Correspondances inter-schémas dans les SIG
Schema mapping in GIS

Yassine Lassoued Snezhana Manoah Omar Boucelma

LSIS, Université de Provence

39, rue Joliot-Curie, 13453 Marseille Cedex 13
{lassoued,manoah,omar}@cmi.univ-mrs.fr

Résumé

L'intégration automatique de données est, en général, un problème difficile. Dans le cas particulier des Systèmes d'Information Géographique (SIG), le problème se complique car il faut assurer une correspondance (appariement) sémantique entre deux sources alors même que le choix des solutions n'est pas évident (ontologies, corrélations des informations, etc.). Nous décrivons dans cet article une approche d'appariement basée sur des méthodes d'apprentissage automatique.

Mots Clef

Intégration de l'information, correspondance inter-schémas, SIG, GML, apprentissage, classification.

Abstract

Semantic integration of Geographic Information Systems (GIS) is a hard topic, especially when there is no special hint that helps in choosing the best formalisms and/or integration architectures. In this paper we describe our approach for GIS schema mapping. Our approach extends existing machine learning approaches for (traditional) data mapping but departs from them due to the nature of geographic data we aim to integrate.

Keywords

Information Integration, schema mapping, GIS, GML, machine learning, clustering.

1 Introduction

L'un des principaux problèmes posés par l'intégration des données consiste à effectuer la correspondance sémantique entre le schéma d'une source de données et un schéma de référence appelé *schéma global* ou *schéma de médiation*. Ce problème est appelé dans la littérature *correspondance inter-schémas* ou *appariement de schémas*.

Étant donné un schéma global G fixé et une nouvelle source de données dont le schéma est noté S, l'appariement consiste à identifier les éléments des deux schémas (S et G) qui se correspondent. D'une manière générale, il s'agit d'établir les règles qui permettent de lier les éléments du schéma S à ceux du schéma G. Ceci permet au médiateur de répondre aux requêtes de l'utilisateur qui sont posées sur le schéma global.

De nombreux travaux se sont intéressés à l'automatisation de l'appariement de schémas mais dans un cadre assez général de sources de données relationnelles ou au mieux XML. Ces travaux ont abouti à des approches assez variées (voir [1] qui présente un aperçu sur les méthodes développées par la communauté Bases de Données). L'appariement au niveau schématique, par exemple, peut utiliser des approches liguistiques telles que la similarité des noms des éléments : égalité de noms, synonymes (voiture $\cong$ automobile), égalité de représentations canoniques (NumEmp $\cong$ numéroEmployé) parties communes (représenté-Par $\cong$ représentant), etc. ou les descriptions fournies dans les schémas. Certaines techniques utilisent un appariement au niveau des instances, par exemple en exploitant un ensemble de mots clés qui peuvent apparaître dans les instances (tels que "excellent état" pour la description) ou la structure du contenu de l'élément (un numéro de teléphone

est un nombre à 10 chiffres), etc.

Parmi les travaux sur l'automatisation de l'appariement de schémas nous avons retenu ceux basés sur des techniques d'apprentissage, et notamment les systèmes (LSD et GLUE) [2, 3, 4] basées sur une technique d'apprentissage multi-stratégies. Ces deux systèmes, bien que différents par les techniques qu'ils emploient, ont des architectures et des fonctionnements globaux assez analogues. Ils cherchent à établir des correspondances 1-1 (bijectives) entre les éléments d'un schéma de source de données et un schéma de médiation vus comme des DTD. Dans ces approches, le problème d'appariement est vu comme un problème de classification qui consiste à associer à chaque élément (ou nœud) de la DTD du document XML source un élément de la DTD de médiation, appelé *étiquette* ou *label*.

1.1 Appariement par apprentissage : une introduction

L'idée clé derrière les approches LSD et GLUE est la suivante : après avoir effectué les correspondances entre le schéma local et le schéma global, l'utilisateur (administrateur du système d'appariement) transmet au système son "savoir-faire", lequel doit être capable d'extraire des informations utiles sur ces correspondances et de proposer des correspondances pour des sources de données futures. Cette transmission est fondée sur des méthodes d'apprentissage. Le système LSD par exemple utilise une approche multi-stratégique d'apprentissage qui consiste en l'emploi de plusieurs apprenants appelés *apprenants de base*. Chacun de ces apprenants apprend à reconnaître les éléments selon un point de vue, par exemple son nom, son contenu, etc. Un *méta-apprenant* est utilisé pour combiner les prévisions des apprenants de base, selon des degrés de confiance qu'il leur accorde.

Le système LSD (respectivement GLUE) procède en deux phases : l'*apprentissage* et la *classification*. Dans la phase d'apprentissage, un appariement manuel entre le schéma d'une source et le schéma de médiation permet d'entraîner les apprenants de base et le méta-apprenant. L'apprentissage est effectué à l'aide d'un échantillon de données. Dans la phase de classification, les apprenants de base essaient de classifier des éléments d'un échantillon de données. Pour chaque élément, le méta-apprenant combine leurs prévisions. Un *convertisseur* (respectivement *estimateur de similarité*) permet par la suite de combiner les décisions du méta-apprenant pour l'ensemble de l'échantillon. Enfin, un *combinateur de contraintes* (respectivement *relaxation labeler*) prend la décision finale en tenant compte d'un ensemble de contraintes d'intégrité. Les principaux apprenants de base utilisés par LSD et GLUE sont le *Name Matcher*, le *Content Matcher*, le *Naive Bayes Learner* et le *XML Learner*. Ces apprenants traitent de l'information textuelle, numérique ou énumérative. Ils exploitent les noms des éléments ou leurs contenus pour reconnaître leurs correspondants dans le schéma global.

