Une approche structurelle pour la reconnaissance de notices bibliographiques

A structural Approach for Library References Recognition

par Yannick CHENEVOY[1] et Abdel BELAID[2]

1. CRID, Université de Bourgogne, Bd Gabriel, F-21000 Dijon. e-mail : CHENEVOY@CRID.U-BOURGOGNE.FR
2. CRIN-CNRS, Campus Scientifique, B.P. 239, F-54506 Vandœuvre-lès-Nancy Cedex. e-mail : ABELAID@LORIA.FR

Résumé

Cet article présente un système de reconnaissance de la structure logique de notices bibliographiques en vue de la conversion rétrospective de catalogues de bibliothèques. Le système est guidé par un modèle de structures de la classe des notices, construit sur la base de spécifications détaillées par la bibliothèque. Le modèle fait intervenir aussi bien des connaissances sur la macro-structure des notices que sur la micro-structure de leur contenu. La reconnaissance de la structure d'une notice consiste à retrouver, à partir d'un flux OCR (Optical Character Recognition), sa structure logique spécifique, conformément aux descriptions du modèle. Le résultat est un flux structuré hiérarchiquement, présentant dans le format UNIMARC, les différents champs de la notice , accompagnés de leur score de confiance. Ce travail a été réalisé dans le cadre du projet européen LIB-MORE associant la société JOUVE et la Bibliothèque Royale de Belgique.

Mots clés : Notices bibliographiques, Reconnaissance de la structure, Analyse de documents, Format Unimarc, SGML.

Abstract

This paper presents a library references recognition system for retrospective conversion of catalogues. The system is guided by a structure model of a reference class, described by an attribute grammar. The analysis method is based on prediction and verification of segmentation hypotheses proposed by the model. The result, given in UNIMARC format, contains the different sub-fields of the reference with their confidence score. This method is enough general to be adapted on any document having a micro-structure. This method has been also used on other kind of documents such as author index and subjects.

Key words : *Library references, Structure recognition, Document analysis, Unimarc Format, SGML.*

1. Introduction

Depuis les années 70, les Bibliothèques ont informatisé leurs fonds documentaires permettant un accès plus facile à leurs ouvrages. Les progrès techniques réalisés par la lecture optique ont incité ces Bibliothèques à se pencher sur la conversion rétrospective de leurs catalogues papier antérieurs à cette période. Ces catalogues sont organisés en notices bibliographiques dont le contenu est une suite d'éléments "normalisés" entre les différentes Bibliothèques, conformément à un formalisme de catalogage commun. Ce formalisme a conduit l'ISBD[1] à faire émerger une norme internationale UNIMARC en 1976. Le travail présenté ici concerne la conversion de catalogues antérieurs à cette date et qui constituent la masse la plus importante de la bibliographie. L'étude s'est portée sur l'année 1973 de la Bibliothèque Royale de Belgique, représentative des difficultés susceptibles d'être rencontrées dans les notices de manière générale.

Sur de tels documents, l'accent a été mis sur la reconnaissance logique faisant intervenir des connaissances très fines sur le contenu (typographiques, lexicales, voire syntaxico-sémantiques) mais aussi sur les erreurs potentielles et les actions possibles de récupération. La difficulté principale consiste à identifier les différents champs et sous-champs, leurs relations de composition et de voisinage, conformément au formalisme standard. C'est un problème typique d'analyse de la structure logique de documents textuels. Dans la littérature, il n'existe pas de travaux similaires de traitement de contenu textuel à partir d'images, mais des rapprochements peuvent être trouvés dans d'autres domaines, notamment pour la reconnaissance des adresses postales [Ker91, Kre88], les textes de loi [Ing80], les adresses des correspondants

1. International Standard Book Documentation.

dans les lettres administratives [Den89, Bay91], etc. La particularité des notices, contrairement aux travaux cités précédemment, est de faire intervenir une variété importante de structures, d'exceptions, de champs facultatifs, répétitifs avec des imbrications multiples et complexes. De plus, la notice est riche en ponctuation qui peut aider, dans certains cas, le système à structurer les différents champs mais qui reste cependant insuffisante pour retrouver la structure profonde de la notice. Il faut interpréter le contenu pour identifier cette structure. Nous verrons plus loin dans le texte, quelques exemples illustrant ces particularités.

2. Structure des notices

2.1. Description générale

La bibliographie étudiée se présente sous la forme de catalogues papier mensuels. Chaque catalogue est divisé en deux parties : la première contient le corps de la bibliographie et la seconde, des indexs relatifs aux auteurs et aux sujets traités (titres, collections, rubriques en français et en néerlandais, etc.). Le corps de la bibliographie contient les notices imprimées sur plusieurs pages, à raison de deux colonnes par page. Chaque notice se présente sous forme d'un bloc de texte de quelques lignes, mais pouvant parfois dépasser la taille d'une colonne et créer des problèmes de débordement d'une colonne à l'autre.

Dans chaque notice, on peut parfaitement identifier une structure physique et une structure logique, comme le montre l'exemple de la figure 1.

	159.962	CDU
Liger-Belair (Gérard). Je suis fakir. ([Par] Gérard Liger-Belair). (*Verviers, Editions Gérard & C°, 1973), 32° carré, couv., ill.,* 158 *p.* (30 *fr.*).		Corps
(Marabout-flash, 352).		Collection
[Titre introductif : Souvenirs, révélations, conseils].		Note
B.D. 14.814 352	*73-2108*	Zref

Figure 1. – Exemple de notice bibliographique.

2.1.1. *La structure physique*

La structure physique est très simple; elle fait ressortir une décomposition verticale en plusieurs parties : le code "CDU"[2], toujours présent à la première ligne et cadré à droite, le corps de la notice représentant la partie la plus dense et la plus difficile à analyser, des notes (concernant le titre, l'auteur, etc.) ou des informations relatives aux collections dans certaines notices (ces deux dernières sont optionnelles), la référence identifiant la notice, à la dernière ligne de celle-ci. Ce numéro est incrémental et permet d'effectuer le lien avec les index d'auteurs et de sujets.

