Inférence semi-automatique et interactive de règles avec ou sans vérité terrain pour la reconnaissance de structure de documents

Cérès Carton

Thèse

THÈSE INSA Rennes
sous le sceau de l'Université Européenne de Bretagne
pour obtenir le grade de
DOCTEUR DE L'INSA DE RENNES
Spécialité : Informatique

présentée par
Cérès Carton
ÉCOLE DOCTORALE : MATISSE
LABORATOIRE : IRISA – UMR6074

Inférence semi-automatique et interactive de règles avec ou sans vérité terrain pour la reconnaissance de structure de documents

Thèse à soutenir le 23 mars 2016
devant le jury composé de :

Rapporteur Josep Lladós
Full Professor à l'UAB (Espagne) / *Rapporteur*

Rapporteur Thierry Paquet
Professeur à l'université de Rouen / *Rapporteur*

Examinateur Christopher Kermorvant
Président de Teklia SAS / *Examinateur*

Examinateur Jean-Marc Ogier
Professeur à l'université de La Rochelle / *Examinateur*

Bertrand Coüasnon
Maître de conférence (HDR) à l'INSA de Rennes / *Directeur de thèse*

Aurélie Lemaitre
Maître de conférence à l'Université de Rennes 2 / *Co-encadrante*

Remerciements

Table des matières

Introduction

Le volume de documents manuscrits, imprimés et numériques produit chaque jour ne cesse d'augmenter. Un intérêt de plus en plus fort est montré pour l'exploitation dématérialisée des documents. Cela peut être dans un contexte industriel ou personnel pour la dématérialisation de chèques, factures, fiches de paie. Il est alors possible de réaliser automatiquement des transactions. La dématérialisation des courriers d'entreprise permet par exemple d'identifier les catégories des documents et d'en extraire des index (numéros de clients, dates, objet, adresse de l'expéditeur, etc.).

La dématérialisation est également utilisée dans le contexte de documents d'archives, permettant un accès facilité pour le public et une conservation facilité des documents, la consultation des documents étant alors non destructive. Cela rend possible un accès simultané et à distance des documents ainsi qu'une exploitation riche des contenus : navigation facilitée, recherche possible d'éléments et mise en correspondance entre certaines parties (index et références par exemple).

Si la seule numérisation permet de conserver les documents, elle ne permet pas leur exploitation efficace. Il faut en effet pouvoir les interpréter. Cette interprétation doit se faire de la manière la plus automatique possible au vue des volumes de documents à traiter et de leur diversité.

Contexte de la thèse

Nous travaillons ici sur une partie spécifique de la reconnaissance du document, la reconnaissance de la structure. La reconnaissance de la structure est une étape préalable nécessaire pour la reconnaissance de contenus. Nous cherchons à extraire la structure physique du document, à étiqueter les blocs extraits puis à construire l'organisation hiérarchique du document et à établir les liens entre les différents éléments. La reconnaissance de la structure nécessite la modélisation ou l'apprentissage de connaissances complexes et hiérarchiques. Pour chaque nouveau type de documents à reconnaitre, une nouvelle modélisation des connaissances est à réaliser.

Les méthodes actuelles ont du mal à résoudre conjointement les deux problématiques de modélisation de connaissances complexes et d'adaptation facile et rapide à un nouveau type de documents. Nous distinguons deux familles principales de méthodes parmi les approches existantes : les méthodes statistiques et les méthodes syntaxiques. Les méthodes statistiques se basent sur un apprentissage automatique d'un modèle statistique à partir d'un échantillon de documents étiquetés. Les méthodes syntaxiques

proposent quant à elles une explicitation des connaissances a priori sur les documents par un expert dans un modèle à base de règles ou de grammaires.

Les méthodes statistiques sont les plus à mêmes de s'adapter facilement à un nouveau type de documents puisqu'elles reposent sur un apprentissage automatique du modèle statistique. Cependant, leur capacité de modélisation des connaissances est limitée et les cas rares sont difficiles à gérer. Les structures modélisées sont simples et les relations spatiales sont limitées à des contextes locaux, ce qui n'est pas adapté aux documents de plus en plus complexes à reconnaître en analyse de la structure de documents. De plus, il n'est pas possible d'utiliser ces méthodes sans vérité terrain annotée sur les documents ; or, la construction d'une vérité terrain annotée est coûteuse.

Les méthodes syntaxiques sont les mieux adaptées à modéliser des connaissances complexes. Cependant, dans le cadre de ces méthodes, c'est un apprentissage manuel des connaissances qui est effectué pour chaque nouveau type de documents. Cet apprentissage manuel est alors long et coûteux, réduisant de ce fait l'adaptabilité à un nouveau type de documents.

Solution proposée

Nous proposons une nouvelle méthode, la méthode Eyes Wide Open (EWO), introduisant une inférence semi-automatique et interactive de règles pour la description progressive de systèmes syntaxiques. Le choix de nous baser sur le formalisme des systèmes syntaxiques permet à notre approche d'avoir un grand pouvoir d'expression et de pouvoir modéliser la connaissance associé à des documents complexes. L'introduction d'une étape d'apprentissage permet d'accélérer l'adaptation du système à un nouveau type de documents, comme c'est le cas dans le cadre des approches statistiques. De plus, cela nous donne une vue exhaustive sur les documents à traiter, ce qui n'est pas possible avec les systèmes syntaxiques actuels. En effet, les approches syntaxiques se basent sur un apprentissage manuel des règles sur un échantillon restreint de documents.

La méthode EWO propose un mécanisme d'apprentissage progressif de la description grammaticale des documents reposant sur trois étapes :

1. Acquisition des données utiles, avec une vérité terrain disponible mais également sans aucune vérité terrain ;
2. Inférence des règles avec un apprentissage à la fois de la structure logique et de la structure physique des documents
3. Assistance à l'intégration dans la description grammaticale des documents.

La méthode EWO donne ainsi la possibilité à l'utilisateur d'inférer des règles, par étapes successives, à un coût minimal sur des données hétérogènes, bidimensionnelles et en grande quantité. L'une des originalités de notre approche est de permettre l'inférence des règles lorsqu'il n'y a pas de vérité terrain annotée disponible sur les documents.

La méthode EWO se base sur des méthodes de clustering combinées à une interaction avec l'utilisateur. Le clustering réalisé sur de grands volumes de documents permet d'extraire automatiquement des redondances, des structures dans les données. L'extraction automatique des redondances donne une vision à la fois synthétique et exhaustive

des données, ce qui n'est habituellement pas possible pour les méthodes syntaxiques. Les structures détectées grâce au clustering sont ensuite manuellement validées par un opérateur humain et les clusters sont également étiquetés par l'opérateur. L'opérateur humain apporte ainsi du sens aux données automatiquement détectées.

Contenu du manuscrit

La première partie présente l'état de l'art.

Le chapitre 1 dresse un panorama des méthodes existantes pour la reconnaissance de la structure de documents. Ce chapitre met en évidence les points forts et les points faibles des deux grandes familles de méthodes que sont les méthodes statistiques et les méthodes syntaxiques. Nous mettons également en avant les limites rencontrées par les méthodes actuelles d'inférence grammaticale.

Le chapitre 2 s'intéresse aux méthodes d'apprentissage non supervisées et en particulier aux méthodes de clustering.

Cette première partie permet donc de situer nos travaux par rapport aux approches de reconnaissance de la structure de documents de la littérature. Elle met en évidence les apports d'une méthode d'inférence de règles semi-automatique et interactive pour l'extraction de connaissances.

La deuxième partie présente la méthode d'inférence semi-automatique et interactive de règles que nous avons proposée.

Le chapitre 3 présente la philosophie de la méthode EWO. Il décrit les propriétés et les caractéristiques de notre méthode. Il permet de présenter comment notre méthode s'introduit dans l'approche manuelle habituellement utilisée pour la description d'un système à base de règles.

Le chapitre 4 présente ensuite le cœur de la méthode d'inférence des règles. La méthode d'acquisition des données dans les cas avec et sans vérité terrain est décrite. L'inférence de la structure logique et physique est ensuite détaillée. Enfin l'intégration de ces connaissances dans la description grammaticale complète des documents est présentée.

Le chapitre 5 décrit la méthode de clustering utilisée, l'Evidence Accumulation Clustering, dans la méthode EWO pour l'acquisition des données ainsi que pour la description de la structure logique et physique des documents.

Le chapitre 6 détaille l'interaction entre l'utilisateur et la méthode afin d'apporter de la sémantique aux données détectées automatiquement.

Dans cette deuxième partie, nous décrivons le fonctionnement complet de la méthode EWO. Nous mettons ainsi en évidence sa généricité et sa facilité d'utilisation pour l'utilisateur. Les mécanismes permettant son utilisation avec et sans vérité terrain sont également décrits.

La troisième partie valide chacun des éléments de la méthode EWO par des évaluations par partie puis globales de la méthode. L'ensemble de ces expérimenta-

tions valide le gain en temps et en qualité des descriptions grammaticales produites en utilisant la méthode EWO.

Le chapitre 7 présente le contexte d'évaluation de la méthode EWO. Nous nous basons sur une méthode syntaxique existante, la méthode DMOS-P (Description et MOdification de la Segmentation avec vision Perceptive). Nous utilisons la méthode EWO dans le processus d'écriture des différentes descriptions grammaticales.

Le chapitre 8 présente une évaluation sur la reconnaissance de la structure de courriers manuscrits en français du corpus RIMES. L'évaluation est faite en deux parties : une évaluation de l'inférence des opérateurs de position seuls puis une évaluation de l'inférence d'une description grammaticale complète dans le cas où une vérité terrain sur les documents est disponible. Le système obtenu avec la méthode EWO est alors comparé à des systèmes purement syntaxiques et des systèmes purement statistiques. Les performances obtenues sont comparables à celles obtenues par les meilleurs systèmes purement statistiques et purement syntaxiques, tout en simplifiant fortement la modélisation des connaissances.

Le chapitre 9 évalue notre méthode dans le cadre particulier des documents de la compétition MAURDOR, où une vérité terrain contenant la segmentation en blocs est connue. Notre système est comparé à un système utilisant à la fois un classifieur et des règles décrites à la main. Nous montrons alors l'apport de la vue exhaustive sur les données permise par la méthode EWO. Les règles décrites avec la méthode EWO obtiennent de meilleurs résultats que celles décrites manuellement.

Le chapitre 10 permet de valider notre méthode dans le cadre de documents pour lesquels nous n'avons pas de vérité terrain annotée disponible. Nous avons pour cela utilisé le corpus d'actes de mariages mexicains de la compétition FamilySearch HIP2013. Une pseudo vérité terrain est produite à moindre coût grâce à la méthode EWO en étudiant les redondances dans un grand volume de documents. En effet, l'annotation manuelle du même volume de documents aurait nécessité la réalisation de 200 fois plus d'actions de la part d'utilisateurs humains.

Première partie

État de l'art

Introduction

Cette première partie est consacrée à l'étude de la bibliographie. Notre domaine d'étude regroupe deux vastes domaines de recherche : l'analyse de la structure de documents et l'apprentissage non supervisé.

Dans le premier chapitre, nous présentons de manière générale les approches utilisées pour l'analyse de la structure de documents que nous regroupons en deux familles principales : les méthodes statistiques et les méthodes syntaxiques. Nous soulignons alors l'intérêt des méthodes syntaxiques, qui présentent un fort pouvoir d'expression notamment pour les structures hiérarchiques. Nous mettons en avant la nécessité d'une étape d'apprentissage automatique dans l'apprentissage du système pour le rendre adaptable facilement à un nouveau type de documents. Nous présentons ensuite les solutions existantes pour l'inférence des méthodes syntaxiques et soulignons les limites actuelles.

Dans le deuxième chapitre, nous présentons les solutions existantes d'apprentissage non supervisé et plus particulièrement de clustering. Cette présentation nous permet de justifier l'intérêt de la méthode de clustering choisie, l'Evidence Accumulation Clustering [FJ02].

Grâce à cette étude bibliographique, nous montrons la nécessité d'introduire une phase d'apprentissage dans la construction des systèmes de reconnaissance de la structure de documents. En montrant les limites de l'inférence grammaticale, nous soulignons la nécessité d'introduire une interaction avec un utilisateur humain qui valide les structures détectées automatiquement avec des méthodes de clustering.

Chapitre 1

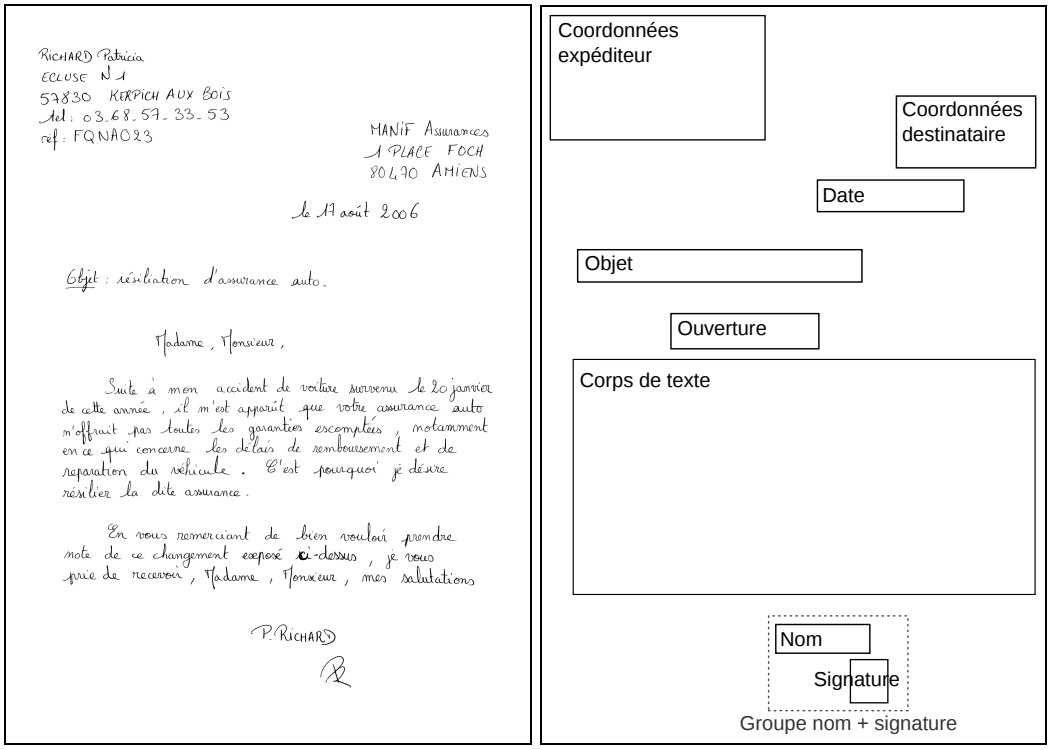
État de l'art des systèmes de reconnaissance de la structure de documents

Dans ces travaux, nous nous intéressons à la reconnaissance de documents et plus particulièrement à l'analyse de la structure logique des documents. Le but principal est de convertir des images de documents en vue de la modification, l'archivage, la recherche, la réutilisation et la transmission de l'information que ces images contiennent. La reconnaissance de la structure logique du document permet de générer une représentation de haut niveau sous la forme d'un document structuré. Il s'agit alors d'extraire la structure physique du document, d'étiqueter les blocs extraits puis de construire l'organisation hiérarchique du document et établir des liens entre les éléments (ordre logique de lecture, liaison entre les illustrations et le texte, imbrication des éléments logiques, etc.). Un exemple de document et de la structure logique associée est présenté dans la figure 1.1.

Les documents à analyser peuvent être de types variés (courriers d'entreprises, factures, articles, etc.) et être plus ou moins structurés. Ainsi, nous pouvons distinguer les documents très structurés comme les tableaux et les formulaires, les documents semi-structurés comme les courriers professionnels, et les documents peu structurés. Pour ces derniers, il n'y a pas de modèle physique qui les caractérisent ce qui complexifie la tâche de détection des objets logiques.

Depuis le début des années 90, la problématique de la reconnaissance de la structure de documents a été largement étudiée. Si au départ les systèmes développés étaient en général dédiés à un domaine spécifique (courriers d'entreprises ou facture par exemple) et à des documents simples, les documents traités sont désormais de plus en complexes [DS14]. De plus, les systèmes développés tendent également à ne plus être dédiés à un domaine spécifique mais au contraire à être utilisables pour des documents variés.

Nous distinguons deux types majeurs d'approches pour les méthodes de reconnaissance de la structure de documents : les méthodes *statistiques* et les méthodes *syntaxiques*. Nous détaillons dans ce chapitre les avantages et les inconvénients de chacune



(a) Image de courrier manuscrit (b) Structure logique du courrier manuscrit

FIG. 1.1 – Exemple de document et de la structure logique associée

de ces approches.

1.1 Méthodes statistiques

Les méthodes statistiques pour la reconnaissance de la structure de documents se basent sur un apprentissage automatique d'un modèle statistique à partir d'un échantillon d'apprentissage étiqueté. L'étiquetage selon les différentes classes possibles est alors appris à partir de cet échantillon. Cet étiquetage peut être appris à différents niveaux, par exemple au niveau pixel (Lemaitre [LGGP07]) ou sur plusieurs niveaux successifs pour arriver au niveau bloc (Montreuil [MNGH10], Chaudhury [CJDR09]). Ces approches ne reposent pas sur des règles ou des heuristiques définies par un expert.

Cette méthodologie d'apprentissage automatique à partir d'un échantillon étiqueté est commune aux nombreuses méthodes statistiques proposées dans la littérature [CFGS03, MBD11, LGGP07, MNGH10, BR08, FSJ+01]. Ces approches se basent sur des formalismes variés tels que les champs aléatoires de Markov ou les CRF (Conditional Random Fields). Quelques contributions s'appuient sur des algorithmes classiques d'apprentissage supervisé. Par exemple, Rangoni utilise un réseau de neurones dynamique perceptif [BR08].

Nous allons détailler ces différentes approches.

1.1.1 Méthodes graphiques probabilistes

Les méthodes graphiques probabilistes consistent en l'étiquetage d'une zone (dont la taille peut varier, le pixel par exemple) en fonction de son voisinage local. Le fait que ces méthodes tiennent compte des informations contextuelles d'un élément explique leur utilisation dans le cadre de la reconnaissance de la structure de documents. Les caractéristiques utilisées peuvent être variées :

- police, taille, couleur des caractères ;
- localisation dans l'image ;
- présence de mot-clé ;
- etc.

Ces méthodes peuvent être basées sur des réseaux bayésiens, des champs aléatoires de Markov ou des champs markoviens conditionnels que nous présentons dans cette section.

1.1.1.1 Champs aléatoires de Markov

Lors d'une modélisation à l'aide des champs aléatoires de Markov, l'image du document est décomposée en cellule. À chacune des cellules est associé un certains nombre de caractéristiques ainsi qu'une étiquette correspondant à la classe de la cellule. Le but est alors d'apprendre quelle est l'étiquette la plus probable d'une cellule en fonction de ses caractéristiques connues et du contexte, c'est-à-dire de son voisinage.

Lemaitre [LGGP07] propose une méthode basée sur les champs aléatoires de Markov avec un étiquetage au niveau pixel pour l'analyse de courrier manuscrit en français. Les caractéristiques de texture et de position sont utilisées. Des étiquettes logiques différentes peuvent coexister dans la même région alors que ce n'est théoriquement pas possible. La cohérence de l'étiquetage au niveau bloc n'est ainsi pas assurée. Les auteurs soulignent que les erreurs dans leur approche proviennent d'un manque d'information globale dans l'analyse (au niveau bloc par exemple). Ils indiquent également qu'un plus grand nombre de données d'apprentissage permettrait d'obtenir de meilleur résultat.

1.1.1.2 Réseaux bayésiens

Le Bourgeois [FSJ+01] propose une méthode combinant réseaux bayésiens et *probabilistic relaxation*. Le réseau bayésien permet d'étiqueter des blocs de texte en se basant sur des caractéristiques géométriques et typographiques. La *probabilistic relaxation* est ensuite utilisée pour améliorer les résultats obtenus par les réseaux bayésiens. Cette méthode itérative tient compte des relations spatiales et hiérarchiques, ce qui apparait comme essentiel pour reconnaître la structure de documents. De plus, cette méthode permet de corriger les erreurs d'un étiquetage préliminaire. Cependant, seuls des résultats préliminaires sont présentés pour cette méthode.

1.1.1.3 Champs aléatoires conditionnels

Les Champs Aléatoires Conditionnels (Conditional Random Fields, CRF ou « champs markoviens conditionnels ») ont été proposés pour l'analyse de données séquentielles. Plusieurs méthodes de la littérature les ont appliqués dans le domaine de l'analyse de documents. Les CRF permettent de dépasser les limites des Champs de Markov.

Montreuil [MNGH10] présente une méthode basée sur la combinaison hiérarchique de CRF. La segmentation de la structure physique est faite à l'aide de trois niveaux de CRF suivis d'une segmentation et d'un étiquetage des blocs avec un modèle CRF. Les auteurs soulignent que l'intégration d'information textuelle permet l'obtention de bons résultats car des blocs avec des étiquettes logiques peuvent avoir les mêmes caractéristiques structurelles. Ils analysent les erreurs de leur système comme dues à l'accumulation d'erreurs aux différents niveaux de segmentation. Ils signalent que le principal avantage de leur méthode est l'utilisation d'une *étape d'apprentissage*, qui permet de prendre en compte la variabilité des documents à analyser. Cela montre l'importance cruciale de l'introduction d'une phase d'apprentissage pour l'obtention de système de reconnaissance de documents efficaces et facilement adaptables à un nouveau type de document.

Chaudhury [CJDR09] présente une méthode de segmentation de documents utilisant également une combinaison hiérarchique de CRF. Le premier niveau de cette hiérarchie permet d'étiqueter chaque pixel selon trois classes : texte, fond et image en utilisant des caractéristiques locales. Des caractéristiques contextuelles permettent ensuite de classer les blocs (titre, auteur, paragraphe, etc.). Les expérimentations consistent en l'analyse d'articles scientifiques, le nombre de documents utilisés n'est pas indiqué.

1.1.2 Avantages et inconvénients

Les méthodes statistiques permettent l'intégration de bruit et d'incertitude qui sont souvent présents dans les documents à analyser. Un autre avantage majeur de ces méthodes est qu'elles permettent un apprentissage automatique du système et donc proposent une conception facilitée du système. Cependant, si cet apprentissage facilite la conception du système, il nécessite des connaissances *a priori* de l'utilisateur par le choix des caractéristiques utilisées dans l'apprentissage du modèle. De plus, une vérité terrain annotée est nécessaire pour effectuer cette étape d'apprentissage. Une vérité terrain annotée est coûteuse à réaliser et par conséquent n'est pas toujours disponible. Quand elle est disponible, elle peut ne pas être d'un volume suffisant pour apprendre tous les cas présents si certains d'entre eux sont rares.

Les méthodes statistiques ne sont en général pas capables de rendre compte de la structure hiérarchique des documents. La capacité à exprimer la structure hiérarchique des documents est cependant cruciale pour la reconnaissance des documents complexes comme les tableaux, les formules mathématiques ou les diagrammes. Les méthodes statistiques permettent de gérer les variations locales mais leur inférence de la structure globale du document est limitée. Par exemple, les relations spatiales pouvant être modélisées avec un CRF sont limitées à de petites portions de l'espace. Ainsi,

Shetty [SSS+07] modélise les relations spatiales entre des voisinages où les voisinages ont approximativement la taille d'un mot. Cependant, pour pouvoir décrire la structure hiérarchique d'un document, une méthode idéale doit aussi être capable de modéliser les relations spatiales entre des éléments plus grands comme des paragraphes, titres, images, etc. et les relations spatiales à distance.

L'un des principaux avantages des méthodes statistiques de notre point de vue est leur capacité à s'adapter à un nouveau type de documents à l'aide d'une phase d'apprentissage automatique sur des données annotées. Cependant, elles ne sont en général pas capable de transcrire la structure hiérarchique globale des documents. De plus, il est difficile d'intégrer de la connaissance sur les documents provenant de l'utilisateur dans le système et le système n'est en général pas compréhensible par un utilisateur humain. Ces aspects sont en particulier permis par les méthodes syntaxiques que nous présentons dans la section suivante.

1.2 Méthodes syntaxiques

Les méthodes syntaxiques proposent une explicitation des connaissances *a priori* sur les documents par un expert dans un modèle. Ce modèle est utilisé pour interpréter les données en entrée afin de correctement segmenter et reconnaitre les documents. Le modèle contient toutes les informations permettant de transformer une structure physique en une structure logique. Ces approches permettent ainsi de tirer profit des spécificités des documents considérés. Elles sont de plus génériques : seul le modèle doit être modifié lorsqu'un nouveau type de documents doit être reconnu.

Nous distinguons dans les méthodes syntaxiques deux types d'approches : les représentations à base de règles et les représentations à base de grammaires.

1.2.1 Représentation à base de règles

Les systèmes de reconnaissance utilisant une représentation à base de règles sont spécifiques à un type de documents [NS95, TA90, DP91, YATT91, Fis91]. Ils utilisent les règles de formatage des documents pour reconnaitre les étiquettes logiques des différents éléments. Ils utilisent pour cela des caractéristiques variées sur la position des éléments (dans la page ou les uns par rapport aux autres), ainsi que des caractéristiques typographiques (tailles, police, etc.).

Les règles étant décrites à la main par un utilisateur humain et adaptées à un type particulier de documents, elles sont difficilement adaptables à un nouveau type de documents. De plus, les représentations à base de règles sont peu flexibles et les règles peuvent devenir assez arbitraires.

1.2.2 Représentation à base de grammaires

Les méthodes à base de grammaires [Con93, KNSV93, IA91, TI94] reposent sur une analogie entre la structure des documents et la syntaxe d'une langue. Cette approche

permet d'utiliser un certain nombre d'outils mathématiques déjà existants et de tenir compte de la nature hiérarchique des documents.

Les systèmes grammaticaux permettent une description sémantique plus précise des relations entre les éléments. En utilisant les approches grammaticales, l'image est segmentée en primitives et l'utilisateur construit un arbre de règles qui décrit comment composer les primitives. La représentation sous forme d'arbre des règles permet une expression des structures récursives comme les structures hiérarchiques.