1.2 Appariement de données géographiques

Notre travail se situe dans le cadre du projet RNTL VirGIS [5] pour l'intégration des données géographiques. Pour assurer l'extensibilité du système d'intégration VirGIS nous avons besoin de rajouter des sources nouvelles à intégrer, sans pour cela remettre en cause le schéma global (de médiation). Ce qui implique de fournir – si possible de façon automatique – les correspondances entre le schéma de la nouvelle source et le schéma global. Pour ce faire, nous nous sommes inspirés des méthodes d'apprentissage utilisés dans les approches LSD et GLUE pour proposer notre système d'apprentissage SIGMA.

A l'instar des systèmes d'appariement existants, SIGMA manipule des sources de données géographiques. Le format retenu pour ces sources est le modèle de données du consortium OpenGIS [6] et son encodage GML (Geography Markup Language) [7]. Un schéma de données GML fournit plusieurs choses : un système de type de données et une Description de Schéma en XML (XSD) avec les notions d'association, d'héritage et de cardinalité. Un schéma de données géographiques est ainsi composé d'un ensemble de *classes* ou *concepts*, liées par des relations d'héritage, d'association ou d'agrégation, et dont les cardinalités sont précisées. Chaque classe représente un type de données géographique et possède un ensemble d'*attributs* ou *propriétés*. Notre problème d'appariement consiste à établir les règles de correspondance entre les éléments d'un schéma de source de données et un schéma global, sachant qu'un élément est une classe ou un attribut de classe.

1.3 Structure de l'article

Cet article est organisé comme suit. La section 2 décrit un exemple réel que nous avons retenu décrire notre approche. Dans la section 3, nous discutons les limitations des approches LSD et GLUE dans le cadre géographique. Notre solution est décrite en section 4. Enfin, nous concluons en section 5 en insistant sur le caractère novateur de notre approche.

2 Un exemple

Pour illustrer l'hétérogénéité sémantique issue des données géographiques, nous considérons trois sources de données BDTQ, CITS et Sherbrooke, du Québec. Il s'agit de sources réelles dont l'intégration fait partie des objectifs du groupe REV !GIS dans leur projet IST-1999-14189. La source BDTQ [8] représente la Base de Données Topographiques du Québec à l'échelle 1/20 000. Elle comprend, initialement, 25 classes couvrant les couvertures : hydrographie, voie de communication, aire désignée, bâtiment, équipement, végétation, forme terrestre, frontière et habillage cartographique. Pour simplifier, nous considérons uniquement les classes concernant l'hydrographie, les bâtiments et les voies de communication décrites dans le schéma S_{BDTQ} de la figure 1.

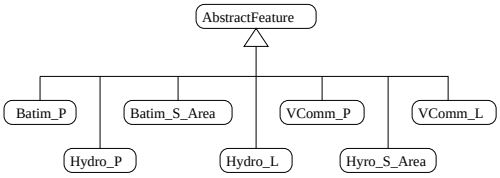

FIG. 1 – Schéma de la source BDTQ

Mis à part les attributs clé et géométrique, toutes les classes du schéma S_{BDTQ} possèdent les mêmes attributs numériques, à savoir : *No_Seq_Elemn, Indicatif, Co_Geomt, Co_Manip, Da_Manip, Zmin, Amax, Echelle, Co_Prodt, Co_Metad, No_Seq_Ratch* et *Toponyme*. Ces classes possèdent un attribut textuel, nommé *Description*, indiquant le type de l'objet géographique. Dans le tableau 1 de l'annexe, nous donnons la signification de chaque classe ainsi que les valeurs possibles de l'attribut *Description*.

La source CITS couvre, à l'échelle 1/50 000, une zone plus étendue. Elle contient initialement 196 classes décrivant à peu près les mêmes couvertures que celles de la source BDTQ, mais dans une *ontologie* assez différente. Pour simplifier, nous en décrivons une petite partie dont le schéma S_{CITS} est illustré dans la figure 2.

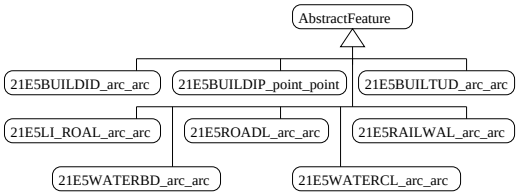

FIG. 2 – Schéma de la source CITS

Comme pour BDTQ, les classes de la source CITS possèdent les mêmes attributs, parmi lesquels *EntityName* qui décrit le type de l'objet. Les valeurs possibles de cet attribut sont données dans le tableau 2 de l'annexe.

La source Sherbrooke représente des données cadastrales couvrant la ville de Sherbrooke et ses alentours. Elle est composée d'une seule classe ayant un attribut, *EntityType*, qui associe à chaque objet géographique un nombre (3, 7, 8, 19 ou 500) indiquant son type.

Nous nous intéressons à intégrer ces sources dans un système dont le schéma global couvre les voies de communication (*Route, Tronçon, Pont, Extrémité, Péage, Tunnel* et *Carrefour*), les bâtiments (*Bâtiment*) et l'hydrographie (*CoursEau* et *Lac*). Les liens entre ces classes sont définis dans le diagramme UML de la figure 3. La classe Route se décompose en tronçons de routes dont les extrémités correspondent à des carrefours, tunnels ou péages. L'attribut *Type* de *Route* ou *Tronçon* sert à définir à quel type de route l'objet appartient. Il s'agit de dire si c'est une *Route, Rue* ou Autoroute. La classe *Bâtiment* regroupe toute sorte de bâtiments fixes. Son attribut *Type* précise le type d'un bâtiment. Contrairement au cas des classes *Route* et *Tronçon*, nous n'avons pas de valeurs prédéfinies de cet attribut.