2.1.2. *La structure logique*

La structure logique est en revanche plus dense. Une *"zone vedette"*, représentant le premier auteur ou le début d'un titre se trouve toujours au début du corps. Pour le reste, la combinatoire des différentes possibilités est très importante. On trouve par exemple, suivant les notices, des *"auteurs principaux"* ou *"secondaires"* (introduits par des expressions caractéristiques) qui peuvent être des personnes physiques ou morales, des *"titres propres"*, *"parallèles"* (écrits dans des langages différents), ou *"par parties"*, des *"sous-titres"*, des *"éditeurs"* avec leur *"adresse"* et la *"date"* d'édition, un champ *"collation"* décrivant les caractéristiques de l'ouvrage (nombre de pages, format, documents d'accompagnement, etc.).

Cette structure est complexe car les champs, outre leurs imbrications étroites et leurs aspects changeants, sont optionnels et leur ordre d'apparition est variable. De plus, la ponctuation qui est le critère de séparation principal peut être soit absente, soit ambigüe. La figure 2 donne un état simplifié de la structure hiérarchique logique du corps. Elle décrit uniquement les composants supérieurs de la hiérarchie sans détailler leur contenu (attributs, affiliation lexicale, style typographique, etc.).

2.2. Le modèle

L'élaboration du modèle a été centrée sur la définition des éléments constituants, appelés objets. Tous les objets ont une structure unifiée autour de trois concepts décrivant le *type* de l'objet, ses *attributs* et les *actions* qui lui sont attribuées. La description formelle de la structure de ces objets est réalisée à l'aide d'une grammaire attribuée, écrite dans le formalisme E.B.N.F. (Extended Backus-Nauer Form).

2.2.1. *Le type*

Il précise la structure de l'objet en donnant la liste de ses composants et leur constructeur, suivant le formalisme développé dans [Che92]. Parmi ces constructeurs, on peut trouver la séquence : haut-bas (*seq_td*), gauche-droite (*seq_lr*) ou logique (*seq*); l'agrégat (*aggr*) ou le choix (*cho*). Chaque objet subordonné peut être suivi par un symbole d'occurrence indiquant s'il s'agit d'un objet optionnel(?), répétitif (+) ou optionnel répétitif (*).

L'exemple suivant indique que "Titre" est une séquence logique d'un titre principal "TitreP" et d'un titre parallèle "TitrePa", optionnel répétitif. Ces deux sous-objets sont séparés par un tiret long, "Tiretlong".

2. Classification Décimale Universelle.

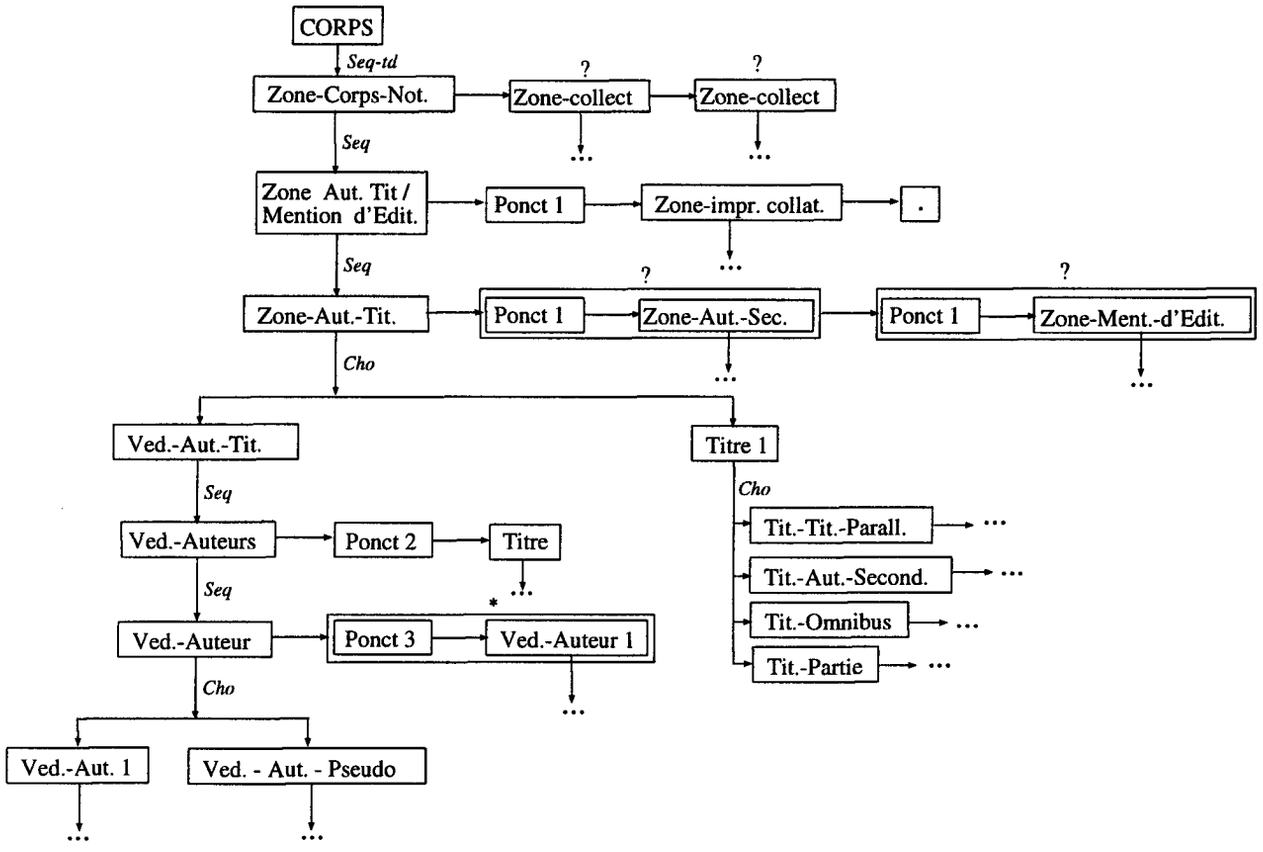


Figure 2. – Description sommaire des éléments de la structure logique des notices.

```
Titre::= Seq TitreP TitrePa*
Sep Tiretlong
Poids TitreP S TitrePa A
```