Conway [Con93] décrit une méthode syntaxique utilisant une grammaire de page. La structure physique est décrite par un ensemble de règles grammaticales spécifiant les relations spatiales entre les éléments et des informations sur la taille de la police, l'alignement ou l'indentation. Les relations spatiales décrites sont du type « sous », « à gauche de », « à droite de », « sur », « à droite de » et « proche de ». La structure logique est quant à elle décrite à l'aide d'une grammaire non contextuelle (Context-Free Grammar). Les descriptions des structures physique et logique sont toutes deux déterministes et l'analyse est faite indépendamment l'une de l'autre.

Krishnamoorthy [KNSV93] applique récursivement des grammaires sur les projections horizontale et verticale de la page à analyser. L'analyse est décomposée en quatre étapes. À la première étape, un seuil est appliqué sur le résultat de la projection pour constituer des atomes. Ces atomes sont regroupés en molécules à la deuxième étape. À la troisième étape, des étiquettes logiques sont attribuées à chacune des cellules. Les cellules contigües avec la même étiquette logique sont ensuite fusionnées à la quatrième et dernière étape.

Afin d'obtenir des méthodes à base de grammaires efficaces pour la reconnaissance de documents, celles-ci doivent tenir compte de la bidimensionnalité et être capable de gérer l'incertitude. Nous présentons maintenant les méthodologies mises en place pour cela.

1.2.2.1 Tenir compte de la bidimensionnalité

L'une des difficultés liées à la représentation à base de grammaires et que les outils mathématiques existants sont adaptés à la représentation de structures monodimensionnelles. Dans les structures monodimensionnelles, les entités sont parfaitement ordonnées dans une direction unique. Dans les structures bidimensionnelles, les entités sont disposées spatialement, et une composante donnée ne possède donc pas un seul prédécesseur et un seul successeur (ce qui est les cas dans les chaînes), mais un certain nombre de voisins.

Des formalismes bidimensionnels ont été proposés, il s'agit essentiellement de grammaires de graphes [GB95, RC96], ou de modèles dérivés des grammaires de graphes (grammaires d'arbre [Bra69], grammaires de web [PR69], grammaires plex [Fed71]). Les grammaires de graphes présentent le plus grand pouvoir d'expression parmi ces formalismes bidimensionnels. Dans cette représentation, les blocs de texte sont les sommets du graphe tandis que les arêtes représentent les relations entre les blocs. Ces formalismes présentent plusieurs inconvénients. Les problèmes de segmentation ne sont pas gérés. Il n'y a pas non plus de recherche de la meilleure solution s'il y a des ambiguï-

tés car il n'y a pas de mécanisme de retour en arrière. Enfin, la production des règles est difficile en raison de la syntaxe complexe utilisée. Cela rend ces formalismes peu utilisables pour des documents complexes nécessitant une description complexe.

Une généralisation des grammaires de chaînes à l'aide d'opérateurs de position a été proposée par Coüasnon dans la méthode DMOS-P [Cou06]. De tels opérateurs de position permettent de modéliser une plus grande variété de relations entre les entités. La syntaxe proposée reste la plus simple possible afin de permettre la modélisation de problèmes complexes par les concepteurs de grammaire.

1.2.2.2 Gérer l'incertitude : les grammaires stochastiques

Une autre limite de ces approches est que les grammaires utilisées pour représenter la structure des documents sont généralement déterministes. il est alors difficile de lever certaines ambiguïtés si l'extraction des composantes physiques n'est pas robuste et présente des erreurs.

Différentes approches basées sur les grammaires stochastiques ont été proposées. Ainsi, Tateisi et Itoh [TI94] proposent une grammaire stochastique pour analyser les documents. Un coût est associé à chaque élément. Le but est alors de construire la structure logique qui présente un coût minimal. Cette recherche exhaustive des solutions est possible car la grammaire utilisée est monodimensionnelle. Mao et Kanungo [MK01] proposent quant à eux l'utilisation d'une grammaire non contextuelle stochastique afin de traiter le bruit et l'incertitude. Cette approche repose elle aussi sur une représentation monodimensionnelle de la structure.

Cependant, une grammaire bidimensionnelle est plus adaptée pour traiter une grande variabilité de type de documents (cf. section 1.2.2.1). Le coût d'incorporation des probabilités dans les grammaires bidimensionnelles peut s'avérer assez élevé en raison de l'explosion combinatoire des solutions possibles à explorer. Maroneze [MCL11] propose l'introduction d'une approche basée sur une grammaire bidimensionnelle localement stochastique. Pour cela, il introduit un opérateur permettant de définir l'introduction locale d'un mécanisme de parsing stochastique. Cette approche permet d'utiliser une grammaire bidimensionnelle tout en gérant l'explosion combinatoire.

Les représentations à base de grammaire permettent de représenter efficacement les structures hiérarchiques et de faire une description sémantique précise des relations entre éléments. Leur pouvoir d'expression est élevé et adapté aux documents complexes. Des approches ont été proposées pour tenir compte de la bidimensionnalité des documents. Plusieurs approches ont montré la possibilité de combiner les approches syntaxiques et statistiques en introduisant un mécanisme stochastique dans la description grammaticale. Cela permet de gérer le bruit et les incertitudes présents dans les documents.

Cependant, les descriptions à base de grammaire sont généralement faites manuellement par un utilisateur humain. Il n'y a alors pas d'étape d'apprentissage automatique pour concevoir le système contrairement aux approches statistiques. Cela limite leur capacité de généralisation à d'autres types de documents puisque la description manuelle

doit alors être refaite.

1.3 Inférence dans les systèmes syntaxiques

Les capacités d'adaptation des méthodes syntaxiques à un nouveau type de documents sont limitées en raison de l'absence d'une étape d'apprentissage automatique dans leur description. En raison de la difficulté de la tâche, peu de méthodes ont été proposées pour apprendre automatiquement les descriptions grammaticales de la structure de documents.

Shilman [SLV05] présente une méthode pour apprendre des modèles grammaticaux non génératifs pour l'analyse de documents. Il focalise ses efforts sur la sélection de caractéristiques et l'estimation de paramètres. L'utilisateur doit spécifier la grammaire de page et fournir un ensemble de pages annotées. Cette méthode permet d'automatiser une partie de l'écriture de la description grammaticale, rendant ce travail plus facile et plus rapidement fait. C'est une première étape intéressante dans la construction d'une étape d'apprentissage pour la description de systèmes grammaticaux. Cependant, l'utilisateur n'est pas aidé pour la description de la grammaire qui n'est pas selon nous une tâche triviale, spécialement avec des corpus de documents à analyser de plus en plus complexes et hétérogènes.

Si l'inférence grammaticale a été peu abordée dans le contexte spécifique de l'analyse de documents, ce sujet a été étudié dans de nombreux autres domaines [dlH05]. La plupart des méthodes d'inférence grammaticale ont été développées pour des grammaires monodimensionnelles. Or, comme nous l'avons indiqué, dans la section 1.2.2.1, nous avons besoin de tenir compte de la bidimensionnalité des documents pour pouvoir décrire efficacement la structure des documents. De plus, les méthodes d'inférence grammaticale ne sont pas robustes au bruit. Un système d'analyse de documents efficace doit être capable de gérer des images bruitées. Enfin, il est difficile pour la plupart des méthodes d'inférence grammaticales de combiner à la fois l'inférence grammaticale et d'autres méthodes ou des connaissances *a priori*. La reconnaissance de documents est un domaine où l'utilisateur possède de nombreuses connaissances *a priori* qu'il nous semble indispensable de pouvoir intégrer pour obtenir des systèmes d'analyse efficaces.

Ces nombreux éléments nous montrent les limites de l'inférence grammaticale dans notre contexte applicatif où nous souhaitons décrire la structure de documents complexes, bidimensionnels et bruités. De plus, dans les méthodes d'inférence grammaticale comme pour les méthodes statistiques présentées dans la section 1.1, il est nécessaire de disposer d'un ensemble d'apprentissage annoté. Or, ce n'est pas toujours le cas, notamment en raison du coût important de production d'une vérité terrain annotée.

1.4 Bilan

Nous pensons qu'il y a un intérêt à exprimer explicitement les connaissances *a priori* sur la structure du document à l'aide d'un formalisme intuitif et intelligible permettant de décrire simplement les règles de structuration du document. Nous souhaitons garantir

l'explicitation et l'externalisation des connaissances descriptives du problème traité. La formalisation des connaissances permet de définir des approches génériques, c'est-à-dire utilisables pour l'analyse de documents structurés de différentes natures.

Bien que peu étudié, l'apprentissage automatique pour inférer des connaissances est indispensable. Les performances des méthodes structurelles seraient significativement améliorées si elles disposaient d'une inférence automatique des connaissances [Ram06]. Comme le soulignent Dengel et Shafait [DS14], la capacité d'un système à gérer des documents complexes de domaines différents vient de sa capacité à extraire la connaissance à partir d'un apprentissage sur une vérité terrain. Cependant, ils soulignent également que les capacités de généralisation des systèmes actuels restent très en dessous de ce qu'un être humain est capable de faire. Ce constat nous conforte dans notre volonté de garder un système compréhensible par l'être humain afin que celui-ci puisse intervenir dans le processus de description du système si nécessaire.

Dans notre contexte, nous nous ajoutons une contrainte supplémentaire qu'il nous semble indispensable de traiter : pouvoir faire cet apprentissage automatique sans vérité terrain annotée. En effet, comme nous l'avons souligné, la création d'une vérité terrain est une tâche coûteuse et fastidieuse, limitant le nombre de documents disponibles pour l'apprentissage. Pour ce faire, nous concentrons nos efforts sur les méthodes d'apprentissage non supervisées et en particulier les méthodes de clustering que nous présentons dans le chapitre suivant. Par cette approche, nous allons détecter des structures dans les données. Afin d'apporter du sens à ces structures, nous proposons de travailler en interaction avec l'utilisateur afin de :

- valider les structures détectées automatiquement ;
- apporter du sens aux données, notamment en étiquetant ces structures avec des libellés porteurs de sémantique.

Chapitre 2

Apprentissage non supervisé : le clustering

L'extraction automatique de connaissances a pour objectif l'acquisition de nouvelles connaissances au sein de grandes quantités de données à l'aide de méthode automatique ou semi-automatique. L'apprentissage non supervisé est en soi une tâche difficile car il n'existe pas d'*a priori* sur les résultats à obtenir. Le volume des données et leur grande variabilité rendent l'extraction de connaissances d'autant plus difficile. Notre but est d'obtenir une meilleure connaissance des données en détectant à la fois les anomalies, et les caractéristiques saillantes.

L'une des techniques majeures de l'extraction de connaissances est le clustering. Le but des méthodes de clustering est de découvrir dans les données des groupes de données homogènes, également appelés clusters. La détection de ces groupes repose sur l'utilisation d'une mesure de similarité entre les observations qui n'est pas facile à spécifier sans connaissance *a priori* sur les données.

Un cluster est un groupe d'individus homogène dans le sens où deux individus proches doivent appartenir au même groupe tandis que deux individus éloignés doivent appartenir à des groupes différents. Un cluster idéal est un ensemble de points *compact*, l'ensemble des objets du groupe sont similaires, et *isolé*, l'ensemble des objets du cluster sont complètement différents des objets des autres clusters. Il s'agit alors de minimiser la distance intra-cluster et de maximiser la distance inter-cluster (figure 2.1). En réalité, un cluster est une entité subjective dont la pertinence et l'interprétation nécessite des connaissances sur le domaine. S'il est possible pour un être humain de détecter des clusters en deux ou trois dimensions, il est nécessaire d'avoir des algorithmes automatiques pour les dimensions supérieures ainsi que pour les données mal séparées [FJ02].

Dans ce chapitre, nous présentons d'abord un panorama des méthodes de clustering existantes. Nous décrivons ensuite un ensemble de méthodes plus récent, les méthodes de clustering ensemblistes. Puis, nous présentons deux problématiques majeures du clustering : la détermination du nombre de clusters et l'évaluation des clusters. Enfin, nous détaillons la méthodologie retenue pour notre méthode d'inférence semi-automatique et interactive de règles pour la description de méthodes syntaxiques de reconnaissance

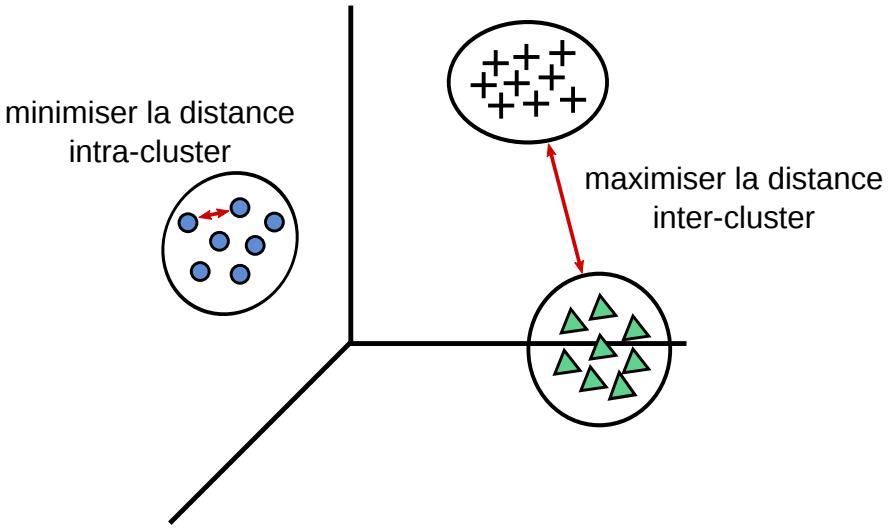


FIG. 2.1 – Définition général du clustering

de la structure de documents.

2.1 Panorama des méthodes existantes

Un très grand nombre d'algorithmes de clustering existent. Classiquement, ces algorithmes sont divisés en deux grands groupes principaux : les algorithmes *hiérarchiques* et les algorithmes *partitifs*. Les algorithmes de clustering hiérarchiques recherchent récursivement des clusters imbriqués les uns dans les autres tandis que les algorithmes partitifs trouvent tous les clusters simultanément et n'imposent pas de structure hiérarchique. Il n'existe pas d'algorithme qui soit capable d'identifier toutes les formes de clusters et de structures que nous pouvons rencontrer en pratique. Chaque algorithme définit une représentation des données. L'algorithme, et donc la représentation, doit être choisi en fonction des données à analyser et du but de l'utilisateur [XW05] [JD88].

Nous présentons ici un rapide état de l'art des grandes familles d'algorithmes de clustering :

- clustering basé sur la minimisation d'une fonction de coût ;
- clustering hiérarchique ;
- clustering basé sur la densité ;
- clustering basé sur les graphes ;
- clustering basé sur les distributions.

Nous nous focalisons sur le « *hard* clustering », c'est-à-dire les algorithmes tels que chaque individu est classé dans une *seule classe*. Il est à noter qu'il existe un ensemble de méthodes, appelé « *fuzzy* clustering », telles que chaque individu n'appartient pas à un seul cluster. Au contraire, dans le cas du fuzzy clustering, nous connaissons pour chaque individu sa probabilité d'appartenir à chaque cluster. Cette approche est utilisée en particulier lorsque les clusters ne sont pas bien séparés. L'un des algorithmes les plus

connus de fuzzy clustering est l'algorithme Fuzzy C-Means [Bez81].

2.1.1 Error-based clustering

Dans cette famille d'algorithme, la recherche des clusters est vue comme un problème d'optimisation. Les clusters sont modifiés jusqu'à ce qu'une fonction de coût soit minimisée.

L'algorithme le plus connu de cette famille est l'algorithme des K-Means (appelé K-Moyennes en français). C'est plus généralement l'un des algorithmes de clustering les plus connus et les plus simples. Le principe de cet algorithme est de trouver la partition qui minimise la distance de chaque point au centroïde du cluster. Le centroïde du cluster correspond à la moyenne des points du cluster. Il se base sur une minimisation de la somme des carrés des distances. C'est un algorithme qui est efficace, puisqu'il présente une complexité linéaire en le nombre d'individus et qui nécessite la définition de peu de paramètres. De nombreuses variations autour de l'algorithme ont été proposées [Jai10]. On peut citer par exemple l'utilisation de la distance de Mahalanobis [LNN97], de la distance L1, ou l'introduction de la théorie des ensembles flous pour obtenir des partitions non exclusives [PB95]. Sa limitation majeure est son incapacité à identifier des clusters avec des formes arbitraires. De plus, même s'il y a peu de paramètres à fixer, il est nécessaire de fixer le nombre de clusters à obtenir. Cette tâche est complexe pour l'utilisateur comme nous le verrons dans la section 2.3. En outre, les K-means sont sensibles à l'initialisation des centres et aux outliers puisque tous les points doivent être affectés à un cluster. Les clusters peuvent donc être fortement déformés par les outliers.

L'algorithme des K-Medoïds [KR87] est une variante basée sur l'algorithme des K-Means pour laquelle le point représentatif du cluster est un point sélectionné parmi les données plutôt que le centroïde de la classe. Cet algorithme est plus robuste aux outliers que les K-Means et est utilisable avec des variables qualitatives. Cependant, cet algorithme présente une complexité algorithmique plus grande que celle des K-Means. Pour des données plus volumineuses, l'algorithme CLARA (Clustering for Large Applications) [KR90], basé sur un échantillonnage des données, a été proposé.

Pour pallier la problématique du nombre de clusters à trouver à fournir en paramètres l'algorithme ISODATA a été proposé [BH67]. Cet algorithme permet d'ajuster automatiquement le nombre K de clusters lors des itérations. Cependant, cet ajustement automatique nécessite de fixer manuellement un plus grand nombre de paramètres et les performances de l'algorithme dépendent énormément de ces paramètres.

2.1.2 Clustering hiérarchique

Les méthodes de clustering hiérarchiques proposent une organisation des données sous la forme d'une structure hiérarchique à partir d'une matrice de similarité. Deux approches existent : les clusterings hiérarchiques agglomératifs et les clusterings hiérarchiques divisifs. Dans le cas des algorithmes agglomératifs, l'approche est ascendante : chaque individu est seul dans son cluster au départ et les paires de clusters sont fusionnées au fur et à mesure jusqu'à ce que tous les individus appartiennent à un seul

et même cluster. Pour les algorithmes divisifs, c'est le contraire l'approche est descendante : tous les individus appartiennent au départ à un même cluster et les clusters sont divisés au fur et à mesure.

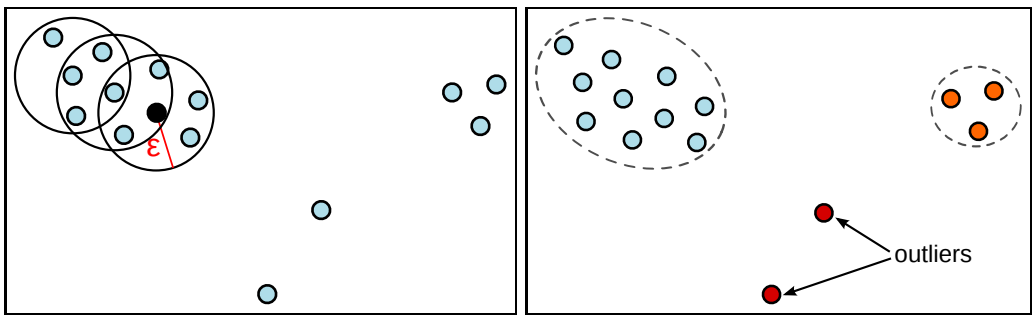
Les méthodes de clustering hiérarchique présentent l'avantage de ne pas nécessiter de fixer le nombre de clusters en paramètre d'entrée de l'algorithme. En effet, les résultats sont présentés sous la forme d'un dendrogramme. La partition finale est ensuite obtenue par l'utilisateur en coupant le dendrogramme à un certain niveau.

L'algorithme de cette famille le plus couramment utilisé est la classification ascendante hiérarchique (CAH). Un des inconvénients majeur de ces approches est que la décision de fusionner deux clusters (ou diviser un cluster) ne peut pas être annulée. De plus, leur application est difficile sur de grands jeux de données en raison de leur complexité algorithmique ($O(n^3)$).

Plusieurs algorithmes hiérarchiques ont été proposés afin de gérer des jeux de données volumineux : CURE [GRS01], ROCK [GRS00], Chameleon [KHK99] et BIRCH [ZRL96]. CURE [GRS01] permet d'obtenir des clusters de formes plus complexes. L'algorithme ROCK [GRS00] est quant à lui utilisable avec des données qualitatives. L'algorithme BIRCH [ZRL96] permet à la fois de gérer des données volumineuses et d'apporter une robustesse aux outliers.

2.1.3 Clustering basé sur la densité

Les méthodes basées sur la densité construisent des clusters comme des régions de l'espace contenant une grande densité de points, séparés par des régions vides ou de densité faible. Dans ces méthodes, les clusters peuvent être de forme arbitraire. De plus, il y a une gestion du bruit contrairement aux K-Means par exemple pour lesquels les outliers ont un impact fort sur la qualité de la partition produite.



(a) Recherche des clusters

(b) Résultat final : deux clusters et deux outliers sont trouvés

FIG. 2.2 – Exemple de fonctionnement de l'algorithme DBSCAN pour un rayon ε et MinPts=3

L'algorithme le plus connu de cette famille est l'algorithme DBSCAN [SEKX98] (Density-Based Spatial Clustering of Applications with Noise). Il prend deux paramètres en entrée : une distance ε, qui représente la taille du voisinage considéré, et $minPts$ le nombre minimum de points devant se trouver dans le rayon ε. Il permet

d'obtenir des clusters de forme arbitraire et est peu sensible aux outliers car les points non compris dans les clusters sont considérés comme du bruit (figure 2.2). Ses performances dépendent par contre des paramètres en entrée.

Des alternatives à DBSCAN existent. Nous pouvons par exemple citer OPTICS [ABKS99] ou DENCLUE [HK99]. OPTICS nécessite les deux mêmes paramètres en entrée que DBSCAN mais permet d'obtenir des clusters de densités différentes contrairement à DBSCAN. DENCLUE est plus rapide que DBSCAN mais nécessite plus de paramètres en entrée et leur choix a un impact fort sur les performances de l'algorithme. DENCLUE est également capable de gérer des données contenant de bruit.

2.1.4 Clustering basé sur les graphes

Cette approche est basée sur la recherche dans le graphe connectant les objets entre eux des arcs à conserver pour former les clusters. Les sommets du graphe sont les données pour lesquelles nous souhaitons obtenir une partition tandis que les arcs représentent les proximités entre les paires. Les clustering hiérarchiques *single linkage* et *complete linkage* peuvent être représentés sous la forme de graphe [JD88].

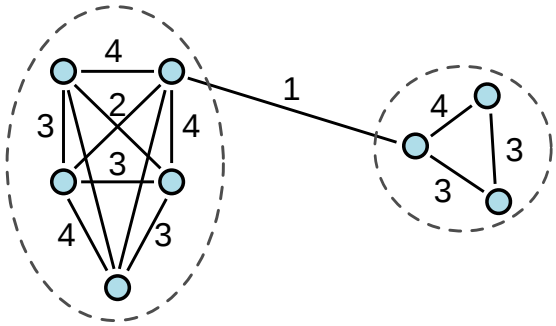


FIG. 2.3 – Exemple illustrant le fonctionnement du Markov Cluster Algorithm

D'autres algorithmes non hiérarchiques sont basées sur la théorie des graphes. Le Markov Cluster Algorithm (MCL) [vD00] par exemple repose sur l'idée suivante : au sein d'un même cluster il existe des connexions fortes entre les individus (figure 2.3). Au contraire, il y a peu de lien entre les clusters. Selon ce principe, lorsque nous démarrons sur un sommet, si nous nous déplaçons aléatoirement alors nous avons plus de chances de rester au sein du même cluster que de changer de cluster. De plus, les poids des arcs sont mis à jour pour renforcer cet effet. Cet algorithme présente l'avantage de ne pas nécessiter une détermination préalable du nombre de clusters. De plus le passage à l'échelle est possible, le MCL clustering est rapide.

2.1.5 Clustering basé sur les distributions

Cette famille d'algorithmes fait l'hypothèse que les données ont été générées en suivant une certaine loi de distribution. Le but est alors de retrouver les paramètres

(inconnus) de cette distribution. La loi de distribution prise comme hypothèse est souvent un mélange gaussien. L'algorithme Espérance-Maximisation est souvent utilisé pour estimer les paramètres d'une densité dans le cadre des mélanges gaussiens. Cet algorithme est sensible aux paramètres d'initialisation et la convergence est lente et peut être vers un optimum local. Le package MCLUST [FRMS12] est un exemple d'implémentation d'algorithme de clustering basé sur les distributions tel que décrit par Fraley et Raftery [CF02].

2.2 Méthodes ensemblistes

Plus récemment, une nouvelle famille d'algorithme de clustering est apparue, les méthodes ensemblistes. Les méthodes ensemblistes de clustering sont inspirées des bons résultats obtenus par les méthodes ensemblistes sur les méthodes supervisées (par exemple le *bagging* ou le *boosting*). L'approche ensembliste part d'un constat simple : un algorithme de clustering impose une organisation aux données, notamment par la mesure de (dis)similarité utilisée sur les données. Le but de la combinaison des résultats de différents clustering est d'améliorer les résultats. Cela repose sur deux idées principales : le fait de combiner les résultats peut permettre de compenser les erreurs d'un seul algorithme et la décision d'un groupe est probablement plus fiable que celle d'un seul individu.

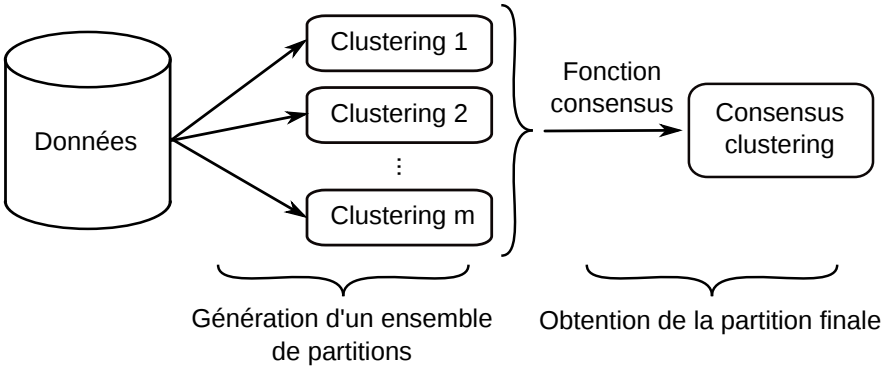


FIG. 2.4 – Schéma de fonctionnement général du clustering ensembliste

Le principe général des méthodes ensemblistes se décompose en deux étapes présentées dans la figure 2.4 :

1. Génération d'un ensemble de partitions sur les données ;
2. Utilisation d'une fonction consensus pour obtenir la partition finale sur les données.

