itemset X est défini comme étant le nombre de séquences du SDB où les articles de X se produit, divisé par le nombre de séquences dans SDB.

(Confiance): SR donnée (X⇒ Y) une règle séquentielle, la confiance des SR est Conf (SR) = Support (XUY) / support (Y)

I.8.4.2 Les règles séquentielles partiellement ordonné

Considérons le PR de la règle (P⇒L). PR est dit être partiellement ordonné si les éléments de P se produisent dans une séquence (dans un ordre quelconque), les éléments aura lieu par la suite dans la même séquence (dans un ordre quelconque). Simplement dit, les exigences de une commande séquentielle à l'intérieur de l'antécédent X et à l'intérieur de la Y conséquente de la règle sont éliminé. Mais l'exigence d'une relation séquentielle entre l'antécédent et en conséquence d'une règle est préservée. [12]

La règle {rs2, rs3} ⇒ {rs4} apparaît en $1^{ère}$ et $4^{éme}$ séquences. Cela signifie que si route secteur rs2 et rs3 apparaît dans une séquence dans un ordre quelconque, il sera suivi par rs4.

Comme tout autre algorithme pour l'extraction des règles séquentielles, deux seuils *minsup* et *minconf* nécessité être réglé par un utilisateur.

Une règle séquentielle partiellement ordonné (X⇒Y) est généré si et seulement si le support de ses articles est supérieur à minsup et sa confiance est supérieure à minconf. Par exemple, considérons-la motif de mouvement SDB. La règle {rs2, rs3} ⇒ {rs4} a un appui de 0,5 car il apparaît pour V1 et V4. De plus, sa confiance est 1 parce que son antécédent seulement apparaît dans V1 et V4.

Les règles séquentielles partiellement ordonnées ont la propriété intéressante d'être plus générale que les règles séquentielles standards.

I.9 Fouille de données spatiale

La FD spatiale est l'application de l'exploration de données pour les modèles spatiaux. Dans le secteur minier de données spatiales, les analystes utilisent l'information géographique ou spatiale pour produire l'intelligence d'affaires ou d'autres résultats. Cela nécessite des techniques et des ressources spécifiques pour obtenir les données géographiques dans des formats pertinents et utiles. Les défis impliqués dans l'extraction de données spatiales

comprennent l'identification de motifs ou de trouver des objets qui sont pertinents pour les questions qui animent le projet de recherche. Les analystes peuvent être à la recherche dans un champ de base de données à grande ou une autre très grand ensemble de données afin de trouver seulement les données pertinentes, en utilisant des outils SIG (système informatique géographique) / GPS ou des systèmes similaires. [12]

I.10 Fouille de données temporelle

FD temporelle peut donc être définie exactement comme la fouille de donnée avec la particularité de bien traiter les données temporelles. Les étapes qui la composent sont aussi les mêmes sauf l'étape de transformation qui est un peu plus poussée pour pouvoir gérer la complication des données temporelles. [13]

I.11 Conclusion

Dans ce chapitre nous avons étendu les notions globales de fouille de données on s'intéresse aux différents algorithmes utilises au DM surtout les règles séquentielles qui ont un partie très important de notre travail qui est la prédiction des routes.

II.1 Introduction

De notre jour à cause des technologies existantes tel que les réseaux wifi et le GPS et les smart phones plusieurs tâches et recherches sont possible à réaliser dans les VANETs[1] un de ces tâches est la prédiction des routes pour les utilisateurs des routes. Dans ce contexte il existe plusieurs recherches et essayes avec différentes technologies et méthodes, tel que les méthodes statiques comme les modèles markovien et les règles d'association.

Dans ce chapitre, on va présenter les définitions de base concernant la prédiction des routes et quelques techniques utilisé pour faire clair ce sujet.

II.2 Définitions

II.2.1 Road network

Un réseau routier est un graphe orienté G (N, RS), où N est l'ensemble des jonctions reliés par des segments de route de l'ensemble de tous les possibles segments de route RS dans une carte donnée.

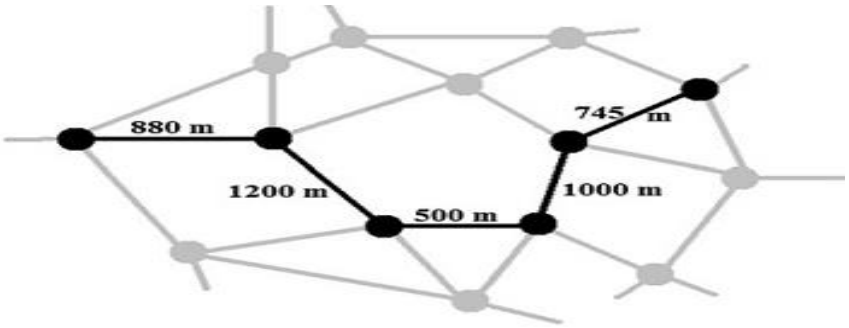

Figure 2.1: trace d'un voyage dans le road network

II.2.2 Segment de route (road segment)

Un segment de route avec un Sr identificateur unique est représenté par un bord unidirectionnel entre deux nœuds. Comme le montre la figure2.2. [14]

[1] **VANET**(véhicule Adhoc Network) :Un réseau VANET est un réseau de communication entre véhicules intelligents équipés de calculateurs, de périphériques réseau et de différents types de capteurs.

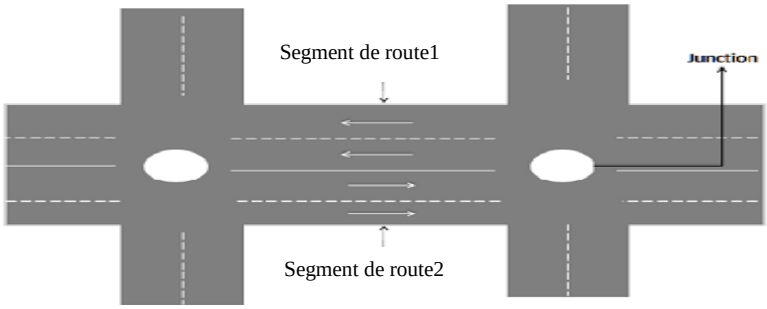

Figure 2-2: segment de route

II.2.3 Séquence de segment (chemin. Motif de mouvement)

On a l'ensemble des routes P= [r_1, r_2, r_3, r_4] est défini par une séquence de segment route traversé par un véhicule pendant son voyage dans une zone géographique. Par exemple dans le tableau 1. Il y a quatre véhicules V:{V1, V2, V3, V4} chaque véhicule a sa route exprimé par l'identificateur sr_i ou la véhicule V3 traverse le segment sr_2suivie par les segments sr6, sr7 successivement. [14]

Véhicule	Motif de mouvement
V1	$sr_0; sr_2; sr_3; sr_4$
V2	$sr_1; sr_2; sr_3; r_5$
V3	$sr_2; sr_6; rs_7$
V4	$sr_2; sr_3; sr_4; sr_6$

Table 2.1 : véhicules et ces segments de route

II.2.4 GPS, DGPS et MapMatching

Systèmes de positionnement global (GPS) ont été utilisés pour fournir des informations sur l'emplacement des objets qui sont équipés d'un récepteur GPS. Avec l'aide de 24 satellites stratégiquement placé en orbite autour de la Terre, des objets GPS compatible (fixes ou mobiles) peuvent être trouvé, la condition que la ligne de vue directe avec les satellites. en recevant les informations provenant de différents satellites, les véhicules peuvent estimer leur position selon cette information reçu avec des techniques telles que l'heure d'arrivée (Time of Arrival) (TOA) .

Cependant, ces informations ne sont pas exacts en raison de l'erreur de localisation connue dans les récepteurs GPS, qui peuvent atteindre jusqu'à 30 m. Cela peut être un problème dans d'applications VANET de types urgence qui nécessitent une localisation précise. Un autre problème avec le GPS est la ligne de contrainte de vue, et cela se produit lorsque des obstacles tels que des bâtiments, des tunnels, ou arbres par exemple obstruent ou peut-être interférer avec la ligne de mire directe des récepteurs, GPS a besoin pour recevoir des informations GPS. Cela pourrait aussi rendre la précision inapplicable pour les applications VANET qui nécessitent des informations de localisation précises.

Afin de gérer les problèmes d'erreur de localisation associée à GPS, une mise à jour technique a été introduite, qui fait usage d'un récepteur stationnaire GPS avec un emplacement connu et les erreurs de localisation similaire à d'autres récepteurs. Cette technique de localisation par GPS, connu sous le nom GPS différentiel (DGPS), fournit les véhicules se déplaçant avec des informations d'erreur différentielle entre la position réelle d'un récepteur GPS fixe et celui qui est reçu d'un satellite pour le même récepteur. Une fois que cette erreur est calculée par le récepteur GPS fixe, il diffuse à tous les autres véhicules afin de synchroniser leurs récepteurs en conséquence.

MapMatching a été considérée comme un outil utilisé pour améliorer la localisation d'un véhicule en mouvement et non pas comme une technique de localisation réelle. En mesurant la position d'un véhicule en mouvement à l'aide d'un dispositif de localisation (par exemple GPS), *MapMatching* peut trouver cet emplacement sur une carte (map) pré-chargée et repérer l'emplacement du véhicule sur ces cartes (maps). La collecte de différents points de localisation au cours d'une période de temps peut donner à l'utilisateur une approximative idée du voyage que ce véhicule a été à travers. [2]

II.3 Etat de l'art

A cause de l'importance de la prédiction des routes et possibilité de la réaliser plusieurs travaux ont été faites et plusieurs modèles ont été proposés pour atteindre une meilleure prédiction . ces modéls peuvent etre regrouper en deux catégories: les modèle statiques et les data maning.

II.3.1 modèle statistique

Plusieurs travaux ont été proposés dans les modéles satistiques comme le modéle markov, nous citerons quelques exemples:

- Simmons (2006) [15] ont utilisé le modèle de Markov caché (HMM) et l'information contextuelle (jour de la semaine, le temps et la vitesse du véhicule) dans un corpus de 46 voyages dans la région du Michigan, aux États-Unis. Le taux de prédictions correctes était de 98%. Cependant, seulement 5% des transitions d'un segment à un autre produit dans les intersections entre les routes, tandis que les 95% restants étaient connectés à un seul autre segment de route, ce qui réduit la difficulté de la prédiction du segment suivant. Pour les 5% de transition est survenue dans les virages, la vitesse de prévisions correctes se situait entre 70% et 80%.
- Le travail de (Uma Nagaraj2011) [16] utilise les modèles statistiques pour la prédiction des routes. Pour chaque modèle ils ont donné une route de véhicule comme entrée alors ils peuvent prédire seulement le segment suivant de la route. Basant sur le modèle de Markov simple ou caché (HMM) pour prédire la route de conduite à court terme, sur la route qui ne prend pas les conditions de circulation dynamiques en compte. Où en utilisant VMM qui déploie PST(Probabilistic Suffix Tree (PST) algorithm) pour générer les modèles de mobilité de prédiction des route à long terme, qui peut être appliqué pour des vrais conditions de circulation pour prédire la future trace de conducteur.
- En (2008) travail de Krumm [17] la mise au point de son modèle est la prédiction à court terme, à savoir, seuls les segments suivants. Son modèle utilise le modèle de Markov pour la prédiction, et après avoir observé les 10 derniers segments parcourus par un utilisateur, il est possible de prédir le prochain avec une précision de 90%. Pour prédire les 10 prochains segments , la précision baisse du taux à 50%.
- (Xipeng Wang 2015) [18] utilise un modèle de Markov de premier ordre pour construire une matrice de probabilité de transition contenant la probabilité associée à chaque liaison. Les algorithmes de réduction des données sur la matrice de probabilité ont été proposées. Un problème majeur lié à cette approche est que l'ensemble de données utilisé pour valider le travail, contient un seul conducteur qui est insuffisant pour représenter l'exactitude de précision approuvé dans leur travail.
- (Attila István Petróczi 2009) [19] propose trois modèles de prédiction, le premier modèle statistique basé sur les itemsets fréquents, deuxieme modèle de Markov de n ordre où n <4 et le troisieme est un Pattern Matching Modèle adaptant .le modèle. La meilleure précision a été rapportée par le second modèle, avec 70%.

