Restructuration automatique de documents dans les corpus semi-structurés hétérogènes

Guillaume Wisniewski*, Ludovic Denoyer*, Patrick Gallinari*
* Laboratoire d'Informatique de Paris 6
8 rue du Capitaine Scott, 75015 Paris
{guillaume.wisniewski, ludovic.denoyer, patrick.gallinari}@lip6.fr

Résumé. L'interrogation de grandes bases de documents semi-structurés (type XML) est un problème ouvert important. En effet, pour interroger un document dont le schéma est nouveau, un système doit pouvoir soit adapter la requête posée au document, soit adapter le document pour pouvoir lui appliquer la requête. Nous nous positionnons ici dans le cadre de la restructuration de documents qui consiste à transformer des documents semi-structurés issus de diverses sources dans un schéma de médiation connu. Nous proposons un cadre statistique général à la problématique de la restructuration de documents et détaillons une instance d'un modèle stochastique de documents structurés appliquée à cette problématique. Nous détaillons enfin un ensemble d'expériences effectuées sur les documents du corpus INEX afin de mesurer la capacité de notre modèle.

1 Introduction

Le développement du document électronique et du Web a vu émerger puis s'imposer des formats de données semi-structurées, tels le XML et le XHTML. Ces nouveaux formats, décrivant simultanément la structure logique des documents et le contenu de ceux-ci, permettent de représenter l'information sous une forme plus riche que le simple contenu et adaptée à des besoins spécifiques. Étant donné l'augmentation rapide du nombre de documents semi-structurés, il est devenu nécessaire d'adapter les méthodes de traitement de données existantes afin de tenir compte des spécificités de ces nouveaux formats ainsi que d'étudier les nouvelles problématiques que ces formats font émerger.

L'initiative INEX (Fuhr et al 2002) propose d'étudier la problématique de la recherche documentaire sur des documents semi-structurés. L'hétérogénéité des structures des données est rapidement apparue comme un obstacle à la conception de systèmes d'interrogation de données semi-structurées issues de différentes sources d'information. Bien que, dans le cadre d'INEX, cette problématique ait été ignorée jusqu'à présent, l'édition 2004 de la campagne d'évaluation propose une nouvelle tâche, la tâche *hétérogène,* qui y est consacrée. Deux solutions peuvent être imaginées pour résoudre ce problème : les systèmes peuvent soit adapter la requête posée au document, soit adapter le document pour pouvoir lui appliquer la requête. Nous adoptons ici la deuxième solution et proposons d'utiliser un *schéma de médiation* pour exprimer l'ensemble des documents considérés dans une structure commune. L'utilisateur n'interagira alors qu'avec ce schéma de médiation. Cette solution nécessite de pouvoir *restructurer* les documents afin d'adapter leur structure au schéma de médiation.

La problématique de restructuration des données est apparue depuis de nombreuses années dans de nombreux domaines tels les entrepôts de données, l'intégration de données, le web sémantique, ... Plus récemment, plusieurs travaux se sont intéressés à l'application de cette problématique aux données semi-structurées et plus particulièrement aux données

XML. À chaque fois, le *schema matching* — action d'associer un élément d'une structure donnée à un élément, sémantiquement équivalent, d'une autre structure — a été identifié comme une étape fondamentale de la reconstruction. Cette tâche est particulièrement difficile, puisqu'elle nécessite d'inférer la sémantique d'un élément à partir de la structure du document ou d'exemples de données. C'est pourquoi elle est aujourd'hui généralement réalisée *à la main*, ce qui explique son coût élevé et son manque de fiabilité (Li et Clifton 2000).

De nombreux travaux en base de données proposent des méthodes pour automatiser la tâche de *matching*. Ces méthodes sont fondées sur l'exploitation de plusieurs types d'informations, comme le nom des éléments, leur type, des métadonnées, … et reposent généralement sur une connaissance a priori du schéma des données à intégrer. La quasi-totalité de ces algorithmes adoptent une approche de type « classification supervisée ». Cette approche est fondée sur une hypothèse d'unicité des éléments (un élément apparaît une et une seule fois dans le document) et suppose la structure cible connue. Les différentes méthodes proposées permettent alors d'obtenir un ensemble de règles d'association entre les éléments de structure des documents et du schéma de médiation. Cependant ces méthodes ne sont efficaces que si la sémantique des éléments est très forte et leur nom explicite. Leur évaluation a été faite sur des corpus de petite taille, relativement peu structurés et ne contenant que très rarement des données textuelles.