Les propriétés attendues d'une méthode ensembliste de clustering sont décrites par Vega-Pons [VR11] :

– robustesse : la partition finale doit obtenir en moyenne de meilleurs résultats que les algorithmes de clustering seuls ;

- cohérence : le résultat de la partition finale doit être très similaire aux résultats combinés de tous les algorithmes de clustering seuls ;
- nouveauté : la méthode ensembliste doit permettre de trouver des solutions qu'un algorithme seul ne peut pas trouver ;
- stabilité : les résultats doivent avoir une sensibilité plus petite au bruit et aux outliers qu'un algorithme seul.

En général, il n'y a pas de contraintes sur la manière dont sont obtenues les partitions. Il est possible par exemple d'utiliser pour construire l'ensemble des partitions :

- le même algorithme de manière identique plusieurs fois, s'il existe une initialisation faite de manière aléatoire (par exemple l'initialisation des centroïdes pour les K-Means) ;
- le même algorithme avec des paramètres d'initialisation différents ;
- différents algorithmes de clustering.

Le choix de la fonction de consensus est par contre beaucoup plus crucial pour obtenir une bonne partition finale. Il existe deux approches principales. La première utilise les *co-occurrences des données*. Dans cette approche, nous analysons combien de fois deux individus appartiennent au même cluster. La seconde cherche une *partition médiane*. La partition finale est obtenue comme étant la solution à un problème d'optimisation, celui de trouver la partition médiane.

Nous nous intéressons ici en particulier à une méthode basée sur une matrice de co-association proposée par Fred et Jain, l'Evidence Accumulation Clustering (nommé EAC clustering dans la suite). Les méthodes basées sur une matrice de co-association reposent sur la construction d'une nouvelle mesure de similarité entre les individus à partir de l'ensemble des partitions. Plus les individus sont proches plus souvent ils sont regroupés ensemble lors des différentes partitions. Cet algorithme permet de stabiliser les clusters instables et n'impose pas de formes aux clusters recherchés. De plus, elle propose une méthodologie pour déterminer automatiquement le nombre de clusters, ce qui est souvent une difficulté dans les méthodes de clustering (cf. 2.3). L'EAC clustering est présenté en détail dans le chapitre 5.

2.3 Détermination du nombre de clusters

De nombreuses méthodes de clustering nécessitent de fournir en paramètre le nombre de clusters K de la partition à obtenir. Déterminer automatiquement le nombre de clusters est une des tâches les plus difficiles du clustering. En effet, cela nécessite une connaissance des données que l'utilisateur ne possède pas toujours à ce moment de l'analyse. De plus, le nombre K de clusters impacte fortement la qualité du clustering obtenu. Si le nombre de clusters est trop important, l'interprétation et l'analyse des clusters peut-être difficile à faire. Au contraire, un nombre trop faible de clusters conduit à une perte d'information. Pour des classes bien séparées, les algorithmes de clustering retrouvent généralement le même nombre de clusters. Le problème se pose principalement dans le cas de chevauchement de classes.

Une première approche souvent utilisée pour déterminer le nombre de clusters de la

partition est de visualiser les données et de fixer manuellement le nombre K de clusters à trouver. Cependant cette tâche n'est pas triviale et cette approche est difficilement utilisable si les clusters ne sont pas bien séparés ou si les données sont en grande dimension.

Une autre approche possible consiste à produire différentes partitions avec différentes valeurs de K. La meilleure valeur de K est ensuite déterminée à l'aide de critères portant sur la qualité des clusters obtenus. Comme critères utilisés pour cette tâche, nous pouvons citer la *statistique du « gap »* proposée par Tibshirani [TGH01], la *silhouette* proposée par Rousseeuw [KR90] ou le critère d'information d'Akaike (AIC) et le critère d'information bayésien (BIC) [SGJ03].

Pour les méthodes ne nécessitant pas de fournir le nombre de clusters à obtenir en paramètre, toutes ne fournissent pas automatiquement une partition à l'utilisateur. C'est alors à l'utilisateur de déterminer lui-même le nombre de clusters de la partition finale. C'est notamment le cas pour la Classification Ascendante Hiérarchique (CAH). La découpe du dendrogramme vient alors après l'exécution de l'algorithme, permettant d'obtenir la partition finale. De nombreux indicateurs classiques existent pour aider l'utilisateur dans cette tâche : le R^2 RSQ, le R^2 semi-partiel SPRSQ, le Cubic Clustering Criterion (CCC), l'inertie intra-classe RMSSTD, le pseudo F PSF et le pseudo t^2 PST2. En pratique, l'utilisateur est souvent amené à combiner les résultats de différents critères pour prendre sa décision. Il peut également utiliser directement le dendrogramme et couper les branches lorsqu'elles sont longues, correspondant à une forte perte d'inertie inter-classe.

2.4 La difficile évaluation des clusters

L'évaluation de la pertinence des clusters trouvés est difficile. Ainsi Jain et Dubes [JD88] soulignent dans *Algorithms for Clustering Data* que :

> The validation of clustering structures is the most difficult and frustrating part of cluster analysis. Without a strong effort in this direction, cluster analysis will remain a black art accessible only to those true believers who have experience and great courage.

En effet, le clustering est une tâche non supervisée. Il n'existe pas de données annotées contenant les clusters à trouver. Nous n'avons donc pas de clusters de référence auxquels comparer les clusters obtenus. Des mesures de qualité des clusters existent et peuvent être classées en deux catégories :

- les mesures dites *internes*, qui évaluent en général la capacité d'un algorithme à produire des clusters qui maximisent la similarité intra-cluster et minimisent la similarité inter-cluster.
- les mesures dites *externes*, qui comparent les résultats obtenus par l'algorithme à des résultats attendus. Ces mesures nécessitent alors un ensemble de données annotées.

2.4.1 Mesures internes

Les mesures dites internes valident la capacité d'un algorithme à produire une bonne partition au sens où nous l'avons défini précédemment, c'est-à-dire des clusters qui maximisent la similarité intra-cluster et minimisent la similarité inter-cluster. Elles considèrent donc à la fois la compacité de chaque cluster ainsi que la séparation des clusters entre eux. Les indices n'évaluant qu'un seul de ces deux aspects sont plus rares. On peut cite par exemple le RMSSTD (Root-mean-square standard deviation) qui évalue la compacité des clusters. Nous présentons ici trois mesures internes d'évaluation des clusters : l'indice de Dunn, la silhouette et l'indice de Davies-Bouldin.

Indice de Dunn : L'indice de Dunn [Dun73] est le rapport entre la distance maximale qui sépare deux éléments classés ensemble et la distance minimale qui sépare deux éléments classés séparément. C'est un indice qui ne repose pas sur une distance particulière et qui peut donc être utilisé dans une grande variété de situations. Un indice de Dunn élevé indique une distance inter-classes élevée et une distance intra-classes faible, ce qui correspond à la définition d'une bonne partition des données.

Silhouette : La silhouette [Rou87] est un indice pouvant être calculé pour chaque objet, cluster ou pour un clustering entier. Cette mesure est comprise entre -1 et 1, une valeur positive indique des clusters compacts et bien séparés. Pour un objet, la silhouette permet d'évaluer si les objets qui appartiennent au même cluster qu'un objet x sont plus proches de x que les objets des autres clusters.

Indice de Davies-Bouldin : L'indice de Davies-Bouldin [DB79] est un indice qui favorise les clusters compacts et bien séparés dans l'espace des données. Une valeur faible indique un clustering de bonne qualité.

L'avantage des mesures internes est qu'elles ne nécessitent donc pas de données annotées et peuvent par conséquent être utilisées quelle que soit la problématique. Le problème des mesures internes est qu'elles ne valident pas que les clusters obtenus ont un sens pour l'utilisateur qui visualisera les résultats. Cela peut être validé avec les mesures externes.

2.4.2 Mesures externes

Une évaluation externe du clustering consiste à comparer les clusters obtenus à des clusters idéaux, annotés dans une vérité terrain. Cette évaluation n'est pas facilement possible dans notre cas étant donné qu'elle nécessite d'avoir une vérité terrain annotée. Si des bases de référence existent, à notre connaissance aucune n'est adaptée à notre problématique d'analyse de la structure de documents. Nous présentons ici trois mesures externes d'évaluation des clusters : l'indice de pureté, le rand index et la F-mesure

Indice de pureté : L'indice de pureté évalue si les clusters contiennent des objets d'une seule classe. L'indice est compris entre 0 et 1, la valeur 1 indiquant que les clusters sont tous purs. L'inconvénient de l'indice de pureté est qu'il surévalue la qualité d'un clustering avec un nombre important de classes. En effet, l'indice de pureté est maximal si le nombre de clusters correspond au nombre d'individus.

Rand index : Le Rand index est une mesure comparant deux clustering. Cet indice correspond au ratio des paires de points pour lesquels les deux algorithmes donnent les mêmes résultats. Plus le Rand index est élevé, plus les clusterings sont proches. Le Rand index pénalise à la fois les faux positifs et les faux négatifs. L'inconvénient majeur du Rand index est qu'il est difficilement comparable. Une version corrigée existe, le Rand index ajusté, dont la valeur est comprise entre 0 et 1. C'est l'une de mesure d'évaluation externe les plus couramment utilisées.

F-mesure : La F-mesure évalue la propension d'un cluster à contenir à la fois tous les objets d'une classe et seulement les objets de cette classe. Elle permet d'affecter différents poids pour les faux positifs et les faux négatifs. De plus, elle prend en compte le nombre de clusters générés, comparé au nombre réels de classe dans la vérité terrain.

2.5 Bilan

De très nombreuses méthodes de clustering existent. Cependant, comme nous l'avons montré, il n'existe pas un algorithme optimal qui permettra d'obtenir des résultats optimaux pour toutes les problématiques. En conséquence, nous nous tournons vers les méthodes ensemblistes qui permettent justement de compenser les erreurs faites par un seul algorithme en combinant les résultats de plusieurs partitions. La méthode EAC clustering que nous avons présentée est une méthode ensembliste, facilement implémentable, et qui propose de plus une partition finale avec une détermination automatique du nombre de clusters. De plus, elle n'impose pas de formes aux clusters trouvés. Ces caractéristiques rendent l'EAC clustering plus facilement adaptable à un grand nombre de jeux de données.

Pour l'évaluation des clusters, nous ne pouvons pas utiliser les mesures externes. En effet, les mesures externes nécessitent d'avoir à disposition une vérité terrain annotée sur les clusters. Or, nous ne possédons pas une telle vérité terrain. Nous utilisons des mesures internes qui nous permettront d'évaluer la bonne compacité et séparabilité des clusters. La sursegmentation pourra être privilégiée dans certains cas que nous présenterons dans le chapitre 5.

Dans tous les cas, il faut noter que même si nous automatisons au maximum la tâche de production des clusters, notamment en proposant une partition finale sans intervention de l'utilisateur sur la détermination du nombre de clusters, la tâche d'interprétation reste à effectuer par l'utilisateur. Cela implique donc l'intégration d'une interaction avec un utilisateur humain.

Conclusion de la première partie

La reconnaissance de la structure de documents est une tâche complexe pour laquelle de nombreuses méthodes ont été proposées. De plus, les documents à analyser sont de plus en plus complexes et variés. Les méthodes syntaxiques de la littérature ont montré leur capacité à exprimer les connaissances complexes, bidimensionnelles, permettant de décrire la structure des documents, et en particulier leur structure hiérarchique. Les connaissances qui sont ainsi exprimées doivent être externalisées pour assurer la généricité des méthodes proposées. Ainsi, le système complet de reconnaissance ne doit pas être modifié lorsqu'un nouveau type de documents doit être analysé. Seul le modèle du document est alors modifié.

Afin d'améliorer les capacités d'adaptation des méthodes syntaxiques, nous proposons d'intégrer une étape d'inférence semi-automatique des connaissances sur la structure des documents. Cette étape existe dans le cadre des méthodes statistiques et est à l'origine des bonnes performances de ces méthodes, malgré leur capacité réduite à exprimer la connaissance sur les documents. Cependant, contrairement aux méthodes statistiques, nous proposons ici une méthode capable d'inférer les connaissances sur les documents avec ou sans vérité terrain annotée disponible sur les documents. Nous tirons pour cela partie des grands volumes de documents auxquels nous avons accès ainsi que de la présence d'un utilisateur humain, le concepteur du système de reconnaissance, qui va interagir avec le système lors de la phase d'apprentissage. L'utilisateur va nous permettre de valider des structures détectées automatiquement et apportera du sens à ces données pour permettre leur intégration dans la description grammaticale sur les documents.

Cette phase d'apprentissage repose sur l'utilisation de méthodes de clustering. Nous sommes en effet dans une configuration d'apprentissage non supervisée : nous cherchons à apprendre des connaissances pour lesquelles il n'existe pas d'étiquetage dans les données d'apprentissage. Nous avons choisi d'utiliser une méthode ensembliste, l'Evidence Accumulation Clustering [FJ02] (EAC Clustering). L'EAC Clustering nous permet d'obtenir des résultats plus robustes que ceux obtenus si une seule partition est effectuée sur les données. De plus, cet algorithme n'impose de formes aux clusters à trouver, ce qui permet de garantir une bonne adaptation de l'algorithme à un grand nombre de jeux de données. Cela participe à la généricité de la méthode que nous présentons ici. Enfin, cette méthode nous permet de minimiser l'intervention de l'utilisateur dans la production des clusters en proposant notamment une détermination automatique du nombre de clusters de la partition à produire.

Dans la suite du document, nous allons détailler cette approche en proposant une méthodologie générique permettant l'inférence de règles pour la reconnaissance de la structure de document. Cette méthode sera ensuite validée dans la troisième partie en étant appliquée à des corpus de documents variés, avec ou sans vérité terrain.

Deuxième partie

Méthode

Introduction

La première partie de ce document a montré l'intérêt de l'inférence de règles pour les méthodes syntaxiques d'analyse de la structure de documents. Nous présentons maintenant la manière dont nous avons défini une méthode générique pour l'inférence semi-automatique et interactive de règles pour la reconnaissance de documents.

Cette partie présente ainsi le cœur de notre travail et expose les concepts que nous avons mis en œuvre pour créer cette méthode complète et générique de reconnaissance de documents, s'appuyant sur l'inférence semi-automatique et interactive de règles, avec ou sans vérité terrain.

Dans le chapitre 3, nous présentons la philosophie de notre approche en détaillant les caractéristiques attendues de notre méthode ainsi que l'approche par construction progressive d'un système de reconnaissance complet. Pour cela, nous décomposons la description grammaticale complète en sous-problèmes. Pour chaque sous-problème une inférence semi-automatique et interactive des règles peut-être faite avec la méthode Eyes Wide Open (EWO). Dans le chapitre 4, nous présentons la méthode EWO que nous avons mise au point pour permettre l'inférence semi-automatique et interactive de règles. Nous abordons ensuite en détails l'algorithme de clustering mis en place dans la méthode EWO (chapitre 5) ainsi que les interactions entre le système et l'utilisateur (chapitre 6).

Chapitre 3

Philosophie du système

Dans ce chapitre, nous présentons le fonctionnement général de notre méthode, la méthode Eyes Wide Open (EWO), pour la description de systèmes syntaxiques de reconnaissance de documents. Cette méthode s'intéresse plus particulièrement à l'introduction de connaissances dans la description, avec ou sans vérité terrain disponible sur les documents. Nous distinguons deux cas :

- lorsque la connaissance est connue *a priori*, notre méthode se base sur une description structurelle de cette connaissance ;
- lorsque la connaissance est au contraire difficile à acquérir, notre méthode propose une combinaison entre méthodes statistiques et structurelles.

Dans le second cas, la méthode EWO se base alors sur une inférence automatique et interactive des règles afin de faciliter l'acquisition de la connaissance. L'interaction se fait avec le *concepteur du système de reconnaissance* de documents, que nous nommons *utilisateur* dans la suite de ce document. Cela permet alors de réduire le temps nécessaire à l'adaptation d'un système de reconnaissance syntaxique à un nouveau type de documents.

La combinaison de méthodes d'apprentissage statistique avec une description structurelle des données et avec une interaction avec l'utilisateur permet de générer des règles apprises dans un contexte complexe pour lequel les méthodes purement statistiques ne sont pas adaptées. En effet, notre méthode dispose du pouvoir d'expression des méthodes syntaxiques, utile pour la description de la structure complexe des documents, tout en permettant d'apprendre certains éléments. Notre méthode peut de plus être utilisée sur des corpus sans vérité terrain, ce qui n'est pas possible pour les méthodes statistiques. Les méthodes statistiques ne peuvent en effet pas se passer d'un échantillon d'apprentissage annoté pour l'entrainement des modèles.

Nous présentons d'abord la méthodologie habituellement utilisée pour la construction d'un système de reconnaissance syntaxique complet. Cette présentation nous permet de mettre en avant les points où un apprentissage automatique présente un intérêt. Nous exposons ensuite les points d'introduction de phases d'apprentissage en utilisant des méthodes d'apprentissage statistique en interaction avec l'utilisateur. Nous identifions ensuite les capacités attendues de notre méthode ainsi que les caractéristiques qui

en découlent. Enfin, nous présentons les éléments à mettre en œuvre pour implémenter la méthode.

3.1 Construction progressive d'un système de reconnaissance complet sans apprentissage

Dans cette partie, nous commençons par présenter la méthodologie habituellement utilisée pour décrire les systèmes à base de règles, où l'utilisateur fait une description manuelle des règles. Puis nous décrivons de quelle façon un système de reconnaissance complet est construit en décomposant la tâche en sous-problèmes. Nous expliquons ensuite comment notre méthode, la méthode Eyes Wide Open, est introduite dans ce processus de construction du système de reconnaissance pour la résolution de certains sous-problèmes. Grâce à la méthode EWO, ces sous-problèmes sont alors résolus de manière semi-automatique et interactive et non plus manuellement par l'utilisateur. Cette décomposition permet de réduire la quantité de données à traiter et de réduire leur variabilité. Ces sous-problèmes sont ceux pour lesquels l'utilisateur possède *peu de connaissances* ou pour lesquels une *grande variabilité* est observée dans les documents à reconnaître.

3.1.1 Construction manuelle d'un système de reconnaissance

La description d'un système de reconnaissance syntaxique est habituellement faite manuellement comme nous l'avons souligné dans le chapitre 1. Lors d'une description manuelle, l'utilisateur sélectionne un échantillon restreint de documents à partir desquels il va décrire les règles grammaticales décrivant les documents. En effet, il n'a pas la capacité d'observer et synthétiser les connaissances extraites de grands volumes de documents et se limite donc à l'analyse d'un petit échantillon de documents. Cet échantillon se compose d'une à quelques dizaines de documents.

Le principal inconvénient de cette sélection faite de manière empirique ou aléatoire est qu'elle ne permet pas de garantir la représentativité de l'échantillon quant aux documents à traiter. La non-représentativité des données affecte ensuite la pertinence des règles décrites par l'utilisateur. Il va notamment être complexe de détecter et décrire les cas rares à partir de l'échantillon de documents sélectionnés.

Afin d'améliorer la pertinence des règles qu'il a pu décrire, l'utilisateur emploie ensuite une approche essai-erreur pour construire un système de reconnaissance efficace (figure 3.1).

(1) L'utilisateur analyse une sélection de documents pour en extraire de la connaissance ;

(2) Il utilise ensuite cette connaissance pour modifier le système de reconnaissance des documents. Le système ainsi obtenu est appliqué sur les documents à reconnaître ;

(3) Les résultats sont ensuite évalués afin de modifier le système de reconnaissance.

Ces trois étapes sont répétées jusqu'à l'obtention d'un système performant.

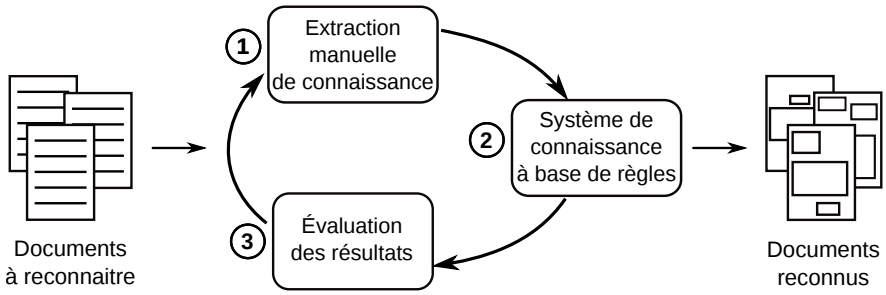


FIG. 3.1 – Description d'un système à base de règles en utilisant une extraction manuelle de connaissance combinée à une approche essai-erreur

Cette approche est complexe et longue pour l'utilisateur et peut nécessiter un grand nombre d'itérations. Ce nombre d'itérations n'est pas connu à l'avance et il est difficile pour l'utilisateur de savoir quand arrêter le processus d'essai-erreur. En effet, à chaque étape, l'utilisateur ne sait pas s'il est parvenu à détecter tous les cas présents dans le corpus à reconnaître puisqu'il n'a pas une vue exhaustive sur les données. De plus, cette technique nécessite, comme nous l'avons indiqué, une évaluation des résultats produits à chaque cycle du système. S'il existe une vérité terrain annotée sur un échantillon représentatif de documents ainsi qu'une métrique adaptée à la tâche à accomplir, alors l'évaluation peut être faite de manière automatique. Si nous ne possédons ni métrique ni vérité terrain, alors l'évaluation se fait elle aussi manuellement et manque alors également d'exhaustivité.

3.1.2 Décomposition en sous-problèmes

La construction d'un système de reconnaissance pour des documents complexes nécessite la description de nombreuses règles grammaticales. Cette description se fait en décomposant le problème principal qu'est la reconnaissance du document complet en différents sous-problèmes plus homogènes. La résolution de chacun des sous-problèmes peut nécessiter l'utilisation de terminaux différents de la grammaire. La décomposition en sous-problèmes permet de réduire la quantité de données à traiter.

Par exemple, dans le cadre de la reconnaissance de journaux de presse ancienne (figure 3.2), la reconnaissance des pages complètes va nécessiter la reconnaissance de :

- paragraphes ;
- titres ;
- articles ;
- séparateurs verticaux et horizontaux ;
- publicités ;
- etc.

Cette décomposition en sous-problèmes est effectuée par l'utilisateur. Elle peut être influencée par la vérité terrain s'il y en a une. Elle est effectuée grâce aux connaissances *a priori* que possède l'utilisateur sur les données à analyser. Le résultat de certains sous-problèmes peut servir à la résolution d'un autre. Par exemple, la capacité à reconnaître

20 centimes LE PLUS FORT TIRAGE DES JOURNAUX DU SOIR 20 centimes

L'INTRANSIGEANT

Le Journal de Paris — Directeur : LÉON BAILBY — Le Journal de Paris

Vendredi 8 Mai 1925

DU MONUMENT A L'AFFICHE

Des noms sur les murs

SIX ANS APRÈS LE TRAITÉ

La Petite Entente se réunit à Bucarest

Le problème posé par l'élection d'Hindenburg et la propagande communiste, sont inscrits à l'ordre du jour de cette conférence

LES RIFFAINS devenaient trop turbulents

Il a fallu montrer aux partisans d'Abd-El-Krim que nous étions les plus forts

LE BUDGET DE 1926

Les projets financiers de M. Caillaux

Le ministre des Finances les exposera samedi en conseil des ministres — Des économies. — Augmentation des impôts sur les revenus mobiliers et les bénéfices agricoles

Nos récoltes sont-elles en danger ?

Dernières nouvelles

Crue de l'Oise

ON REPARE...

ILS TIRENT sur les examinateurs

LES HEURES NOUVELLES

Un discours inutile

DANS LA CITÉ DES ARTS NOUVEAUX

PLACE SAINT-SULPICE

La Foire St-Germain

AU PETIT PALAIS

Du Poussin à Corot...

DÉCORATION

BENTOT Le Sidérateur par JULES HOCHE

LES ÉLECTIONS MUNICIPALES

Les partis prennent — position — pour le second tour

"N'importe qui ne doit pas pouvoir faire souffrir des êtres vivants"

Le repas hebdomadaire des "gymnotes"

les séparateurs d'articles va être utilisée pour la reconnaissance des articles. Ces éléments vont influencer l'ordre de résolution des sous-problèmes. Lors de la construction sans apprentissage d'un système, chaque sous-problème est résolu de manière manuelle par l'utilisateur.

3.2 Construction d'un système complet avec apprentissage semi-automatique et interactif

Une fois la décomposition en sous-problèmes déterminée, l'utilisateur va chercher à résoudre chacun d'entre eux. Grâce à la décomposition en sous-problèmes, les données à traiter sont plus homogènes et il devient donc possible de les traiter avec des méthodes statistiques non supervisées. Au contraire, si l'on cherche à décrire l'ensemble du document en une seule fois, la variabilité des données ne permettra pas de détecter des structures et des répétitions utiles dans les données pour la description des documents. Deux situations se présentent à l'utilisateur pour la résolution de chaque sous-problème selon les connaissances *a priori* qu'il possède.

Si une *forte connaissance a priori* existe sur le sous-problème à résoudre, alors une expression manuelle des règles grammaticales peut être faite. C'est par exemple le cas pour les paragraphes d'un article de journal qui répondent à des règles communes et connues. L'utilisateur n'a alors pas nécessairement besoin d'analyser les documents et synthétiser les informations obtenues pour produire les règles grammaticales. Sa connaissance *a priori* est suffisante pour décrire la règle grammaticale de reconnaissance des paragraphes.

Si l'utilisateur *n'a pas une connaissance a priori* suffisante du sous-problème et/ou une *grande variabilité* peut être observée alors la méthode EWO sera utilisée. C'est le cas par exemple pour les séparateurs horizontaux d'articles dans la presse ancienne. Ces éléments présentent en effet une grande variabilité. L'utilisateur ne sera alors pas capable de décrire manuellement et rapidement une règle grammaticale permettant de reconnaitre ces séparateurs avec une précision et un rappel suffisants. L'intérêt de l'utilisation de la méthode EWO est alors de permettre une inférence interactive et automatique des règles pour la résolution du sous-problème. La méthode EWO fournit à l'utilisateur une vue exhaustive des données à traiter sur l'ensemble des pages contrairement à ce qui est possible avec une approche manuelle. Le temps nécessaire à l'écriture de la règle de grammaire est alors fortement diminué par rapport à une description manuelle de la règle, pour produire une règle avec une meilleure précision et un meilleur rappel.