- Dans (Disheng Qiu 2013) [20] les auteurs ont mis en place une approche probabilistique basé sur un modèle de Markov caché (HMM). Ils ont essayé de surmonter certaines limites liées aux données recueillies à partir d'appareils GPS tels que l'incertitude, et donc d'améliorer la précision de prédiction de route et le prédiction de destination. Comme mentionné dans la section précédente, une limitation de problème lié à des modèles de Markov , avec l'hypothèse que le route suivant ne dépend que de la route réelle courant pour un conducteur.
- Dans (Francisco Dantas.2015) [21] ont proposé la prédiction des routes et destinations par l'utilisation de technique de partial matching (PPM). La base de donnée a été créé à partir des déplacements réels et capturés à l'aide d'une application installé dans les smart phones des participant de ce travail ,les résultats obtenu pour les 15% des route l'occurrence est de 32.02%.et 50% des routes 45.06% est obtenu .et 85% des routes 46.02%.

II.3.2 modéle data minning

L'application de data mining dans le cadre de la prédiction de route a deux formes : les réseaux de neurones et l'extraction des motifs séquentielles.

- parmi les travaux base sur les réseaux de neurones on trouve deux architectures de réseau de neurones ont été utilisé: l'architecture de feed-forward (Tomás Mikluscák 2012) [22], et l'architecture bidirectionnelle récurrente (Alexandre de Brébisson 2015) .
- En (Alexandre de Brébisson 2015) [23] une approche des réseauxneurones presque entièrement automatisé pour prédire la destination d'un taxi basé au début de sa trajectoire et les métadonnées associées (heure de départ, id du pilote, l'information de client). un réseau neuronal bidirectionnel récurrent est utilisé pour encoder la representation de préfixe de taxis,et les metadonnées associées.
- Dans (Tomáš Mikluščák, Michal Gregor, and Aleš Janota.2012) ils sont travaillés sur des algorithmes et méthodes pour l'utiliser dans la prédiction des routes, ces méthodes sont base sur les réseaux des neurones, et utilisent les routes parcourus par la voiture pour la data set, deux data set avec différentes tailles sont utilisé dans ce modèle. L'occurrence approche de 60%.
- (Chen L. 2010) [24] ont tenté de prédire en même temps la destination et le futur itinéraire d'une personne. Depuis les données d' un vrai GPS, les auteurs ont proposé de regrouper les lieux importants que l'utilisateur peut déroger ou aller , en

utilisant FBM (Forward-Backward Matching) algorithme de clustering. Les trajectoires sont abstraites puis extraites les modèles de mouvement à l'aide d'un algorithme de CRPM étendu. Les facteurs importants doivent être considérés et fixés lors de l'extraction des motifs et des règles séquentielles (1) Seuil(minsup) La valeur requise pour le modèle d'extraction et (2) la confiance associée à des règles générées.

- (JOSH JIA-CHING YING 2013) [25] a défini un nouveau type de motif fréquent, appelle GTS pattern (motif de GTS), qui prend en compte les comportements des utilisateurs motivés par les traces géographiques(Geaographic-triggered) ,les traces temporels(Temporal-triggered) et les traces sémantique. Sur la base du modèle GTS, et il a proposé un nouveau cadre de frame work appelé GTS-LP prédire le prochain emplacement d'un utilisateur mobile.
- (FernandoTerroso-Saenz1 2015) [26] Le travail base sur un modèle de mobilité en ligne. par l'utilisation d'une architecture client /server. la solution proposée dans ce modèle introduit un nouveau mécanisme de prédictions. il donne définition d'une nouvelle méthode basé sur la vitesse et l'abstraction du route, et traiter les événements obtenus avec le clustering, basé sur la densité qui sont stockés sous forme multi graphe, tout en ligne.
- (Kohei Tanaka et al2009) [27] ont proposé l'utilisation d'une nouvelle méthode de prédiction de destination cette méthode est un système de navigation faire la prédiction de destination et fourni des informations au l'utilisateur de véhicule (temps d'arrive). Ils ont évalué la méthode (AM (Alternative way Method).DM *(*Departure Method*)*. BM (Basic Method) CM (Context Method .HM)(*hybridmethod)*)
  basé sur un système prototype. Ils ont clarifié les situations dans lesquelles la méthode proposée fonctionne bien, pour l'intégration de ces méthodes proposées de différentes façons de changement de la situation.
- Le travail de (Huei-Yu .L et al) [28] est basé sur le mode de transport et de comportement sémantiques pour prédire le prochain emplacement d'un utilisateur. Cette technique utilise le modèle de Markov caché pour trouver la relation entre les utilisateurs de mode de transport et leur comportement sémantique. On prend en consédération qu'ils ne demandent pas d'histoire de données de voyage. Ils

peuvent faire une prédiction, même si les utilisateurs mobiles n'ont jamais été à cet endroit avant et pas d'autres données de formation existent.

- (Merah A 2013) [2] dans ce travail, il a proposé initialement les schémas de communication pour recueillir l' historiques des chemins de véhicules. Après avoir recueilli tous ces chemins et la détermination de support minimal (minsup), les plus fréquent chemins parcourus par véhicules peuvent être extraits et utilisés comme motif de mouvement des véhicules. Selon minimum valeur de confiance (minconf), un ensemble de règles séquentielles sont générées à partir de ces motifs extraits par la suite .et la de prédection de la future route d'un véhicule. Ce travail qui est le pus proche de notre travail .

Travail	**Cathégorie**	**Modèle de prediction**	**destination ou route**	**Exactitude**
Simmons (2006)	Stastique	Modéle de Marckov HMM	Les deux	Entre 70 et 80 %
Uma Nagaraj,Nivedita N.Kadamdié (2011)	Stastique	Modéle de Marckov HMM	Route	Pas mentionne
Krumm(2008)	Stastique	Marckov modéle	Route	70%
XipengWang (2015)	Stastique	Modèle de Marckov SMM	Route	38% et couvrage de prediction 61%
Attila István Petróczi (2009).	Stastique	Modéle de Marckov SMM,HMM,VMM	Route	70%
Francisco Dantas. (2015)	Stastique	PPM	Les deux	Routes 15%-50%-85% 32.02%- l'occurrences : 45.06%-64.02%
Disheng Qiu (2013),	Stastique	Modéle de Marckov HMM ,VMM	Les deux	70%

Chen L. (2010)	Data mining	CRPM (contuniuos route pattern mining)	Route	71.86%
Fernando Terroso-Saenz1 · Mercedes Valdes-Vela(2015)	Data mining	Clustering basé sur densite	Route	8%-90%
TomášMikluščák. (2012)	Datamining	Reseau de neurones	Les deux	60%
Alexandre de Brébisson (2015)	Datamining	Reseau de neurones	Destination	Pas mentionne
JOSH JIA-CHING YING	Data mining	Régles sequentielle	Route	Pas mentionne
Huei-Yu .L et a al	Modéle de Marckov Semantic pattern	HMM Semantic pattern (sematic behavior label)	Route	58.2%-61.3% 67.5%-68.3%
Merah A 2013	Data mining	Motif sequentielles	Mouvement (route)	43.92%
Kohei Tanaka et al(2009)	Trip matching	BM,AM,DM,HM	Destination	86%-84%-72%-95%

Table 2.2: Table de travaux de prédiction des routes

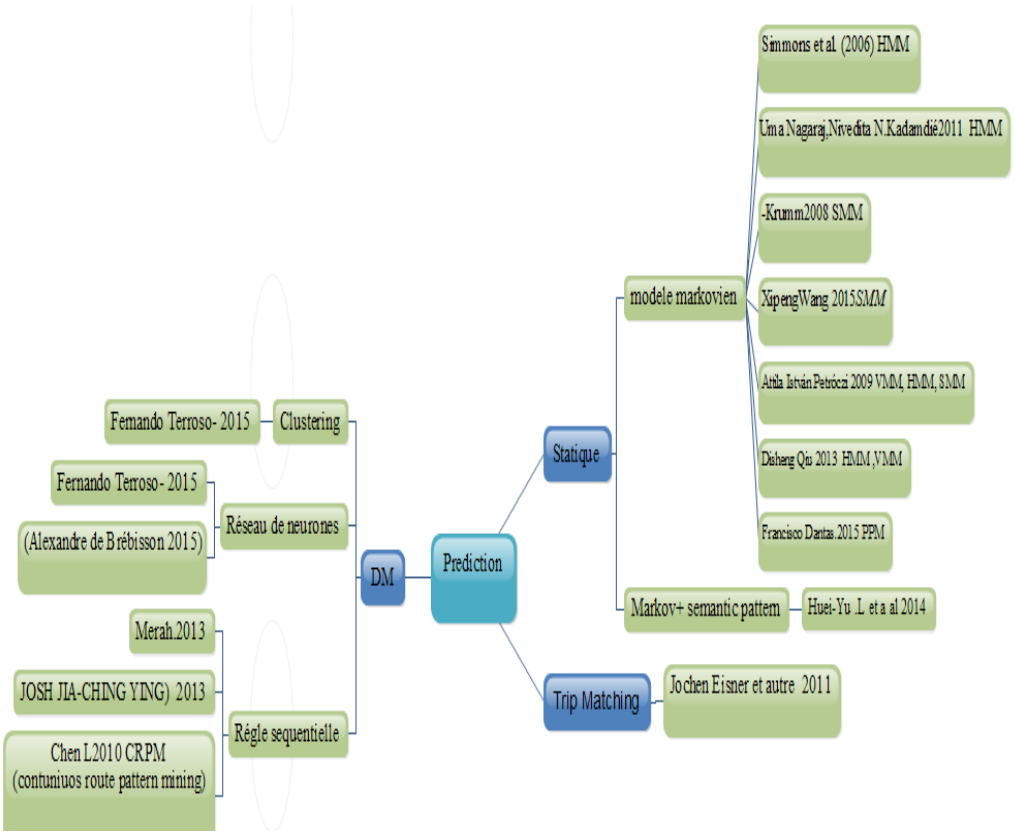

Figure 2.3: schema résume les travaux précedent

II.4 Conclusion

Dans ce chapitre on a défini des concepts liés à la prédiction des routes pour arrive à bien comprendre la prédiction et ces applications. La prédiction des routes est un de ces applications. Plusieurs travaux ont étaient faite, parmi ces travaux le travail de [2] comme il montre les travaux situé dans ce chapitre. Ils ont utilisé différents modèles et technique comme le modèle de Markov, l'extraction des règles séquentielles selon les avantages de chaque modèle, est pour but d'amélioré la prédiction de route et atteindre une prédiction réalisable avec plus d'exactitude dans les résultats.