3 Limitations des approches existantes dans un cadre géographique

La difficulté d'intégrer des données géographiques est liée à plusieurs facteurs : la richesse sémantique de l'information géographique, son

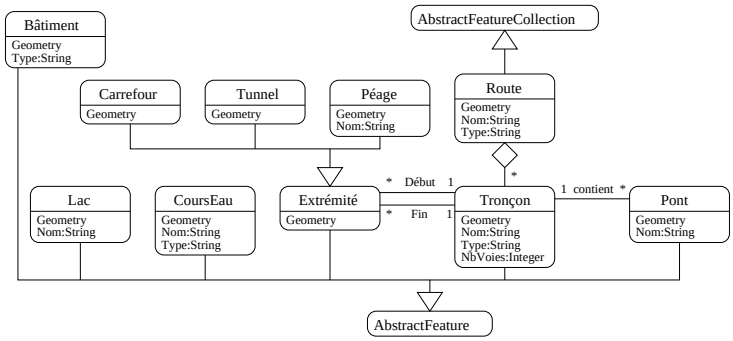

FIG. 3 – Schéma de médiation

hétérogénéité *ontologique*, son caractère pauvre (généralement) en information textuelle et les problèmes liés à la "connaissance métier" des concepteurs de bases de données géographiques. Effectuer des correspondances inter-schémas est un travail difficile à automatiser. Dans ce qui suit, nous nous référons à l'exemple que nous avons décrit dans le paragraphe 2 pour identifier les causes de cette difficulté et les limitations des systèmes LSD et GLUE dans le cadre géographique.

3.1 Problèmes liés à la nomenclature

D'abord, il est important de rappeler que, d'une part, malgré les travaux sur la standardisation de la nomenclature des éléments géographiques (voir EDIGEO [9, 10]), les fournisseurs de bases de données géographiques "ne se sentent pas obligés" de respecter ces standards. Non seulement la différence de nomenclature est importante, mais aussi une nomenclature peu significative pour l'utilisateur final peut être employée. C'est le cas de la source Sherbrooke où les types des objets sont numériques et, par conséquent, n'ont aucune signification. L'exemple CITS montre aussi des noms de classes peu significatifs et difficiles à interpréter : des noms composés de chiffres, d'abréviations et de mots tels que "arc" et "point".

Ensuite, il est difficile de gérer la polysémie de la nomenclature pour établir des correspondances plus précises. Ce problème relève aussi de l'hétérogénéité ontologique. En fait, un terme ou un concept peut être défini de plusieurs manières selon l'ontologie de la source. Par exemple, dans la source BDTQ, le concept de bâtiment regroupe les maisons fixes et mobiles. Cependant, dans le schéma global, il ne regroupe que les bâtiments fixes. Selon le choix de la définition d'un terme (nom de classe ou attribut), les correspondances entre le schéma de la source et celui de médiation se font différemment.

Les problèmes liés à la nomenclature constituent une limitation pour les apprenants textuels qui pourraient voir leurs performances baisser devant la richesse de nomenclature ou même s'annuler face à une nomenclature non significative (cas des données cadastrales de Sherbrooke).

3.2 De XML à GML

Une autre limitation de l'approche LSD, pour les données géographiques, provient de différences fondamentales entre l'appariement dans le cadre de données XML générales et celui dans le cadre de données géographiques GML. Des données générales XML ont une structure arborescente où des nœuds s'agrègent récursivement jusqu'à obtenir des feuilles qui sont des attributs de types simples, textuels ou numériques. Ces feuilles présentent souvent des caractéristiques telles que leur nom, les structures particulières de leurs contenus, etc. Ils sont eux-même souvent caractéristiques de leurs nœuds parents, i.e., les nœuds parents se distinguent par leurs feuilles ou leurs fils. Établir les correspondances entre ces feuilles, d'une part, est un problème bien adapté aux apprenants utilisés par LSD ou GLUE et, d'autre part, facilite la correspondance entre les nœuds parents.

Quant à la structure de données géographiques GML, elle fait intervenir des relations autres que l'agrégation (héritage et association) et une notion de cardinalité. De plus, la majeure information sur les objets géographiques (instances de classes) réside souvent dans l'attribut géométrique qui est de type assez complexe et, par conséquent, ne peut pas être traîté par les éléments du système LSD. Le reste des attributs des classes ne sont

pas forcément caractéristiques aux classes comme le montrent nos exemples où toutes les classes géographiques ont les mêmes attributs. Par conséquent, la correspondance entre les attributs n'aide pas à établir la correspondance entre les classes. Enfin, dans le cadre géographique, les attributs reflètent essentiellement le point de vue du producteur et de sa "connaissance métier" : ils peuvent relever de la partie méta-information ou tout simplement de domaines indépendants de ceux visés par le schéma global (orienté utilisateur). C'est le cas de la source BDTQ où les seuls attributs intéressants pour le schéma global sont l'attribut géométrique, le *Toponyme* et la *Description*.