Le contexte des travaux effectués jusqu'à présent est donc très différent de celui de la recherche d'informations structurées. Dans ce domaine, les corpus considérés sont généralement très grands, les documents ont un contenu textuel important et ne respectent pas l'hypothèse d'unicité précédemment décrite : on trouve par exemple plusieurs éléments de type *paragraphe* dans un document. Ainsi, bien que les travaux effectués en base de données puissent servir de guide, il est nécessaire de développer des méthodes de restructuration spécifiques et adaptées aux corpus de documents semi structurés textuels.

Dans cet article, nous proposons une modélisation des documents structurés textuels par un modèle stochastique basé sur les réseaux bayésiens. Nous montrons ensuite comment, à l'aide d'un formalisme proche des modèles de langages, il peut être adapté à la problématique de restructuration. Nous présentons ensuite un ensemble d'expériences sur le corpus INEX afin de mesurer la capacité de ce modèle à restructurer des documents XML.

2 État de l'art

L'automatisation de l'intégration de données et du *schema matching* a été étudiée depuis longtemps par la communauté base de données. (Rahm et Bernstein 2001) présente une synthèse des travaux effectués. Ces travaux peuvent se regrouper en deux grandes catégories : la première regroupe des modèles qui travaillent à partir d'une connaissance *a priori* du schéma des données (DTD, schéma des bases de données, …), alors que la seconde regroupe des systèmes dont les approches utilisent des exemples de données. Par exemple, le système *SemInt* (Li et Clifton 2000) se base sur la similarité entre noms d'éléments ; le modèle *Similarity Flooding* (Melnik et al 2002) repose sur une analyse de la structure des données. (Kurgan et al 2002) constitue un exemple de travaux de la deuxième catégorie. D'autres travaux, comme (Madhavan et al 2003) préfèrent généraliser des exemples de *matchings* fait à la main.

À l'heure actuelle, le système *LSD* (Doan et al 2003) semble être une approche très intéressante qui propose d'utiliser des techniques issues de l'apprentissage afin d'obtenir un

ensemble de règles d'association entre les éléments d'un schéma d'entrée et les éléments du schéma de médiation. Cette approche repose sur une combinaison de différents systèmes d'apprentissage. Chacun de ces systèmes permet de caractériser un type d'information (nom du nœud, structure du document, contenu des nœuds, ...). Ils sont entraînés sur un ensemble d'exemples de documents exprimés à la fois dans le schéma d'origine et dans le schéma de médiation. La manière optimale de combiner ces différents systèmes est, elle aussi, apprise à partir de cet ensemble d'apprentissage. Les performances de ce modèle sont assez élevées et l'approche est particulièrement souple : de nouveaux types d'informations peuvent très facilement être pris en compte. Elle est aussi très générale puisqu'elle a été appliquée à des données XML, à des bases SQL mais aussi à des ontologies.

Tous ces algorithmes utilisent une approche de type classification supervisée des nœuds d'un document structuré qui permet d'attribuer une étiquette à chaque nœud (un tag dans le cas des documents XML). Cependant, ce type d'approche ne permet la reconstitution des documents dans le schéma de médiation uniquement si ce dernier est suffisamment simple. En effet, dans le cas de schémas complexes, il est nécessaire, en plus d'attribuer une étiquette à chacun des nœuds, de les « positionner » correctement les uns par rapport aux autres dans le document final. C'est pour cela que ces approches sont difficilement utilisables dans le contexte des documents textuels qui sont souvent très complexes. Seul (He et Chang 2003) considère les documents dans leur ensemble et réalise une véritable reconstruction des documents.

Le problème de l'évaluation du *schema matching* reste un problème ouvert : il n'y a pas, pour le moment, de cadre d'évaluation bien défini et chaque auteur propose ses propres mesures d'évaluation. (Do et al 2002) donne un aperçu des différentes méthodes d'évaluation utilisées.

3 Cadre de notre travail

Dans ce travail, nous nous intéressons au problème de la restructuration de documents dans le cadre de la recherche d'informations structurées dans les corpus de document XML. Notre objectif est d'arriver à convertir automatiquement de nouveaux documents dont le schéma est inconnu *a priori* dans un schéma de médiation avec lequel l'utilisateur interagira pour l'interrogation. Cette transformation devra donc conserver la sémantique de chacun des éléments du document. Ce contexte nous a amené à faire plusieurs hypothèses :

- les documents que nous considérons sont supposés venir tous de domaines similaires. On s'intéressera par exemple à la restructuration d'articles de journaux scientifiques parlant d'informatique. Les éléments qui composent ces documents sont donc sémantiquement relativement proches et on peut supposer qu'il existera toujours un élément de la structure de médiation correspondant à un élément de la structure d'origine.