2.2.2. *Le poids*

L'utilisateur peut affecter à chacun de ces objets un poids symbolique pris dans un intervalle qu'il aura défini, par exemple [A,Z]. Ces poids sont ensuite convertis sur la base d'une formule de correspondance symbolique numérique. Ces poids interviendront dans le calcul des scores *a posteriori* et permettront de privilégier certaines solutions par rapport à d'autres. Dans l'exemple précédent, "TitrePa" a un poids (A) plus fort que celui de "TitreP" (S), car il est considéré comme plus important par l'utilisateur du fait de son caractère optionnel (et donc, de sa rareté). Il est à noter que importance n'est pas ici synonyme de probabilité mais de valeur de pondération servant comme critère de choix en cas d'ambiguïté dans la structuration.

2.2.3. *Les attributs*

Ils permettent de préciser le contenu de l'objet, concernant son aspect typographique, son contexte linguistique et sa sémantique. Chaque attribut est identifié par un nom suivi d'une liste de valeurs, éventuellement pondérées. Parmi ces attributs, on peut citer :*Nature* (string, line, word, char, etc.), *Mode* (capital, numeric, alphabetic, punctuation, etc.), *Style* (bold, italic, standard, etc.), *Lex* (pour l'affiliation lexicale), etc.

La pondération est donnée ici par des valeurs numériques car elle se veut être plus précise que la pondération symbolique (considérée comme une estimation). Contrairement aux poids symboliques, cette pondération intervient dans le calcul des scores *a priori* qui permettent soit de choisir les hypothèses à traiter en priorité, soit d'éliminer d'office les hypothèses trop éloignées des données analysées.

Par exemple, `Mode -num 5 -punct 3`, décrit le mode par l'absence de numérique ou de ponctuation ("-" pour exprimer la négation). La pondération associée au numérique est plus importante, car elle est plus discriminante que la ponctuation.

2.2.4. *Les actions*

Elles correspondent à des tâches spécifiques attachées aux objets, à la manière des "*méthodes*" utilisées dans les langages orientés objets. On distingue les actions de *pré-analyse* et les actions de *post-analyse*.

Il y a deux types d'actions de *pré-analyse* : les *stratégies locales* et les *pré-conditions*. Les premières correspondent à des actions localisées sur des objets dont la structure est très complexe, et dont l'analyse par la stratégie générale serait coûteuse. Ce sont des heuristiques adaptées à certaines situations locales. Celles-ci disposent de sources de connaissances spécifiques (lexiques, règles contextuelles, etc.) et fonctionnent indépendamment de la stratégie générale. Les secondes permettent d'étudier les conditions sous-lesquelles les objets seront analysés ou non par la stratégie générale, comme par exemple, vérifier la présence d'un style ou d'un mot particulier.

Les actions de *post-analyse* correspondent également à deux types de traitement en fin d'analyse d'un objet : finalisation et rattrapage. Les premières sont appliquées en cas de succès de l'analyse (résultat conforme au modèle) et permettent de finaliser un traitement sur un objet, par exemple en le restituant dans un format donné, ou en éliminant les hypothèses résiduelles de l'agenda (table d'hypothèses à gérer). Les secondes sont appliquées en cas d'échec de l'analyse (aucune hypothèse n'a été validée) pour ré-étudier un autre contexte d'analyse possible. Par exemple, en cas d'échec de la segmentation, dû au choix de seuils trop restrictifs; ceux-ci sont remis en cause pour un nouveau découpage. Un autre exemple concerne le rétablissement de la ponctuation manquante, par une prise en compte du contexte grammatical (utilisation des majuscules pour localiser les débuts de champs). Nous reviendrons plus loin sur la stratégie employée ainsi que sur les cas de réussite et d'échec.

2.2.5. *L'héritage*

On distingue essentiellement deux types d'héritage : la succession et la référence externe. Dans le premier cas, l'héritage des attributs entre un objet englobant et ses objets subordonnés est réalisée par filiation directe, conformément au type du constructeur. Dans le cas d'un "choix", tous les attributs sont transmis aux objets subordonnés. Pour les séquencements, seuls les attributs *typographiques* (style, mode, etc.) sont hérités vers les objets subordonnés. Les attributs décrivant la structure géométrique (position, largeur, hauteur, etc.) sont filtrés en fonction du type de structuration physique. Par exemple, une séquence de haut en bas ne transmettra que la largeur aux objets subordonnés; les attributs de positionnement dans une page, région ou ligne ne sont hérités que pour le premier objet subordonné d'une séquence.

L'héritage par référence externe est obtenu par le constructeur `Import`. Dans ce cas, les caractéristiques de l'objet subordonné (objet importé) sont héritées par le terme courant. Dans ce type d'héritage, il n'y a pas de restriction dans les attributs importés. De plus, le constructeur et les objets subordonnés de l'objet importé sont également transmis au terme courant. L'exemple suivant décrit l'objet `Notes` comme une répétition de l'objet `AutreNotes` dont les caractéristiques sont importées. Les attributs `Sep`, `Action` et `Style` de l'objet `Notes` remplacent les éventuels attributs du même nom hérités de `AutreNotes`.

```
Notes : := Import AutreNotes+
Sep Tiretlong B
Style standard
Action +VerifyStartWith(--,FALSE)
```

3. Fonctionnement du système

La figure 3 présente le schéma général du système. Le système est composé de quatre modules principaux : *prétraitement*, *filtrage*, *analyse structurelle* et *post-traitement*. Nous allons détailler dans la suite le fonctionnement des trois premiers modules. Le post-traitement a été réalisé en entreprise en mode semi-automatique et concerne tous les aspects de vérification-correction des résultats et leur restitution à la bibliothèque dans le format UNIMARC. Ce module tient compte des scores de reconnaissance structurelle afin d'isoler rapidement les zones douteuses ou ambigües.