3.3 Hétérogénéité ontologique et différence de granularités

Dans notre cadre, nous effectuons des correspondances entre les éléments de deux structures GML, que l'on représente avec des diagrammes UML pour une meilleure illustration. Ces diagrammes sont souvent de granularités différentes (granularité de classification). De plus, ils font intervenir non seulement des agrégations mais aussi des notions d'association et de généralisation / spécialisation. D'une part, les correspondances 1-1 ne sont plus assurées. Les notions de décomposition, et héritage et les différences d'ontologies engendrent assez souvent des correspondances multiples, du type $1 - n$ ou même $m - n$. Pour illustrer cela, reprenons l'exemple de la source BDTQ. Remarquons que les classes *Batim_P* et *Batim_S_Area* correspondent toutes les deux à la classe *Bâtiment* du schéma global. Les classes *VComm_P* et *VComm_L* correspondent aux classes *Tronçon*, *Tunnel* et *Pont*. D'autre part, une correspondance entre deux classes peut se définir avec une condition sur les valeurs de leurs attributs (problème d'imbrication). Par exemple, la classe *VComm_L* de BDTQ correspond à la classe *Pont* du schéma global avec la condition

Description = '*Pont*'

ou aussi la classe *Batim_P* correspond à la classe *Bâtiment* avec la condition

Batim_P.Description $\neq$ '*Maison mobile*' et *Batim_S_Area.Description* $\neq$ '*Serre*'.

3.4 Le facteur géométrie

Les données géographiques contiennent un aspect géométrique. De fait, les types géographiques ont souvent des propriétés géométriques pouvant les distinguer les uns des autres. Ces propriétés ne portent pas uniquement sur des objets vus isolément les uns des autres (forme d'un objet, mesures, etc.), mais aussi sur une vue d'ensemble des données (ou une partie du moins) qui permet de voir comment ces objets sont disposés dans l'espace. Par exemple, quand nous regardons des routes (ou tronçons de routes) nous remarquons qu'il s'agit de lignes assez régulières et allongées c'est ce qui pourrait les distinguer d'autres objets (bâtiments, lacs, etc.). Ces routes ou tronçons, vue dans leur ensemble, forment souvent une sorte de grille assez régulière (angles quasiment droits), c'est ce qui les distingue des cours d'eau par exemple. Aussi, une rue peut se différencier d'une autoroute par sa longueur mais aussi par sa situation dans un milieu urbain. De même, les bâtiments peuvent être vus comme des objets, ponctuels ou polygonaux de forme assez géométriques (angles droits et un petit nombre de côtés) et ayant des surfaces d'un ordre de grandeur certain, formant le plus souvent, dans des zones urbaines, des regroupements où certains sont assez proches les uns des autres et éventuellement séparés par des rues, etc. Si certaines de ces propriétés sont difficiles à décrire formellement, d'autre le sont moins et peuvent être calculées : longueur, aire, allongement, densité, liens topologiques ou métriques simples, etc.

Utiliser la géométrie dans une approche d'appariement n'est pas sans risque à cause notamment des problèmes liés à la qualité de la source de données et la manière de représenter les objets géographiques. En fait, l'aspect géométrique des données géographiques est directement lié à leur *qualité*, en particulier à l'échelle de représentation des objets. Les sources de données étant de qualités hétérogènes, nous avons souvent affaire à des représentations différentes, des géométries hétérogènes ou des géométries peu "parlantes" (ponctuelles par exemple). Par exemple, des bâtiments peuvent être représentés par des objets ponctuels (s'il sont de tailles petites par rapport à l'échelle ou pour d'autres raisons) ou polygonaux. Cette hétérogénéité de représentation géométrique peut apparaître au sein de la même source de données. C'est le cas de la source BDTQ qui offre deux représentations des bâtiments *Batim_P* (ponctuels) et *Batim_S_Area* (surfaciques). Certaines représentations (par exemple la représentation ponctuelle des bâtiments) sont moins "parlantes" parce qu'elles ne permettent pas de calculer des ca-

ractéristiques des objets représentés telles que la surface, la longueur, etc. C'est ce qui pose une difficulté de plus pour reconnaître géométriquement les objets géographiques et qui pourrait, par conséquent, limiter les performances d'apprenants géométriques.

4 Notre solution

Nous décrivons dans cette section le système SIGMA basé notamment sur le concept d'apprenant géographique qui utilise la géométrie.

Pour améliorer la précision de l'appariement vis à vis du problème de correspondances multiples dû aux différences de granularités (cf. paragraphe 3.3), nous proposons une étape préliminaire de *raffinement* ou *extension* du schéma local. Il s'agit d'identifier des sous-classes des classes initiales pour construire un nouveau schéma appelé *schéma raffiné* ou *étendu*. Le but de définir un tel schéma est d'obtenir un schéma de niveau de granularité plus élevé facilitant et améliorant l'appariement avec le schéma global. Ainsi, l'étape d'appariement s'effectuera ultérieurement entre ce nouveau schéma raffiné et le schéma global.

Dans la suite, nous décrivons les étapes de raffinement et d'appariement pour un schéma de données noté S et un schéma global G.

4.1 Raffinement

Dans cette étape, nous cherchons à raffiner la classification du schéma S en construisant un schéma étendu S^* ayant un niveau de granularité supérieur et contenant tous les types géographiques couverts par la source. La construction d'un tel schéma est basée sur la notion d'*attribut discriminant* ou *propriété discriminante*. Un attribut discriminant pour une classe donnée est "un attribut dont les valeurs permettent de discriminer des sous-classes différentes au sein de cette même classe". Même si cette définition n'est pas formelle, reconnaissance d'un attribut discriminant est assez intuitive en pratique. Souvent, il s'agit d'un attribut qui associe une description ou une valeur (parmi un nombre assez limité de descriptions ou valeurs) à chaque instance de la classe en question. C'est le cas des attributs *Description* des classes de BDTQ, *EntityName* de CITS et *EntityType* de la source Sherbrooke.