- nous supposerons que nous disposons d'un ensemble de documents exprimés dans le schéma de médiation pour pouvoir effectuer l'apprentissage. Cette supposition nécessite soit de convertir une partie des documents à la main, soit d'utiliser le schéma d'un des corpus que l'on cherche à intégrer comme schéma de médiation.

Nous avons aussi choisi de ne pas utiliser d'informations a priori sur les schémas des documents (disponible par exemple dans les DTD, pour les documents XML) dans la mesure où celles-ci ne sont que rarement disponibles. Les *matchings* sont déduits directement du contenu et de la structure des documents du corpus.

4 Modèle stochastique de documents semi-structurés

Dans ce paragraphe, nous décrirons un modèle génératif de documents semi-structurés. Ce modèle est inspiré de celui présenté dans (Denoyer et Gallinari 2004) qui a été utilisé dans la classification supervisée de documents XML.

4.1 Modèle génératif de documents semi-structurés

Nous adoptons la représentation traditionnelle des documents semi-structurés sous forme d'arbre. À chaque nœud du document correspond un nœud du graphe et chacun des nœuds est composé d'une information de contenu et d'une étiquette. La figure 1 donne un exemple de représentation d'un document arborescent.

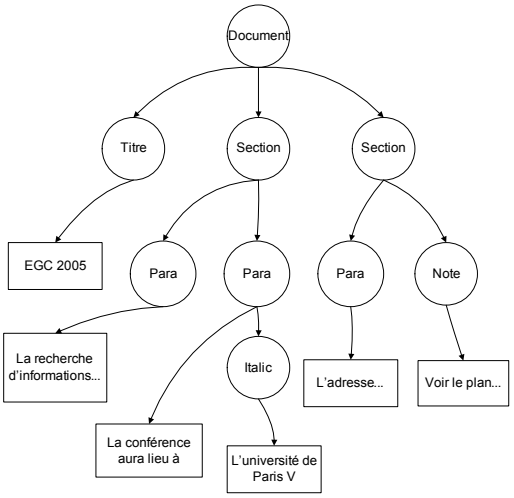

Figure 1 : Exemple de documents semi-structurés. Les informations de contenu apparaissent dans les rectangles ; les nœuds sont représentés par des cercles dans lesquels apparaissent les étiquettes.

Ainsi, pour un document d composé de $|d|$ nœuds, chaque nœud n_i peut s'écrire : $n_i = (s_i, t_i)$ où s_i représente l'étiquette du nœud et t_i le contenu de celui-ci. Nous avons alors, si nous appelons θ l'ensemble des paramètres du modèle :

$$\begin{aligned} P(d \mid \theta) &= P(n_1,...,n_{|d|} \mid \theta) \\ &= P\left((s_1,t_1),...,(s_{|d|},t_{|d|}) \mid \theta\right) \\ &= P(s_1,...,s_{|d|} \mid \theta) \cdot P(t_1,...,t_{|d|} \mid s_1,...,s_{|d|},\theta) \end{aligned} \qquad (1)$$

$P(d \mid \theta)$ peut donc se décomposer en un produit de deux termes : le premier décrivant la structure du document et le second le contenu de celui-ci. La probabilité de structure $P(s_1,...,s_{|d|} \mid \theta)$ décrit la manière dont la structure du document est générée et $P(t_1,...,t_{|d|} \mid s_1,...,s_{|d|},\theta)$, la probabilité de contenu, la manière dont le contenu de chaque nœud est généré. Il est impossible d'estimer directement ces deux probabilités. Nous sommes donc

amenés à proposer des hypothèses simplificatrices afin de réduire la complexité du problème. Ces hypothèses permettront d'insérer dans nos modèles certaines connaissances a priori, spécifiques aux problèmes ou aux corpus.

4.1.1 Probabilité de contenu

Nous ferons l'hypothèse que le contenu d'un nœud ne dépend que de l'étiquette de celui-ci. Posons $t_i = \left(w_i^1, \ldots, w_i^{k_i}\right)$, où les $\left(w_i^j\right)_{j \in 1,k_i}$ sont l'ensemble des mots apparaissant dans le i-ème nœud. En utilisant un modèle Naïve Bayes, on obtient alors :

$$P(t_1,...,t_{|d|} \mid s_1,...,s_{|d|},\theta) = \prod_{i=1}^{|d|} P(t_i \mid s_i,\theta) = \prod_{i=1}^{|d|}\prod_{j=1}^{k_i} P(w_j \mid s_i,\theta)$$