III.1 Introduction

Nous avons présenté, dans les précédents chapitres, un aperçu des techniques de la fouille de données en concentrant le plus sur les règles séquentielles suivis par un état de l'art des travaux dans la littérature qui traitent le problème de prédiction des routes

Le présent chapitre sera consacré à la présentation de notre nouvelle approche de prédiction des routes basée sur les règles séquentielles. elle est composé de deux principales parties:

Partie 1- (Présentation de l'approche) Nous exposons, dans cette partie, la démarche de prédiction qu'on a proposé en détaillant les différents modules de notre Framework de prédiction développé.

Partie 2- (Etude expérimentale) Afin de valider et tester l'approche proposée, nous avons mené plusieurs expériences. Nous présentons dans cette partie les résultats de ces expérimentations en variant une liste des paramètres relatifs à notre système de prédiction.

III.2 Partie 1 : Présentation de l'approche

III.2.1 Les règles séquentielles pour la prédiction des routes (ARSPR)

Nous allons présenter, dans cette section, notre nouvelle **A**pproche basée sur les **R**ègles **S**équentielles pour la **P**rédiction des **R**outes (ARSPR). Nous commençons par la présentation de l'architecture générale d'ARSPR (voir figure 3.1). Nous exposons, par la suite, les deux modules de système: préparation des données et prédiction. Ce dernier module est composé de son tour en deux sous modules: 1) génération des règles séquentielles et 2) prédiction des routes basée sur les règles générées.

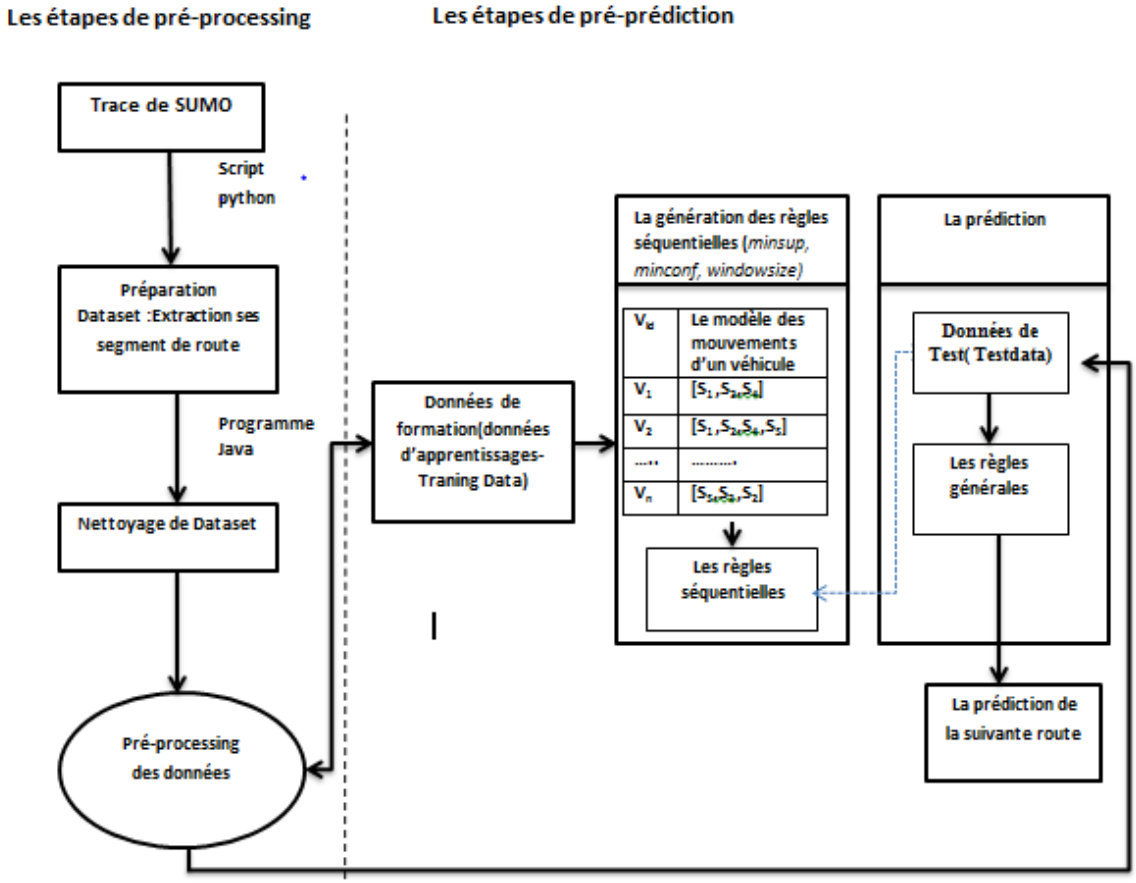

Figure 3.1: Architecture de système de prédiction ARSPR

III.2.2 La préparation des données (pré-processing)

Ce module a pour but de préparer le jeu de données (dataset) pour la prédiction des routes. Il consiste à générer initialement la trace de mobilité des véhicules en basant sur un scenario de mobilité avec SUMO 0.17 [29]. Un ensemble des scripts pythons et un programme java sont utilisés, par la suite, afin de nettoyer la trace et préparer la dataset finale en préservant que les informations requises pour la prédiction. Les principales étapes de cette phase sont présentées dans figure 3.2.

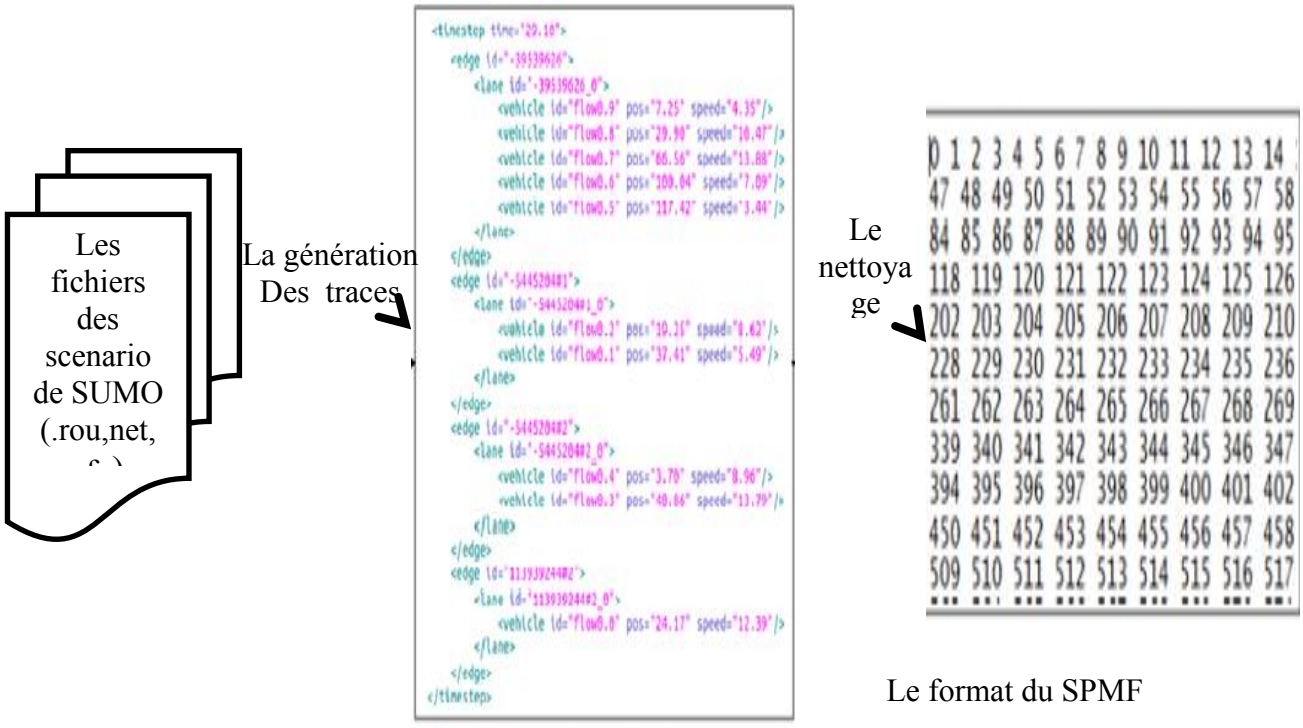

Figure3.2: Préparation de jeu de données (Dataset)

III.2.3 Génération de la trace de mobilité

Cette étape est basée sur un scénario de SUMO composé de plusieurs fichiers (un fichier de configuration (.cfg), fichier pour définir les véhicules et leurs types ainsi que les routes, etc.

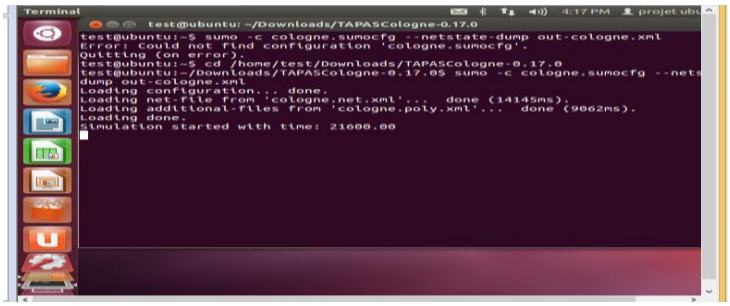

Figure 3.3: La génération de dataset

Elle consiste à générer la trace de mobilité des véhicules en exécutant la command suivante:

```
sumo -c cologne.sumocfg --netstate-dump out-cologne.xml
```

Où *tapas.cfg* c'est fichier de configuration, *output.xml* c'est le fichier contenant la trace générée. La figure 3.5 présente la syntaxe générale de fichier généré ou elle présente les champs suivants:

Timestamp time: c'est le temps

Vehicle ID: l'identifiant du véhicule

EdgeID: c'est l'identifiant de segment de route

LaneID: Identifiant de la voie (voir sumo pour plus de détails)