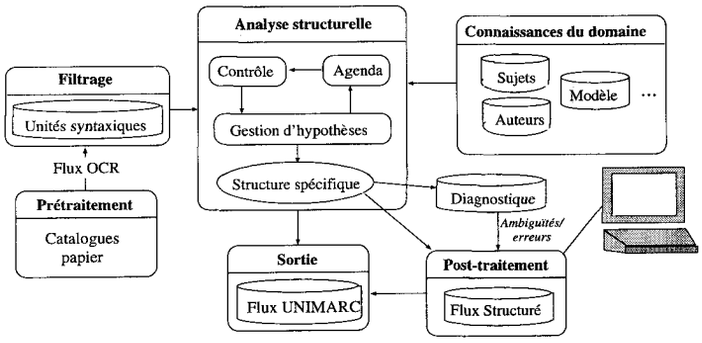


Figure 3. – Schéma général de l'analyse.

3.1. Prétraitement

Le *prétraitement*, réalisé en entreprise (JOUVE), traite de la numérisation des catalogues, de leur segmentation en notices et de la reconnaissance des caractères de celles-ci. A l'issue de ce prétraitement, chaque notice se présente sous la forme d'un flux de caractères ASCII balisé par des étiquettes de type SGML[3] [Iso86]. Ces balises indiquent le début et la fin d'une zone d'intérêt. Par exemple, <DOC ...>...</DOC> indiquent le début et la fin du document, <LIG ...>...</LIG> indiquent les limites d'une ligne, <LEX ...>...</LEX> indiquent l'affiliation lexicale et <I ...>...</I> marquent les limites d'une zone en italique.

La figure 4 donne le flux OCR de la notice de la figure 1. On peut remarquer dans cet exemple l'imprécision des résultats de l'OCR, par exemple pour le style du prénom à la quatrième ligne. L'analyse doit prendre en compte ce type d'erreur par des connaissances contextuelles plus générales.

3. Standard General Markup Language.

```
<DOC TY=N PROV=ENRLEX EG=OK NPN=2085 NDN=2114 IMA=users/brb/juin73/images>
<PAG NP=1 NOM=0008.ima> <COL XHG=63 YHG=1900 XBD=1027 YBD=2912> <NOT NON=2108 EN=OK>
<LIG XHG=870 YHG=2215 XBD=1000 YBD=2266 YBSL=2256 ST=t> <RED F=85.69>159.962</LIG>
<LIG XHG=149 YHG=2260 XBD=1001 YBD=2313 YBSL=2298 ST=p> <B>Liger-Belair</B> <I>(Gérard).</I>
<LEX L=GFR,GNL> <RED F=50.00>Je <LEX L=GFR>suis <LEX L=GGB,GFR>fakir. <LEX L=GGB,GFR,GNL>
<RED F=99.99>([Par] <RED F=99.97>Gé-</LIG> <LIG XHG=148 YHG=2304 XBD=1002 YBD=2356 YBSL=2342
ST=p>rard Liger-Belair). <RED F=89.99> <I>(Verviers, <LEX L=GGB> <RED F=100.00>Editions
<LEX L=GNL> <RED F=99.99>Gérard</I> <I>&amp;amp;</I> </LIG> <LIG XHG=151 YHG=2350 XBD=1000
YBD=2403 YBSL=2388 ST=p> C..., <RED F=83.33>1973), 32... <LEX L=GFR,GNL> <I>carré, <RED F=66.66>
couv.,ill.,</I><RED F=99.97>158 <RED F=25.00>p. (30 <I>fr.</I>).</LIG> <LIG XHG=149 YHG=2406
XBD=549 YBD=2457 YBSL=2443 ST=p> <LEX L=GGB,GFR,GNL> Marabout-flash, 352).</LIG> <LIG XHG=148
YHG=2447 XBD=1001 YBD=2499 YBSL=2485 ST=p> <LEX L=GFR> <RED F=99.99>[Titre <LEX L=GFR>
introductif : <LEX L=GGB,GFR> <RED F=89.99>Souvenirs, <LEX L=GFR>révélations, <LEX L=GGB,GFR,GNL>
con-</LIG> <LIG XHG=149 YHG=2494 XBD=245 YBD=2545 YBSL=2530 ST=p>seils].</LIG> <LIG XHG=148
YHG=2546 XBD=1002 YBD=2599 YBSL=2584 ST=t> <RED F=43.75>B.D. 14.814 <RED F=99.97>352<S N=15>
<I>73-2108</I> </LIG> </NOT> </DOC>
```

Figure 4. – Flux OCR de la notice de la figure 1.

3.2. Filtrage

Le filtrage permet, à partir du flux de données, d'isoler les unités syntaxiques (mots, fragments de mots, ponctuation, etc.) puis d'extraire leurs attributs (style, affiliation lexicale, mode, etc.). Chaque unité syntaxique est rangée dans une table d'analyse, accompagnée de ses attributs, de son ordre d'apparition et de sa longueur en nombre de caractères.

Le filtrage tient compte des règles de formattage et permet de retrouver les unités syntaxiques au-delà des découpages physiques (fin de ligne, espace, tiret, etc.). Il identifie les différents modes des unités syntaxiques (numérique, majuscule, etc.) et tient compte du balisage du flux d'entrée pour extraire tous ces attributs.

3.3. Analyse structurelle

Elle fonctionne suivant un processus de prédiction-vérification d'hypothèses de segmentation en champs. La segmentation est fondée sur l'analyse des commencements et fins possibles des champs de la notice courante en fonction des connaissances du modèle. Pour chaque champ analysé, les hypothèses correspondantes sont rangées dans un agenda, puis prises en compte de manière opportuniste pendant l'analyse (c'est le rôle du contrôle). Le résultat de l'analyse est une instance du modèle, appelée structure spécifique. Nous allons détailler dans la suite le fonctionnement de la stratégie générale.