La construction du schéma raffiné S^* est obtenu en recherchant les attributs discriminants des classes du schéma S et en dénombrant leurs valeurs possibles. Considérons une classe C de S, pour fixer les idées la classe Hydro_S_Area de la source BDTQ. Si T est un attribut discriminant de la classe C (pour notre exemple il s'agit de l'attribut *Description*) et $D(T)$ est l'ensemble des valeurs possibles de T, alors la classe C de S peut être remplacée dans S^* par les sous-classes désignées par les couples (C, d), où $d \in D(T)$. Dans notre exemple, la classe Hydro_S_Area est étendue par l'ensemble des sous classes :

(Hydro_S_Area, Barrage), (Hydro_S_Area, Buse), ..., (Hydro_S_Area, Ligne virtuelle de plan d'eau)

Le raffinement du schéma S repose sur le choix des propriétés discriminantes. Pour simplifier nous supposons qu'une classe admet au plus une propriété discriminante. Dans les exemples rencontrés en pratique, ceci reste généralement suffisant et un attribut discriminant se distingue souvent par des particularités telles que son nom (de la forme "Description", "Type", "EntityType", etc.), son type (souvent textuel descriptif ou énuméré) et ses valeurs (souvent référent à des types géographiques). Ainsi, la recherche d'une propriété discriminante peut être facilitée par l'emploi d'un système d'apprenants. Nous définissons un système d'apprenants de base *Name Learner* et un *Content Learner* analogues à leurs homonymes définis dans LSD [2, 3]. Ces deux apprenants sont entraînés à l'aide d'exemples de noms et de valeurs d'attributs discriminants enrichis par un dictionnaire de synonymes, d'abréviations et de combinaisons classiques. Étant donné une classe C de S, le *Name Learner* (respectivement *Content Learner*) associe à chaque attribut T de C un score $S_n(T)$ (respectivement $S_c(T)$) reflétant le degré auquel il croit que T est l'attribut discriminant de C en se basant sur son nom (respectivement un ensemble de ses valeurs).

Lors de la phase de recherche d'un attribut discriminant pour une classe C, un méta-apprenant permet de combiner les prédictions $S_n(T)$ et $S_c(T)$ pour l'attribut T à l'aide de coefficients W_n et W_c et d'une valeur seuil σ de la façon suivante :

$$S(T) = max(0, W_n S_n(T) + W_c S_c(T) - \sigma)$$

Les coefficients W_n et W_c reflètent les degrés de confiance des apprenants respectifs *Name Learner* et *Content Learner*. La valeur seuil σ est un seuil (positif) au dessous duquel le score final de l'attribut s'annule. Ces différents coefficients sont

obtenus grâce à une phase d'entraînement. Dans cet entraînement, le méta-apprenant demande aux apprenants de base de prédire pour chaque élément d'un ensemble E d'attributs s'il s'agit ou pas d'un attribut discriminant. Connaissant le résultat, il peut ainsi juger leurs capacités à prédire le caractère discriminant des attributs et choisir un seuil. Il affecte alors les degrés de confiance W_n et W_c aux apprenants ainsi qu'une valeur σ au seuil minimal.

Un agent décisif classe les attributs de scores positifs dans l'ordre décroissant. Le premier étant le plus susceptible d'être l'attribut discriminant il est proposé par cet agent à l'utilisateur à travers une interface permettant de montrer de visualiser des exemples de valeurs de cet attribut. L'utilisateur confirme ou refuse le choix. Dans le cas d'un refus le choix suivant est proposé, jusqu'à validation ou échec.

Une fois la propriété discriminante est identifiée, le schéma S peut être raffiné, nous obtenons le schéma étendu S^*.

4.2 Appariement

Dans cette étape, nous effectuons les correspondances entre le schéma étendu S^* construit dans l'étape précédente et le schéma global G. Nous utilisons une technique analogue à celle employée par LSD.

SIGMA utilise pour l'appariement les apprenants de base de LSD, des apprenants géométriques, un apprenant GML, un méta-apprenant, un convertisseur et un combinateur de contraintes. Il prévoit une phase d'apprentissage des modules apprenants et une phase de classification.

Phase d'apprentissage. Supposons que l'on dispose d'une source de données initiale s_0 dont le schéma raffiné est noté S_0. Nous fournissons dans un premier temps un appariement manuel entre les éléments du schéma S_0 et ceux du schéma G. Ensuite, nous extrayons un échantillon de données de la source P qui permettra d'entraîner les apprenants de base et les apprenants géométriques. Les apprenants géométriques ne participent qu'à la reconnaissance des classes à caractère géométrique et pas des attributs. Chaque apprenant est entraîné à reconnaître les éléments du schéma S pour pouvoir prévoir, ultérieurement, les correspondances des éléments d'une nouvelle source avec le schéma global G.

En fait, chaque apprenant a une capacité à reconnaître un certain élément du schéma G. Pour savoir combiner les décisions des apprenants de base et décider sur la classification d'un nouvel élément, le méta-apprenant a besoin de quantifier la confiance qu'il accorde à un apprenant vis-à-vis un d'élément donné de G. Ainsi, dans la phase d'apprentissage, il demande aux apprenants de base et, éventuellement, aux apprenants géométriques de classifier les éléments d'un échantillon de données. Connaissant les vrais correspondances avec le schéma global, il affecte à chaque apprenant A et chaque élément l du schéma global un degré de confiance W_l^A fonction de sa capacité à reconnaître cet élément.

Phase de classification. Une fois les apprenants entraînés, ils sont prêts à effectuer les liens entre la nouvelle source s dont le schéma étendu est S^* et le schéma de médiation G. Supposons que l'on veuille classifier un élément e de la schéma S^*. Alors, nous commençons par extraire un échantillon δ d'instances de e à partir de la source s. Ensuite, nous considérons chaque élément de l'échantillon un par un. De chaque élément d de l'échantillon, nous envoyons l'information appropriée concernant e. Par exemple, si un apprenant L s'intéresse au nom de l'élément, nous lui envoyons le nom de e, s'il s'intéresse à la valeur (contenu), nous lui envoyons la valeur de e pour l'élément d de l'échantillon. Un module de calculs géométrique permet d'envoyer les informations nécessaires pour les apprenants géométriques.