4.1.2 Probabilité de structure

L'estimation de la probabilité de structure est inspirée des grammaires probabilistes d'arbre (Carrasco et Rico-Juan 2002). La structure de l'arbre est décrite par une grammaire de type CFG, associant à chaque nœud la liste ordonnée de ses fils : par exemple, la règle *document*→*titre section section* indique qu'un nœud d'étiquette *document* aura trois enfants d'étiquettes respectives *titre* et *section* puis encore *section*. Nous considérons le processus génératif dans lequel l'auteur d'un document structuré découpe un document en plusieurs parties étiquetées puis, récursivement, redécoupe chacune de ces parties en sous-parties. Soit $childrentag(n_i)$ l'ensemble ordonné des étiquettes des enfants d'un nœud. La probabilité structurelle d'un document calculée par un tel modèle stochastique s'exprime alors ainsi

$$P(s_1,\ldots,s_{|d|} \mid \theta) = \prod_{i=1}^{|d|} P(childrentag(n_i) \mid s_i,\theta)$$

Cette hypothèse d'indépendance du premier ordre est du même type que celle faite par les *Hidden Tree Markov Model* (Diligenti et al 2003)

L'équation précédente permet de modéliser les documents par un réseau bayésien comme le montre la figure 2 qui illustre les dépendances entre les variables aléatoires décrivant le document. Ce réseau bayésien permettra de mieux comprendre la méthode de restructuration présentée dans la partie 5.

4.1.3 Probabilité du document

L'équation (1) peut donc finalement se réécrire :

$$P(d \mid \theta) = \prod_{i=1}^{|d|}\left(P(childrentag(n_i) \mid s_i,\theta) \cdot \prod_{j=1}^{k_i} P(w_j \mid s_i,\theta) \right)$$

4.2 Apprentissage

L'apprentissage du modèle nécessite que l'on puisse estimer :

- $\forall i \in \left[1,|d|\right], P(childrentag(i), s_i)$: la probabilité de trouver un ensemble de nœuds ordonnés sous un tag donné ;
- $\forall s \forall w P(w \mid s)$ la probabilité de trouver un mot sous un nœud donné.

Pour cela nous allons maximiser le logarithme de la vraisemblance du modèle sur le corpus d'apprentissage *D* :

$$L_D = log\left(\prod_{d \in D} P(d|\theta)\right)$$

Ceci revient à résoudre l'équation : $\nabla_\theta L_D = 0$ sous les contraintes assurant que les sommes des différentes probabilités estimées soient égales à 1.

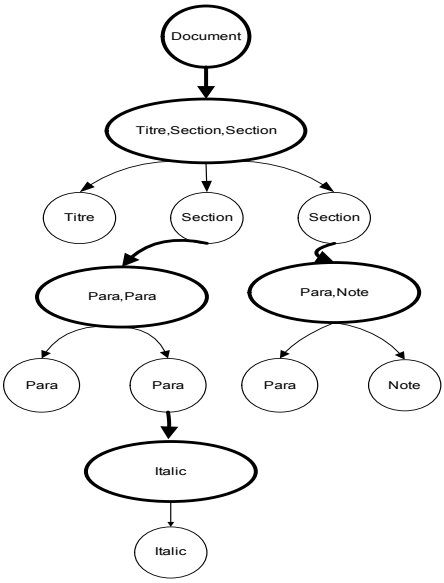

Figure 2 : Le réseau bayésien correspondant au calcul de la probabilité de structure pour le document de la figure 1. Les flèches plus épaisses correspondent aux probabilités $P(childrentag(n_i) \mid s_i, \theta)$. *Les flèches normales ne sont qu'une réécriture permettant de faire le lien avec le document XML.*

Le système précédent admet une solution analytique. Les valeurs lissées des estimateurs robustes que l'on peut dériver de cette solution sont :

$$P(s_1,...,s_m \mid s, \theta) = \frac{N^s_{s_1,...,s_m}}{N^s} + \varepsilon \qquad P(w \mid s, \theta) = \frac{N^s_w + 1}{N^s + |V|}$$

où :

- $N^s_{s_1,...,s_m}$ est le nombre de nœud dont l'étiquette est *s* et les enfants sont $s_1,\ldots,s_m$
- N^s est le nombre de nœuds avec l'étiquette *s*
- N^s_w est le nombre de fois où le mot *w* apparaît sous le nœud *s*
- $|V|$ est la taille du vocabulaire