La figure 3.3 présente la méthodologie globale de construction d'une description grammaticale complète. Lorsque la méthode EWO est employée pour résoudre un sous-problème, cela permet de générer les règles correspondant au sous-problème traité. Ces règles sont intégrées dans la description grammaticale complète. La résolution du sous-problème permet également d'enrichir l'ensemble des données étiquetées disponibles par l'acquisition de nouvelles données (cf. section 4.1.4) ou par l'enrichissement de données étiquetées existantes (cf. section 4.1.3). Ces données étiquetées sont alors utilisables

pour résoudre les sous-problèmes restants. L'utilisateur peut lui aussi intervenir sur la description grammaticale pour la compléter/la modifier en intégrant ses connaissances sur les documents à traiter.

3.3 Caractéristiques attendues de la méthode EWO

Dans cette partie, nous identifions les capacités attendues de la méthode pour réaliser une inférence interactive et automatique de règles avec ou sans vérité terrain pour la résolution de sous-problèmes dans des contextes spécifiques. À partir de ces capacités, nous détaillons les propriétés que doit avoir notre système. Pour chacune de ces propriétés, nous détaillons l'impact sur l'implémentation de la méthode Eyes Wide Open.

3.3.1 Capacités attendues de la méthode EWO

Nous proposons ici une méthode capable de *modéliser des connaissances complexes* portant sur des données hétérogènes, bidimensionnelles et en grande quantité. Cette modélisation des connaissances doit nous permettre ensuite d'inférer des règles à un coût minimal. Pour cela, nous proposons d'inférer des règles correspondant à un sous-problème de reconnaissance du document (cf section 3.1.2) ce qui nous permet de réduire la combinatoire. Si l'inférence est réalisée sur un sous-problème, elle est par contre réalisée sur un grand volume de données ce qui nous permet d'avoir une connaissance exhaustive des documents contrairement à ce qui est possible habituellement. En effet, la construction d'un système de reconnaissance est faite habituellement soit sur des données non annotées de volume très restreint soit sur des données annotées manuellement, en volume également restreint en raison du coût de l'annotation manuelle.

3.3.2 Propriétés de la méthode

Nous présentons ici les trois grandes propriétés de notre méthode : généricité, intégration de connaissance *a priori* et utilisation possible sans vérité terrain.

3.3.2.1 Généricité

La méthode EWO doit être générique, c'est-à-dire que la méthode doit pouvoir être utilisée efficacement sur n'importe quel type de documents à analyser. Cette nécessité de généricité a un impact sur les méthodes statistiques utilisées, que ce soit pour la détection de redondances dans les données comme pour leur analyse pour l'inférence grammaticale.

Les méthodes statistiques employées doivent donc s'adapter à des données variées, dont les caractéristiques seront également variées : variables qualitatives ou quantitatives mais également structure globale des données. De plus, afin de faciliter la généricité de la méthode EWO, le nombre de paramètres à fixer pour les analyses effectuées doit être le plus réduit possible. La capacité d'un utilisateur à fixer correctement les valeurs

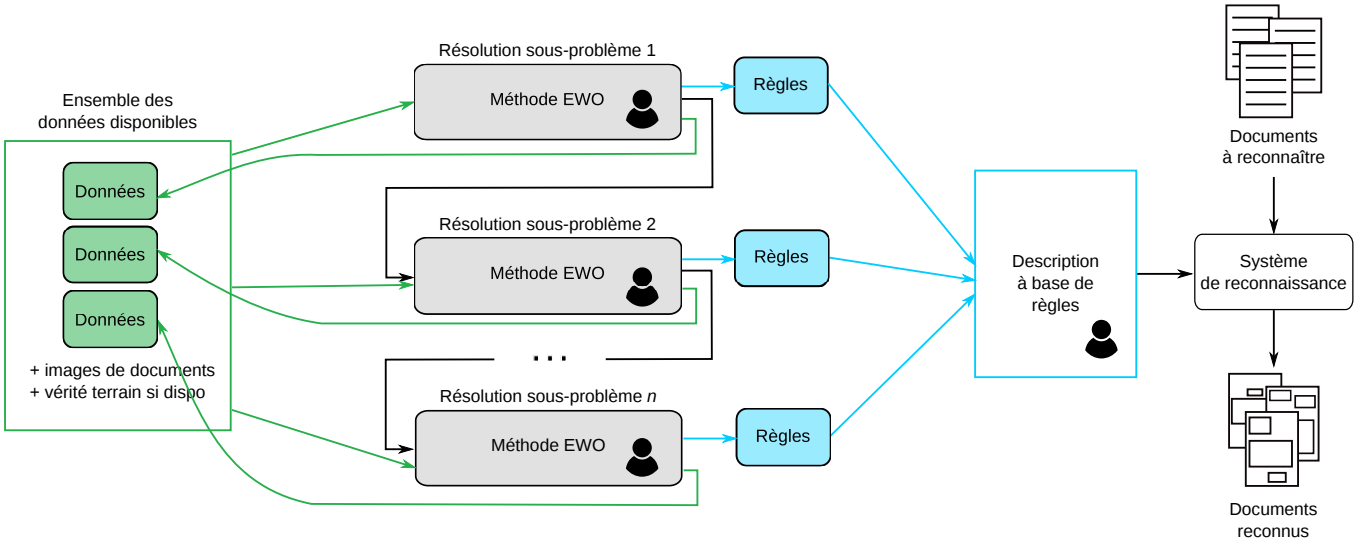


FIG. 3.3 – Méthodologie globale de la construction d'une description grammaticale avec la méthode EWO (cf. figure 4.1) par une décomposition en sous-problèmes

des paramètres provient d'une connaissance approfondie des données. Or la méthode EWO a justement pour but d'aider l'utilisateur à acquérir rapidement des connaissances sur un nouveau type de documents à reconnaître dans le but d'en faire la description grammaticale.

3.3.2.2 Intégration de connaissance *a priori*

Les méthodes syntaxiques pour l'analyse de la structure de documents présentent l'avantage de donner une description grammaticale de documents compréhensible par un utilisateur humain. Il lui est donc possible d'intégrer des connaissances qu'il possède *a priori* sur les documents.

En améliorant la capacité des méthodes syntaxiques à s'adapter à un nouveau type de documents, nous voulons garder cette capacité à introduire facilement des connaissances *a priori* de l'utilisateur. Pour cela, il faut notamment s'attacher à ce que les parties de code générées automatiquement par la méthode EWO le soient sous une forme compréhensible par un utilisateur humain. Ainsi, il pourra continuer à modifier directement la description grammaticale si cela est nécessaire. Cette contrainte a également un impact sur les méthodes statistiques utilisées qui doivent proposer des résultats compréhensibles par l'utilisateur. Nous minimisons l'utilisation de systèmes de type « boite noire » dans notre méthode. Ce genre de système n'est notamment pas utilisable pour l'inférence de la structure logique des documents. L'utilisateur peut alors difficilement injecter de la connaissance et de la sémantique dans la structure logique automatiquement inférée. Pour l'inférence de la structure physique, une plus grande souplesse est laissée à l'utilisateur sur les méthodes utilisables. Des systèmes de type « boite noire » sont alors utilisables.

3.3.2.3 Gestion de l'absence de vérité terrain

La constitution d'une vérité terrain est une tâche longue et couteuse à réaliser. Pour ces raisons, il n'est pas toujours possible d'avoir une vérité terrain afin de faire l'inférence des règles. De plus, en raison du coût de sa constitution, une vérité terrain ne sera possible que sur un nombre relativement limité de documents. Or, les documents à analyser peuvent présenter une grande variabilité. Cette variabilité peut porter sur des variations au sein d'un même type de documents, par exemple des fiches de paie provenant de différentes entreprises. Elle peut également porter sur des corpus de documents contenant des documents de type varié. C'est par exemple le cas du corpus de documents Maurdor (présenté au chapitre 9) qui contient des courriers imprimés et manuscrits, des formulaires mais aussi des cartes géographiques ou des tickets de caisse. Plus la variabilité des documents est importante plus la quantité de documents annotés nécessaire à l'inférence grammaticale est importante.

L'apprentissage automatique de système de reconnaissance de documents sans vérité terrain est une tâche particulièrement difficile (cf. section 1.3). Nous proposons ici une méthode utilisable également en l'absence de vérité terrain annotée sur les documents. Notre méthode se base sur l'analyse d'un grand volume de documents afin

de pouvoir utiliser les redondances dans la structure des documents. Cette capacité favorise également la généricité de la méthode, la rendant applicable à n'importe quel corpus de documents.

3.4 Mise en œuvre

La méthode EWO se base sur une coopération entre l'utilisateur et le système. L'utilisateur emploie la méthode EWO lorsque sa connaissance sur un sous-problème de la description du document est insuffisante pour en faire la description manuellement. L'utilisateur par sa connaissance du problème prend des décisions, notamment sur le rejet ou la conservation de groupes d'observations, et apporte du sens aux données. Le système permet quant à lui la détection de redondances dans les données, leur analyse et leur description pour l'inférence de règles grammaticales ainsi que la minimisation de la confusion des règles inférées. Le système permet d'effectuer une analyse exhaustive des documents utilisés pour l'apprentissage, avec ou sans vérité terrain.

3.4.1 Clustering

La détection de redondances dans les données ainsi qu'une partie de l'inférence de règles grammaticales repose dans la méthode EWO sur l'utilisation de méthode de clustering. En effet, nous sommes ici dans le cadre d'un apprentissage non supervisé des données. Nous cherchons à détecter des structures et redondances dans les données qui n'ont pas été explicitement annotées, que l'on possède une vérité terrain ou non sur les documents. Cette analyse se fait sur de grands volumes de données, donnant une connaissance exhaustive des documents à l'utilisateur avec ou sans vérité terrain.

Les méthodes de clustering mises en œuvre sont fortement influencées par la généricité de la méthode que nous proposons. En effet, si de nombreuses méthodes de clustering existent, elles ne sont pas toutes efficaces sur tous les jeux de données. Dans notre approche, nous ne savons pas à l'avance quelle est la forme des clusters recherchés. La méthode de clustering utilisée doit pouvoir s'adapter à ces différentes possibilités et cela sans nécessiter l'intervention de l'utilisateur. Pour cela, nous avons choisi d'utiliser la méthode de clustering introduite par Fred et Jain, l'Evidence Accumulation Clustering (EAC) [FJ02] introduite dans le chapitre 2. De plus, en utilisant cette méthode, le nombre de clusters est également déterminé automatiquement. L'utilisateur intervient alors seulement pour apporter du sens aux structures de données détectées.

Dans le chapitre 4 décrivant la méthode EWO, nous détaillerons les points d'utilisation du clustering dans notre méthode. L'implémentation sera présentée dans le chapitre 5.

3.4.2 Interaction utilisateur

L'interaction avec l'utilisateur est un élément clé de la méthode EWO. En effet, comme nous l'avons souligné dans l'état de l'art (section 1.3), une inférence grammaticale entièrement automatisée serait très complexe dans le cadre de grammaires en deux

dimensions. De plus, nous ne pourrions pas répondre aux caractéristiques nécessaires de notre système que sont l'intégration de connaissance *a priori* ainsi qu'une inférence possible en l'absence de vérité terrain.

La conception d'une méthode entièrement automatisée est alors trop complexe. En effet, les données que nous utilisons ici de par leur volume et leur hétérogénéité nécessitent l'utilisation d'une méthode semi-automatisée. La méthode EWO requiert une interaction avec un utilisateur humain. Cependant cette interaction reste limitée et repose essentiellement sur un apport de sémantique de la part de l'utilisateur sur des structures de données détectées automatiquement par la méthode EWO. Le but de notre méthode est de faciliter l'apprentissage et la création d'un système de reconnaissance adapté à un nouveau type de documents. L'interaction limitée avec l'utilisateur nous permet de conserver l'intérêt d'une méthode d'apprentissage.

Dans le chapitre 4 décrivant la méthode EWO, nous détaillerons les points d'interaction avec l'utilisateur dans notre méthode. Le fonctionnement de l'interaction sera ensuite détaillé dans le chapitre 6. L'ensemble de la méthode est évaluée dans les chapitres 8, 9 et 10.

Chapitre 4

Méthode Eyes Wide Open

Dans ce chapitre, nous focalisons notre présentation sur la méthode Eyes Wide Open (EWO). Cette méthode est utilisée pour la résolution automatique et interactive de sous-problèmes de la description grammaticale. Nous décrivons ici les trois grandes étapes de cette méthode d'inférence automatique et interactive de règles présentées dans la figure 4.1 :

1. Acquisition des données ;
2. Inférence des règles ;
3. Intégration dans une description grammaticale.

La méthode EWO est utilisée de manière progressive lors de la description d'un système de reconnaissance complet de documents. Cette méthodologie globale a été décrite dans le chapitre 3. L'utilisation de la méthode EWO présente un intérêt chaque fois que la connaissance *a priori* de l'utilisateur et/ou la variabilité des données ne permet pas facilement une description par l'utilisateur. L'utilisateur peut alors inférer les règles grammaticales du sous-problème considéré (cf. figure 4.1). La résolution de chaque sous-problème lui permet donc d'enrichir la description grammaticale complète.

Nous attirons l'attention du lecteur sur le fait que dans ce chapitre nous indiquons l'utilisation d'un algorithme de clustering à plusieurs endroits de la méthode. La présentation détaillée de l'algorithme de clustering est effectuée dans le chapitre suivant. Dans ce chapitre, nous restons génériques afin de présenter la méthodologie générale de résolution automatique et interactive d'un sous-problème de la description grammaticale. Nous rappelons que la méthode doit satisfaire les propriétés suivantes (cf. section 3.3.2 :

- généricité ;
- intégration de connaissance *a priori* ;
- utilisation possible avec ou sans vérité terrain.

4.1 Acquisition des données

Afin de réaliser l'inférence des règles, un ensemble de données doit être collecté (Fig 4.1 - Acquisition des données). Ces données seront ensuite analysées pour y détecter les

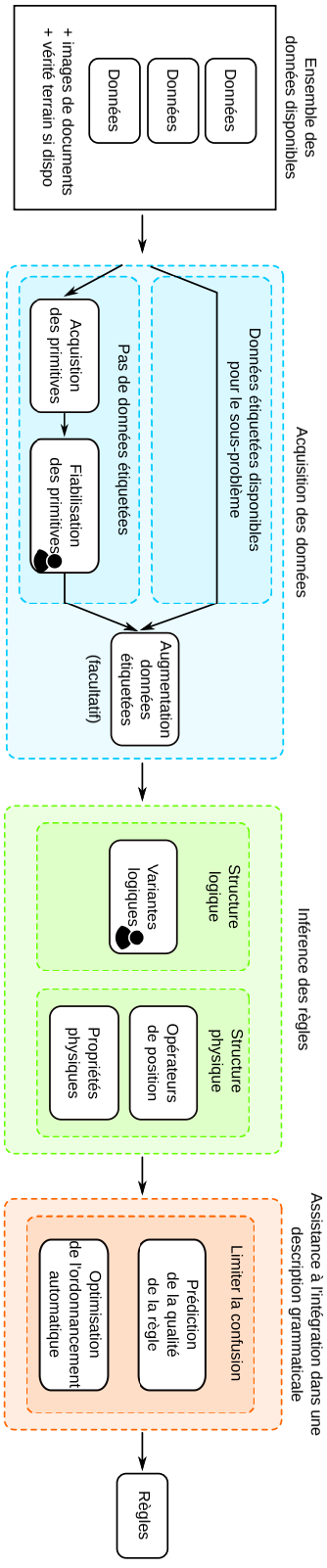


FIG. 4.1 – Méthodologie globale de la méthode EWO

redondances et les structures. Deux cas se présentent alors selon qu'une vérité terrain annotée des documents est disponible ou non. Si une vérité terrain est disponible, ce qui est le cas le plus simple, alors elle est directement utilisable et est considérée comme fiable. Dans le deuxième cas, il n'y a pas de vérité terrain disponible. Il faut alors constituer un ensemble de données qui pourra être utilisé à la place de la vérité terrain. Quel que soit le mode d'acquisition des données, l'inférence des règles et l'assistance à l'intégration dans une description grammaticale sont ensuite effectuées de la même manière.

Dans cette section, nous présentons les mécanismes mis en place avec ou sans vérité terrain pour préparer les données dans l'objectif d'inférer des règles pour décrire un système de reconnaissance de la structure de documents.

4.1.1 Données utiles

Comme pour toute méthode d'analyse de données, il faut fournir en entrée de la méthode EWO des données adaptées au problème que nous cherchons à résoudre. Ces données proviennent d'un échantillon de documents représentatifs des documents finaux à traiter. La *représentativité* des documents d'apprentissage est *indispensable* pour la constitution d'une description grammaticale adaptée. Notre méthode est adaptée pour l'analyse de grands volumes de documents. L'échantillon fourni par l'utilisateur peut donc être grand s'il dispose de beaucoup de documents.

La méthode EWO est une méthode adaptée à l'analyse des données en largeur. Elle n'est pas adaptée à leur analyse en profondeur. Nous devons donc fournir des éléments correspondants à la granularité à analyser. Si l'utilisateur veut apprendre une règle grammaticale permettant de décrire les séparateurs horizontaux d'articles dans des archives de presse, il doit fournir des éléments horizontaux parmi lesquels se trouvent des séparateurs. Nous effectuons alors une analyse *en largeur* : pour un niveau d'élément nous recherchons les différentes variations existantes. Par contre, l'analyse *en profondeur* n'est pas possible dans la méthode EWO. L'utilisateur ne peut pas fournir des composantes connexes en cherchant à obtenir une règle grammaticale sur les séparateurs horizontaux.

Les données fournies doivent donc être adaptées au sous-problème à résoudre. Les données que nous fournissons doivent également nous permettre de résoudre les deux tâches de la reconnaissance que sont la segmentation et l'étiquetage. Pour résoudre la tâche d'étiquetage, il faut que nous possédions des exemples des éléments à étiqueter dans nos données d'apprentissage. Par contre, l'analyse seule de ces éléments va masquer les problèmes de segmentation. Nous avons donc besoin de leur associer des éléments correspondants aux terminaux de la grammaire, qui vont être effectivement utilisés pour segmenter les éléments dans le processus de reconnaissance.

4.1.2 Cas où la vérité terrain est disponible

Lorsqu'une vérité terrain annotée est disponible pour les documents à reconnaître, elle est constituée au minimum pour chaque élément de sa boite englobante ainsi que

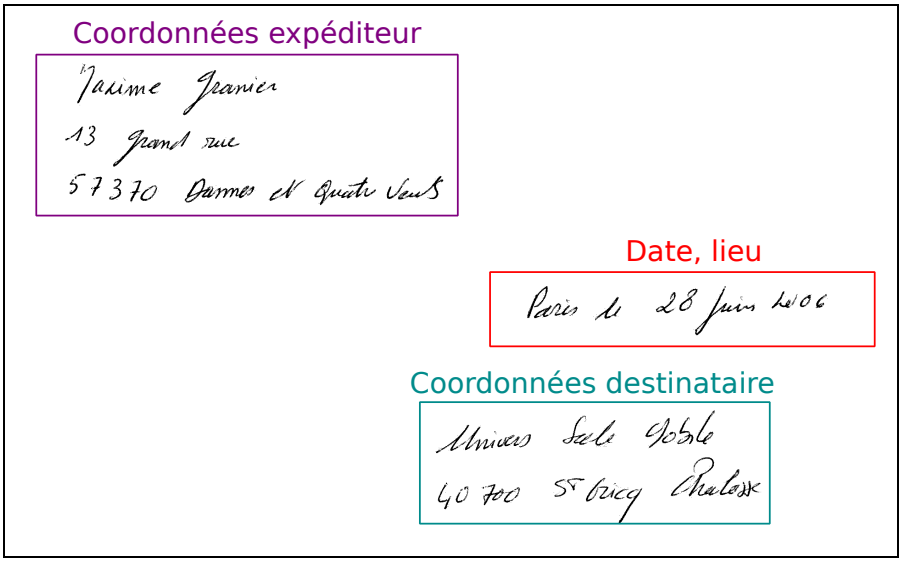
de son étiquette logique (figure 4.2(a)). Cet étiquetage peut être complété par de nombreuses informations comme la transcription du texte de l'élément, des regroupements d'élément comme par exemple une signature et le nom qui lui est associé, l'ordre de lecture entre différents éléments, etc. Dans les faits, en raison du coût de l'étiquetage, il est assez rare d'avoir une vérité terrain aussi précise.

La vérité terrain ainsi constituée représente alors les éléments logiques déjà segmentés. Lors de la reconnaissance de la structure d'un document, c'est à la fois la segmentation et l'étiquetage logique des éléments qui sont réalisés. Le niveau de granularité contenu dans la vérité terrain ne permet alors pas, tel quel, d'inférer des règles qui vont permettre de segmenter correctement les éléments. Par exemple, nous ne disposons pas en général de la liste des pixels ou des composantes connexes qui constituent un élément. Afin de permettre l'apprentissage des propriétés physiques qui permettront d'effectuer la segmentation des éléments, la méthode EWO permet de procéder à une *augmentation de la vérité terrain.*

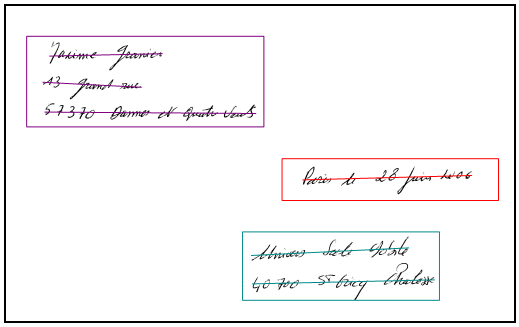
4.1.3 Augmentation de la vérité terrain

Pour procéder à l'augmentation de la vérité terrain, les terminaux de la grammaire sont ajoutés à la vérité terrain. En effet, ce sont les terminaux de la grammaire qui vont effectivement être utilisés lors de l'analyse pour procéder à la segmentation. N'importe quelle primitive d'analyse peut être ajoutée à la vérité terrain si sa position dans le document est connue. Elles peuvent être directement extraites de l'image comme c'est le cas des composantes connexes et des segments de droite, ou être le résultat d'un précédent système de reconnaissance comme des lignes de texte ou les résultats d'un OCR. Dans la figure 4.2, nous pouvons observer un exemple où une vérité terrain contenant seulement les boites englobantes avec les étiquettes logiques des éléments (figure 4.2(a)) est augmentée avec les lignes de texte extraites dans la page (figure 4.2(b)) ainsi que les composantes connexes (figure 4.2(c)).

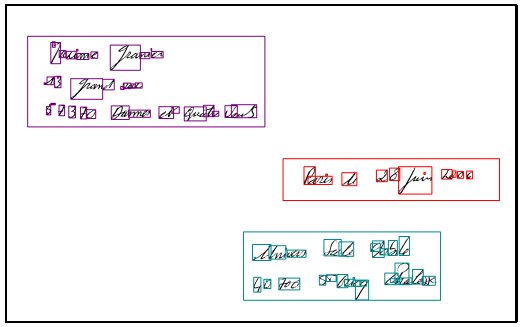
En utilisant la position des primitives d'analyse, nous affectons automatiquement à chaque zone annotée de la vérité terrain les primitives qui sont contenues dans la zone. Cette augmentation de la vérité terrain permet d'inférer par la suite la structure physique des éléments beaucoup plus précisément. Cependant, ces nouvelles connaissances ont un coût : elles ne sont pas complètement fiables puisqu'elles ont été produites automatiquement sans vérification, contrairement à la vérité terrain produite manuellement. Par exemple dans le cas de lignes de texte détectées dans un document, certaines lignes peuvent être détectées de manière incorrecte comme montré dans l'exemple de la figure 4.3. Dans notre méthode, nous ne proposons pas de technique pour supprimer ces éléments erronés de la vérité terrain augmentée. C'est la robustesse de l'analyse, notamment via la détection des valeurs extrêmes, qui permettra d'obtenir une bonne inférence des règles. Cette robustesse est également indispensable pour gérer les corpus où il n'y a pas de vérité terrain disponible (cf. section 4.1.4).



(a) Éléments logiques étiquetés dans un courrier manuscrit (vérité terrain existante)



(b) Lignes affectées aux éléments logiques



(c) Composantes connexes affectées aux éléments logiques

FIG. 4.2 – Exemple d'augmentation de la vérité terrain pour des courriers manuscrits en français

4.1.4 Cas sans vérité terrain

Une méthode efficace et utilisable sur n'importe quel jeu de données doit être capable de fonctionner sans vérité terrain annotée en entrée. Il est actuellement très difficile de faire de l'apprentissage automatique sans vérité terrain. Nous n'avons pas trouvé de méthode d'analyse de la structure de documents capable d'apprendre sans vérité terrain (cf. chapitre 1). Pour rendre possible l'inférence de règles dans le cas où il n'existe pas de vérité terrain annotée pour les documents à reconnaître, nous proposons de *construire une pseudo vérité terrain*. Les données de la pseudo vérité terrain sont ensuite utilisées dans le reste de la méthode EWO exactement comme des données issues d'une vérité terrain annotée.

Pour construire une pseudo vérité terrain, nous proposons d'utiliser de grands volumes de données parmi lesquelles nous recherchons automatiquement des répétitions,

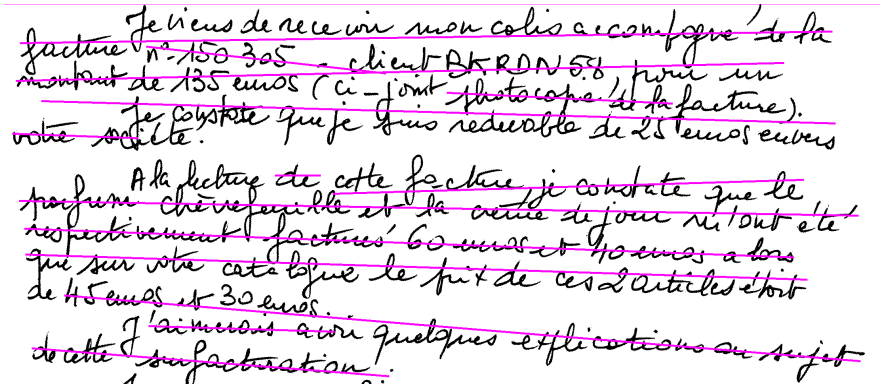


FIG. 4.3 – Exemple d'erreur sur les lignes détectées et affectées à un élément logique « corps de texte » dans un courrier manuscrit

des structures redondantes. Cette recherche de répétitions est effectuée sur des terminaux de la grammaire. Ces primitives d'analyse peuvent être de types variés. Elles peuvent être porteuses d'une sémantique dans le cas par exemple de résultats d'un OCR, mots-clés détectés avec du *word spotting*, etc. Elles peuvent également ne pas avoir de sémantique associée dans le cas d'éléments extraits de l'image comme les composantes connexes de l'image ou les segments de droite par exemple. Les primitives à utiliser dépendent des éléments logiques pour lesquels nous cherchons à écrire une règle de grammaire comme nous l'avons présenté dans la section 4.1.1.