Ensuite, chaque apprenant associe à chaque élément du schéma global un *degré de confiance* $c_{e,l}^A(d)$ représentant son degré de correspondance avec l'élément e compte tenu de la donnée d.

En se basant sur les paramètres de confiance accordés aux apprenants vis-à-vis de chaque élément du schéma G, le méta-apprenant combine les prédictions de tous les apprenants pour l'élément e et issues de l'élément d de l'échantillon δ. Il calcule pour chaque élément l du schéma G un coefficient $c_{e,l}(d)$ indiquant le degré auquel il croit associer l'élément e à l compte tenu de la données d :

$$c_{e,l}(d) = \sum_A W_l^A c_{e,l}^A(d)$$

Le même processus est répété sur le reste de l'échantillon. Ainsi, nous obtenons une liste de prédictions contenant une prédiction par élément

de l'échantillon δ. Le convertisseur combine ces prédictions en calculant leur moyenne et associe à chaque étiquette l un coefficient d'association $C_{e,l}$:

$$C_{e,l} = \frac{1}{|\delta|} \sum_{d \in \delta} c_{e,l}(d).$$

Nous obtenons ainsi une matrice de correspondances $(C_{e,l})_{e \in S^*, l \in G}$ entre les éléments du schéma étendu de la source et ceux du schéma global. L'étape finale de l'appariement consiste à associer une étiquette à chaque élément du schéma S^*. Ceci est effectué par un combinateur de contraintes dont le rôle est de rechercher un étiquetage correspondant aux coefficients de correspondances les plus grand tout en satisfaisant un ensemble de contraintes d'intégrité sur les élément des schémas. Ces contraintes sont définies par l'utilisateur (administrateur du système). Elles sont assez variées. Une contrainte possible pourrait être de la forme : "Si les éléments (nœuds) a et b de S^* correspondent respectivement aux éléments *Route* et *Route.Type* du schéma global, alors b doit être un attribut de a dans le schéma de la source".

Apprenants de base. Nous passons en revue les apprenants que nous utilisons.

- **Name Matcher** Il classifie un élément XML (vu comme nœud d'un arbre) en se basant sur son nom, étendu avec des synonymes ainsi que tous les noms des éléments qui lui sont hiérarchiquement supérieurs à partir de la racine. Il utilise la technique du "1-plus proche voisin" comme le système Whirl, développé dans [11].
- **Content Matcher** Il utilise également Whirl. Cependant, il se base sur le contenu de l'élément GML au lieu de son nom. Il est bien adapté aux éléments textuels longs, tels que descriptifs, et aux éléments admettant des valeurs assez distinctes et descriptives, telles que des couleurs, etc. Il n'est pas adapté aux éléments ayant de petites valeurs numériques.
- **Naive Bayes Learner** Il exploite les fréquences de mots [12] et il est très efficace quand il existe des mots fortement indicatifs pour le *label* en question. Il est efficace aussi dans le cas d'un grand nombre de mots faiblement indicatifs. Il voit son efficacité baisser quand il s'agit de champs numériques ou courts (code postal, couleur, etc.).
- **GML Learner** Le principe de son fonctionnement est analogue à celui de l'apprenant XML de LSD, à la différence qu'il exploite la sémantique des relations entre les éléments (héritage, agrégation, association, cardinalité).

Apprenants géométriques. L'idée derrière l'apprentissage géométrique est que les objets géographiques d'une classe donnée possèdent souvent des caractéristiques géométriques les distinguant des objets d'autres classes. Ainsi, les apprenants géométriques essayent d'exploiter ces propriétés pour classifier des instances des classes du schéma de la source. Nous distinguons deux type de propriétés géométriques : les propriétés géométriques individuelles qui concernent un objet géographique pris isolément (exemples de l'aire, la longueur, le périmètre, etc.) et les propriétés géométriques d'ensemble qui caractérisent un ensemble d'objets (exemples de la densité, la densité surfacique, etc.). À la suite de cette distinction, nous proposons dans ce paragraphe deux apprenants géométriques de base.

Apprenant géométrique individuel Cet apprenant exploite les propriétés géométriques des objets pour les classifier. Ces propriétés permettant une éventuelle classification sont très nombreuses mais certaines d'entre elles sont plus distinctives que d'autres dans la reconnaissance d'un certain type d'objets. Pour illustrer cela, considérons un système très simple avec deux classes géographiques : les bâtiments et les routes surfaciques. Intuitivement, les routes se distinguent des bâtiments par leurs diamètres et surfaces. Si nous considérons uniquement ces deux critères, un modèle mathématique simple de cette idée de classification géométrique peut s'illustrer par la figure 4-a. Dans ce modèle, la classification d'un objet surfacique o d'aire A et diamètre Δ est faite selon la position de (A, Δ) par rapport aux droites $\mathcal{D}_B$ et $\mathcal{D}_R$. Si (A, Δ) est situé au dessous de $\mathcal{D}_B$ alors o sera considéré Bâtiment. Si (A, Δ) est situé au dessus de $\mathcal{D}_R$ alors Route. Remarquons que les classifications ne sont pas forcément disjointes, i.e. un objet peut être considéré à la fois Bâtiment et Route (zone entre $\mathcal{D}_B$ et $\mathcal{D}_R$). Ce genre de raisonnement se traduit naturellement par un système d'apprentissage basé sur un réseau de neurones [13, 14, 15] comme illustré dans la figure 4-b. Nous ne détaillons pas ici la technique des réseaux de neurones, mais nous en montrons la structure (couches et nœuds) utilisée sur l'exemple simple puis dans notre cas général. Dans le réseau neuronal représenté en figure 4-b, des nœuds d'entrée correspondent aux