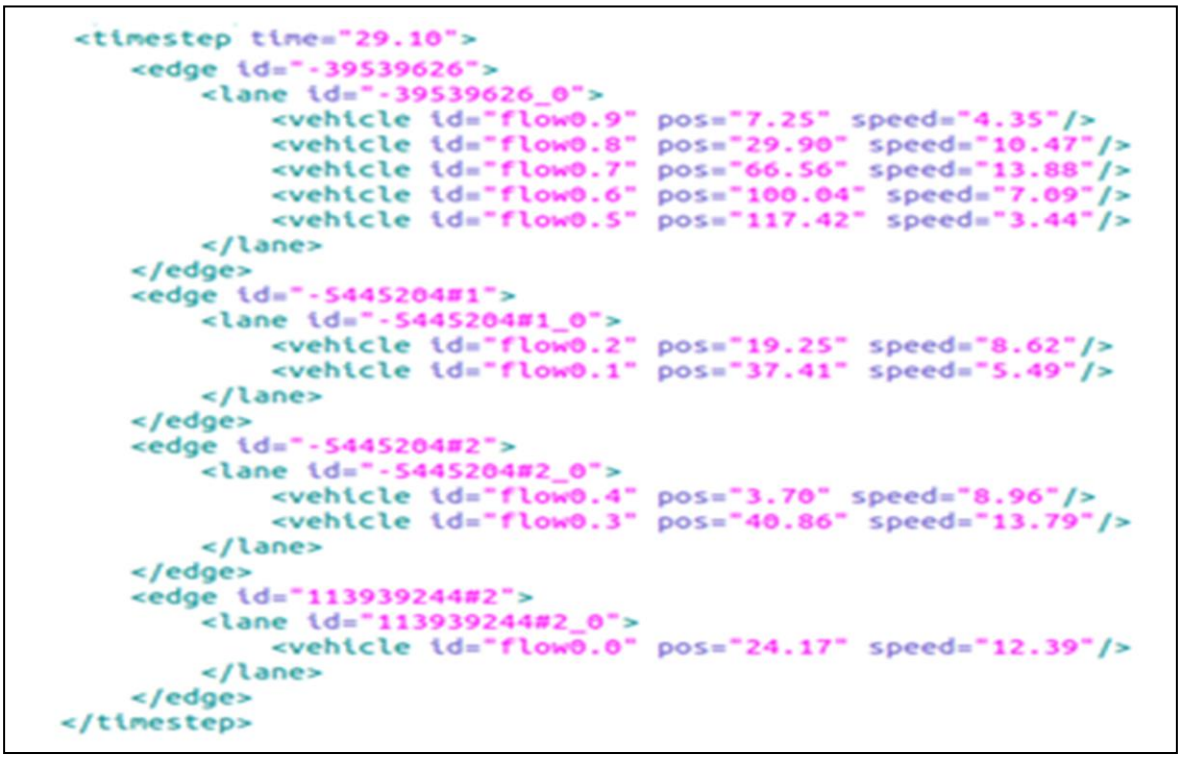

Figure 3-4: Une partie de fichier output.xml

III.2.4 Nettoyage de la trace

Cette étape a pour but de construire la liste des segments de routes traversés par chaque véhicule. Elle commence par une analyse grammaticale (parsing) de fichier XML généré de l'étape précédente en préservant que les champs *vehicule ID* et *edges* (liste des segments de routes). Cette analyse a été établie à l'aide d'un script python nommé *vehlane.py*, ou une modification sur le script original a été effectuée afin de réduire le temps de parsing. Avec la commande suivante.

```
python vehLanes.py out-cologne.xml output-final.xml
```

Le fichier XML "output-final *.xml* " c'est le fichier générer de la phase précédente qui va être soumet à un processus de nettoyage éliminant les segments consécutifs répétés et par la suite être convertis au format de l'outil SPMF [30] implémentant les algorithmes *RuleGen*

[31] et *TRuleGrowth* [31]. La figure 3.5 illustre cette conversion ainsi que le fichier généré dans le format conforme au SPMF. [32]

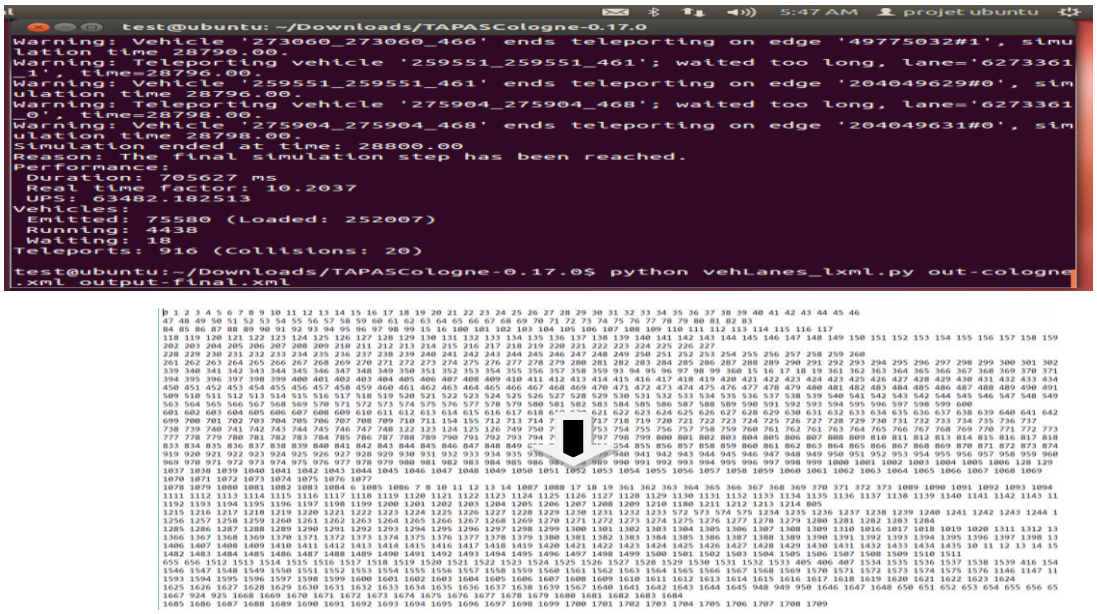

Figure 3.5: La dataset générée au format SPMF

Remarque : Il est important de noter ici que la génération et la préparation de dataset prend un temps considérable (des heures >3 avec la dataset qu'on a utilisée TAPASCologne) et nécessite aussi l'installation des packages spécifiques (lxml, etc.).

III.2.5 Le module de prédiction

Ce module prend en entrée le jeu de données utilisé, l'algorithme sélectionné ainsi qu'une liste des paramètres requises pour la prédiction (training ratio, minsup, minconf, window-size, etc) et donne en sortie l'estimation des mesures de performances de prédiction des routes. Le processus de prédiction, résumé dans algorithme A, est subdivisé en deux principales étapes:

1. Génération des règles séquentielles.
2. Prédiction des routes basée sur les règles extraites.

III.2.5.1 Génération des règles séquentielles

Étant donné une base de séquences (jeu de données dans notre cas) et des seuils *minsup* et *minconf*, un ensemble des règles séquentielles ayant un support et une confidence respectivement plus élevé que *minsup* et *minconf* sont générés. Ces règles ont la forme X, Y⇒ Z. L'utilisateur doit choisir l'algorithme de génération des règles séquentielle à appliquer entre: 1) règle séquentielle standard en avec RuleGen, [31] 2) règles séquentielles

partiellement ordonnées avec une contrainte de fenêtre (windowsize) de temps par TRuleGrowth. [32]

noter que TRuleGrowth défini un paramètre de taille de fenêtre (window size). Avec cet algorithme, l'utilisateur peut générer des règles de la forme X-> Y ou X et Y doivent être proche les uns aux autres en respectant le temps. Par exemple, un utilisateur veut générer les règles apparues dans trois itemsets consécutifs. L'algorithme RuleGen [25] doit être également modifié afin qu'il inclue ce paramètre (window size). Pour prédire le future segment de route, nous ne considérons que la règle séquentielle avec un seul élément dans son conséquent.

III.2.5.2 Prédiction à base des règles séquentielles extraites

Les règles extraites sont utilisées, dans cette phase pour prédire le prochain élément (segment de route) qui va suivre une séquence de mobilité de test Ts. La prédiction est assurée en deux étapes suivantes:

Etape 1: Balayage et vérification de Ts avec les antécédents de toutes les règles générées dans l'étape précédente. Si une règle R contient Ts elle sera retenu et ajouter dans un ensemble des règles de correspondance RC contenant toutes les règles vérifiant cette condition exigence. Par exemple :

Soit I = {RS1, RS2, RS3,…..RSn} ensemble des routes traversé par un véhicule qui constituent le database,. S est l'ensemble des séquences. La table suivante représente une table de séquence contient 4 séquences.

ID séquence	la séquence
S1	(RS1), (RS1, RS2, RS3), (RS4),(RS3,RS6)
S2	(RS1, RS4), (RS3), (RS2, RS3), (RS1, RS5)
S3	(RS5,RS6),(RS1,RS2),(RS4,RS6),(RS3),(RS2)
S4	(RS5),(RS7),(RS1,RS6),(RS3),(RS2) ,(RS3)

Table 3.1 : table de séquence.

La génération des règles séquentielles standard avec l'algorithme (**RuleGen**) L'algorithme RuleGen besoin comme entrées *(*minsup [0%,100%] et min conf [0% ,100%]) et la data base de séquence. On applique l'algorithme *RuleGen* sur cette database avec min sup = 75% et minconf =50%, on obtient presque 21 règles séquentielles générés.

La régle	support	confiance
{RS1 } ==> { RS1 }{ RS2 }	100%	100%
{ RS1 } ==> { RS1 }{ RS3 }	100%	100%
{ RS1 } ==> { RS1 }{ RS3 }{ RS2 }	75%	75%
{ RS1 } ==> { RS1 }{ RS3 }{ RS3 }	75%	75%
{ RS2 } ==> { RS1 }{ RS2 }	100%	100%
{ RS2 } ==> { RS2 }{ RS3 }	75%	75%
{ RS2 } ==> { RS3 }{ RS2 }	75%	75%
{ RS2 } ==> { RS1 }{ RS3 }{ RS2 }	75%	75%
{ RS3 } ==> { RS1 }{ RS3 }	100%	100%
{ RS3 } ==> { RS2 }{ RS3 }	75%	75%
{ RS3 } ==> { RS3 }{ RS2 }	75%	75%
{ RS3 } ==> { RS3 }{ RS3 }	75%	75%
{ RS3 } ==> { RS4 }{ RS3 }	75%	75%
{ RS3 } ==> { RS1 }{ RS3 }{ RS2 }	75%	75%
{ RS3 } ==> { RS1 }{ RS3 }{ RS3 }	75%	75%
{ RS4 } ==> { RS4 }{ RS3 }	75%	100%
{ RS1 }{ RS2 } ==> { RS1 }{ RS3 }{ RS2 }	75%	75%
{ RS1 }{ RS3 } ==> { RS1 }{ RS3 }{ RS2 }	75%	75%
{ RS1 }{ RS3 } ==> { RS1 }{ RS3 }{ RS3 }	75%	75%
{ RS3 }{ RS2 } ==> { RS1 }{ RS3 }{ RS2 }	75%	100%
{ RS3 }{ RS3 } ==> { RS1 }{ RS3 }{ RS3 }	75%	100%

Table 3.2 : les règles générées par Rule Gen

la règles { RS4 } ==> { RS4 }{ RS3 } dit: si le motif séquentielle { RS4 } que la véhicule traversé alors la véhicule passe sur les segments RS4,RS 3.