3.3.1. Espace d'analyse

Les informations mémorisées dans la table des unités syntaxiques sont déterminantes pour la suite de l'analyse car elles permettent d'identifier les différents espaces d'analyse pour les hypothèses à vérifier. Ainsi, un espace d'analyse est identifié par un numéro de début et un numéro de fin d'unité syntaxique. Par la suite, l'analyse se réfèrera toujours à cette table pour retrouver le contenu propre aux hypothèses étudiées pour un espace de recherche donné.

3.3.2. Génération d'hypothèses

L'analyse des indices image est basée sur l'étude des initiales et des finales des objets subordonnés pour l'hypothèse courante dans l'espace d'analyse [Moh79].

Soit par exemple la règle : `O := Seq A B C`

Il s'agit de rechercher dans l'espace d'analyse correspondant à l'objet `O` toutes les initiales et finales des objets subordonnés `A`, `B` et `C`. A partir de ces indices, on construit toutes les sous-zones potentielles qui correspondent aux différentes combinaisons d'initiales et de finales. Dans cet exemple, pour chaque finale possible de `A`, il faut rechercher une initiale de `B` qui la suit immédiatement. Pour cette combinaison, il faut choisir une finale de `B` qui précède une initiale de `C`, etc.

Cette étude conduit à la création de nouvelles hypothèses sur des fragments de l'espace d'analyse appelés *fragment de contenu*. Cette méthode nous assure de converger vers les fragments terminaux en limitant le contexte de recherche et en focalisant l'analyse sur des fragments de plus en plus précis, jusqu'à la reconnaissance des fragments terminaux.

3.3.3. Choix des hypothèses

A chaque hypothèse est associé un score de confiance *a priori* basé sur la correspondance entre les attributs donnés par le modèle pour cette hypothèse et ceux trouvés dans son espace d'analyse. Ce score fixe un niveau de validité de l'hypothèse en cours, permettant de la retenir ou de la rejeter. Soit O une instance spécifique d'un objet caractérisé par les attributs at. Son espace recouvre le flux représenté par les unités syntaxiques us. L est la longueur en caractères de l'espace de recherche courant ($L(us)$ est la longueur de us). $C(at, us)$ est la fonction d'évaluation de l'attribut at sur l'unité us, comme par exemple, le score de reconnaissance de l'attribut "*italique*", par l'OCR, sur l'unité syntaxique "editeur" ou le score d'appartenance d'un mot à un lexique. $W(at)$ est le poids accordé à l'attribut at dans le modèle. Le score de confiance

a priori de O est calculé de la manière suivante :

$$S(O) = \frac{\sum_{at}\sum_{us} C(at,us) \times W(at) \times L(us)}{\sum_{at} W(at) \times L}$$

Cette formule fournit un score normalisé qui tient compte de la longueur des unités syntaxiques, de leurs attributs et de l'importance accordée à ses attributs par l'utilisateur.

Un premier filtrage permet de rejeter les hypothèses dont le score *a priori* est trop faible. S'il reste plusieurs hypothèses, on choisit la "meilleure" d'entre elles. Pour cela, plusieurs stratégies sont envisageables. On peut décider d'étudier, par exemple, l'hypothèse qui possède le meilleur score de confiance afin de repousser le plus tard possible les échecs. Cette stratégie se révèle inadaptée pour des documents où la structuration est très fine et très complexe car elle peut conduire à une explosion combinatoire du nombre d'hypothèses. Pour ce type de documents, une stratégie "en profondeur d'abord" combinée avec des heuristiques locales aux objets se révèle plus directe. On choisira donc les hypothèses à vérifier dans l'ordre où elles sont créées.

Chaque fois qu'une hypothèse doit être vérifiée, les éventuelles actions de préconditions pour l'objet générique correspondant sont exécutées. Si elles ne sont pas vérifiées, l'hypothèse est rejetée avec un score *a posteriori* nul. Si l'objet générique possède une action de pré-analyse de type heuristique, celle-ci est également exécutée. Dans le cas contraire, c'est à l'analyseur d'étudier les découpages possibles en fonction du constructeur.

3.3.4. *Validation et mise à jour des scores*

Chaque fois que l'analyse d'un fragment est terminée, un score *a posteriori* est associé à l'hypothèse correspondante. Ce score est propagé vers le haut de la structure spécifique (arbre). La propagation des scores intervient donc sur des fragments non terminaux de la structure spécifique. Si tous les objets subordonnés O_i d'un tel fragment F ont été analysés, son score *a posteriori*, SC_F, est également mis à jour suivant la formule :

$$SC_F = \sum_{i=1}^{n} SC_{O_i} \times P_{O_i}$$

où P_{O_i} représente le poids de l'objet subordonné O_i. L'analyse est terminée lorsqu'il n'y a plus d'hypothèse à étudier, c'est-à-dire lorsqu'on a pu calculer le score *a posteriori* de R, où R est la racine du modèle générique. Pour ce qui concerne les objets terminaux, SC_F est égal à $S(F)$ (le score *a priori* devient le score *a posteriori*).

Lors de la validation d'une hypothèse, si son score *a posteriori* est insuffisant, les éventuelles actions de rattrapage sont alors exécutées.

3.4. Stratégies locales

Nous montrons ici quelques exemples d'actions exécutées avant l'analyse générale. En fonction du status retourné par ces actions, elles peuvent jouer le rôle de préconditions, auquel cas, l'analyse se poursuit normalement, ou de stratégies locales, court-circuitant la stratégie générale. Lorsqu'une action joue le rôle de stratégie locale, elle a le contrôle de nouvelles hypothèses (décompositions possibles de l'objet courant) à émettre.