Les primitives d'analyse ainsi extraites fournissent un grand nombre de données à analyser. Cependant ces données, contrairement à une vérité terrain annotée, ne peuvent être utilisées directement. En effet, elles ne sont pas fiables, de nombreuses données ne correspondent pas aux éléments que l'utilisateur souhaite analyser pour procéder à l'inférence de règles. L'acquisition des données dans le cas sans vérité terrain se décompose alors en deux étapes comme nous pouvons le voir sur la figure 4.4. D'abord, nous procédons à une extraction des primitives puis nous fiabilisons ces primitives. Les primitives fiabilisées constituent alors la pseudo vérité terrain.

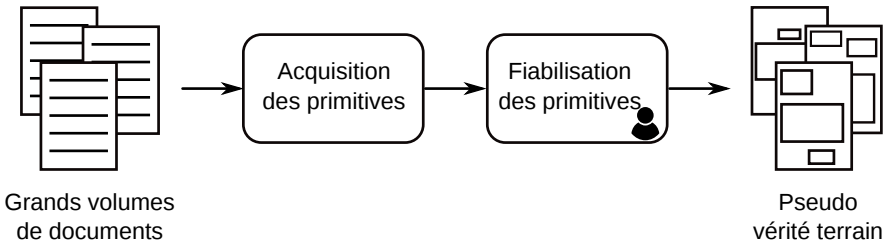


FIG. 4.4 – Processus d'acquisition des données sans vérité terrain (détail de Fig. 4.1 - Acquisition des données)

4.1.4.1 Extraction des primitives

L'extraction des primitives est une étape permettant de détecter des éléments intéressants mais introduisant également beaucoup de bruit dans l'analyse. Les primitives sont sélectionnées par l'utilisateur en fonction des règles qu'il veut inférer (cf. section 4.1.1). L'extraction des primitives peut être faite à l'aide d'un traitement global du document comme avec un OCR. Elle peut également être réalisée avec le système de reconnaissance de documents en cours de construction.

Par exemple, nous cherchons à détecter des mots clés dans un document d'archives d'état civil avec des modèles de points d'intérêt. La figure 4.5 nous montre les résultats obtenus dans plusieurs documents pour la détection du mot clé « comparecen ». La figure 4.5(a) nous montre un document où une seule instance du mot clé est trouvée et correspond effectivement au mot clé « comparecen » recherché. C'est la cas idéal où aucun bruit n'est introduit dans l'analyse. Les figures 4.5(b) et 4.5(c) montrent quant à elles deux exemples de document présentant plus d'un mot clé détecté comme « comparecen ». Certaines des occurrences ont donc été détectées à tort. C'est le cas le plus courant obtenu après extraction des primitives. Dans ce cas, nous ne pouvons pas utiliser directement les éléments obtenus grâce à l'étape d'extraction de primitives pour constituer la pseudo vérité terrain. Il est également possible de ne rien trouver dans le document alors qu'un élément est effectivement présent comme c'est le cas dans la figure 4.5(d). Dans ce cas la pseudo vérité terrain est partielle pour le document.

4.1.4.2 Fiabilisation des primitives

Pour construire la pseudo vérité terrain, nous ajoutons une étape supplémentaire après l'extraction de primitives : la fiabilisation des primitives. La fiabilisation des primitives consiste à supprimer de l'analyse le bruit, c'est-à-dire les occurrences qui ne correspondent pas aux éléments pour lesquels nous souhaitons faire l'inférence.

La fiabilisation des primitives permet de constituer une pseudo vérité terrain sur laquelle la méthode EWO s'appuie pour poursuivre l'analyse. Elle se base sur la combinaison d'un clustering automatique des données et d'une interaction avec l'utilisateur. Le clustering des données permet de mettre en avant des ensembles homogènes de données. Cette homogénéité peut concerner la combinaison de différentes propriétés disponibles sur les éléments comme la taille des éléments, leur position ou tout autre propriété connue sur les données (taille de la police du texte, espacement vertical ou horizontal des éléments, etc.). Le détail de la méthode de clustering employée est présenté dans le chapitre 5.

Les clusters sont ensuite présentés à l'utilisateur sous la forme de quelques exemples représentatifs du cluster. Une mesure d'homogénéité est également fourni à l'utilisateur afin de s'assurer que les exemples sont effectivement bien représentatifs des éléments du cluster. Sur la base de ces exemples représentatifs, l'utilisateur prend la décision de *conserver ou non toutes les occurrences du cluster*. L'interaction avec l'utilisateur est détaillée dans le chapitre 6.

Cette approche permet d'obtenir la pseudo vérité terrain nécessaire à l'inférence des règles à moindre coût. En effet, au lieu d'avoir à regarder toutes les primitives pour dé-

terminer si elles sont conservées ou non pour l'analyse, l'utilisateur regarde un nombre très limité de primitives, quelques exemples seulement par cluster. Le nombre de documents pour lesquels une pseudo vérité terrain peut être obtenue est donc beaucoup plus important que dans le cas d'une vérité terrain obtenue manuellement. L'inconvénient majeur de cette approche est que la pseudo vérité terrain construite est *moins fiable qu'une vérité terrain annotée manuellement*. Une comparaison du coût de construction de la pseudo vérité terrain à une vérité terrain annotée manuellement est présentée dans le chapitre 10. Dans l'exemple présenté, le coût de construction est 200 fois moins élevé pour la pseudo vérité terrain.

4.1.4.3 Pseudo vérité terrain

Une fois l'extraction des primitives et la fiabilisation terminée, l'ensemble des primitives fiabilisées obtenues constitue la pseudo vérité terrain. Nous pouvons également procéder à une augmentation des données avec les terminaux de la grammaire comme présentée dans la section 4.1.3.

Cette pseudo vérité terrain est utilisée dans le reste de l'analyse de la même manière que la vérité terrain. C'est la robustesse de l'analyse qui nous permet de pallier la moins bonne fiabilité de la pseudo vérité terrain en comparaison à une vérité terrain manuellement annotée.

La construction de la pseudo vérité terrain, comme la construction de la description grammaticale complète, se fait de manière progressive (cf. section 3.1.2). À chaque fois que l'utilisateur veut résoudre un sous-problème il peut utiliser les primitives déjà contenues dans la pseudo vérité terrain. S'il a besoin de nouvelles primitives, celles-ci vont s'ajouter à l'ensemble des primitives déjà à sa disposition, après avoir été fiabilisées (cf. figure 3.3).

4.2 Inférence des règles

Dans la construction d'un système de reconnaissance de la structure de documents, les structures logique et physique des documents doivent être décrites. La méthode EWO permet à l'utilisateur d'inférer des éléments pour ces deux structures. Nous détaillons dans cette partie les méthodes mises en place pour l'inférence des règles (Fig. 4.1 - Inférence des règles).

L'inférence complète d'une règle est construite de la manière suivante : inférence logique (cf. 4.2.1), puis inférence des positionnement (cf. 4.2.2.1) et inférence des propriétés physiques (cf. 3.3.2). Les règles sont construites progressivement. Les exemples donnés dans cette section vont détailler chacune des étapes menant à l'inférence complète de la règle.

4.2.1 Apprentissage des variations logiques

L'apprentissage de la structure logique d'un document consiste en l'apprentissage du rôle et de la nature de chaque élément ainsi que de l'ensemble des liens hiérarchiques

et/ou logiques qui les lient les uns aux autres. La structure logique du document décrit le contenu intellectuel du document. La structure logique peut être complexe à déterminer automatiquement car elle exprime souvent des connaissances de l'utilisateur sur la constitution des documents.

Nous présentons ici un exemple de description logique. Nous souhaitons décrire la structure de courriers manuscrits comme celui présenté dans la figure 4.6. La description logique décrit qu'un tel courrier est composé de 7 éléments : les coordonnées expéditeur, les coordonnées destinataire, la date et le lieu, l'ouverture, le corps de texte, la signature et le post-scriptum. La description d'un courrier manuscrit est alors la suivante :

```
courrier ::
   coordonneesExpediteur &&
   coordonneesDestinataire &&
   dateLieu &&
   ouverture &&
   corpsDeTexte &&
   signature &&
   ps.
```

L'une des tâches de la description de la structure logique est la description des variations logiques des règles. À une étiquette logique peuvent être associés différents éléments dont la description va varier, en terme de positionnement et de propriétés physiques par exemple. Chacune de ces variantes sera décrite par une variation de la règle logique. La détection des variations logiques d'une règle nécessite donc une connaissance exhaustive du corpus. Nous proposons de détecter les variations logiques grâce à un clustering non supervisé des données effectué dans EWO. La méthode employée est non supervisée puisque nous désirons détecter une structure non connue dans la vérité terrain si elle est présente. La méthode de clustering choisie est détaillée ultérieurement dans le chapitre 5.

Chaque groupe détecté par l'algorithme de clustering peut permettre de définir une variation logique de la règle que nous cherchons à décrire. La décision d'utiliser le groupe pour définir une variation logique est prise par l'utilisateur. L'utilisateur visualise quelques exemples d'éléments du cluster pour valider sa pertinence et ensuite le nommer de manière pertinente. Cette étape de validation et de nommage ne peut pas être réalisée automatiquement mais nécessite au contraire l'intervention de l'utilisateur qui va apporter du *sens aux données*. L'interaction est détaillée dans le chapire 6.

Dans le cas des courriers manuscrits, nous devons décrire l'élément logique « coordonnées expéditeur ». Nous recherchons donc grâce à la méthode EWO si différentes variantes logiques existent pour cet élément. Nous nous basons pour cela sur les propriétés contenues dans la vérité terrain. Nous utilisons les dimensions (hauteur et largeur) de la boite englobante des éléments « coordonnées expéditeur ». Deux groupes sont détectés grâce à la méthode de clustering et présentés à l'utilisateur à l'aide d'un nombre restreint d'exemples représentatifs. Après observation des exemples, l'utilisateur peut étiqueter le cluster 1 (figure 4.7) comme représentant des « références clients » des co-

ordonnées expéditeur tandis que le cluster 2 (figure 4.8) représentent des « coordonnées postales ». Chacune de ces variations logiques pourra ensuite être décrite séparément.

Dans la description grammaticale, nous avons alors pour la règle `coordonneesExpediteur` deux variations que nous allons enrichir avec l'inférence de la structure physique proposée par la méthode EWO :

```
% Variation logique 1 correspondant aux références clients
coordonneesExpediteur ::=
    referenceClient.

% Variation logique 2 correspondant aux adresses postales
coordonneesExpediteur ::=
    coordonneesPostales.
```

Dans un courrier, nous pouvons avoir un bloc expéditeur contenant l'adresse postale, un bloc expéditeur contenant les références client ou les deux.

Cas particulier des valeurs extrêmes

Le but de notre méthode est de faciliter la description d'un nouveau type de documents en donnant une vision exhaustive des données à l'utilisateur. Dans notre analyse, il nous faut alors détecter les valeurs extrêmes. Les valeurs extrêmes peuvent provenir soit d'une valeur atypique soit d'une erreur dans la vérité terrain ou la pseudo vérité terrain. Dans les deux cas, ce sont des éléments importants qu'il nous faut détecter et pouvoir présenter à l'utilisateur.

Les valeurs extrêmes sont considérées dans notre analyse comme un groupe particulier de données à traiter. Ces observations ne sont pas introduites dans l'analyse globale en raison de leur fort impact possible sur la qualité de l'analyse. En effet, de nombreux indicateurs statistiques et méthodes sont peu robustes à la présence de valeurs extrêmes. Pour la détection des outliers, nous proposons d'utiliser une détection s'appuyant sur l'écart-type. Nous considérons alors que les valeurs qui n'appartiennent pas à l'intervalle $[moy - t \times SD; moy + t \times SD]$ sont des valeurs extrêmes. Pour le coefficient t, $t = 3$ si le nombre d'observations est supérieur à 80, $t = 2,5$ sinon. Cette méthode de détection des valeurs extrêmes est simple à utiliser et est bien connue pour ne pas détecter toutes les valeurs extrêmes. Cela limite donc le risque de supprimer des observations intéressantes pour l'analyse.

L'utilisateur a alors accès à ces valeurs extrêmes. Il peut ainsi facilement choisir d'introduire les cas rares dans la description grammaticale. Il peut également détecter des erreurs dans la vérité terrain ou la pseudo vérité terrain. Dans le chapitre 9.3, nous présentons comment notre méthode nous a permis de détecter de nombreuses erreurs dans la vérité terrain du corpus de la compétition Maurdor.

Dans cette section, nous avons montré que la méthode EWO permet à l'utilisateur d'avoir une vue à la fois exhaustive et synthétique sur les données. La détection automatique des cas généraux et des cas rares présents dans le corpus de documents permet

l'inférence de la structure logique des documents. Nous détaillons dans la section suivante la suite de l'inférence de la règle avec l'apprentissage de la structure physique.

4.2.2 Apprentissage de la structure physique

La structure physique permet de décrire comment le système peut reconnaitre les éléments de la structure logique. L'apprentissage de la structure physique d'un document consiste en l'apprentissage du positionnement des éléments, leur agencement les uns par rapport aux autres ainsi que l'ensemble de leurs propriétés physiques. La structure physique du document décrit l'apparence visuelle du document. La description de la structure physique nécessite notamment de fixer la valeur de nombreux paramètres.

L'apprentissage de la structure physique permet d'avoir une vision exhaustive des données pour la détermination des paramètres ce qui est très difficile dans le cadre d'une description effectuée manuellement. Les cas rares sont notamment difficiles à détecter dans une approche manuelle. En effet, dans une approche manuelle, nous sélectionnons un petit échantillon de documents pour faire l'apprentissage (d'une dizaine à quelques dizaines de documents). La probabilité d'observer les cas rares est alors faible. De plus, si nous les observons, nous risquons de sur-estimer leur importance dans le corpus. Une approche par apprentissage automatique permet d'avoir une vision à la fois exhaustive et précise des documents.

Dans cette section, nous présentons la méthode mise en place pour l'apprentissage de la structure physique. Nous décrivons d'abord le fonctionnement général du positionnement des éléments dans une description grammaticale. Puis, nous détaillons la méthode proposée pour apprendre automatiquement les positionnements. Enfin, nous présentons la méthode pour l'apprentissage des propriétés physiques des éléments.

4.2.2.1 Positionnement des éléments

Pour l'analyse de documents structurés, il est utile de modéliser et évaluer la position des différents composants entre eux et au sein de la page. Le positionnement de ces composants est utilisé durant l'analyse de la structure pour définir l'orientation et l'ordre d'analyse. La qualité des informations spatiales affecte donc les performances de l'analyse. Les frontières des zones d'analyse ne sont pas utilisées pour identifier la classe des éléments mais la bonne définition du positionnement des éléments permet un bon rappel tout en limitant la combinatoire.

Les zones 2D qui sont nécessaires pour guider l'analyse sont difficiles à définir. Dans les méthodes syntaxiques, elles sont souvent définies manuellement ce qui présente des inconvénients. Un opérateur humain doit observer les images de documents pour définir une zone appropriée d'après ces exemples. De plus, les cas rares ne sont pas observés et l'analyse des erreurs doit alors être faite afin d'ajuster les zones définies. L'ajustement des paramètres est chronophage et n'est pas une tâche facile.

La description d'un opérateur de position consiste en la définition de 6 paramètres différents : les coordonnées du point haut gauche (X_a, Y_a), les coordonnées du point bas droit (X_b, Y_b) et les coordonnées du point d'ancrage (X_p, Y_p). Le point d'ancrage permet

d'indiquer dans quel ordre analyser les différents éléments de la zone. Les différents éléments contenus dans la zone sont ainsi analysés du plus proche du point d'ancrage au plus éloigné.

Dans la méthode EWO, chaque composant d'une page est représenté sous la forme de sa boîte englobante. C'est une représentation qui est souvent utilisée dans les méthodes existantes de reconnaissance de la structure de documents. Nous tenons alors compte de deux point pour représenter l'objet : les coordonnées du coin haut gauche et les coordonnées du coin bas droit, comme illustré dans la figure 4.9.

Plusieurs variantes physiques peuvent correspondre à un opérateur de position. Par exemple, dans la description de lettres manuscrites, une catégorie peut regrouper les post-scriptum et les pièces jointes (appelée « PS/PJ »). La figure 4.10(b) représente une vue normalisée de toutes les occurrences de « PS/PJ »présentes dans l'ensemble d'apprentissage. Deux groupes distincts existent. Si nous ne tenons pas compte de ces groupes pour le calcul de l'opérateur de position, comme c'est fait dans la figure 4.10(c), nous obtenons alors pratiquement toute la page comme zone d'intérêt. Au contraire, quand les deux groupes sont pris en compte et une variante produite pour chaque groupe (cf. figure 4.10(d)), alors l'opérateur de position obtenu est bien plus précis.

4.2.2.2 Apprentissage automatique des positionnements

L'apprentissage des opérateurs de position est découpé en plusieurs étapes d'analyse automatique :

1. L'utilisateur demande explicitement l'opérateur de position P d'un élément ;
2. Calcul de P grâce à une détection automatique des groupes : une ou plusieurs variantes physiques sont produites. Les frontières de chaque variante sont inférées ;
3. Détermination du point d'ancrage pour chaque variante ;
4. Détermination de l'ordre d'analyse des variantes ;
5. Introduction de l'opérateur de position P dans la structure logique définie par l'utilisateur. L'utilisateur peut revenir à la première étape pour un autre élément.

Comme c'est fréquemment le cas dans la littérature, nous utilisons deux modes de positionnement dans un document : positionnement absolu et positionnement relatif. Un positionnement absolu consiste à décrire la position de chaque élément par rapport à un élément fixe, ici la page entière, indépendamment de la position des autres composants. Cette approche est bien adaptée pour des éléments dont la position est stable dans la page. Le positionnement relatif est plus associé à notre perception des similarités dans les arrangements spatiaux. Par exemple, dans le cas des courriers manuscrits, nous sommes capables d'exprimer des relations telles que « la date/le lieu sont au dessus des coordonnées destinataires ». Le positionnement relatif s'intéresse à la position des éléments entre eux et non pas à leur position absolue dans la page.

a - Positionnement absolu Le positionnement absolu consiste à décrire la position d'un élément dans la page entière indépendamment de la position des autres éléments.

Cette approche est particulièrement utile pour démarrer l'analyse, lorsque aucun élément n'a été trouvé.

Lorsqu'un positionnement absolu d'un élément est requis, les données contenues dans les données d'apprentissage sont alors immédiatement utilisables. Nous utilisons les documents où l'élément recherché est effectivement présent pour apprendre le positionnement. Si plusieurs occurrences de l'élément logique sont présentes dans le document, chaque instance est utilisée et considérée indépendamment des autres. Les documents où l'élément logique est absent ne sont pas utilisés mais leur effectif ainsi que les exemples concernés sont mis à la disposition de l'utilisateur pour d'autres analyses si nécessaire.

b - Positionnement relatif Le positionnement relatif consiste en la description de la position d'un élément en fonction d'un autre. Nous appelons *relatif* l'élément que nous cherchons à positionner en fonction d'un autre qui est lui appelé *référence*. Ces données doivent d'abord être filtrées puis nous procédons à une translation du système de coordonnées.

Étape 1 - Filtrage des données Lorsqu'un positionnement relatif d'un élément logique est requis, toutes les données ne peuvent pas être utilisées. En effet, l'analyse du positionnement est faite au sein d'une paire référence-relatif. Or, ces paires ne sont pas disponibles dans tous les documents. Le tableau 4.1 présente les différents cas possibles. Lorsqu'il n'y a pas de relatif présent dans le document celui-ci n'est pas retenu dans l'analyse. Si un ou plusieurs relatifs sont présents mais pas de référence, nous proposons à l'utilisateur l'inférence d'un opérateur de position absolue. Enfin, nous avons les cas où l'inférence d'un opérateur de position relatif est possible.

	0 référence	1 référence	Plusieurs références
0 relatif	∅		
1 ou plusieurs relatifs	opérateur de position absolue	OK	OK si paires explicitées dans la vérité terrain

TAB. 4.1 – Description des cas possibles pour un document pour la constitution des paires référence-relatif

Si une seule référence est présente dans le document, il n'y a pas d'ambiguïté pour les constitutions des paires. Soit $(rel_1, \ldots, rel_n)$ l'ensemble des éléments relatifs d'un document et ref l'unique élément référence du document, alors l'ensemble des paires étudié est :

$$\{\forall i \in \{i, \ldots, n\}, (rel_i, ref)\}$$

Si plusieurs références sont présentes dans le document, la constitution des paires référence-relatif n'est possible que si les paires sont explicitement renseignées dans la vérité terrain. On utilise alors exclusivement les paires de la vérité terrain dans l'analyse.

Étape 2 - Prétraitement Lorsque les données utilisables ont été sélectionnées, nous réduisons la référence à son coin haut gauche (X_d, Y_d). Une translation du système de coordonnées est effectuée. Les coordonnées des éléments relatifs sont modifiées pour que l'origine se situe à ce point (X_d, Y_d). En utilisant ces nouvelles coordonnées pour la boite englobante de l'élément relatif, nous pouvons alors procéder comme dans le cas du positionnement absolu.

Soit (X, Y) les coordonnées d'un point P dans le système de coordonnées de la page et (X_d, Y_d) les coordonnées du point relatif servant de nouvelle origine du repère. Alors dans le nouveau repère les coordonnées (x, y) du point P sont :

$$x = X - X_d$$
$$y = Y - Y_d$$

c - Détection automatique des variantes Comme nous l'avons présenté, il est nécessaire d'avoir des opérateurs de position pouvant être composés de plusieurs zones distinctes. Afin de conserver une méthode d'apprentissage automatique des opérateurs de position, nous devons être capable de détecter automatiquement si plusieurs opérateurs différents doivent être inférés, un pour chaque zones. Notre méthode ne peut pas s'appuyer sur une prédétermination manuelle du nombre de zones d'intérêt. Le nombres de groupes est déterminé en s'appuyant sur la détection des maxima locaux dans un histogramme. Le nombre total de maxima locaux correspond au nombre de zones différentes dans la page. Dans l'exemple de la figure 4.11, deux maxima locaux sont détectés amenant à construction de deux groupes différents. Cette détection est faite via une méthode de recherche séquentielle présentée par Lerddaradsamee [LJ12]. Cette méthode nous permet de détecter des zones spatiales disjointes dans la page contrairement aux tests menés avec la méthode de clustering présentée dans le chapitre 5.

Dans la figure 4.12, nous représentons les résultats obtenus pour la détection automatique des groupes pour le positionnement absolu des éléments « date, lieu » dans un corpus de courriers manuscrits en français. Comme nous pouvons l'observer dans la figure 4.12(b), deux groupes distincts sont détectés, l'un positionné en haut à droite de la page, l'autre positionné en dessous du premier, également à droite de la page.

L'analyse se poursuit alors pour chaque groupe : nous détectons les frontières des zones ainsi que le point de vue à adopter pour le parcours des éléments de la zone.

d - Méthode basée sur la densité pour le calcul des frontières de zones Lorsque les groupes ont été détectés et les valeurs extrêmes supprimées, les frontières de chaque zone sont alors définies. Si nous prenons la zone englobant toutes les boites englobantes des éléments (figure 4.13(a)), la zone risque d'être trop large et d'introduire des erreurs dans l'analyse et d'augmenter la combinatoire. Nous introduisons la notion de densité pour définir les frontières de la zone de l'opérateur de position afin de maitriser la combinatoire tout en maximisant le rappel (figure 4.13(b)).

Pour utiliser la notion de densité pour la définition des frontières de la zone, nous nous appuyons sur une division de l'espace à l'aide d'une grille. Dans cette grille, la

largeur d'une case représente un pourcent de la largeur de la page et la hauteur d'une case représente un pourcent de la hauteur de la page (figure 4.14(a)). Pour chaque case, nous comptons le nombre de boites englobantes ayant une intersection non vide avec cette cellule (figure 4.14(b)). Cette information est utilisée comme une approximation de la densité. Puis, si une case contient moins d'éléments qu'un seuil fixé, alors nous fixons l'effectif de la case à zéro (figure 4.14(c)). Toutes les cases qui ont une densité non nulle sont ensuite utilisées pour définir l'opérateur de position. Les zones sont des unions d'unité non vides de l'espace. La forme finale de la zone peut alors être variée. Nous présentons dans la figure 4.14(d) le résultat obtenu pour une zone rectangulaire définie par ses coins haut gauche et bas droit.

e - Comment gérer la confusion ? Une fois les zones déterminées, il nous faut déterminer les points d'ancrage pour chacune des zones ainsi que l'ordre de parcours des zones. Même si la zone contient effectivement l'élément recherché, elle peut également contenir de nombreux autres éléments qui apportent de la confusion dans notre analyse. En effet, ces autres éléments peuvent être sélectionnés à la place de l'élément recherché.

Une manière de gérer ces risques de confusion est de définir des règles et des conditions permettant de déterminer quel élément est celui que nous recherchons parmi les autres. Cependant, les propriétés peuvent ne pas différer suffisamment pour que nous puissions choisir facilement le bon élément. Une autre manière de gérer ces risques de confusion consiste à créer des opérateurs de position qui les minimisent en utilisant l'ordre de parcours des zones ainsi que le point de vue. Dans ce but, nous introduisons dans l'analyse l'*indice de confusion* que nous cherchons à minimiser.