Les génération des règles séquentielles avec (TRuleGrowth): les entrés de cette algorithme sont : database , minsup [0% ,100%] (pourcentage), minconf [0%,100%] (pourcentage), window_size [*entier* > 0] la paramètre windowsize est le nombre des itemset consécutifs.

On a appliqué l'algorithme **TRuleGrowth** sur le database (table 3.1) on obtient 4 règles séquentielles on donne minsup = 30% et min conf = 80% avec window-size =3.

Régle	Support	Confiance
{RS1 } ==> {RS2 }	80 % (4 séquences)	100 %
{RS1 } ==> {RS2, RS3 }	80 % (4 séquences)	100 %
{RS1 } ==> {RS3 }	80 % (4 séquences)	100 %
{RS4 } ==> {RS3 }	60 % (3 séquences)	100 %

Tables 3.3 : les règles générées par *TRuleGrowth.*

Par exemple: on peut dire que si le véhicule est passé par la route RS1 alors elle passe par les segments RS2 et RS3, d'après la règles {RS1} ==> {RS2, RS3} qu'il a une confiance de 100%.

Etape 2 : Effectuer la prédiction en sélectionnant l'une des règles de RC. Cette sélection est basée sur le calcule d'un poids ou score défini comme suit:

Score (R) = (c_1 Conf (R) + c_2 Sup (R)) × longueur (R).

Où conf (R), sup (R) et longueur (R) représentent, respectivement, la confiance, le support et le nombre d'éléments dans R. Les paramètres c1, c2 sont deux constantes fixées lors des expérimentations et générant de meilleurs résultats. Une fois la meilleure règle est sélectionnée, la conséquence de la règle est renvoyée. Par exemple, supposons qu'un véhicule donné V_i traverse les segments rs2 et rs3. En tenant la règle rs2, rs3⇒ rs4 générée de l'étape précédente et qui corresponds avec les segments traversés, le module de prédiction donne comme résultat de prédiction est le segment rs4.

Algorithme de notre système de prédiction
Entrée *Minsup:* support minimal *Minconf:* confiance minimale *BDS:* database de séquence (motif de mobilité pour chaque véhicule) *VCSegment:* le segment de route actuel de la voiture ***Sortie*** *le futur segment de route*
Procédure génération-Règle-séquentielle (SDB, minsup, minconf)

début

Si (algo est RuleGen)// cas de règle standard

EP= *extraction de motif fréquents séquentiels (BDS , minsup)*

SR= générer de règles séquentielles standard(EP ,minconf)

Si *(algo est TRuleGen)//cas de règles séquentielles partiellement ordonnées*

POSR=générer règles séquentielles(BDS,minsup,minconf)

Fin

// La prédiction est similaire pour les deux types des règles séquentielles

Procédure de prédiction (SR,VCSegement)

Pour chaque règle *r* de *SR*

règles-généré= Vérifier *VCSegment* avec antécédent de *r*

Si (Les règles-génère ={})

Retourner prédiction échoué

Sinon

Pour chaque règle MR dans règles-générés

Calculer Score(MR)=(C_1conf(MR)+C_2sup(MR))*longueur(MR)

Fin Pour

Choisir la meilleure règle BMR qui a le score maximal

Retourner la conséquence de BMR

Fin

III.3 Partie 2 : Etude expérimentale

III.3.1 Étude expérimentale

L'objectif principal de cette partie et l'étude de performances de notre system de prédiction basée sur les règles séquentielles. Nous visons à expérimenter et comparer la prédiction des routes avec deux types des règles (standard avec l'algorithme RuleGen [31] et partiellement ordonnées avec TRuleGrowth. [32]

Cette section décrit initialement l'enivrement de travail. Une description de jeu de données utilisé est illustrée par la suite. Afin d'étudier les performances, deux mesures de performances ont été présentés suivi par des tests comparatifs selon plusieurs paramètres, visant la scalabilité de notre système de prédiction.

III.3.2 Environnement de travail

On a utilisé Ubuntu 12.04 LTS sur Vmware (Workstation pro 12) [33] sur une machine physique dotée d'un processeur Core i5-4200 U1.6 GHZ CPU et un disque dur 500 GB. Pour générer les règles séquentielles nous avons utilisé une implémentation JAVA des algorithmes RuleGen et TRuleGrowth disponibles dans. [34]

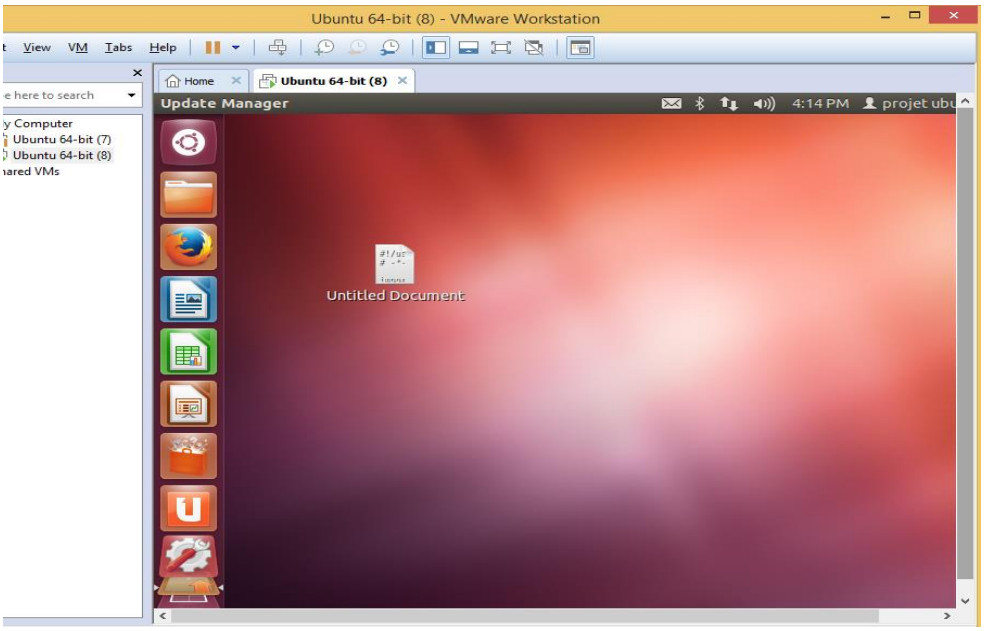

Figure 3.6: L'environnement de travail

III.3.3 Le jeu de données (Dataset)

Dans notre étude expérimentale, on va commencer par la génération de Dataset de TAPASCologne à l'aide de SUMO [29] (Simulation of Urban Mobility) version 0.17. Le logiciel SUMO (Simulation of Urban Mobility) est un outil de simulation de trafic facilitent l'évaluation des changements d'infrastructure ainsi que les changements de politique avant de les appliquer sur la route.

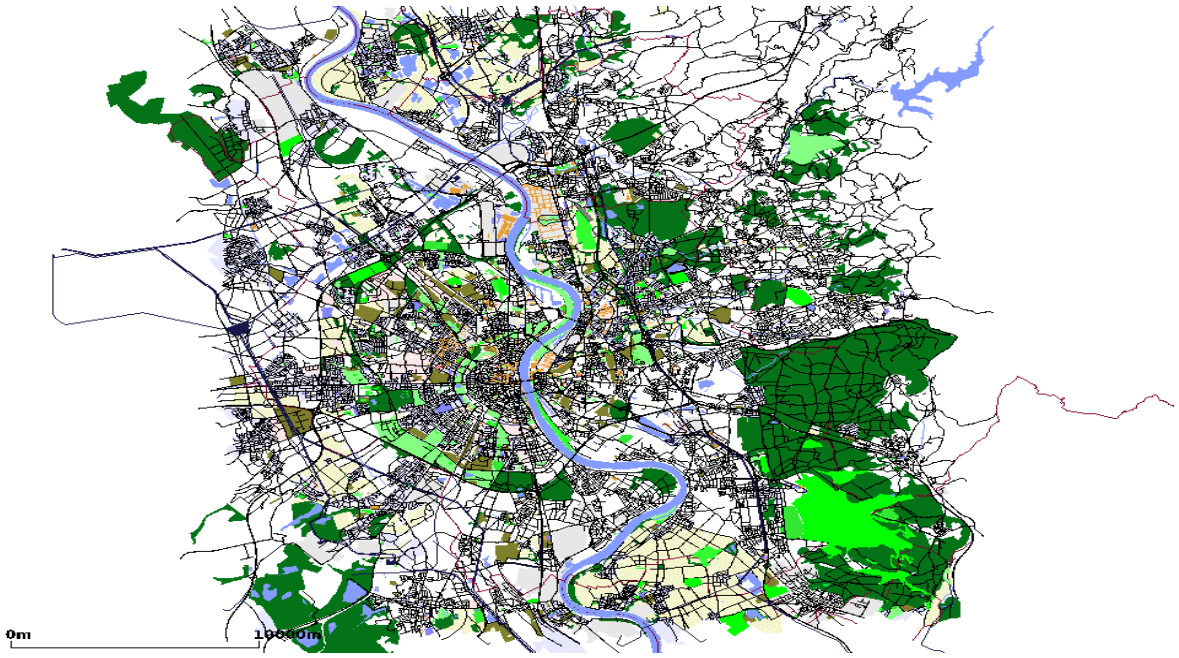

Figure 3.7: Aperçu de TAPSCologne avec SUMO

III.3.4 Description de Dataset

TAPASCologne [32] est une initiative de l'institut des systèmes de transport au Centre aérospatial allemand (ITS-DLR), qui vise à reproduire, avec le plus haut niveau de réalisme possible, la circulation automobile dans la zone urbaine plus grande de la ville de Cologne, en Allemagne. La trace synthétique résultante de la circulation automobile dans une ville de Cologne couvre une région de 400 kilomètres carrés pour une période de 2 heures, comprenant plus de 70.000 déplacements en voiture individuelle.