3.4.1. *Recherche d'auteurs*

Dans les notices bibliographiques étudiées, il convient, entre autres, d'identifier les auteurs secondaires de l'ouvrage. Contrairement aux auteurs principaux, les auteurs secondaires sont introduits par une expression particulière ("par", "introduit par", "illustration de", etc.). Il convient de reconnaître cette expression et de vérifier que ce qui suit correspond à un auteur. Le problème ici vient du fait que les auteurs ne sont pas forcément présentés sous le même format dans l'index et à l'intérieur des notices. De plus, la liste des expressions n'est pas exhaustive. Il convient donc d'appliquer une analyse syntaxique fine pour reconnaître ces auteurs secondaires, comme le montre l'exemple suivant :

```
ZATZME : := Seq ZAT ZME?
Sep Ponct1
Action +InitAuteurs(Expressions,IndexAuteurs,...)
```

Les paramètres `Expressions`, `IndexAuteurs`, etc. correspondent à une liste de lexiques utilisés par la stratégie locale `InitAuteurs`

3.4.2. *Recherche de style*

Dans le but de minimiser le nombre d'hypothèses émises en cours d'analyse, nous avons dû développer certaines heuristiques permettant de découper un champ en recherchant des caractéristiques typographiques. L'exemple ci-dessous montre une action qui découpe l'objet courant à la première ponctuation précédant le début d'une zone italique. Ce prédécoupage autorise en fait une analyse par parties de la notice et fait l'économie de nombreuses hypothèses qui, de toute façon, auraient échoué.

```
ZATX : := Seq ZATZME ZIC
Sep Ponct
Action +SplitField(italic,Ponct)
```

3.4.3. *Suppression d'hypothèses inutiles*

Certains objets à reconnaître sont facilement identifiables (par exemple une ville trouvée dans un atlas de villes). Il est intéressant dans ce cas de supprimer toutes les hypothèses en attente qui contiennent, dans un autre contexte, le même espace de recherche. L'action `KillAmbiguities` dans l'exemple ci-dessous, est activée si l'objet `MotEd1` est parfaitement reconnu. Elle parcourt

l'arborescence de la structure spécifique et supprime toutes les hypothèses en attente qui contiennent le même contenu que `MotEd1` et qui n'appartiennent pas à d'autres instances de `MotEd1`. Cette action doit être utilisée avec prudence car toute nouvelle hypothèse sur cet espace, autre qu'une instance de `MotEd1`, sera interdite.

```
MotEd1 ::= Terminal
Alex Edition //opl. tir. uitg. éd, etc.
Type mot
Action KillAmbiguities() RestituteField()
```

3.5. Restitution du flux de sortie

Lorsque l'analyse est terminée, il convient de parcourir l'arborescence de la structure spécifique afin de produire un flux structuré en sortie. Le parcours est effectué en profondeur d'abord. La structure est représentée par un balisage au sens SGML. Chaque balise de début de champ est paramétrée par le score *a posteriori* de l'analyse. Si pour un même objet spécifique, plusieurs hypothèses de décomposition ont réussi, il y a alors une ambiguïté dans la structure reconnue.

Certains types de documents sont intrinsèquement ambigüs. Par exemple, dans la séquence :

```
A := Import B+
Sep Virg
B := Terminal
```

A est une répétition d'objets B, séparés par une virgule. Mais rien n'indique qu'une virgule ne puisse pas apparaître à l'intérieur d'un B, d'où l'ambiguïté. De plus, si pour un objet spécifique rencontré lors du parcours, aucune hypothèse de décomposition n'a été vérifiée, il s'agit alors d'une erreur. Les erreurs peuvent provenir soit de l'OCR (caractères séparateurs non reconnus, style ou mode mal identifié, etc.). Elles peuvent également se produire si le document analysé correspond mal au modèle générique (champ obligatoire absent, styles non conformes, etc.). Un balisage spécial est réservé pour restituer chaque ambiguïté, de même que pour circonscrire les zones mal reconnues, ceci en vue d'une correction semi-automatique ultérieure.

Le schéma de la figure 5 résume les principaux modes de fonctionnement de l'analyse structurelle aussi bien en ce qui concerne la stratégie générale que les stratégies particulières (locales, de rattrapage, de restitution, etc.).

4. Résultats

Cette approche a été testée pour la conversion des catalogues techniques de la Bibliothèque Royale de Belgique [Bel94]. Ce projet a été financé par la commission européenne, dans le cadre du programme Bibliothèques, LIB-MORE.

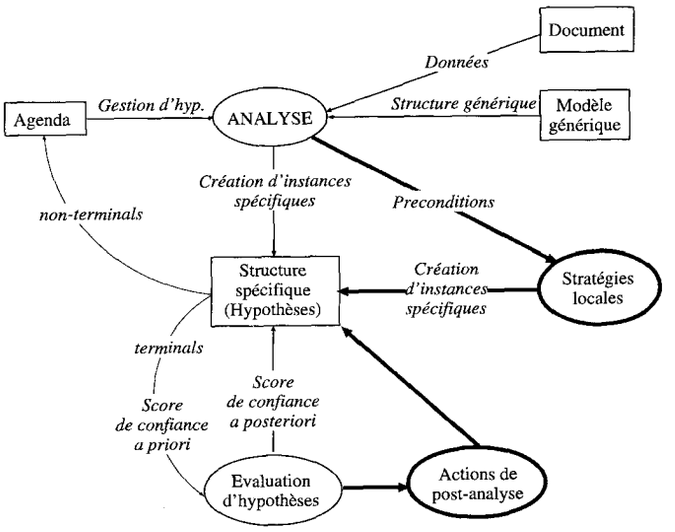


Figure 5. – Schéma de fonctionnement de l'analyse structurelle.

Un premier travail a consisté à modéliser les index d'auteurs et de sujets afin de reconnaître leur structure. Le flux produit en résultat a permis de constituer les lexiques d'auteurs et de sujets. D'autres lexiques ont dû être construits manuellement comme les atlas des villes, les abréviations, les expressions introduisant les auteurs secondaires, etc.