5 Modèle de restructuration de documents

Les modèles de langages sont des modélisations statistiques stochastiques qui cherchent à modéliser le langage spécifique à une tâche donnée. Ces modèles ont été particulièrement utilisés dans le cadre de la recherche documentaire. Pour cette tâche, on cherche à évaluer la pertinence d'un document d pour une requête q à l'aide de la probabilité $P(q \mid d)$. Cette dernière correspond à la probabilité que la requête q ait été générée par un modèle de langage du document d. Nous proposons d'adopter une démarche similaire au problème de la restructuration : pour chaque document d^{orig} que l'on souhaite transformer, nous allons chercher à évaluer à quel point un document d est une restructuration de ce document. Pour cela, nous allons déterminer $P(d \mid d^{orig}, \theta)$, la probabilité que le document d ait été généré par un modèle de langage du document d^{orig}. On pourra alors définir la nouvelle structure du document comme étant :

$$d_{optim} = \arg\max_{d} P(d \mid d^{orig}, \theta)$$

Le calcul de cette probabilité est fondé sur le modèle de document semi-structuré que nous avons présenté au paragraphe précédent. Résoudre cette équation revient à considérer tous les documents d possibles. Comme leur nombre peut devenir rapidement très grand, il est nécessaire de faire des hypothèses simplificatrices. Nous considérons le processus génératif suivant : afin d'obtenir une restructuration d'un document, nous allons tout d'abord transformer la structure du document d'origine afin de l'exprimer dans le schéma de médiation, puis, une fois cette structure définie, nous « distribuerons » l'information de contenu dans les différents nœuds du document final. Ce processus est illustré par le réseau bayésien de la figure 3.

L'ensemble de ces hypothèses permet de simplifier l'expression de $P(d \mid d^{orig}, \theta)$:

$$\begin{aligned} P(d \mid d^{orig}, \theta) &= P\Big((t_1, s_1), \ldots, (t_{|d|}, s_{|d|}) \mid (t_1^{orig}, s_1^{orig}), \ldots, (t_{|d|}^{orig}, s_{|d|}^{orig}), \theta \Big) \\ &= P(s_1, \ldots, s_{|d|} \mid s_1^{orig}, \ldots, s_{|d|}^{orig}, \theta) \cdot P(t_1, \ldots, t_{|d|} \mid t_1^{orig}, \ldots, t_{|d|}^{orig}, s_1, \ldots, s_{|d|}, \theta) \end{aligned}$$

On retrouve la décomposition en probabilité de contenu / probabilité de structure du modèle de documents semi-structurés présenté en partie 4.

Dans ce travail, nous nous limiterons à une instance simplifiée du problème de restructuration : nous supposerons que ni l'arbre de structure du document, ni son contenu ne sont modifiés ; seules les étiquettes du document doivent être exprimées dans le schéma de médiation. À chaque nœud de d^{orig} correspond donc un unique nœud de d.

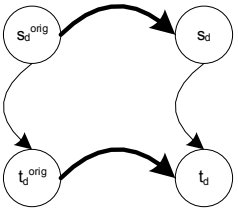

Figure 3 : Dépendances conditionnelles entre le document initial d^{orig} et le document final

d. Les flèches en gras indiquent les dépendances entre document reconstruit et document d'origine ; les flèches normales symbolisent les dépendances entre structures et contenu à l'intérieur d'un document.

5.1 Probabilité de contenu

$P_{contenu} = P(t_1,\ldots,t_{|d|} \mid t_1^{orig},\ldots,t_{|d|}^{orig}, s_1,\ldots,s_{|d|},\theta)$ correspond à la probabilité de restructuration du contenu. Nous supposerons que le contenu d'un nœud ne dépend que de l'étiquette de celui-ci. Comme nous avons fait l'hypothèse que le contenu des nœuds n'était pas modifié, on peut poser : $P(t_i \mid t_i^{orig}, s_i) = 0$ si $t_i \neq t_i^{orig}$. L'équation précédente peut donc se réécrire :

$$P_{contenu} = \prod_{i=1}^{|d|} P(t_i \mid s_i, \theta)$$

5.2 Probabilité de structure

La figure 4 décrit le réseau bayésien correspondant à la probabilité de structure. Nous considérons que les étiquettes des nœuds enfant d'un nœud n_i dépendent à la fois de l'étiquette de n_i et des étiquettes de ces nœuds dans le document original. Cette hypothèse permet de réécrire la probabilité de restructuration structurelle sous la forme :

$$P(s_1,\ldots,s_{|d|} \mid s_1^{orig},\ldots,s_{|d|}^{orig},\theta) = \prod_{i=1}^{|d|} P\left(childrentag(n_i) \mid s_i, childrentag(n_i^{orig}),\theta\right)$$

Comme nous ne possédons pas de données d'apprentissage exprimées à la fois dans le schéma de médiation et dans le schéma original, nous ne pouvons estimer la probabilité $P\left(childrentag(n_i) \mid s_i, childrentag(n_i^{orig}),\theta\right)$. C'est pourquoi, nous considérons la simplification suivante : $P\left(childrentag(n_i) \mid s_i, childrentag(n_i^{orig}),\theta\right) = P\left(childrentag(n_i) \mid s_i,\theta\right)$ qui nous permet d'utiliser les paramètres du modèle de documents de la partie 4.