Dataset	**Pays**	**Surface (km2)**	**Nombre de véhicule**	**Heures**	**La longueur moyenne de chemin**
TAPASCologne	Allemagne /Cologne	400	70.000	2h	110

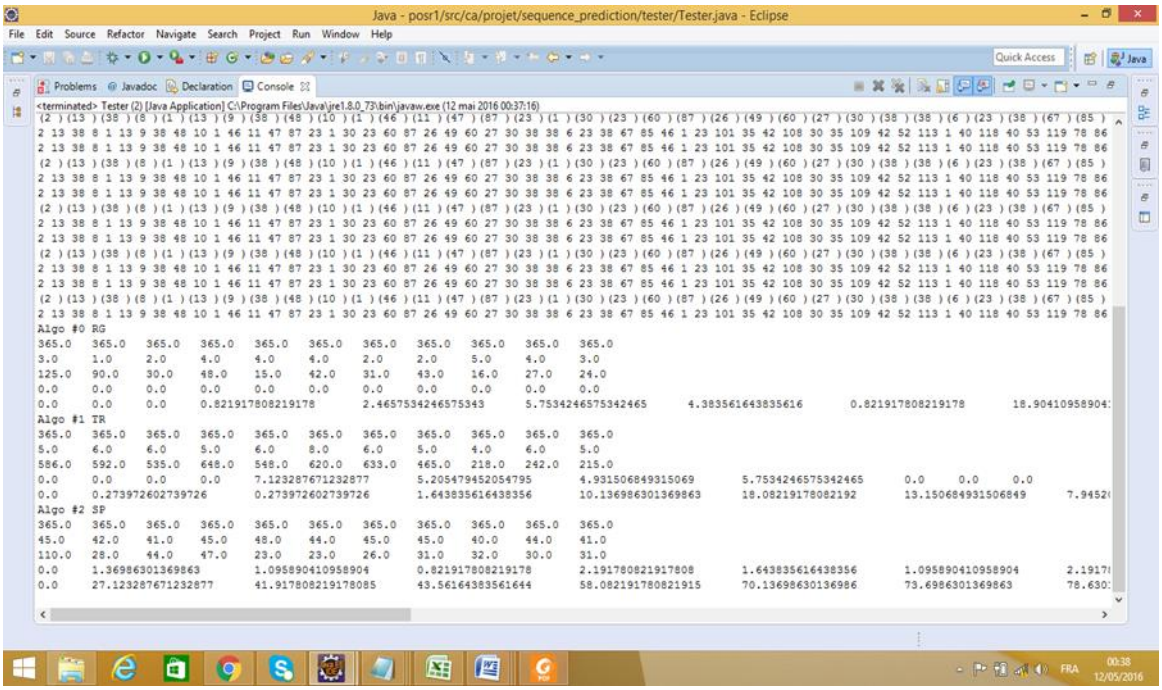

Figure 3.8 : les règles générées avec les taux de prédictions avec l'agorithme(RuleGen et TRuleGrowth)

III.3.5 Les paramètres d'évaluation

Pour mesurer et comparer les performances de notre système de prédiction *ARSPR*, nous avons utilisé deux mesures largement utilisés pour évaluer les systèmes de prédiction.

Précision globale ou l'exactitude: Elle est définie par le nombre des futures routes prévues avec succès sur le nombre total des séquences de test.

$$\text{Précision globale ou exactitude} = \frac{\text{Nombre de la bonne prediction}}{\text{Nombre de sequnece}}$$

Couverture: Elle représente le nombre de séquences où la prédiction a été effectuée par le nombre des séquences de teste. Cette mesure indique si une règle de correspondance est trouvée pour une séquence prédiction ou non.

$$\text{Couverture} = \frac{\text{Nombre des règles de correspondance}}{\text{Nombre des séquences de test}}$$

III.3.6 Résultats

Cette section représente les résultats obtenus lors de notre étude des performances des règles séquentielles (TRuleGrowth, RuleGen) pour la prédiction des routes en jouant sur les paramètres essentiels suivants en variant:

- Taux d'apprentissage (Training_ratio).
- Taille de fenêtre (window-size).

- Minsup et Minconf .
- Nombre de véhicules (Scalabilité)
- Un paramètre nommé *ToKeep* désignant les N derniers segment de routes a utilisé pour effectuer la prédiction pour chaque séquence. IL prend les valeurs de [0-11].

III.3.6.1 Taux d'apprentissage (Training_ratio)

La dateset est divisée en deux sous-ensembles, un ensemble d'apprentissage (training) et un ensemble de test selon le taux d'apprentissage (Training-ratio) qui indique le pourcentage des séquences de l'ensemble de données utilisé pour l'apprentissage. On a varié ce paramètre d'une valeur de 10% jusqu'à 90% et le paramètre ToKeep de 1 à 10 avec un M*insup= 0.002*, Minconf =0.5 et W*indow-Size = 25.*

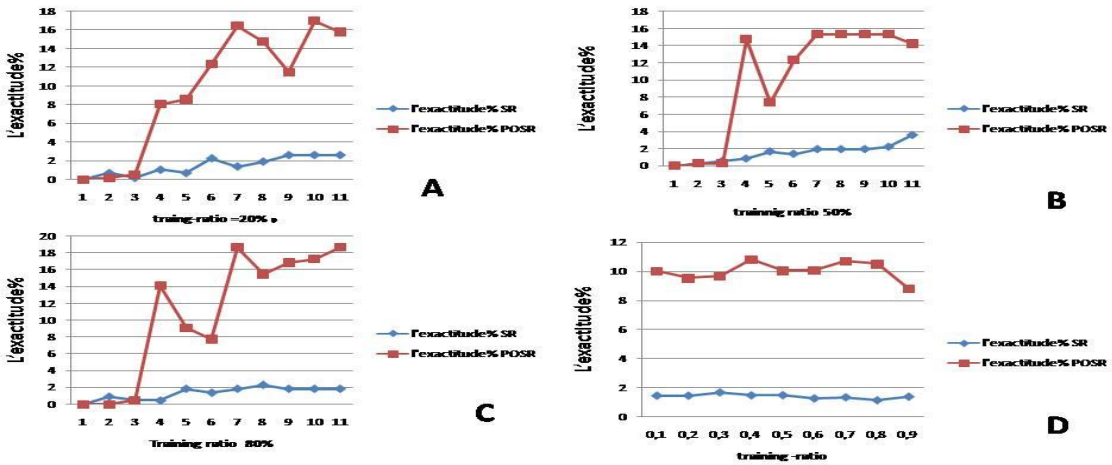

Figure 3.9: L'exactitude par rapport le training-ratio

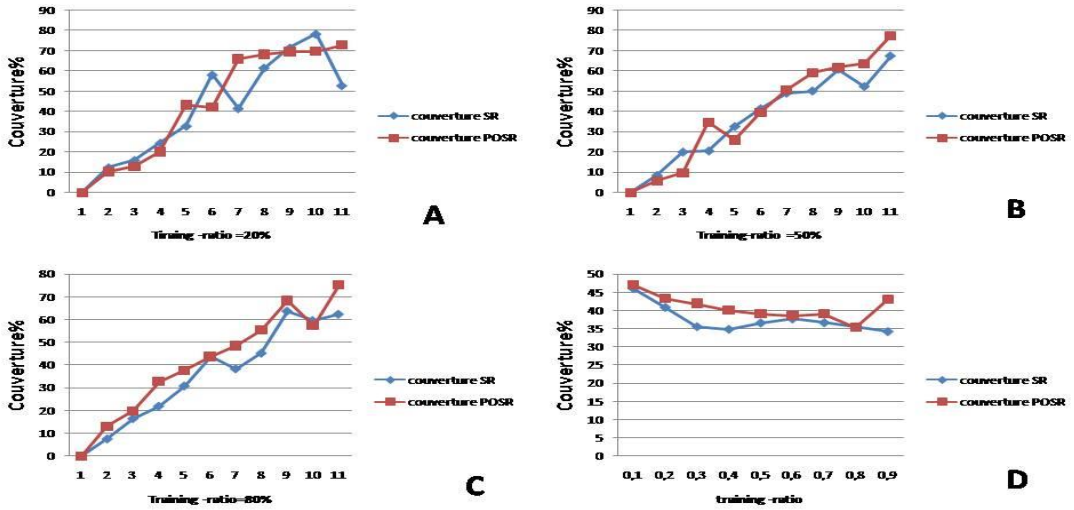

Figure 3.10: La couverture par rapport le training-ratio

D'après les résultats de la figure 3.10, on observe l'influence du paramètre de training-ratio à l'exactitude. La grande différence entre eux est que les prédictions basées sur les règles séquentielles partiellement ordonnées (POSR) étaient de 10% à 12% plus précise que des prédictions basées sur les règles séquentielles standard (SR). On peut observer que cela est vrai même si un ensemble d'apprentissage plus petit est utilisé pour POSR.

La figure 3.11 résume le taux de couverture estimé pour SR et POSR. Les résultats obtenus indiquent que le taux de couverture de POSR est plus élevé par rapport au SR ce qui justifier la bonne précision rapportée de POSR. On a remarqué qu'avec un taux traning ratio de 50% on peut atteindre une couverture de 80%.

III.3.6.2 Taille de fenêtre (window-size)

Le paramètre *Window-Size* permet de spécifier que les modèles doivent se produire dans un nombre maximal d'item sets consécutifs. Impact de ce paramètre sur la performance de les algorithmes SR ET POSR est que POSR donne une exactitude plus élève 7 fois que SR par exemple dans B et C qui l'exactitude de POSR 8 fois SR. Le paramètre se varier entre 5 et 40 on donnant *Training –ratio= 0.7 et minsup= 0.002 et min conf =0.5.*