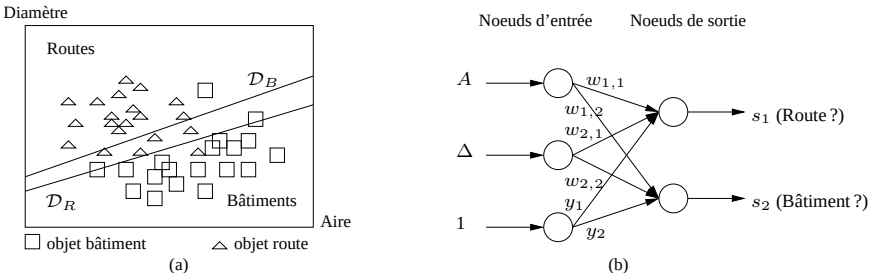

FIG. 4 – Exemple simple d'apprentissage de reconnaissance des bâtiments et routes

propriétés de l'objet o. La dernière entrée est une entrée constante (qui vaut 1). À chaque classe (ici bâtiment et route) correspond un nœud de sortie. La valeur de cette sortie est *vrai* si l'objet est jugé correspondant à la classe en question et *faux* sinon. Les sorties sont calculées en fonction de combinaisons linéaires des entrées et des poids réglables $w_{i,j}$ et y_i, $(i, j \in \{1, 2\})$. Dans la phase d'apprentissage, les valeurs des poids sont réglées en partant d'un échantillon d'objets dont nous connaissons les classes.

De la même manière, si nous disposons d'un schéma global avec n classes, nous définissons un apprenant neuronal dont la couche de sortie comprend n nœuds, chacun correspondant à une classe. La couche d'entrée contient un nœud par paramètre géométrique. L'apprenant exploite les propriétés géométriques individuelles des objets. Pour rechercher les correspondants d'une classe t d'une source de données dans le schéma global, l'apprenant géométrique part d'un échantillon d'instances de t. Il essaie de classifier chaque instance de l'échantillon en se basant sur l'ensemble de ses propriétés géométriques. Nous proposons les propriétés géométriques de base suivantes :

- **Type géométrique** Correspond à l'un des types géométriques du modèle GML (0 :Abstract, 1 :Point, 2 :Line, 3 :Polygon, 4 :MultiPoint 5 :MultiLineString, 6 :MultiPolygon).
- **Aire** Notée A, est l'aire totale de l'objet.
- **Diamètre** Noté Δ, est la plus grande distance entre deux points de l'objet.
- **Longueur** C'est la somme des longueurs de tous les composants de l'objet.
- **Périmètre** Désigne la somme des périmètres de tous les composants de l'objet.
- **Allongement** C'est le rapport du diamètre et de l'aire de l'objet.
- **Concavité** C'est le rapport de la longueur de l'objet et de son diamètre.
- **Compacité** Notée c, mesure le degré de ressemblance à un cercle ou disque. Elle est proportionnelle au rapport entre le carré du diamètre de l'objet et son aire. Plus précisément, elle se calcule de la façon suivante :

$$c = \frac{\pi}{4}\frac{\Delta^2}{A}.$$

  Plus la compacité est proche de 1 plus l'objet est circulaire.
- **Complexité** Désigne le rapport du nombre de sommets et de la longueur pour les objets linéaires ou multi- et le rapport du nombre de sommets et du périmètre pour les objets surfaciques. Pour le type ponctuel, elle est nulle.
- **Connexité** C'est le nombre de composantes pour un objet d'un type multiple.

L'apprentissage de l'apprenant consiste à régler ses poids de sorte à réduire le plus possible son erreur de prévision. Il s'effectue, comme pour les apprenants de base, à l'aide d'échantillon de données de la source dont les classes sont connues.

Apprenant géométrique ensembliste Cet apprenant exploite des propriétés caractérisant les objets géographiques vus dans leur ensemble. Considérons un type d'objets t du schéma local étendu. Pour rechercher les correspondants de t dans le schéma global, l'apprenant géométrique commence par un ensemble d'échantillons de données du type t. Un échantillon est choisi en délimitant une zone (rectangulaire) de taille "raisonnable" autour d'une instance de t choisie aléatoirement parmi les données. L'échantillon est composé de l'ensemble des instances qui chevauchent le contour de la zone. Pour cet échantillon, l'apprenant essaie de classifier t selon les paramètres suivants :

- **Densité** Désigne le rapport entre le nombre d'instances de t et l'aire de la zone délimitée.

– **Densité surfacique** Désigne le rapport entre la somme des aires des éléments de l'échantillon et l'aire de la zone délimitée.
– **Adjacence** Notée α, mesure le taux d'adjacence entre les éléments de l'échantillon. Elle est définie par :

$$\alpha = \frac{2a}{n(n-1)}$$

où a désigne le nombre de paires d'éléments adjacents et n le nombre d'éléments de l'échantillon.

L'apprenant géométrique ensembliste pourrait exploiter de nombreuses autres propriétés telles que l'aire moyenne, la longueur moyenne, le diamètre moyen, etc. des éléments de l'échantillon. Le principe de fonctionnement de l'apprenant ensembliste est semblable à celui de l'apprenant individuel.