5.3 Modèle de reconstruction

Au final, la structure optimale du document reconstruit s'obtient en résolvant :

$$d_{optim} = \arg\max_d P(d \mid d^{orig},\theta) = \arg\max_d \left(\prod_{i=1}^{|d|} P\left(childrentag(n_i) \mid s_i,\theta\right) \cdot \prod_{i=1}^{|d|} P(t_i \mid s_i,\theta) \right)$$

La manière optimale de résoudre cette équation s'inspire des méthodes de programmation dynamique utilisées lors de l'inférence des modèles de Markov cachés ou des grammaires stochastiques (algorithme de Viterbi). La complexité de cet algorithme est linéaire par rapport au nombre d'étiquettes possibles, au nombre de nœuds dans le document et au nombre de règles de dérivation apprises.

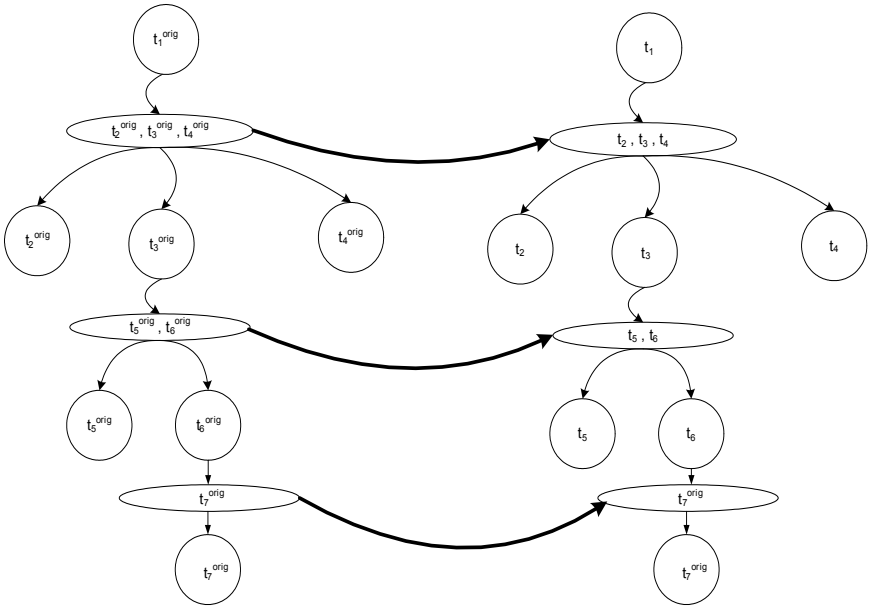

Figure 4 : Dépendances entre variables aléatoires décrivant la structure pour le modèle de restructuration. Les flèches en gras indiquent les dépendances entre document d'origine et document reconstruit et les flèches normales indiquent les dépendances entre éléments à l'intérieur d'un modèle de documents.

6 Expériences et résultats

Nous décrirons dans ce paragraphe les différentes expériences que nous avons effectuées pour tester les performances de notre algorithme. Nous allons dans un premier temps présenter les expériences menées, avant de discuter du problème de l'évaluation et de détailler les résultats obtenus. Nous étudierons plus particulièrement l'intérêt de considérer simultanément l'information de structure et l'information de contenu.

6.1 Protocole

Pour tester la validité de notre modèle, nous avons effectué plusieurs séries d'expériences à partir du corpus d'INEX. Ce corpus a été rassemblé dans le cadre d'une campagne d'évaluation des moteurs de recherche XML et regroupe près de 12 000 articles scientifiques au format XML, ce qui représente plus de 7 000 000 de noeuds. Les expériences ont été menées sur le corpus original d'INEX qui comporte 139 tags ainsi que sur une version d'INEX ne comportant que cinq tags. En effet, dans la mesure où un système de restructuration est développé pour répondre à un problème précis, nous pensons que l'évaluation de ce système doit tenir compte des spécificités du problème abordé. Dans le contexte de la recherche d'informations structurées, seul un petit nombre de tags apparaît dans les requêtes des utilisateurs ; nous cherchons donc à évaluer les performances de notre algorithme sur un corpus n'ayant qu'un nombre réduit de tags.