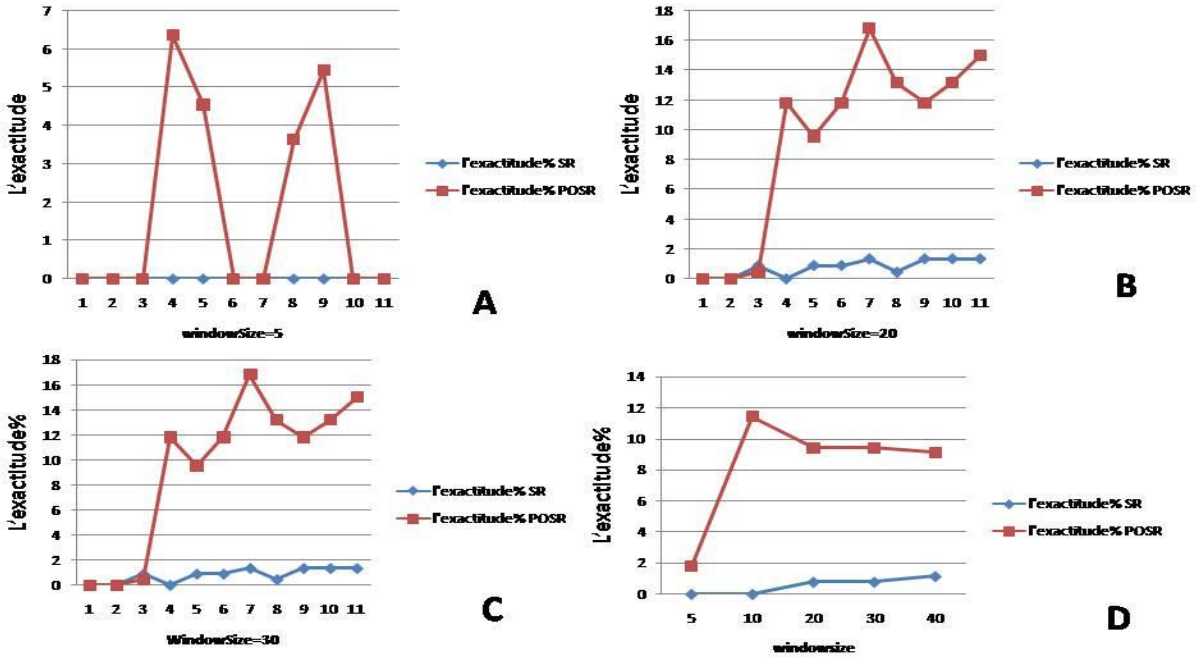

Figure 3.11: l'exactitude par rapport windowsize

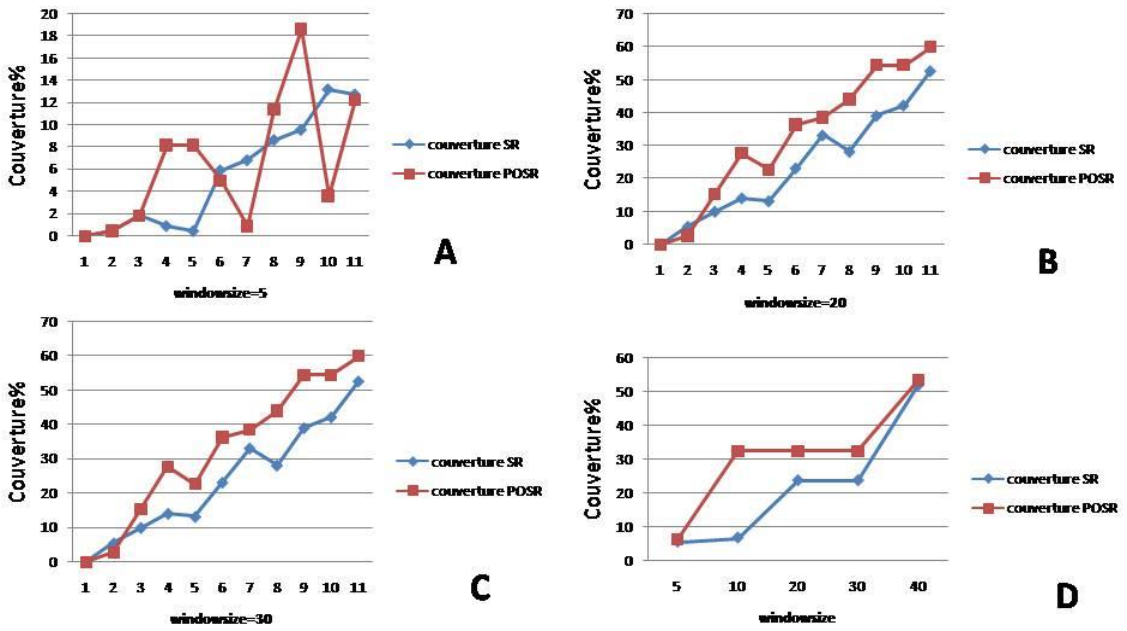

Figure 3.12: la couverture par rapport windowsize

D'après Les diagrammes A B C D dans la figure 3.13 qui représentent la couverture obtenues lorsque on varier le paramètres de *windowsize* entre 5 et 40, on remarque que le taux de couverture des règles atteindre jusqu'au 60% avec une préférence concrète pour POSR.

III.3.6.3 Le minsup

Le minsup est le seuil de support, dans cette partie de l'étude expérimentale on varier les valeurs de minsup entre 0.0005 et 0.003 et tenant le *windowsize= 25* et *training ratio= 0.7* min conf= 0.5 avec le paramètre *to keep* de 1 -10 on voir les résultats comme suit :

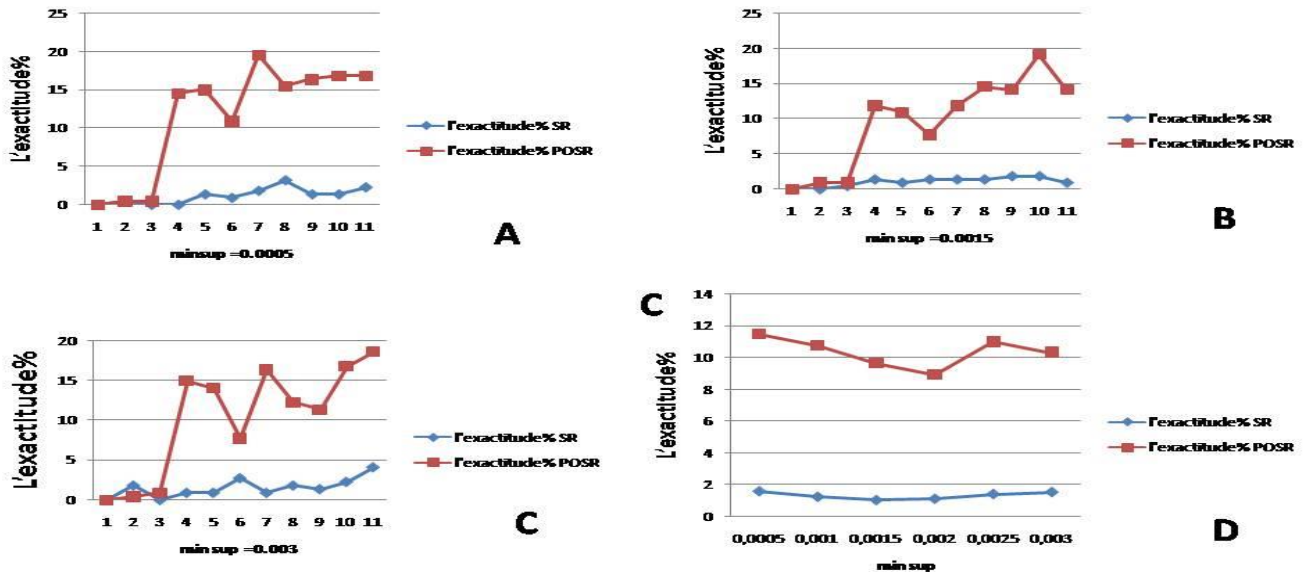

Figure 3.13: l'exactitude par rapport Min sup

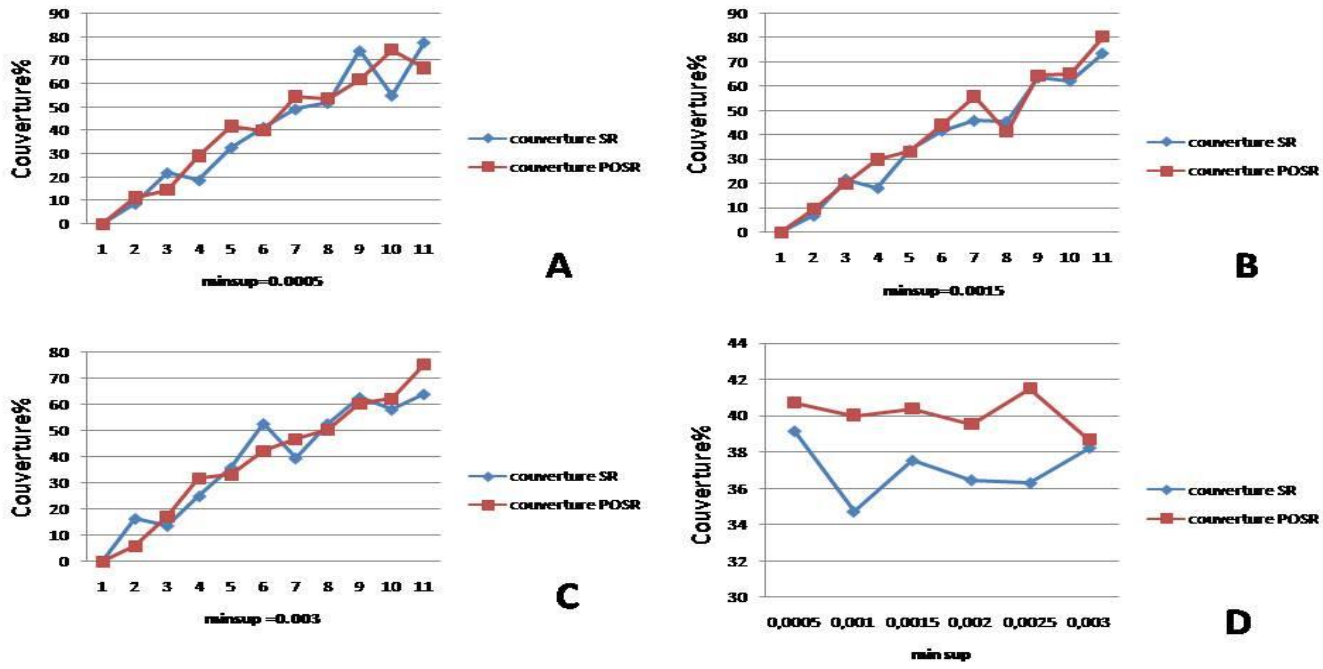

Figure 3.14: la couverture par rapport Min sup

D'après les A, B, C, D diagrammes de figure 3.13 on observe que l'exactitude obtenue par POSR est meilleur que SR qui peut atteindre au taux de 10 fois plus que SR.

Dans D qui résume les différents exactitudes obtenues on remarque que l'exactitude est plus avec POSR que avec SR de 10%, donc presque 6 fois plus. On donnant windowsize =25 Minsup =0.002 minconf= 0.5 et to keep =1de 10.

L'autre facteur qui est la couverture. La figure 3.14 représente l'impact de paramètre min sup sur le facteur de la couverture des règles, les observations obtenant que jusqu'à 80% des règles générés. POSR donne les taux plus que SR.