L'indice de confusion L'indice de confusion est un indicateur qui permet de savoir combien d'éléments d'un autre type que celui recherché sont présents dans la zone. Il est défini comme :

$$confusion(l, zone) = \sum_{i=1}^{n} l[i] \cap zone \neq \varnothing$$

l contient tous les éléments à l'exception de ceux du type recherché. Dans le cas de positionnements relatifs, nous devons exclure à la fois les références et les relatifs pour obtenir la confusion.

f - Ordre des zones Quand plus d'une zone a été trouvée, nous analysons les éléments en parcourant chacune des zones tant que nous n'avons pas trouvé l'élément recherché. Pour minimiser les erreurs, nous devons commencer par les zones qui présentent l'indice de confusion le plus petit. En procédant de cette manière, nous limitons les possibilités de choisir un élément d'un autre type que celui recherché. Nous calculons donc l'indice de confusion dans chacune des zones trouvées pour l'opérateur de position. Les zones sont ensuite analysées par valeur croissante d'indice de confusion.

Par exemple, nous détectons deux zones différentes pour l'élément « date, lieu » dans un courrier manuscrit en français. Dans la figure 4.15, nous avons représenté ces

deux zones sur un même courrier. Une zone se trouve complètement en haut à droite du document tandis que la deuxième zone se trouve également dans la partie supérieure droite du document, sous la première. L'indice de confusion nous indique d'analyser la première zone d'abord car peu de confusion sont possibles avec d'autres éléments. Au contraire, la deuxième zone contient de nombreux autres éléments pouvant être confondus avec l'élément « date, lieu », et notamment l'« ouverture » du courrier ainsi que les « coordonnées destinataire ».

g - Choisir le point d'ancrage Lorsque nous cherchons à reconnaître un élément dans un document, nous ne devons pas seulement savoir dans quelle partie de l'image le rechercher. Nous devons également définir dans quel ordre nous allons parcourir les éléments de cette zone, c'est-à-dire déterminer le point de vue utilisé. Les éléments seront alors parcourus dans la zone du plus proche au plus éloigné. Nous choisissons ici de déterminer le meilleur point de vue automatiquement, à l'aide de l'indice de confusion afin de minimiser les risques d'erreurs. Pour définir le meilleur point de vue, nous calculons l'indice de confusion pour un ensemble prédéterminé de points de vue. Nous choisissons alors le point de vue qui minimise l'indice de confusion.

Dans l'exemple de « date, lieu » dans les lettres manuscrites, la figure 4.16(a) présente une représentation synthétique de toutes les boîtes englobantes des éléments pouvant apporter de la confusion. Pour augmenter la lisibilité de la figure, l'exemple se base sur un sous-ensemble de 30 courriers. Dans l'expérimentation réelle (cf. chapitre 8), nous avons utilisé l'ensemble des 900 courriers disponibles. La figure 4.16(b) représente elle les éléments « date, lieu » que nous recherchons pour ces 30 courriers. Le point de vue est choisi afin de maximiser les chances de trouver l'élément recherché tout en minimisant les risques de sélectionner un autre élément de la page. Dans ce but, nous choisissons ici de parcourir les éléments de la zone du haut vers le bas.

La méthode EWO permet alors de générer les deux variantes suivantes pour l'opérateur de position, après une interaction avec l'utilisateur limitée uniquement à nommer les opérateurs de position inférés :

```
coinHautDroit :-
   Xp = 0,
   Yp = -100,
   Xa = 43% largeur de la page
   Ya = 0,
   Xb = 100% largeur de la page,
   Yb = 16% hauteur de la page.

hautDroit :-
   Xp = 0,
   Yp = -100,
   Xa = 30% largeur de la page
   Ya = 13% hauteur de la page,
```

```
Xb = 100% largeur de la page,
Yb = 45% hauteur de la page.
```

4.2.2.3 Propriétés physiques des éléments

Dans une description grammaticale, les propriétés physiques des éléments que l'on souhaite reconnaitre doivent être décrites pour pouvoir correctement segmenter l'image de document. De nombreux paramètres sur les propriétés physiques des éléments doivent être déterminés. La détermination manuelle des paramètres est longue et difficile. C'est pourquoi nous intégrons dans EWO une détermination automatique des paramètres définissant les propriétés physiques des éléments.

Les propriétés physiques que nous recherchons sont les propriétés propres des éléments que nous cherchons à décrire. Nous ne sommes pas dans le contexte d'un classifieur où nous recherchons alors au contraire des critères discriminant une classe par rapport aux autres. Dans ce contexte, nous utilisons donc des statistiques descriptives sur les variables à notre disposition pour chaque variation logique de la règle afin d'apprendre les propriétés physiques. La visibilité globale sur les observations et la division des observations en variations logiques homogènes (section 4.2.1) permet la pertinence des propriétés ainsi inférées. Nous utilisons en particulier l'intervalle interquartile.

Lors de l'apprentissage des propriétés physiques des éléments, il est crucial d'utiliser les informations ajoutées grâce à l'augmentation de la vérité terrain et qui sont liées aux primitives de la grammaire. En effet, ce sont les propriétés physiques sur les terminaux de la grammaire qui vont permettre de segmenter correctement le document. La détermination des paramètres physiques se fait en mode interactif : l'utilisateur pose une question, portant sur la valeur d'un paramètre physique, et obtient une réponse de la part du système, qui est la valeur de ce paramètre. L'interaction est présentée de manière détaillée dans le chapitre 6.

Si nous considérons par exemple la description des coordonnées expéditeur dans un courrier manuscrit, nous obtenons la règle grammaticale suivante en utilisant seulement les boites englobantes des éléments :

```
% Variation logique des coordonnées expéditeur correspondant aux
% coordonnées postales
coordonneesPostales ::=
    AT(hautGauche) &&
    2 à 5 lignes &&
    0% <= hauteur <= 32% hauteur de la page &&
    0% <= largeur <= 50% largeur de la page.
```

Cette règle grammaticale ne permet pas une bonne segmentation du bloc coordonnées expéditeur comme nous pouvons l'observer sur l'exemple présenté dans la figure 4.17. La ligne correspondant à l'objet du courrier est comprise dans le bloc coordonnées expéditeur trouvé. Lorsque nous utilisons les lignes de texte, de nombreuses autres propriétés physiques peuvent être inférées. La règle grammaticale complète finalement obtenue est la suivante :

```
coordonneesExpediteur ::=
    AT(hautGauche) &&
    2 à 5 lignes &&
    Variation maximum sur l'interlignes = 115px &&
    Variation maximum sur l'alignement à gauche des lignes = 226px &&
    0% <= hauteur <= 32% hauteur de la page &&
    0% <= largeur <= 50% largeur de la page.
```

L'ajout des propriétés physiques basées sur les primitives d'analyse permet alors la bonne segmentation des blocs de texte.

4.3 Assistance à l'intégration dans une description grammaticale

Lors de la création d'une description grammaticale, l'utilisateur recherche les propriétés propres de chaque type d'éléments. Cette description peut amener à des confusions avec d'autres éléments du document qui ont des propriétés similaires. Lorsqu'une grammaire est définie manuellement, une approche essai-erreur est utilisée pour déterminer les confusions possibles entre les éléments. Cette approche essai-erreur implique que l'utilisateur doit :

- utiliser le système de reconnaissance ;
- utiliser une métrique, si disponible, pour évaluer les résultats.

Ces deux étapes demandent du temps. Cependant, pour obtenir une grammaire entièrement fonctionnelle l'utilisateur doit nécessairement réduire la confusion des différentes règles de grammaire. La réduction de la confusion repose sur deux aspects : la modification des règles de grammaire et la détermination de l'ordre optimale des règles (Fig. 4.1 - Assistance à l'intégration dans une description grammaticale).

4.3.1 Prédiction de la qualité de la règle

Nous chercherons ici à diminuer le temps nécessaire à l'optimisation des règles de grammaire. Dans EWO, nous intégrons un système permettant de simuler le comportement de la description grammaticale en dehors du système de reconnaissance complet afin de diminuer le temps nécessaire à la conception de la description grammaticale. Pour ce faire, les règles grammaticales sont approximées par des requêtes. Les requêtes sont ensuite appliquées sur la vérité terrain annotée ou la pseudo vérité terrain. Ce processus permet de mettre en avant les confusions entre les différents éléments logiques que nous cherchons à détecter. Les règles grammaticales peuvent alors être modifiées afin de minimiser la confusion sans utiliser le système de reconnaissance.

Avec cette méthode, le temps nécessaire pour l'amélioration des règles grammaticales est considérablement diminué. En effet, la méthode EWO produit les résultats d'une requête en quelques secondes contrairement à l'exécution du système de reconnaissance et l'analyse de ces résultats qui peut être beaucoup plus couteux en temps. La

méthode EWO donne la possibilité à l'utilisateur d'interagir en temps réel pour modifier les règles. De plus, l'utilisateur peut facilement adapter la description grammaticale en fonction de ses objectifs (précision, rappel ou équilibre entre les deux).

4.3.2 Optimisation de l'ordonnancement automatique des règles

Une manière de réduire la confusion est de trouver un ordonnancement approprié des règles de la description grammaticale. Quand un élément est segmenté et étiqueté avec le système de reconnaissance de documents, l'élément peut être consommé. Il ne sera alors plus disponible pour le reste de l'analyse. C'est pourquoi nous devons d'abord consommer les éléments pour lesquels la règle de grammaire présente une bonne précision afin de minimiser la confusion. L'ordonnancement des règles a un impact déterminant sur les performances globales du système.

Dans EWO, lorsque l'utilisateur est satisfait des règles de grammaire qu'il a conçues, il peut demander un ordonnancement automatique des règles. Pour effectuer cet ordonnancement automatique des règles, nous proposons l'algorithme suivant :

1. Appliquer chaque règle sur tous les éléments disponibles
2. Calculer la précision de chaque règle
3. Sélectionner la règle présentant la meilleure précision
4. Consommer les éléments correspondants à l'application de cette règle
5. Retourner à l'étape 1 tant que toutes les règles n'ont pas été ordonnées

Cet ordonnancement automatique des règles permet d'optimiser automatiquement la structure logique de la description grammaticale en minimisant les confusions possibles. Afin d'accélérer le processus d'ordonnancement, nous n'utilisons pas les règles dans le système de reconnaissance mais les requêtes dans EWO.

Dans l'exemple des courriers manuscrits en français, l'ordonnancement automatique de la description grammaticale permet d'obtenir l'ordre suivant :

```
lettre ::=
   ouverture &&
   corpsDeTexte &&
   signature &&
   coordonneesExpediteur - adresse &&
   coordonneesDestinataire &&
   ps &&
   dateLieu &&
   objet &&
   coordonneesExpediteur - reference client.
```

Nous pouvons remarquer que ce n'est pas un ordre qui aurait été trouvé facilement manuellement.

4.4 Bilan

Nous avons présenté la méthode Eyes Wide Open qui permet la résolution automatique et interactive de sous-problèmes de la description grammaticale (cf. figure 4.1).

Nous avons détaillé les trois grandes étapes de cette méthode :

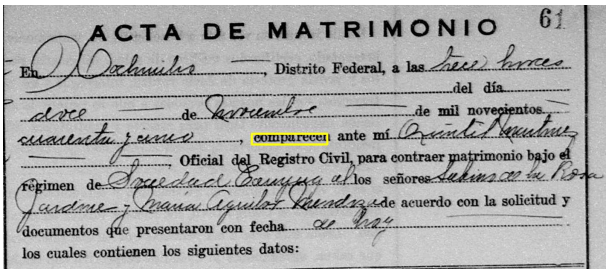
1. Acquisition des données ;
2. Inférence des règles ;
3. Intégration dans une description grammaticale.

Grâce à ces travaux, il est désormais possible d'inférer les règles en inférant à la fois la structure logique et la structure physique correspondant à un sous-problème de la description grammaticale.

Notre approche pour l'acquisition des données permet de réaliser l'inférence des règles que nous ayons une vérité terrain annotée disponible sur les documents ou non. Nous avons détaillé comment nous procédons à l'acquisition des données sans vérité terrain, à l'aide d'une fiabilisation de primitives en interaction avec l'utilisateur.

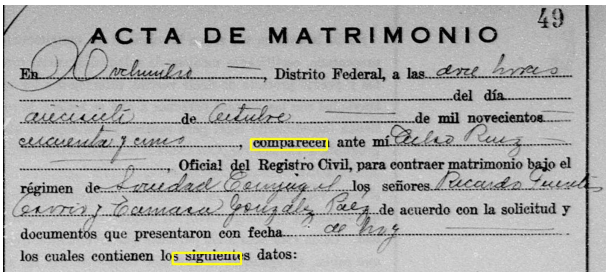
La méthode EWO est indépendante de toute méthode existante et peut donc être utilisée pour l'étape d'apprentissage de n'importe quelle méthode syntaxique. De plus, l'étape d'acquisition des données quand il n'y a pas de vérité terrain annotée sur les documents peut s'appliquer dans un cadre plus général puisqu'elle ne dépend pas de la méthode d'inférence utilisée après.

Notre méthode permet à l'utilisateur d'avoir une vue exhaustive sur les documents à reconnaitre, contrairement à ce qui était possible lors d'une description manuelle des règles. En effet, seul un petit échantillon de documents étaient alors analysés par l'utilisateur, ne garantissant pas une bonne représentativité des documents à traiter. Cette vue exhaustive va faciliter et accélérer le travail de création de la description grammaticale.

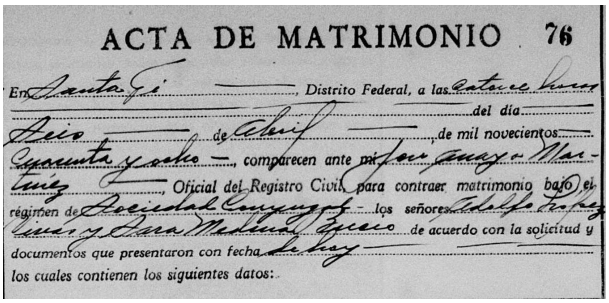


(a) Cas idéal : une seule primitive trouvée, correspondant à l'élément recherché

(b) Exemple de bruit : le mot-clé recherché est effectivement trouvé ainsi que d'autres occurrences de bruit

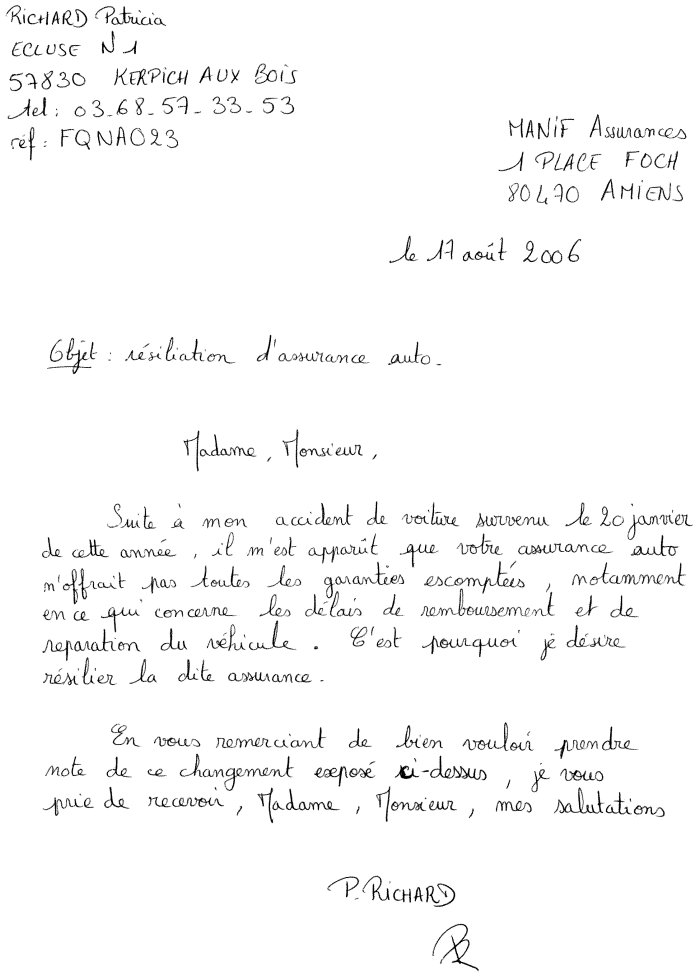


(c) Exemple de bruit : le mot-clé recherché est effectivement trouvé ainsi que d'autres occurrences de bruit



(d) Exemple de document où aucune occurrence du mot-clé recherché n'est trouvée

FIG. 4.5 – Exemple de mots clés « comparecen » détectés dans différents documents avec une méthode de *word spotting*


RICHARD Patricia
ECLUSE N 1
57830 KERPICH AUX BOIS
tel : 03-68-57-33-53
réf : FQNA023

MANIF Assurances
1 PLACE FOCH
80470 AMIENS

le 11 août 2006

Objet : résiliation d'assurance auto.

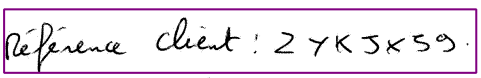
Madame, Monsieur,

Suite à mon accident de voiture survenu le 20 janvier de cette année, il m'est apparût que votre assurance auto n'offrait pas toutes les garanties escomptées, notamment en ce qui concerne les délais de remboursement et de reparation du véhicule. C'est pourquoi je désire résilier la dite assurance.

En vous remerciant de bien vouloir prendre note de ce changement exposé ci-dessus, je vous prie de recevoir, Madame, Monsieur, mes salutations

P. RICHARD

FIG. 4.6 – Exemple de document dont on effectue la description logique


Référence client : 2YKJK59

(a) Exemple 1

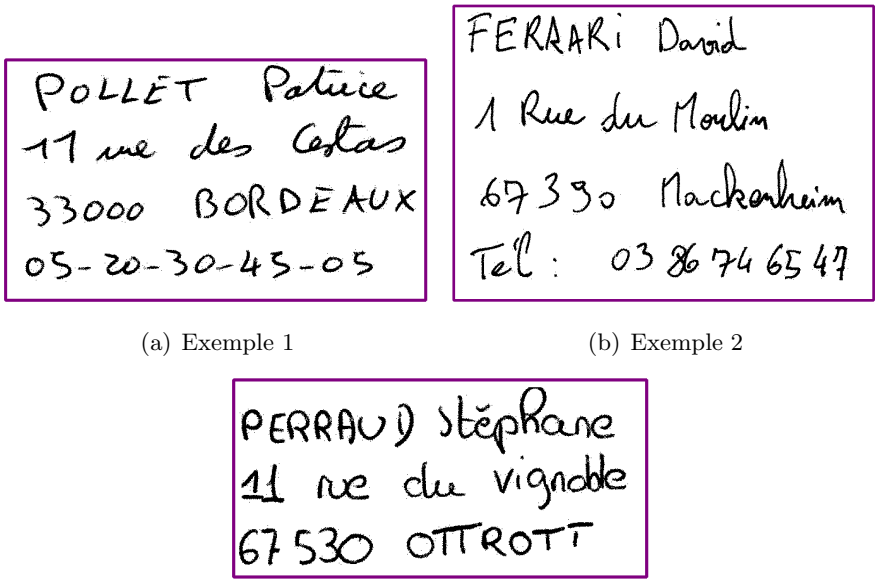

Réf. Client : AQGAZ79

(b) Exemple 2


référence client : ZJSL000

(c) Exemple 3

FIG. 4.7 – Exemples représentatifs présentés à l'utilisateur pour le cluster 1. Ce cluster est étiqueté « référence client » par l'utilisateur.



(a) Exemple 1

(b) Exemple 2

(c) Exemple 3

FIG. 4.8 – Exemples représentatifs présentés à l'utilisateur pour le cluster 2. Ce cluster est étiqueté « coordonnées postales » par l'utilisateur.

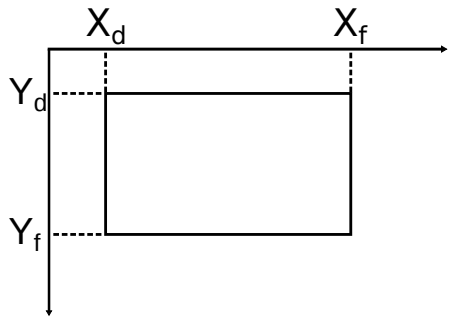
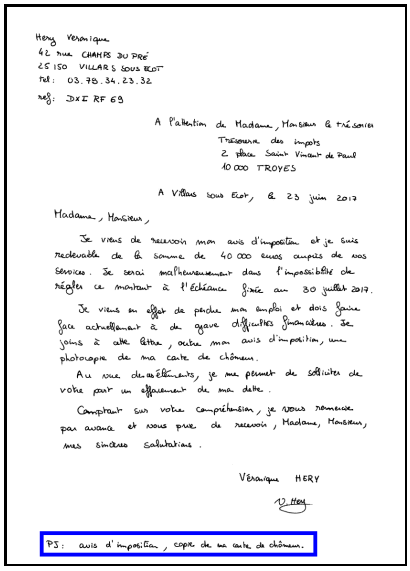
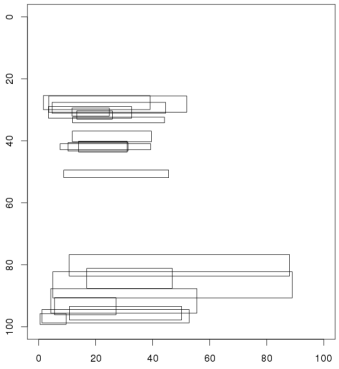


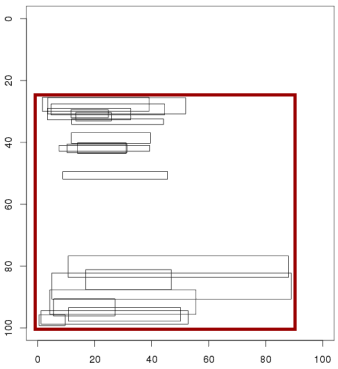
FIG. 4.9 – Représentation des objets par leur boite englobante



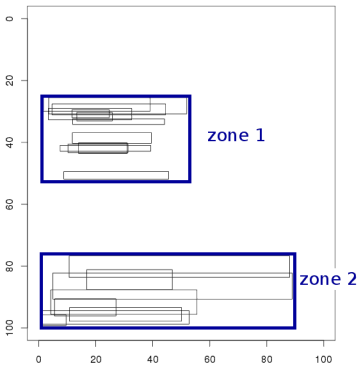
(a) Exemple de courrier contenant un élément PS/PJ



(b) Positionnement normalisé de PS/PJ dans un corpus de 300 pages



(c) Zones obtenues quand l'opérateur de position ne tient pas compte des variantes physiques



(d) Zones obtenues lorsque les variantes physiques des opérateurs de position sont détectées

FIG. 4.10 – Exemple d'inférence de l'opérateur de position de l'élément « PS/PJ », montrant l'intérêt de la définition de plusieurs zones pour un opérateur de position. La boite englobante de chaque occurrence de PS/PJ est représentée par un rectangle dans une page normalisée.