L'objectif de ces expériences consiste à évaluer l'intérêt de considérer à la fois l'information de structure et l'information de contenu. Pour cela nous avons effectué une reconstruction de chacun des corpus en prenant en compte successivement :

- que les informations de contenu (modèle *Contenu*) ;
- les informations de structure (modèle *Structure*) ;
- les informations de structure et de contenu (modèle *StructureContenu*).

À notre connaissance, aucune expérience de restructuration n'a été menée sur des corpus de documents textuels du type INEX. Afin de disposer d'éléments de comparaison, nous avons donc décidé de comparer nos résultats à ceux de modèles *naïfs*. Un modèle purement aléatoire ne présente que peu d'intérêt (le nombre de tags correctement reconstruits est inversement proportionnel au nombre de tags dans le vocabulaire). Nous avons donc utilisé certaines propriétés simples des documents pour déterminer le choix d'une étiquette. En particulier, nous avons remarqué que les documents du corpus INEX étaient composés d'un préambule, du corps de l'article et d'une bibliographie et que donc, la profondeur pouvait influer sur le choix de l'étiquette. Pour tester cette influence, nous avons utilisé comme élément de comparaison un modèle simple (modèle *Simple)* qui apprend la probabilité de trouver un tag sachant la profondeur du nœud et choisit les tags suivant cette loi de probabilité.

6.2 Évaluation et résultats

Pour le moment, nous utilisons deux mesures d'évaluation :

- le rappel, tel qu'il est défini en recherche d'informations. Il permet d'évaluer directement le nombre de nœuds qui ont été correctement étiquetés. C'est la mesure d'évaluation généralement utilisée dans les travaux de *schema matching*.

- le pourcentage de documents dont plus d'un certain pourcentage de noeuds a été correctement reconstruit. Cette mesure d'évaluation est plus adaptée à un système développé dans le cadre de la recherche d'informations structurées. En effet, il n'est pas nécessaire de reconstruire parfaitement un document pour pouvoir l'interroger.

Le tableau 1 détaille la valeur du rappel pour les différentes expériences. Notre modèle *StructureContenu* arrive à restructurer 86% des nœuds pour le corpus à 5 tags et 65% des nœuds pour le corpus à 139 tags.

Les expériences menées sur le corpus INEX montrent clairement que la prise en considération simultanée de l'information de structure et de l'information de contenu par notre algorithme améliore sensiblement les performances de reconstruction : dans le cas du corpus 139 tags, le modèle *StructureContenu* reconstruit correctement près de 65% des tags, alors que le modèle *Structure* et le modèle *Contenu* arrivent à reconstruire moins de 50% des nœuds. Pour le modèle *Contenu*, nous avons distingué le cas où l'on ignorait les nœuds vides de contenu qui ne peuvent être classifiés correctement et le cas où l'on prenait en considération tous les nœuds, les nœuds vides étant considérés comme mal classés.

Les performances sur le corpus à 139 tags sont encourageantes si on les compare aux résultats du modèle *Simple* : on arrive à retrouver l'étiquette de sept fois plus de nœuds avec notre modèle. Par contre, fort logiquement, la différence entre le modèle aléatoire et notre

	Contenu, avec tous les nœuds	Contenu sans les nœuds vides	Structure	Structure et contenu
Nombre de nœuds	≈ 5 000 000	≈ 3 500 000	≈ 5 000 000	≈ 5 000 000
5 tags	58%	81%	72,90%	86,50%
139 tags	27,80%	38,70%	49,70%	65,30%

Tableau 1 : Résultat de notre modèle sur le corpus d'INEX en utilisant le rappel comme mesure d'évaluation. Différents types d'informations sont considérés.

modèle est nettement moins significative pour le corpus à 5 tags. Le tableau 2 détaille la comparaison entre le modèle *Simple* et le modèle *StructureContenu*.

Modèle *StructureContenu*		Modèle *Simple*	
corpus à 139 tags	corpus à 5 tags	corpus à 139 tags	corpus à 5 tags
65,30%	86,50%	9,5%	79,30

Tableau 2 : Comparaison entre notre modèle et un modèle Simple*. La mesure d'évaluation utilisée est le rappel.*

La figure 5 présente les résultats réalisés avec la seconde mesure d'évaluation pour le corpus avec 139 tags. Le modèle *StructureContenu* est capable de retrouver les étiquettes de 90% des nœuds pour 40% des documents; avec le corpus à 5 tags, on arrive à reconstruire à 90% près des 90% des documents.