Le modèle générique des notices a ensuite été construit manuellement à partir des spécifications de la Bibliothèque Royale sur la base du mois de juin 1973. Le modèle ainsi construit comporte 226 objets et possède une profondeur de 24 niveaux depuis la racine jusqu'au terminal le plus éloigné. Les actions spécifiques au modèle sont au nombre de 36 pour les notices, 1 pour l'index des auteurs et 5 pour l'index des sujets. Elles représentent moins de 20 % du code développé.

Le système a été testé en laboratoire sur le catalogue du mois de juin 1973, qui comporte environ 400 notices. Le taux de reconnaissance atteint 85 % avant le post-traitement de récupération.

Le résultat de l'analyse de la notice de la figure 1 est donnée dans la figure 6. Les codes 675, 200, etc. correspondent aux balises UNIMARC des champs "CDU", "vedette", etc. Les autres balises commençant par $ correspondent aux sous-champs. Les valeurs après QSTR correspondent aux scores de confiance entre 0 et 10000. Le code 903 est annoncé en cas d'erreur, 902 en cas d'ambiguïté. On remarquera au passage les transformations dans le flux de sortie : nom de l'auteur, éditeur, adresse, date, prix, etc. Les indicatifs paramétrant les balises (I=...) permettent de préciser la nature des champs, comme par exemple une vedette auteur (I=0b) dans le champ UNIMARC 200.

La principale difficulté rencontrée fût la phase de conception du modèle. Les raisons sont nombreuses. Tout d'abord, le texte des notices rédigé en pré-ISBD est complexe car il contient un grand nombre d'éléments dans des successions variées et pouvant être optionnels. D'autre part, les séparateurs entre les différents champs sont ambigüs, comme par exemple la ponctuation qui

```
<675 I=bb QSTR=10000> <$a QSTR=10000>159.962</$a> </675>
<200 I=0b QSTR=9834>  <$f QSTR=9487>Gérard Liger-Belair</$f> <$a QSTR=10000>Je suisfakir</$a></200>
<700 I=b0 QSTR=10000> <$a QSTR=10000>Liger-Belair</$a> <$b QSTR=10000>Gérard</$b> </700>
<210 I=bb QSTR=9705>  <$a QSTR=10000>[Verviers]</$a> <$c QSTR=9519>[Editions Gérard & C...]</$c>
                      <$d QSTR=10000>[1973]</$d> </210>
<215 I=bb QSTR=9750>  <$d QSTR=7353>32... carré</$d> <$c QSTR=8601>couv., i11.</$c>
                      <$a QSTR=10000>158 p.</$a> </215>
<010 I=bb QSTR=10000> <$d QSTR=10000>30 BEF</$d> </010>
<225 I=2b QSTR=10000> <$a QSTR=10000>Marabout-flash</$a> <$v QSTR=10000>352</$v> </225>
<517 I=0i QSTR=10000> <$a QSTR=10000>Souvenirs, révélations, conseils</$a> </517>
<900 I=bb QSTR=9772>  <$a QSTR=10000>B.D. 14.814 352</$a> <$b QSTR=9285>73-2108</$b> </900>
```

Figure 6. – Analyse structurelle de la notice de la figure 1.

peut parfois jouer la double fonction de séparateur et de simple ponctuation. Enfin, s'agissant de faire correspondre ce format pré-ISBD au format de sortie UNIMARC, toutes les règles de restitution ne sont pas clairement identifiées surtout en ce qui concerne la zone des "titres" et les "mentions de responsabilité".

Dans de tels documents où la structure logique est complexe, de nombreux champs restent ambigüs si l'on se contente d'une description hiérarchique en terme de champs et de sous-champs. De plus, une telle description peut conduire à une explosion combinatoire du nombre d'hypothèses générées et rendre le coût de l'analyse inacceptable. Il est difficile, par exemple, de différencier un "titre omnibus" d'un "titre propre" suivi de "sous-titres". Même l'être humain s'y perd lorsqu'il s'agit d'une langue différente de la sienne. Des sources de connaissances supplémentaires sont nécessaires, ici, pour une reconnaissance optimale.

Nous avons inclus dans le modèle des stratégies locales permettant de limiter le champ d'analyse. Ces heuristiques ont permis d'atteindre un temps d'analyse de l'ordre de quelques secondes par notice.

C'est ici que nous atteignons les limites de ce système. Les stratégies locales ont été développées sur la base d'un échantillon de notices (mois de juin 73). De ce fait, elles ne couvrent pas l'ensemble des possibilités susceptibles d'être rencontrées, conduisant inévitablement à des erreurs ou à des ambiguïtés. L'être humain est toujours nécessaire pour la reconnaissance des notices résiduelles car il dispose encore actuellement de connaissances linguistiques et sémantiques supérieures à celles de la machine. Nous pensons que cette différence représente la prochaine frontière à franchir pour la reconnaissance optimale de documents à structure logique dense et complexe.

Un prototype industriel, issu de ce système, a été mis au point par la société JOUVE. Le modèle a été étendu à tous les catalogues de l'année 1973. Sur 4548 notices traitées, une intervention manuelle a été nécessaire pour 33% d'entre elles, soit pour lever des ambiguïtés, soit pour restructurer complètement la notice. 5.4% des notices ont dû être retournées à la bibliothèque, à cause de leur non conformité avec les spécifications fournies. Les principaux problèmes rencontrés tant sur le mois de test que sur l'année complète proviennent des "titres et mentions de responsabilité", de la "zone d'adresse" et de la "zone de collection". Ces erreurs ne sont pas issues uniquement de la structuration, mais également des défaillances de l'OCR qui, malgré son renforcement par association de plusieurs systèmes les plus performants du commerce, produit des erreurs sur la ponctuation, le parenthésage, les tirets et le style qui sont les indices de base pour la structuration automatique.