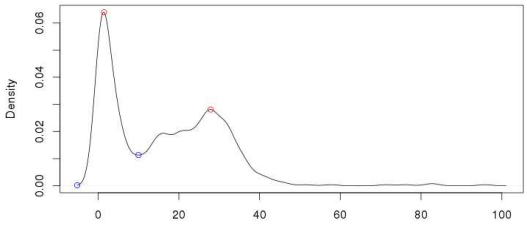
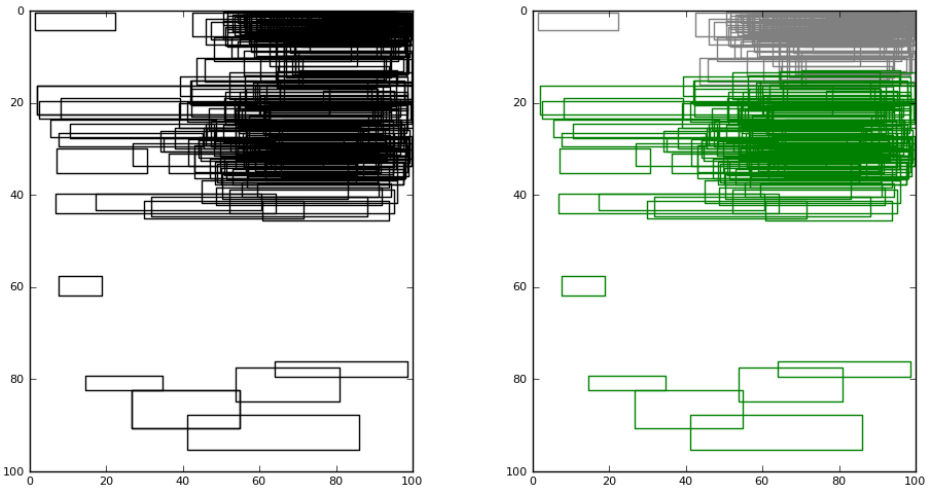


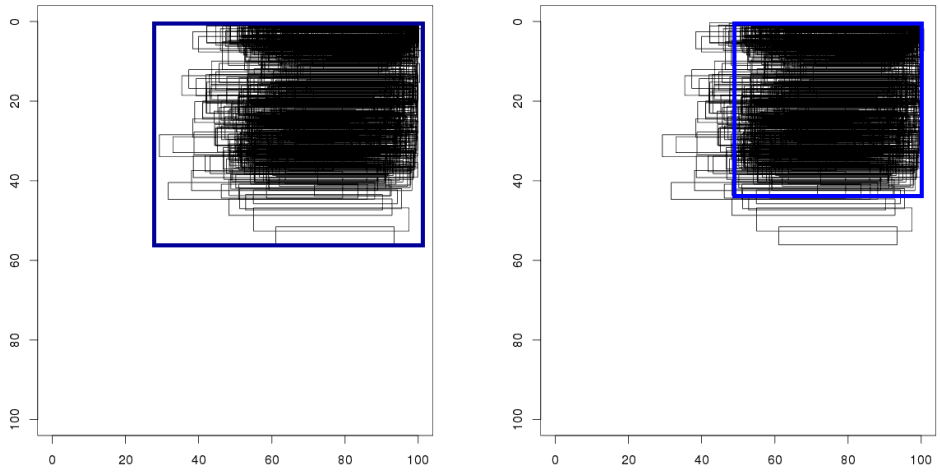
FIG. 4.11 – Détection de deux groupes par la méthode de la détection des maxima locaux dans un histogramme



(a) Représentation synthétique des boites englobantes de tous les éléments « date, lieu »

(b) Représentation des deux groupes détectés pour le positionnement des éléments « date, lieu »

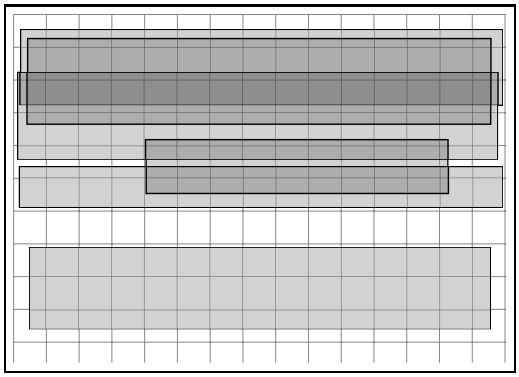
FIG. 4.12 – Exemple de détections des groupes pour les positionnements des éléments « date, lieu » dans un corpus de courriers manuscrits en français



(a) Représentation de la zone d'opérateur de position englobant tous les éléments

(b) Représentation de la zone d'opérateur de position réduite grâce à la densité

FIG. 4.13 – Exemple d'ajustement des frontières d'une zone avec une méthode basée sur la densité



(a) Représentation des boites englobantes des éléments

2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	3	3	3	3	3	3	3	3	3	2
2	2	2	2	3	3	3	3	3	3	3	3	3	2
1	1	1	1	2	2	2	2	2	2	2	2	2	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1	1	1	1	1	1	1

(b) Grille de densité obtenue avant seuillage

2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	3	3	3	3	3	3	3	3	3	2
2	2	2	2	3	3	3	3	3	3	3	3	3	2
				2	2	2	2	2	2	2	2	2	

(c) Grille de densité obtenue après seuillage (seuil = 1)

2	2	2	2	2	2	2	2	2	2	2	2	2	2
3	3	3	3	3	3	3	3	3	3	3	3	3	3
3	3	3	3	3	3	3	3	3	3	3	3	3	3
2	2	2	2	3	3	3	3	3	3	3	3	3	2
2	2	2	2	3	3	3	3	3	3	3	3	3	2
				2	2	2	2	2	2	2	2	2	

(d) Zone de l'opérateur de position obtenu (en vert)

FIG. 4.14 – Exemple de calcul des frontières d'une zone d'un opérateur de position

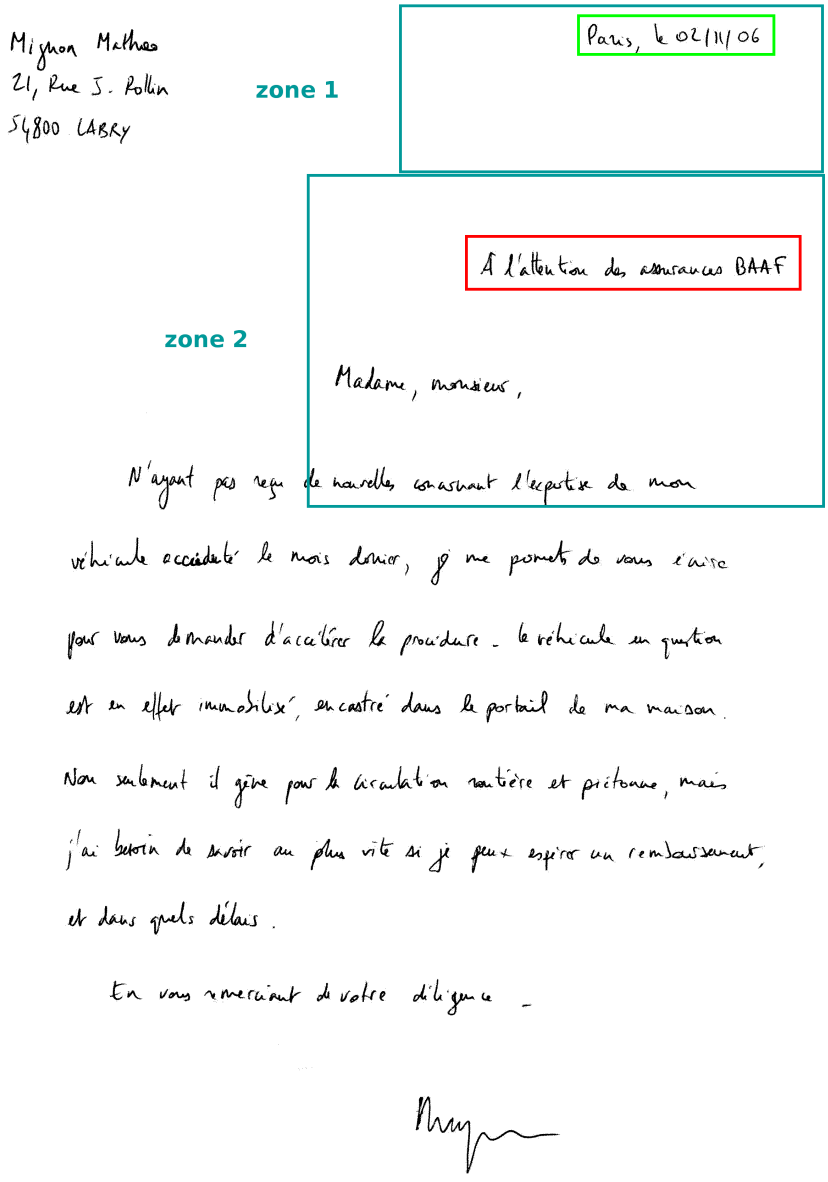
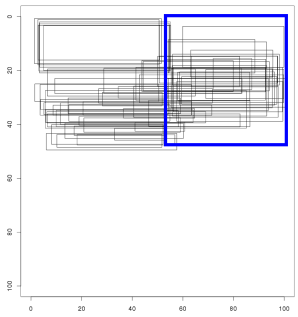
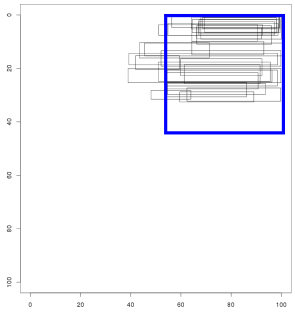


FIG. 4.15 – Exemple de minimisation de la confusion grâce à l'ordre des zones : une recherche de l'élément « date/lieu » d'abord dans la zone 1 (`coinHautGauche`) puis dans la zone 2 (`hautGauche`) permet de réduire les risques de confusion avec un autre élément logique



(a) Représentation de tous les éléments pouvant apporter de la confusion



(b) Représentation des éléments recherchés (« date, lieu »)

FIG. 4.16 – L'opérateur de position de « date, lieu » est analysé du haut vers le bas pour maximiser les chances de sélection l'élément « date, lieu » tout en minimisant les risques de sélectionner un autre élément

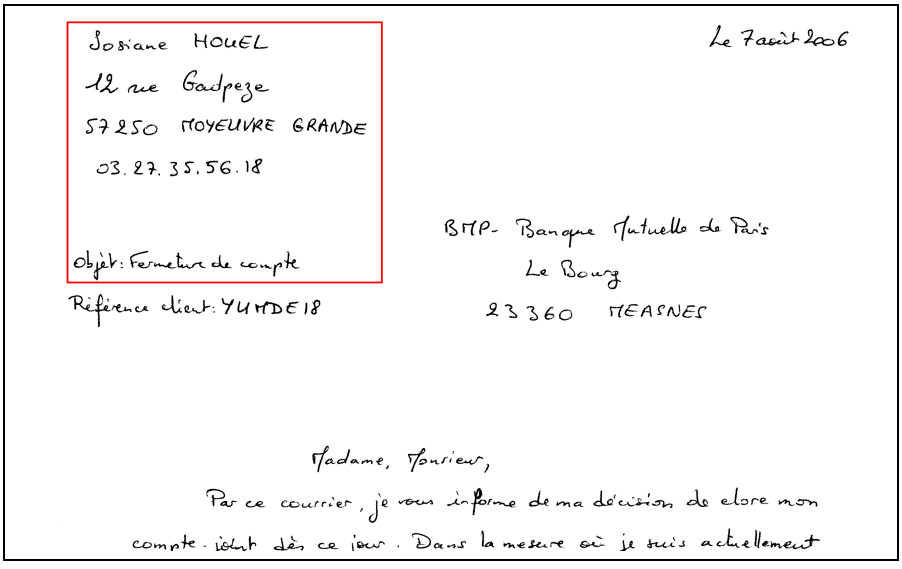


FIG. 4.17 – Exemple de bloc coordonnées expéditeur mal segmenté

Chapitre 5

Clustering

La capacité de la méthode EWO à détecter des structures intéressantes dans les données est cruciale pour son bon fonctionnement. Pour cette détection, nous utilisons un algorithme de clustering, l'Evidence Accumulation Clustering [FJ02], qui offre de nombreuses bonnes propriétés le rendant adapté à notre problématique.

Dans ce chapitre, nous présentons d'abord l'Evidence Accumulation Clustering et nous détaillons pourquoi cet algorithme est adapté à notre problématique. Nous présentons ensuite les points d'intégration de l'algorithme de clustering dans la méthode EWO.

5.1 Evidence Accumulation Clustering

L'Evidence Accumulation Clustering (dénommé EAC clustering par la suite) est un algorithme de clustering de la famille des méthodes ensemblistes. Dans cette famille d'algorithmes, plusieurs partitions différentes sont obtenues sur le même jeu de données et nous cherchons à obtenir une partition consensus qui agrège les résultats. Ces méthodes s'inspirent des nombreuses méthodes ensemblistes existant en classification supervisée comme le *bagging* ou le *boosting*. Le processus de combinaison des différentes partitions peut compenser les erreurs d'une partition. De plus, la décision d'un groupe est alors considéré comme plus fiable que la décision d'un seul algorithme.

L'idée à l'origine de l'EAC clustering est que si deux individus appartiennent au même cluster alors ils vont probablement être regroupés ensemble dans différentes partitions. Nous pouvons alors construire une nouvelle mesure de similarité entre les individus en nous basant sur le nombre de fois où deux éléments sont regroupés dans le même cluster. Un algorithme de clustering est ensuite utilisé avec cette mesure de similarité pour produire la partition finale.

Cette technique permet de stabiliser les clusters instables en moyennant leur réponse et est très efficace pour trouver des clusters ayant des formes non conventionnelles (spirales par exemple). En effet, en utilisant l'EAC clustering, nous n'imposons pas de formes aux clusters recherchés contrairement à la plupart des algorithmes de clustering.

5.1.1 Fonctionnement

Nous rappelons ici le fonctionnement de l'EAC clustering introduit par Fred et Jain [FJ02]. Nous supposons que nous cherchons à obtenir une partition pour n individus. Pour cela, nous allons construire N partitions intermédiaires permettant d'aboutir à la partition finale des données.

Construction des partitions Les N partitions intermédiaires sont construites en utilisant :

- le même algorithme avec les mêmes paramètres ou
- le même algorithme avec des paramètres différents ou
- différents algorithmes de clustering.

Combiner les partitions : construction de la matrice de co-association La construction de la matrice de co-association revient à la création d'une nouvelle mesure de similarité entre les observations. En utilisant les co-occurrences des individus dans le même cluster comme mesure de leur association, les N partitions effectuées sur les n individus correspondent à une matrice de co-association C de taille $n \times n$ telle que :

$$C(i,j) = \frac{n_{ij}}{N}$$

où n_{ij} correspond au nombre de fois où la paire (i,j) se trouve dans le même cluster parmi les N partitions.

Détermination de la partition finale La partition finale est construite à l'aide d'une Classification Ascendante Hiérarchique appliquée sur la matrice de co-association construite lors de la combinaison des partitions. La détermination du nombre de clusters de la partition finale se fait à l'aide du critère du temps de vie maximum (maximum lifetime criterion). Nous découpons donc le dendrogramme au niveau de la partition qui perdure le plus longtemps. Dans l'exemple de la figure 5.1, une partition en trois clusters est privilégiée, la valeur l_3 étant la plus grande.

5.1.2 Justification du choix

L'Evidence Accumulation Clustering a été choisi car il répond à plusieurs des contraintes que nous avons mis en avant dans la section 3.3.2. L'EAC clustering assure la *généricité* de notre méthode. En effet, cet algorithme de clustering a la particularité de ne pas imposer de représentation aux données, notamment en imposant une forme aux clusters recherchés. Cela nous permet de l'utiliser au sein de la méthode EWO sur n'importe quel nouveau corpus de documents sans avoir besoin de changer d'algorithme.

De plus, l'EAC clustering nous permet de résoudre automatiquement la problématique de détermination du nombre de clusters à trouver. Cette problématique bien connue est en effet une des plus complexes du clustering. La détermination du nombre

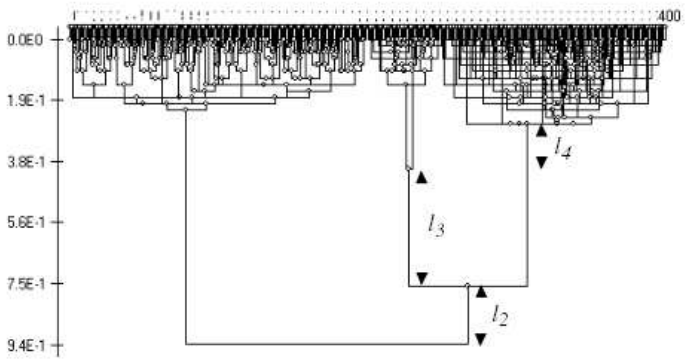


FIG. 5.1 – Illustration du découpage de la partition finale selon le critère du *maximum lifetime criterion* menant à un découpage en 3 clusters (extrait de l'article *Data Clustering Using Evidence Accumulation* de Fred and Jain [FJ02])

de clusters à trouver nécessite en effet de bonnes connaissances sur les documents, ce que l'utilisateur ne possède pas forcément au moment de l'utilisation de la méthode EWO. La méthode EWO sert en effet au contraire à donner une vue à la fois exhaustive et synthétique sur les données à traiter sur les documents à traiter pour faciliter la description grammaticale des documents.

5.1.3 Notre implémentation

Dans l'implémentation que nous avons choisi, nous utilisons pour la partie de construction des partitions l'algorithme des K-moyennes. La valeur de K est choisi aléatoirement à chaque étape du split. Cette technique est celle qui obtient les meilleurs résultats en comparaison à un choix fixe de K [FJ02]. En effet, lorsque K a toujours la même valeur, il existe un intervalle restreint qui donnera une partition finale correspondant à un découpage significatif pour l'utilisateur. De plus, l'algorithme des K-moyennes est un algorithme souvent utilisé pour lequel de nombreuses implémentations performantes existent.

Pour la création de la partition finale, nous avons choisi d'utiliser comme mesure de dissimilarité inter-classe le lien moyen (*average link*). En effet, les auteurs proposent l'utilisation de deux mesures de dissimilarité inter-classe différentes : le saut minimum ou le lien moyen. Le saut minimum obtient de meilleures performances que le lien moyen lorsque les clusters sont bien séparés. Notre méthode devant être générique et s'adapter facilement à un nouveau jeu de données, nous avons favorisé le lien moyen qui correspond aux situations que nous rencontrons en général dans les données.

L'inconvénient majeur de cette méthode est qu'elle a une complexité quadratique en temps et en mémoire en fonction du nombre d'observations $O(n^2)$. Afin d'améliorer ses performances sur des jeux de données plus volumineux, nous utilisons dans notre implémentation une matrice creuse pour la matrice de co-association comme proposé par Lourenço [LFJ10].

5.1.4 Présentation de la partition à l'utilisateur

Les clusters obtenus sont présentés à l'utilisateur pour lui apporter une connaissance à la fois synthétique et exhaustive des données à analyser. Nous avons choisi de présenter les clusters à l'utilisateur au travers d'exemples représentatifs. Ces exemples représentatifs sont des individus proches des centroïdes de chaque cluster.

Ces exemples représentatifs sont accompagnés d'une mesure de la qualité de la partition produite. Afin de permettre à l'utilisateur de s'assurer que les exemples fournis sont effectivement bien représentatifs de l'ensemble des individus du cluster, des indicateurs sur la dispersion des individus dans le cluster sont fournis. Nous avons choisi comme indicateur de dispersion l'écart-type.

5.2 Points d'intégration dans EWO

Lors de l'analyse des données par la méthode EWO, deux points nécessitent l'utilisation d'une méthode de clustering pour extraire des connaissances :

- la fiabilisation des données lorsqu'il n'y a pas de vérité terrain annotée disponible sur les données ;
- la détection automatique des variations logiques.

Nous détaillons dans cette section ces deux points d'insertion de l'algorithme de clustering.

5.2.1 Fiabilisation des données

Lors de la fiabilisation des données présentée dans la section 4.1.4, l'EAC clustering est utilisé afin de supprimer des données par cluster afin de constituer une pseudo vérité terrain.

L'utilisation de l'EAC clustering dans le cadre de la fiabilisation des données nous permet de sur-segmenter si nécessaire les données. En effet, nous ne désirons pas avoir trop de clusters à visualiser mais nous avons besoin d'avoir une cohérence suffisante dans chaque cluster pour que l'utilisateur puisse facilement prendre une décision fiable sur la conservation de toutes les occurrences du cluster à partir de seulement quelques exemples représentatifs. Cependant, une trop grande sur-segmentation des données conduirait à l'analyse de très nombreux clusters ce qui réduirait l'intérêt de la méthode par rapport à une visualisation manuelle de chaque occurrence. C'est pourquoi nous proposons une partition finale sans sur-segmentation et laissons la possibilité à l'utilisateur de sur-segmenter si cela s'avère nécessaire.

5.2.2 Détection automatique des variations logiques

L'EAC clustering est également utilisé lors de la détection des variations logiques présentée dans la section 4.2.1. Dans le cadre de la détection automatique des variations logiques, nous ne cherchons pas à sur-segmenter la partition finale. En effet, cela conduirait à la production d'un grand nombre de variantes de règles non nécessaire à la

description et la reconnaissance des documents. La partition finale utilisée dans cette partie est donc produite en utilisant le critère du *maximum lifetime criterion*.

5.3 Bilan

Nous avons introduit dans la méthode EWO l'utilisation d'un algorithme de clustering en deux points distincts : lors de la fiabilisation des données et lors de la détection automatique des variations logiques. Le clustering nous permet de faire émerger automatiquement des structures de données qui sont ensuite visualisées par l'utilisateur afin de leur donner un sens. Ainsi, les données mises en avant peuvent être conservées ou non dans l'analyse. Si l'utilisateur choisit de les conserver dans l'analyse, il peut ensuite attribuer à chaque cluster un libellé porteur de sémantique afin de réaliser la description grammaticale des documents. Le clustering nous permet ainsi d'avoir une vision à la fois *synthétique* et *exhaustive* des données puisque seuls quelques exemples par cluster sont visualisés par l'utilisateur.

L'algorithme de clustering choisi ici est l'Evidence Accumulation Clustering introduit par Fred et Jain [FJ02]. Cet algorithme nous permet de répondre à nos exigences de *généricité*, puisqu'il n'impose pas de formes aux clusters recherchés, mais nous permet aussi de ne pas solliciter l'utilisateur pour fixer les paramètres du clustering.

La minimisation du nombre de paramètres à fixer par l'utilisateur est cruciale dans notre méthode puisque nous proposons à l'utilisateur, par la méthode EWO, d'acquérir de la connaissance sur les documents à traiter. La détermination des paramètres est une tâche qui nécessite des connaissances sur les données que l'utilisateur n'a pas. Dans notre méthode, aucun paramètre n'est fourni ici par l'utilisateur. Pour les partitions intermédiaires, chaque partition est construite avec des paramètres déterminés de manière aléatoire. Pour la détermination du nombre de clusters de la partition finale, nous utilisons le critère proposé par Fred et Jain, le *maximum cluster lifetime*.

Chapitre 6

Interaction utilisateur

Dans ce chapitre, nous détaillons les points d'interaction entre l'utilisateur et la méthode EWO (cf. figure 4.1). Pour chaque moment d'interaction, nous précisons à quelle question l'utilisateur doit répondre, quelles sont les connaissances nécessaires à l'utilisateur et enfin nous présentons aux lecteurs des extraits de l'interface afin de visualiser ce qui est présenté à l'utilisateur.

L'interaction est intégrée dans notre méthode car comme nous l'avons présenté dans les chapitres précédents la problématique étudiée est trop complexe pour permettre une inférence entièrement automatique de la description grammaticale. Notre méthode doit être capable de gérer de gros volumes de données bidimensionnelles, complexes et bruitées. De plus, la méthode doit pouvoir fonctionner sur des corpus où aucune vérité terrain n'est disponible du fait du coût prohibitif de l'annotation manuelle d'une vérité terrain.

L'interaction avec l'utilisateur est effectuée en trois points différents de la méthode EWO que nous allons maintenant détailler :

1. Fiabilisation des primitives ;
2. Détection des variations logiques ;
3. Inférence des opérateurs de position.

6.1 Fiabilisation des primitives

L'étape de fiabilisation des primitives est réalisé lorsqu'il n'y a pas de vérité terrain étiquetée disponible (cf. section 4.1.4). Cette étape nécessite une interaction forte avec l'utilisateur. Nous rappelons d'abord le principe général de la fiabilisation des primitives.

Afin de fiabiliser les primitives, une partition de toutes les occurrences des primitives extraites automatiquement est effectuée à l'aide de l'Evidence Accumulation Clustering [FJ02]. Nous obtenons ainsi des clusters d'occurrences semblables. La décision de conserver ou supprimer les occurrences est ensuite effectuée par cluster. L'intervention de l'utilisateur se fait au niveau de la prise de décision de la conservation ou de la suppression de chaque cluster.

Pour la fiabilisation des primitives, l'utilisateur doit donc répondre à la question fermée suivante : *« conserve-t-on les observations contenues dans le cluster pour la constitution de la pseudo vérité terrain ? »*. Pour cela, il est nécessaire à l'utilisateur de savoir quelles sont les primitives qu'il considère comme correctes pour la suite de l'analyse.

En pratique, la fiabilisation se décompose en trois étapes différentes :

1. Sélection des variables à utiliser pour le clustering ;
2. Visualisation de la proposition de partition ;
3. Visualisation des clusters.

Pour la présentation des extraits de l'interface, nous nous basons sur l'exemple de la fiabilisation des primitives correspondant au mot-clé « comparecen » dans des actes de mariages mexicains d'archive (cf. chapitre 10).

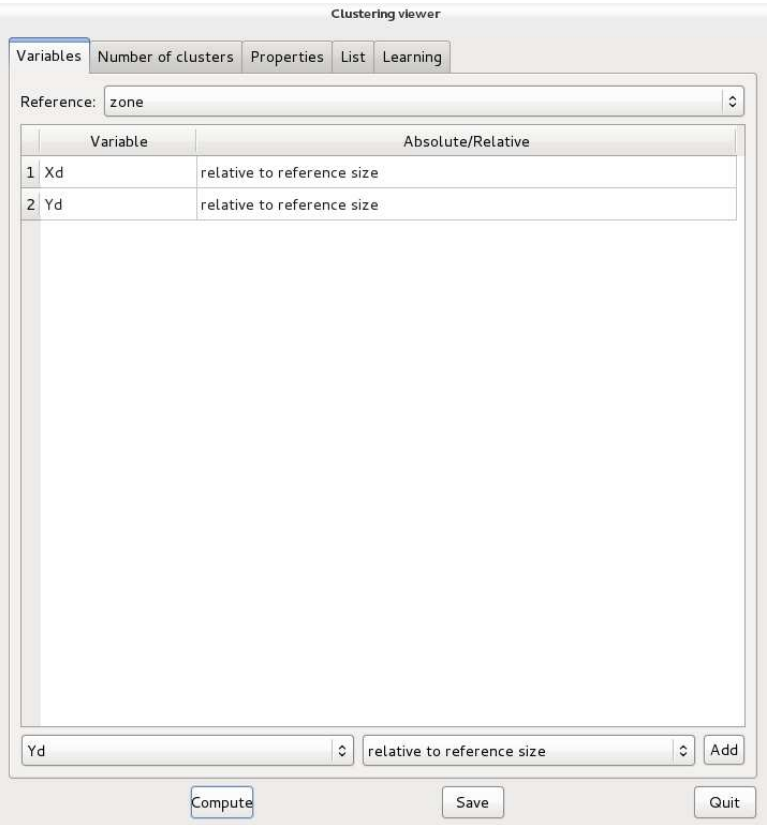


FIG. 6.1 – Exemple de sélection des variables pour le clustering : le clustering est effectué selon la position du mot-clé « comparecen » dans une zone de référence selon l'axe des abscisses et des ordonnées

1 - Sélection des variables à utiliser pour le clustering : cette sélection est effectuée par l'utilisateur en fonction des primitives à fiabiliser. Pour fiabiliser les primitives

du mot-clé « comparecen », nous effectuons le clustering sur la position du mot-clé en utilisant les coordonnées (X_d, Y_d) du point haut gauche de sa boite englobante selon l'axe des abscisses et des ordonnées. L'écran associé à cette sélection des variables est présenté dans la figure 6.1.

2 - Proposition de partition : Le dendrogramme est affiché à l'utilisateur et la méthode EWO propose un découpage du dendrogramme pour l'obtention de la partition finale des données à partir du *maximum lifetime criterion* (cf. chapitre 5). La répartition du nombre d'individus par cluster de la partition proposée est affichée. Dans notre exemple, la méthode EWO nous propose une partition en 11 classes (figure 6.2). L'utilisateur peut *sur-segmenter* les données en choisissant un autre découpage du dendrogramme s'il juge que les clusters obtenus ne sont pas suffisamment compacts pour permettre une prise de décision fiable.