III.3.6.4 le paramètre de Nombre des véhicules

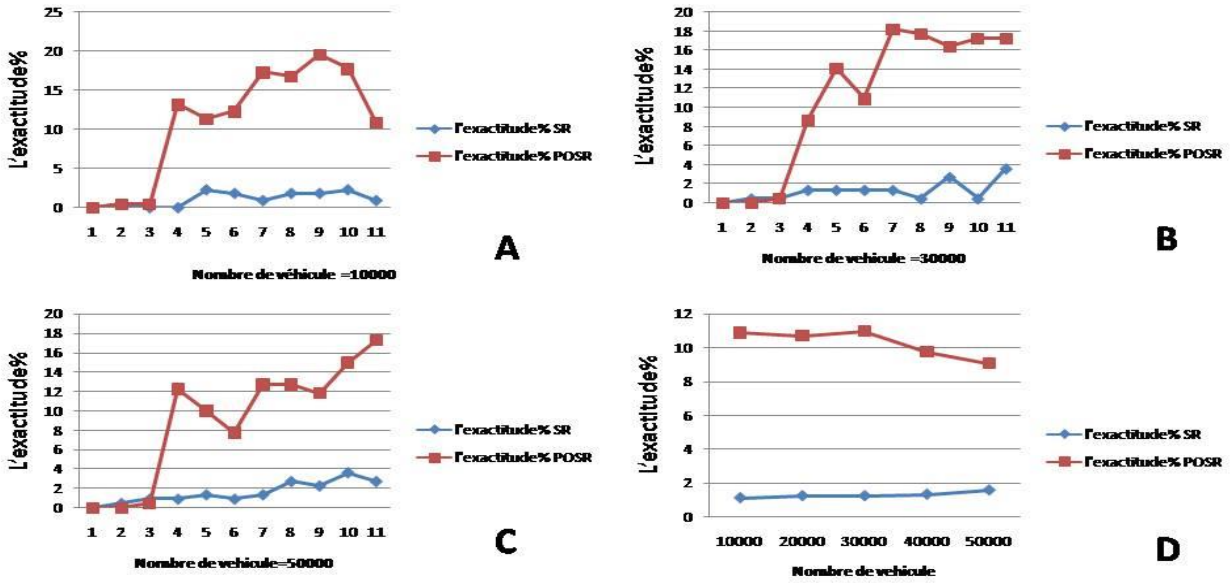

Figure 3.15 : l'exactitude par rapport le nombre de véhicule

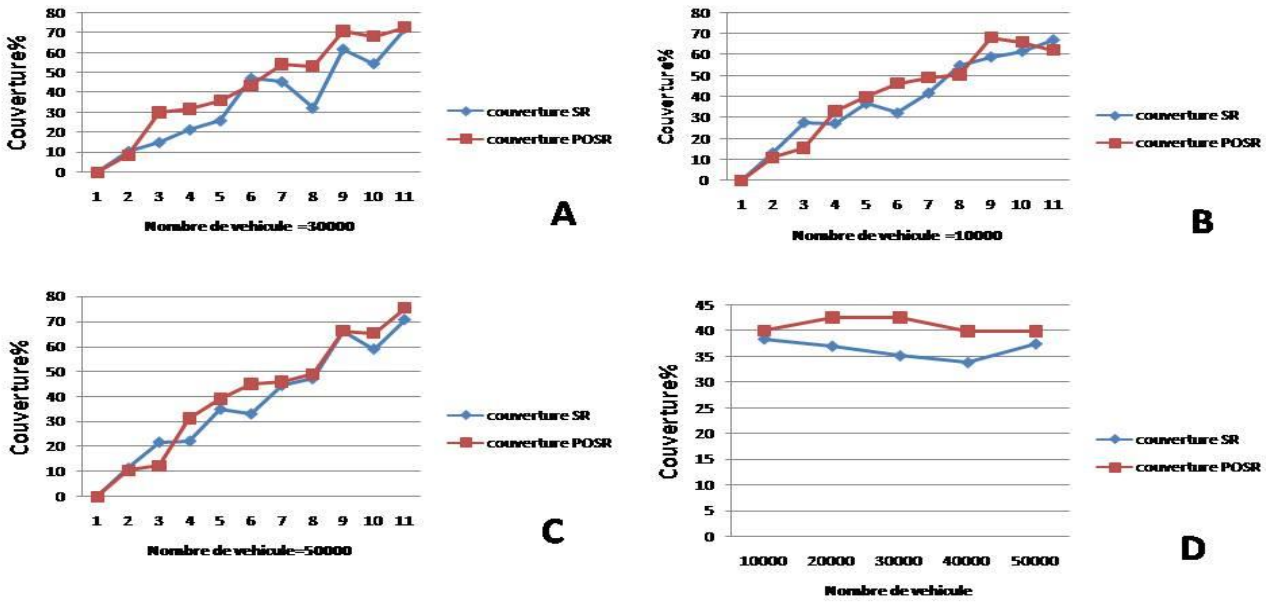

Figure 3.16: la couverture par rapport le nombre de véhicule

dans ce partie d'étude expérimentale on étudie l' influence de le paramètres de nombre de véhicule sur le performance de POSR et SR, les diagrammes A, B, C, D dans figure 3.15 représente l'exactitude, qui est plus élève avec POSR que SR et arrive jusqu'à 10% différence entre POSR et SR, on prend en considération que le nombre de véhicule se varier entre 10000 et 50000 et de Minsup 0.002, min conf =0.5 avec windowsize =25 et Training-ratio =0.7 et to keep = 1de 10.

la figure 3.16 représente l'état de facteur de couverture par rapport au nombre de véhicule, l'influence de ce facteur montre que les résultats obtenues par POSR et SR sont approximatives avec préférence pour POSR qui atteindre jusqu'au 70% par POSR .

III.4 Conclusion

Dans ce chapitre étaient la partie pratique de notre travail, on mettant les deux algorithmes RuleGen et TRuleGrowth sous expérience afin d'étudier la performance de chacun a l'aide des métriques spécifiques (l'exactitude, couverture) , les expérimentation sous fait sur une dataset réel *TAPASCologne.* D'après les différents tests TRuleGrowth a donné des résultats meilleurs que RuleGen donc la performance de POSR et meilleur que SR a la prédiction des routes.

Conclusion générale

Dans ce travail, nous sommes intéressés au problème de la prédiction des routes. Pour cela, nous sommes attachés dans ce mémoire à proposer une nouvelle approche basée sur les techniques de datamining et plus précisément les règles séquentielles pour résoudre ce problème de sélection. Cette approche est composée de deux partie: (1) la génération des règles séquentielles et par la suite (2) la prédiction des futures routes en basant sur les règles générées. Pour augmenter l'exactitude de notre prédiction nous avons ainsi proposé d'appliquer les règles séquentielles partielement ordonnées avec l'algorithme TRuleGrowth. Cet algorithme a démontré leur efficacité, par rapport aux règles séquentielles standard dont le fait ou il a amélioré les performances de prédiction. Pour cela, il est sensé d'être plus adapté au problème de prédiction des routes par rapport au règles standard, vue la notion de window qu'il l'applique lors de génération des règles.

Perspectives

En complément de la recherche effectuée, le travail réalisé dans ce mémoire ouvre diverses perspectives de recherche:

(a) Dans ce mémoire, nous sommes restreints à utiliser notre modèle pour le problème de prédiction des route. Cependant, les règles séquentielles peuvent constituer une bonne solution aux autres problèmes de prédiction des mouvements tels que la prédiction de destination et location.

(b) Notre approche peut être utilisée avec d'autres techniques de data mining ou même statistiques afin d'améliorer l'exactitude.

Bibliographie

1. **Deguchi, Y.** *Hev charge/discharge control system based on car navigation information.* s.l. : SAE Convergence International Congress & Exposition on Transportation Electronics), 2004.

2. **Merah, Amar Farouk.** *Vehicular Movement Patterns: A Sequential Patterns Data Mining Approach Towards Vehicular Route Prediction* .

3. **philippe-fournier-viger.** *introduction-to-sequential-rule-mining.* s.l. : data-mining.philippe-fournier-viger.com.

4. **DJEFFAL , Dr. Abdelhamid.** *Cours Fouille de données avancée* . s.l. : Université Mohamed Khider - Biskra , 2014/2015.

5. **PREUX, Ph.** *Fouille de donnees Notes de cours* . s.l. : Universite de Lille 3.

6. **GREYC, François RIOULT.** Fouille de données orientée motifs,usages . s.l. : Équipe Données-Documents-LanguesCNRS UMR 6072Université de Caen Basse-NormandieFrance.

7. **HANANE, AMIRAT.** *these de magistaire .Les techniques de fouille de données pour la conception physique des entrepôts de données : nouvelle approche et étude comparative.* s.l. : Universtie Amar tlidji, 2013.

8. **Georges El Helou, Charbel Abou khalil.** *Data Mining Techniques d'extraction des connaissances* . 16 février 2004.

9. **DIDAY, Johanna GARCIA ,Johanna GOLD.** *DATAMINING Quel auteur doit-on éditer ?* . s.l. : Université Paris Dauphine UFR Informatique de Gestion, 2005.

10. **Marascu , Alice.** *Extraction de motifs séquentiels dans les flux de données.*

11. **Philippe Fournier-Viger, Ted Gueniche, and Vincent S. Tseng.** *Using Partially-Ordered Sequential Rules to Generate More Accurate Sequence Prediction.*

12. *https://www.techopedia.com/definition/30595/spatial-data-mining.*

13. **MOUTACALLI, Mohamed Tarik.** *Une approche de reconnaissance d'activités utilisant.*

14. **HANANE, AMIRAT.** *Partially ordered sequential rules for route prediction.*

15. **Reid Simmons, Brett Browning, Yilu Zhang , Varsha Sadekar.** *Learning to Predict Driver Route and Destination Intent* . 2006.

16. **Uma Nagaraj, Nivedita N.Kadam.** *Study Of Statistical Models For Route Prediction Algorithm In VANET.*

17. **Krumm , John.** *A Markov Model for Driver Turn Prediction* .

18. **Xipeng Wang, Yuan Ma, Junru Di, Yi L Murphey, Shiqi Qiu,.** *Building efficient probability transition matrix using machine learning from big data for personalized route prediction.*

19. **Attila Istvcn Petróczi, Csaba Gàspàr-Papanek.** *Route Prediction on Tracking Data to Location-Based Services* . s.l. : EUNICE, 2009. 69-77.

20. **Disheng Qiu, Paolo Papotti, Lorenzo Blanco.** *Future Locations Prediction with Uncertain Data.*

21. **Francisco Dantas N. Neto, Cláudio de Souza Baptista1, Cláudio E. C. Campelo.** *Prediction of Destinations and Routes in Urban Trips with Automated Identification of Place Types and Stay Points* . s.l. : (UFCG)/(IFPB), 2015.

22. **Tomás Mikluscák, Michal Gregor, Ales Janota.** *Using Neural Networks for Route and Destination Prediction in Intelligent Transport Systems.* 2012.

23. **Alexandre de Brébisson, Étienne Simon , Alex Auvolat , Pascal Vincent, and Yoshua Bengio.** *Artificial Neural Networks Applied to Taxi Destination Prediction* . s.l. : PKDD/ECML, 2015.

24. **Ling Chen, Mingqi Lv , Qian Ye , Gencai Chen , John Woodward.** *A personal route prediction system based on trajectory data mining* .

25. **Zaki, M.J.** *SPADE: An Efficient Algorithm for Mining Frequent Sequences.* 2001.

26. **Fernando Terroso-Saenz, Mercedes Valdes-Vela ,Antonio F. Skarmeta-Gomez.** *Online route prediction based on clustering of meaningful velocity-change areas* . 2015.

27. **Kohei Tanaka, Yasue Kishino, Tsutomu Terada, Shojiro Nishio.** A Destination Prediction Method Using Driving Contexts and Trajectory for Car Navigation Systems .

28. **Huei-Yu Lung, Chih-Heng Chung, and Bi-Ru Dai.** *Predicting Locations of Mobile Users Based on Behavior Semantic Mining* .

29. https://sourceforge.net/projects/sumo/files/sumo/version%200.17.0/.

30. http://www.philippe-fournier-viger.com/spmf.

31. philippe-fournier-viger.http://www.philippe-fournier-viger.com/spmf/index.php?link=algorithms.php.

32. https://sourceforge.net/projects/sumo/files/traffic_data/scenarios/TAPASCologne/.

33. http://www.vmware.com/fr/products/workstation/workstation-evaluation.

34. http://www.philippe-fournier-viger.com/spmf/index.php?link=algorithms.php.