5. Conclusion

La représentation du modèle, notamment la possibilité d'inclure des actions spécifiques aux objets a permis d'assouplir le système de reconnaissance, le rendant plus indépendant du document analysé. Les méthodes spécialisées du domaine, comme les mécanismes de rattrapage ou le choix de la représentation des résultats se trouvent intégrées au modèle, laissant le système d'analyse se concentrer sur le choix des objets à analyser (calcul des scores, gestion des hypothèses, etc.), bref, sur la stratégie proprement dite. De plus, ces actions peuvent correspondre à des stratégies locales ou des heuristiques qui permettent d'éviter d'analyser l'objet par la stratégie générale si celui-ci est vraiment particulier. L'utilisation du système pour la reconnaissance des index des auteurs et des index des sujets a permis d'enrichir les connaissances contextuelles utiles pour la reconnaissance des notices. Cet apport démontre que nous disposons ici d'un système efficace et évolutif capable de réutiliser ses résultats comme sources nouvelles de connaissances pour d'autres types de documents.

La difficulté majeure rencontrée dans ce projet s'est révélée être la modélisation du document. En effet, le degré de finesse des réglages, notamment en ce qui concerne les poids accordés aux objets et aux attributs rend difficile la tâche du concepteur de modèles lorsque ceux-ci sont flous ou complexes. Une perspective intéressante pour la suite de ces recherches concerne un système d'aide semi-automatique à la conception de modèles. Un tel système doit pouvoir fournir au concepteur des indices concernant les éventuelles ambiguïtés contenues dans le modèle. Il doit également disposer d'outils de mesures adaptés afin d'alléger la tâche du concepteur. Un système d'apprentissage de modèles de macro-structures a été développé dans notre équipe [Aki95]. Il montre tout l'intérêt d'étendre l'apprentissage à la micro-structure et donc à la structure logique.

Les résultats obtenus sont encourageants. Ils ne pourront cependant être améliorés que par l'apport de nouvelles sources de connaissances, notamment linguistiques et sémantiques. Une description hiérarchique des objets modélisés sera toujours restrictive dans le cas de certains documents dont la structure logique est complexe.

BIBLIOGRAPHIE

[Aki95] T.O. Akindele. Vers un système de construction automatique de modèles génériques de structures de documents. Doctorat de l'Université de Nancy I, janvier 1995.

[Bay91] T. Bayer, J. Franke, U. Kressel, E. Mandler, M. Oberlander, and J. Schurmann. *Towards the Understanding of Printed Documents*, pages 3–35. In Structured Document Image Analysis, H. Baird, H. Bunke and K. Yamamoto (eds.), Springer-Verlag, 1991.

[Bel94] A. Belaïd, Y. Chenevoy, and J. C. Anigbogu. Qualitative Analysis of Low-Level Logical Structures. In *EP'94*, volume 6, pages 435–446, Darmstadt, Germany, Apr. 1994.

[Che92] Y. Chenevoy. Reconnaissance structurelle de documents imprimés : études et réalisations. thèse de doctorat de l'INPL, décembre 1992.

[Den89] A. Dengel and G. Barth. ANASTASYL : A Hybrid Knowledge-based System for Document Layout Analysis. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*, pages 1249–1254, Detroit MI., 08 1989.

[Ing90] R. Ingold. Structures de documents et lecture optique : une nouvelle approche. Thèse de doctorat, Ecole Polytechnique Fédérale de Laussane, 1990.

[iso86] Internationnal Standard Organization. *Information Processing, Text and Office Systems, Standard Generalized Markup Language (SGML)*, 1986.

[Ker91] S. M. Kerpedjiev. Automatic Extraction of Information Structures from Documents. In *International Conference on Document Analysis and Recognition (ICDAR) St-Malo, France*, volume 2, 1991.

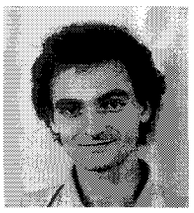
[Kre88] J. Kreich, A. Luhn, and G. Maderlechner. Knowledge-based Interpretation of Scanned Business Letters. In *Proceedings of IAPR Workshop on Computer Vision*, pages 417–420, Tokyo, 1988.

[Moh79] R. Mohr. Descriptions structurées et analyse de formes complexes - Application à la reconnaissance de dessins. Thèse d'état, Centre de Recherche en Informatique de Nancy, 1979.

Manuscrit reçu le 1er février 1995.

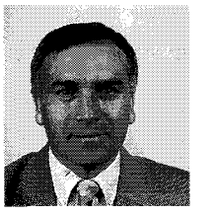
LES AUTEURS

Yannick CHENEVOY



Yannick Chenevoy a obtenu une thèse de Doctorat en Informatique de l'Institut National Polytechnique de Lorraine (INPL) en 1992. Il a débuté ses recherches au Centre de Recherche en Informatique de Nancy (CRIN) sous la direction de A. Belaïd, sur les problèmes de l'analyse et de la reconnaissance structurelle des documents imprimés. Il est maintenant Maître de Conférence à l'Université de Bourgogne et il continue ses recherches sur le même thème au Centre de Recherche en Informatique de Dijon (CRID).

Abdel BELAID



Abdel Belaïd a fait ses études universitaires à l'Université Henri Poincaré (UHP) de Nancy et sa recherche au Centre de Recherche en Informatique de Nancy (CRIN) où il a obtenu sa thèse de 3e cycle en 1979 et sa thèse d'état en 1987. Après quelques années d'enseignement en tant qu'assistant, puis maître assistant à l'UHP, il est Chargé de recherche au CNRS depuis 1984. Ses domaines de recherche sont le traitement d'images et la reconnaissance des formes. Il est co-auteur d'un livre intitulé : Reconnaissance des formes : méthodes et applications. Il anime un groupe de recherche READ autour de plusieurs projets sur l'analyse de documents et la reconnaissance de l'écriture. Il est membre de SPECIF et de l'Association Française pour la Cybernétique Economique et Technique (AFCET).