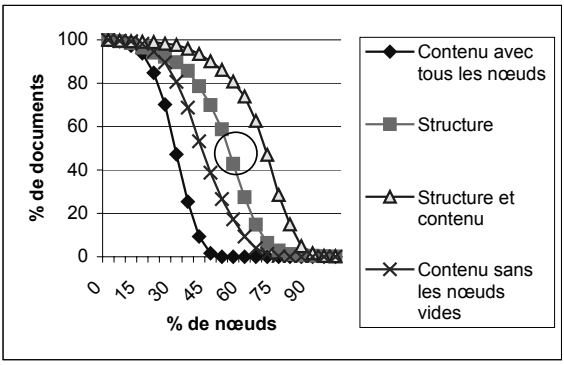

Figure 5 : Performances sur le corpus à 139 tags. Sur ces courbes, on peut lire (point entouré) que le modèle Structure permet de retrouver plus de 60% des étiquettes des nœuds pour 40% des documents du corpus de test

7 Conclusion

Nous avons présenté un modèle génératif de documents structurés. Nous avons ensuite proposé un formalisme statistique pour la tâche de restructuration et vu comment ce modèle pouvait être utilisé dans le cadre de la restructuration simple où l'on cherche à exprimer un document dans un schéma de médiation sans modifier sa structure arborescente ni son contenu. Nous avons enfin montré que notre modèle permet d'obtenir de bons résultats sur le

corpus INEX et que l'utilisation simultanée de l'information de structure et de contenu était indispensable pour obtenir de bonnes performances.

La tâche de restructuration, bien qu'abordée dans la communauté BD, est aujourd'hui assez nouvelle dans le cadre de la recherche d'informations et du traitement des documents textuels. Cette problématique est une problématique émergente dont les enjeux sont importants aussi bien pour le stockage efficace de documents semi-structurés que pour leur interrogation par des utilisateurs. Le modèle présenté dans cet article constitue aujourd'hui une des premières approches de l'utilisation des modèles d'apprentissage à la tâche de restructuration.

This work was supported in part by the IST Programme of the European Community, under the PASCAL Network of Excellence, IST-2002-506778. This publication only reflects the authors' views.

Références

Carrasco R.C. et Rico-Juan J.R. (2002), A similarity between probabilistic tree languages : application to XML document families, Pattern Recognition, 36(9):2197–2199, 2002.

Denoyer L. et Gallinari P. (2004), Bayesian Network Model for Semi-Structured Document Classification, Information Processing and Management, 2004.

Diligenti M., Frasconi P. et Gori M. (2003), Hidden Tree Markov Model for Document Image Classification, IEEE Transaction on Pattern Analysis and Machine Intelligence,

Do H.H., Melnik S. et Rahm E. (2002), Comparison of Schema Matching Evaluations, Proceedings of the GI-Workshop "Web and Database", Erfurt, 2002.

Doan A., Domingos P et HALEVY A. (2003), Learning to match the schemas of data sources : A multistrategy approach, Machine Learning, 50(3):279–301, 2003.

Fuhr N., Govert N., Kazai G. et Lalmas M. (2002), INEX : Initiative for the Evaluation of XML Retrieval, Proceedings ACM SIGIR 2002 Workshop on XML and Information Retrieval, 2002.

He B. et Chang K.C.C. (2003), Statistical Schema Matching accross Web Query Interfaces, Proceedings of ACM Sigmod'03, 2003

Kurgan L., Swiercz W. et Cios K.J. (2002), Semantic Mapping of XML Tags Using Inductive Machine Learning, Proceedings of the 2002 International Conference on Machine Learning and Applications (ICMLA'02), pp. 99-109, Las Vegas, NV, 2002

Li W.S. et Clifton C. (2000), Semint : a tool for identifying attribute correspondences in heterogeneous databases using neural networks, Data Knowledge Engineering, 33(1)

Madhavan J., Bernstein P., Chen K., Havely A. et Shenoy P. (2003), Corpus-based Schema Matching. Workshop on Information Integration on the Web, p59–66, 2003.

Melnik S., Molnia-Garcia H. et Rahm E. (2002), Similarity Flooding: A Versatile Graph Matching Algorithm, Proceedings of the International Conference on Data Engineering

Rahm E. et Bernstein P. (2001), A survey of approaches to automatic schema matching, VLDB Journal, 10(4) :334–350, 2001.

Summary

Querying heterogeneous XML document collections is an open problem. In order to query a document from a new source of information, we can either transform the query or transform the document into a mediation schema known by the user. In this article, we are interested in the second solution. We propose to study the task of fitting a document to a mediation schema. We first introduce a generative stochastic model of semi-structured documents. We then show how this model can be extended for the *document mapping* task. We then describe preliminary experiments performed on the INEX collection.