5 Conclusion

Dans cet article, nous avons présenté une approche d'intégration automatique de données. Nous nous sommes intéressés aux correspondances qui peuvent être découvertes lorsqu'une source de données est candidate à l'intégration dans une architecture de médiation. L'approche est basée sur une technique d'apprentissage multi-stratégies, qui combine plusieurs apprenants pour déduire le degré de similarité entre concepts et favoriser l'émergence de règles automatiques de correspondance. Notre approche utilise les standards de l'OpenGIS (modèle de données et GML).

Le travail présenté dans cet article fait partie d'un travail plus large mené dans le cadre du projet RNTL VirGIS sur l'intégration des données géographiques.

Références

[1] Philip A. Bernstein Erhard Rahm. On matching schemas automatically. *VLDB Journal*, 2001.

[2] Pedro Domingos Anhai Doan and Alon Levy. Learning source descriptions for data integration. In *Proceedings of the International Workshop on The Web and Databases (WebDB)*, 2000.

[3] Pedro Domingos Anhai Doan and Alon Halevy. Reconciling schemas of disparate data sources : A machine-learning approach. In *Proc. of ACM SIGMOD Conf. on Management of Data*, 2001.

[4] Pedro Domingos Anhai Doan, Jayant Madhavan and Alon Halevy. Learning to map between ontologies on the semantic web. In *Proc. of the Int. (WWW) Conf.*, 2002.

[5] Infrastructure logicielle pour l'interopérabilité de systèmes d'information géographique. URL : http ://www.telecom.gouv.fr/rntl/FichesA/-Virgis.htm, October 2000.

[6] Opengis consortium. URL : http ://www.opengis.org.

[7] Ron Lake Simon Cox, Adrian Cuthbert and Richard Martell. Geography markup language (gml) 2.0. URL : http ://www.opengis.net/gml/01-029/GML2.html, February 20th 2001.

[8] Base de données topographiques du québec (bdtq), 1 :20,000. URL : http ://felix.-geog.mcgill.ca/heeslib/bdtq_20000.html, July 2001.

[9] Nomenclature d'échange du cnig associée à la norme edigéo. URL : http ://www.cnig.fr/cnig/infogeo/france/-Normalisation/nomenclature.cnig.1.fr.html, 1995.

[10] Ministère de l'Économie de Finances et de l'Industrie. *Standard d'échange de objets du plan cadastral informatisé*, 2001 December.

[11] W. Cohen and H. Hirsh. Joins that generelize : Text classification using whirl. In *Proc. of the Fourth Int. Conf. on Knowledge Discovery and Data Mining*, 1998.

[12] P. Domingos and M. Pazzani. Beyond independence : Conditions for the optimality of the simple bayesian classifier. In Morgan Kaufmann, editor, *Proceedings of the Thirteenth International Conference on Machine Learning*, pages 105–112, Bari, Italy, 1996.

[13] M.Wynne-Jones R. Rohwer and F. Wysotzki. *Machine Learning, Neural and Statistical Classification*, chapter Neural Networks, pages 84–106. Ellis Horwood, February 1994.

[14] A. Krogh J. Hertz and R. Palmer. *Introduction to the Theory of Neural Computation*. Addison-Wesley, 1991.

[15] P.D. Wasserman. *Neural Computing, Theory and Practice*. Van Nostrand Reinhold, 1989.

Annexe

Classe	Signification	Valeurs possibles de l'attribut *Description*
Batim_P	Bâtiment ponctuel	Bâtiment, Bâtiment en construction, Bâtiment en ruines, Maison mobile, Serre
Batim_S_Area	Bâtiment de surface	Bâtiment, Bâtiment en construction, Bâtiment en ruines, Serre
VComm_P	Voie de communication ponctuelle	Pont
VComm_L	Voie de communication linéaire	Autoroute, Autoroute à axes fusionnés, Chemin, Chemin non carrossable, Chemin carrossable non pavé, Chemin carrossable pavé, Bretelle, Gué, Écran antibruit, Mur de soutènement, Pont Pont couvert, Passerelle, Route collectrice, Route collectrice non pavée, Route collectrice pavée, Route d'accès aux ressources, Route d'accès aux ressources non pavée, Route d'accès aux ressources pavée, Route locale, Route locale non pavée, Route locale pavée, Route nationale, Route nationale non pavée, Route nationale pavée, Route régionale, Route régionale non pavée, Route régionale pavée, Rue, Rue non pavée, Rue pavée, Traverse, Tunnel, Pont d'étagement, Voie de communication, Voie de communication abandonnée, Voie de communication en construction, Voie ferrée
Hydro_P	Hydrographie (point)	Chute, Écueil, Rapide
Hydro_L	Hydrographie (ligne)	Barrage, Buse, Canal, Chute, Cours d'eau, Cours d'eau intermittent, Écueil, Rapide
Hydro_S_Area	Hydrographie (surface)	Barrage, Canal, Cours d'eau, Écluse, Île, Lac Mare, Réservoir hydroélectrique, Ligne virtuelle de plan d'eau

TAB. 1 – Description des classe de BDTQ

Classe	Signification	Valeurs de l'attribut *EntityName*
21E5BUILDID_arc_arc	Bâtiment linéaire	Building
21E5BUILDIP_point_point	Bâtiment ponctuel	Building
21E5BUILTUD_arc_arc	Zone urbaine	Built-UpArea
21E5LI_ROAL_arc_arc	Route à usage limité	Limited-UseRoad
21E5ROADL_arc_arc	Route	Road
21E5RAILWAL_arc_arc	Voie ferrée	RailWay
21E5WATERBD_arc_arc	Hydrographie (ligne)	WaterBody
21E5WATERCL_arc_arc	Cours d'eau	WaterCourse

TAB. 2 – Description des classe de CITS