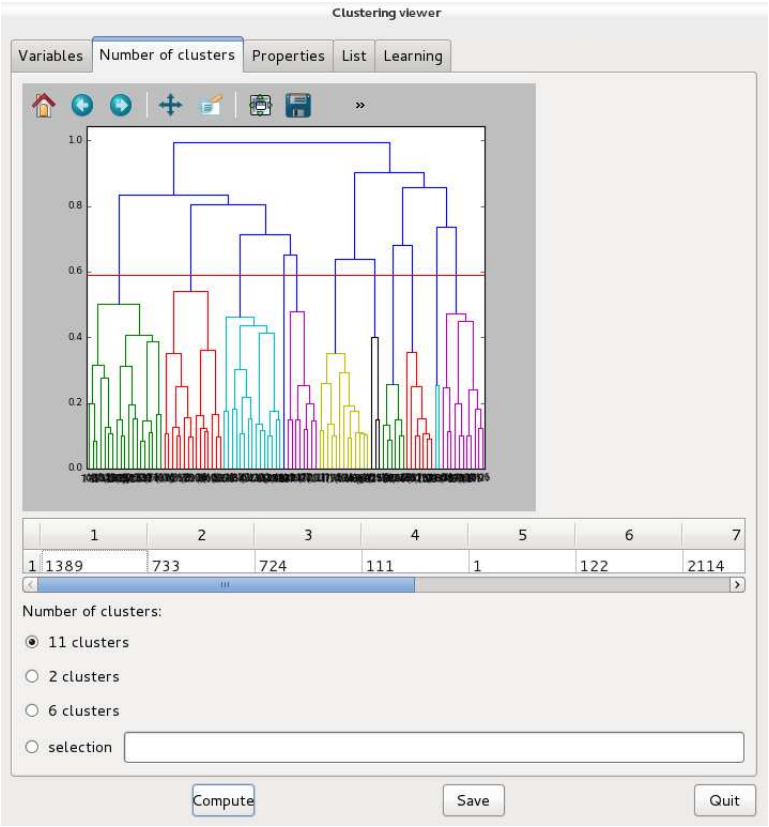


FIG. 6.2 – Exemple d'affichage du dendrogramme obtenu pour la partition finale. Un découpage en 11 clusters est proposé automatiquement par la méthode EWO

3 - Visualisation des clusters : Pour chaque cluster, le système sélectionne 6 exemples représentatifs du cluster et les présentent à l'utilisateur. L'utilisateur prend

alors la décision de conserver ou supprimer toutes les occurrences du cluster de l'analyse.

Dans notre exemple, le cluster présenté dans la figure 6.3 est rejeté par l'utilisateur. Dans l'interface, il lui suffit alors de cocher une case (encadrée en vert) pour indiquer que toutes les occurrences qu'il contient sont donc supprimées de la pseudo vérité terrain. En effet, les exemples nous montrent que le cluster ne contient pas des occurrences du mot-clé « comparecen » mais des occurrences correspondant au mot « Ocupación ». L'utilisateur supprime ainsi 2114 occurrences en une seule prise de décision et interaction avec la méthode EWO.

L'utilisateur peut également accéder à la liste des toutes les occurrences contenues dans le cluster et ainsi visualiser d'autres éléments que ceux présentés en tant qu'exemples représentatifs. Cela est notamment utile dans le cas de petits clusters (en nombre d'individus) présentant des données atypiques.

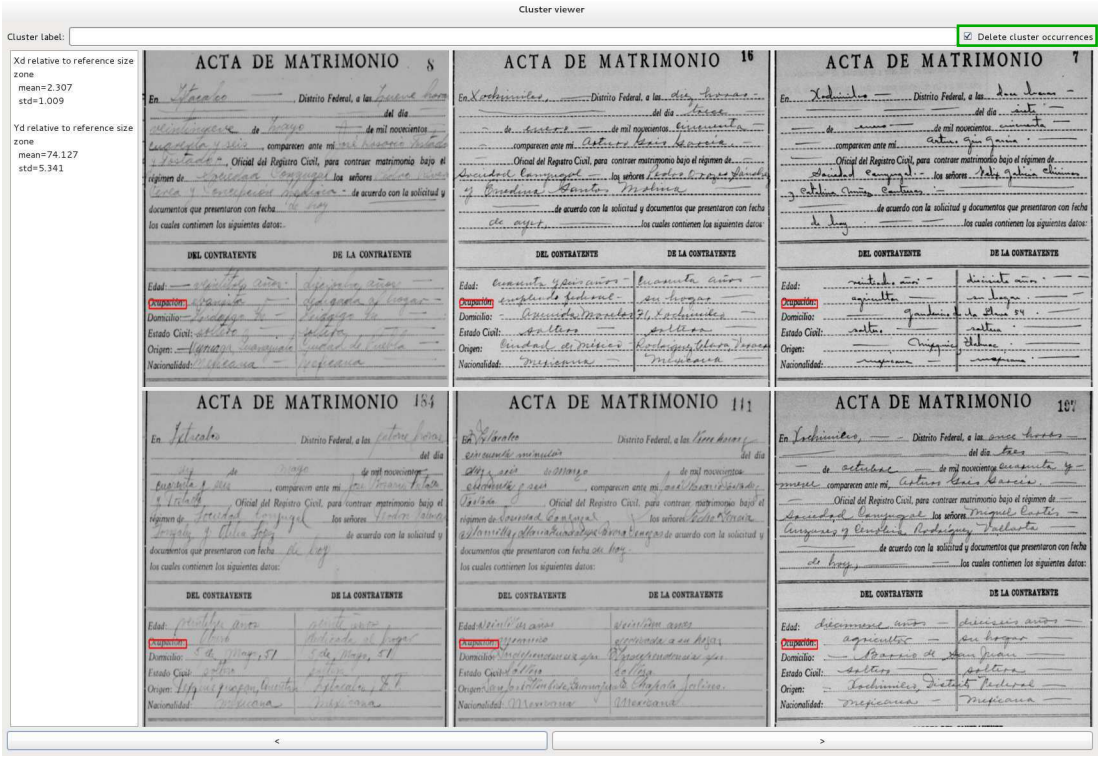


FIG. 6.3 – Exemple d'affichage d'un des clusters à l'utilisateur : sur la base des exemples présentés, l'utilisateur décide de supprimer toutes les occurrences du cluster.

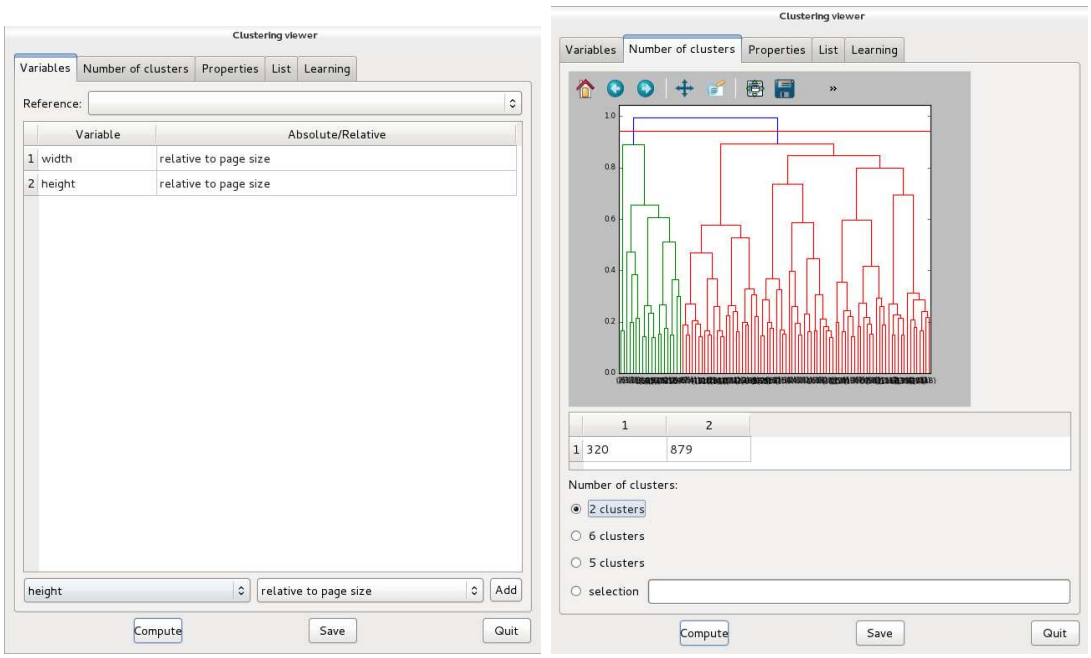
Lorsque l'utilisateur a visualisé chacun des clusters dans l'interface, l'ensemble des occurrences à conserver est déterminé et la pseudo vérité terrain est constituée.

6.2 Détection des variations logiques

La détection des variations logiques est basée sur un principe similaire à celui de la fiabilisation des primitives (cf. section 4.2.1). Une partition des occurrences de la vérité terrain ou de la pseudo vérité terrain est effectuée grâce à l'Evidence Accumulation Clustering [FJ02]. Les clusters formés par cette partition sont ensuite présentés à l'utilisateur.

Cependant, dans le cas de la détection des variations logiques, l'utilisateur ne répond plus à une question fermée visant à conserver ou rejeter les occurrences du cluster. Sa tâche consiste ici à attribuer un *libellé à chaque cluster*. Le libellé attribué est porteur d'une sémantique. Il permettra à l'utilisateur d'indiquer dans la description grammaticale quel type d'éléments sont décrits avec cette variante de règle. Le cluster peut éventuellement être rejeté de l'analyse si l'utilisateur ne souhaite pas modéliser les observations qu'il contient.

Nous retrouvons donc les mêmes trois étapes que pour la fiabilisation des primitives (section 6.1). Nous prenons ici l'exemple de la détection des variations logiques pour l'élément « coordonnées expéditeur » dans des courriers manuscrits.



(a) Exemple de sélection des variables pour le clustering : le clustering est effectué selon la largeur et la hauteur de la boite englobante

(b) Affichage du dendrogramme obtenu pour le clustering des élément « coordonnées expéditeur ». Une partition en deux classes est proposée par la méthode EWO.

FIG. 6.4 – Extraits de l'interface graphique pour l'interaction entre l'utilisateur et la méthode EWO lors de la détection des variantes logiques

1 - Sélection des variables à utiliser pour le clustering : Les variables sélectionnées par l'utilisateur pour effectuer le clustering sont la hauteur et la largeur de la boite englobante des éléments « coordonnées expéditeur » (figure 6.4(a)).

2 - Proposition de partition : Le dendrogramme est affiché à l'utilisateur et la méthode EWO propose automatiquement un découpage du dendrogramme pour l'obtention de la partition finale. La méthode EWO propose ici une partition en deux classes (figure 6.4(b)).

3 - Visualisation des clusters : Pour chaque cluster, la méthode EWO sélectionne et présente des exemples représentatifs. L'utilisateur attribue ensuite à chacun des clusters un libellé porteur de sémantique.

Dans notre exemple, le cluster présenté dans la figure 6.5 représente la variation logique des coordonnées postales. Il suffit alors à l'utilisateur de nommer la variante logique dans le champ encadré en vert sur la figure. Une fois le nommage effectué, l'utilisateur visualise le second cluster détecté.

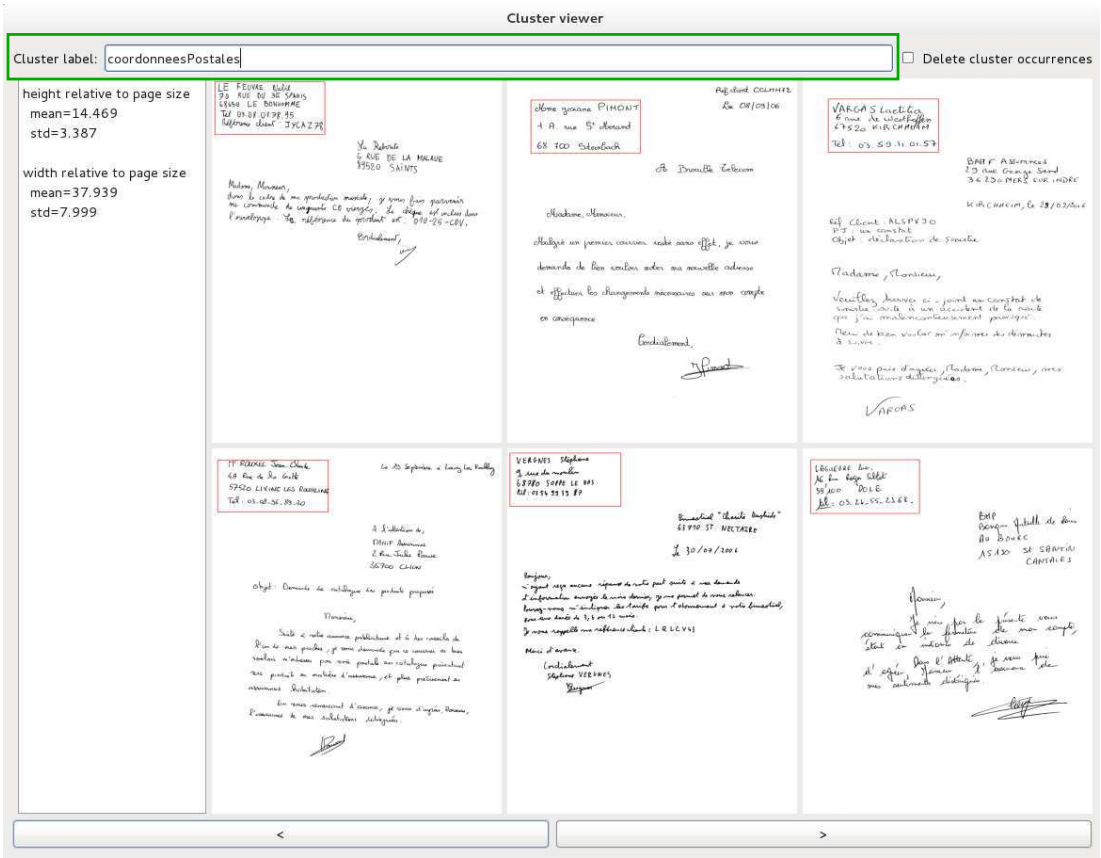


FIG. 6.5 – Présentation d'un cluster à l'utilisateur pour la détection des variations logiques correspondant aux « coordonnées expéditeur » dans des courriers manuscrits

Lorsque l'utilisateur a visualisé tous les clusters dans l'interface, l'ensemble des variantes logiques a été détecté et nommé. L'utilisateur peut alors poursuivre l'inférence progressive de la règle dans la méthode EWO en inférant la structure physique de l'élément (positionnement et propriétés physiques).

6.3 Opérateurs de position

Lors de l'apprentissage de la structure physique des documents, des opérateurs de position sont inférés (cf. section 4.2.2.1). L'utilisateur intervient dans cette inférence en déterminant l'opportunité d'utiliser un opérateur de position *absolu ou relatif*. Lors du choix d'un opérateur de position relatif, c'est l'utilisateur qui indique quels éléments doivent être utilisés comme référence pour le positionnement. La référence est influencée par l'ordre des règles et le rappel et la précision obtenus pour chaque type d'éléments.

En pratique, il y a donc deux interactions différentes entre l'utilisateur et la méthode EWO pour l'inférence des opérateurs de position :

1. Choix d'un opérateur de position absolu ou relatif ;
2. Nommage de chacune des variantes automatiquement détectées par la méthode EWO de l'opérateur de position. Comme dans le cas de la détection des variantes logiques (cf. section 6.2), le libellé est porteur d'une sémantique.

Nous présentons les extraits de l'interface sur l'exemple de l'inférence d'un opérateur de position absolu correspondant à l'élément logique « date, lieu » dans des courriers manuscrits.

Lorsque l'opérateur de position est inféré, les différentes variantes sont présentées à l'utilisateur de manière synthétique dans une page normalisée (figure 6.6 - encadré rouge). Dans notre exemple, deux variantes physiques différentes ont été automatiquement détectées. L'utilisateur peut également instancier sur les différents documents de l'ensemble d'apprentissage l'opérateur de position inféré (figure 6.7).

Si différentes variantes existent pour l'opérateur de position, l'ordre optimal lui est indiqué (figure 6.6 - encadré vert). Pour l'opérateur de position de « date, lieu », l'ordre optimal proposé par la méthode EWO est :

1. Variante représentée en gris, correspondant au coin haut droit de la page ;
2. Variante représentée en vert, en haut à droite de la page sous la variante représentée en gris.

Pour chacune des variantes, l'utilisateur indique le nom de la variante en indiquant un libellé porteur d'une sémantique (figure 6.6 - encadré bleu). Dans notre exemple, la variante représentée en gris est nommée « coinHautDroit » par l'utilisateur.

Le code de l'opérateur de position à intégrer dans la description grammaticale est alors généré (figure 6.6 - encadré jaune) en utilisant le libellé fourni par l'utilisateur. Ce code permet de définir l'opérateur de position dans la description grammaticale. Il comprend les coordonnées des coins haut gauche et bas droit définissant sa boite englobante ainsi que les coordonnées de son point d'ancrage.

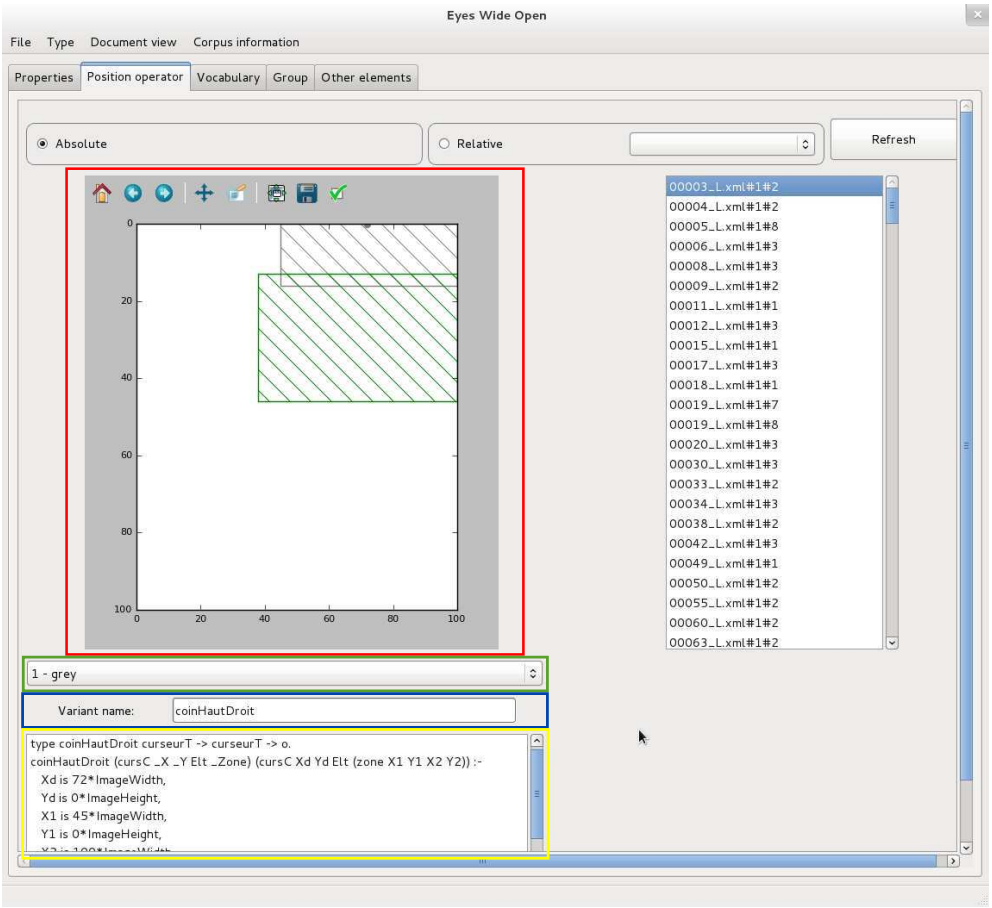


FIG. 6.6 – Exemple d'inférence d'un opérateur de position absolu : la méthode EWO représente les différentes variantes détectées dans une page normalisée (encadré rouge). L'ordre optimal inféré est indiqué à l'utilisateur (encadré vert). L'utilisateur nomme chacune des variantes détectées par la méthode (encadré bleu) et le code correspondant est automatiquement généré (encadré jaune).

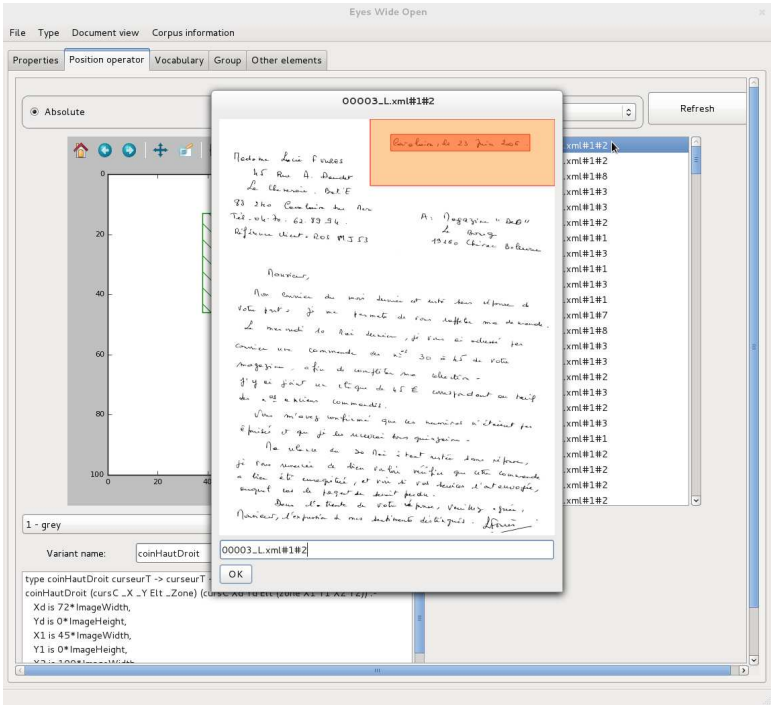


FIG. 6.7 – Exemple d'instanciation d'un opérateur de position absolu sur un document

6.4 Bilan

Dans ce chapitre, nous avons présenté le mode d'interaction entre la méthode EWO et l'utilisateur. Dans notre méthode, l'utilisateur est sollicité le moins possible afin d'avoir une méthode d'extraction de la connaissance pour la description de système de reconnaissance de documents la plus générique et automatique possible. L'utilisateur est sollicité afin d'apporter du sens aux structures de données détectées automatiquement dans le corpus traité.

Les questions posées à l'utilisateur sont principalement de deux types : 1) « Faut-il conserver les données présentées ? » 2) « Quel libellé porteur de sens peut être attribué au groupe de données visualisé ? ». Cette interaction limitée avec l'utilisateur nous permet de constituer l'ensemble de données utilisées pour l'inférence des règles, de déterminer la structure logique des documents à reconnaitre et d'inférer les positionnements.

Conclusion de la deuxième partie

Dans cette partie, nous avons présenté EWO, une nouvelle méthode de construction d'un système de reconnaissance complet avec apprentissage semi-automatique et interactif. Cette étape d'apprentissage permet la modélisation de connaissances complexes sur des données hétérogènes, bidimensionnelles en grande quantité. La méthode proposée est indépendante de tout système de reconnaissance de documents syntaxique.

Décomposition en sous-problèmes

La méthodologie complète repose sur une décomposition en sous-problèmes de la description globale du document et une résolution progressive des sous-problèmes (cf. figure 3.3). Chaque sous-problème est alors résolu manuellement par l'utilisateur si ses connaissances sont suffisantes, ou par apprentissage semi-automatique et interactif à l'aide de la méthode EWO si ses connaissances ne sont pas suffisantes et/ou que la variabilité des données est trop importante. La résolution de chaque sous-problème permet de générer des règles à intégrer dans la description grammaticale complète ainsi que des données étiquetées qui viennent s'ajouter aux données déjà connues. Ces données augmentent nos connaissances sur les documents et sont utilisables pour la résolution des sous-problèmes restants.

L'utilisation d'EWO permet d'avoir une vue exhaustive sur les documents à reconnaitre, contrairement à une approche manuelle qui se base en général sur un petit échantillon de documents. Cet échantillon n'est alors pas forcément représentatif de l'ensemble des documents à traiter. La méthode EWO permet de plus un apprentissage sans vérité terrain annotée manuellement. La méthodologie d'acquisition des données est réutilisable quelle que soit la méthode utilisée pour la reconnaissance des documents.

Inférence automatique et interactive de règles

La méthode EWO se décompose en trois étapes principales :

1. Acquisition des données ;
2. Inférence des règles ;
3. Assistance à l'intégration dans une description grammaticale.

Ces trois étapes nous permettent de proposer une solution complète partant des documents à analyser pour arriver aux règles qui permettront de résoudre le sous-problème

considéré. L'inférence globale d'une règle est faite progressivement en inférant la structure logique, puis en inférant les positionnements et les propriétés physiques. Pour ce faire, la méthode EWO repose sur deux éléments majeurs : l'émergence automatique de structures grâce à un algorithme de clustering et une interaction avec l'utilisateur pour donner un sens à ces structures détectées automatiquement.

Émergence automatique de structures

Le clustering permet de donner une vision à la fois synthétique et exhaustive des données. Comme nous l'avons montré dans cette partie, nous nous sommes attachés à minimiser l'intervention de l'utilisateur pour la réalisation du clustering, notamment en minimisant le nombre de paramètres fixés manuellement. En effet, il faut déjà avoir de bonnes connaissances sur les données pour être capable de fixer les paramètres d'un algorithme de clustering. Nous avons pour cela utilisé l'EAC clustering [FJ02].

L'EAC clustering présente plusieurs avantages nous permettant de répondre aux contraintes que nous nous sommes fixées. De par son approche ensembliste, l'EAC clustering n'impose pas de formes aux clusters détectés. Cela nous garantit une bonne adaptabilité quel que soit le jeu de données à analyser ce qui favorise la généricité de notre approche. De plus, l'EAC clustering propose une méthode pour déterminer le nombre de clusters de la partition automatiquement. Cela nous permet donc de minimiser l'intervention de l'utilisateur lors de la construction de la partition.

Apporter du sens aux données

L'émergence de structures de données est réalisée de manière la plus automatique possible dans la méthode EWO. Cependant, pour pouvoir utiliser ces structures de données pour réaliser l'inférence des règles, il nous est indispensable de solliciter l'utilisateur. Ces sollicitations vont nous permettre d'apporter du sens aux données détectées automatiquement.

Cependant, si l'interaction avec l'utilisateur est indispensable, nous nous sommes attachés à la minimiser. Son rôle ici est de répondre principalement à deux types de question : « Faut-il conserver les données présentées ? » et « Quel libellé porteur de sens peut être attribué au groupe de données visualisé ? ». Ces sollicitations de l'utilisateur sont utilisées lors de la construction de l'ensemble de données étiquetées et lors de la détection des variations logiques pour l'inférence de la structure.

Nous allons montrer dans la partie suivante trois descriptions grammaticales construites à l'aide de la méthode EWO. Ces descriptions grammaticales permettent de valider notre méthode d'apprentissage semi-automatique et interactif de règles, avec et sans vérité terrain.

Troisième partie

Expérimentations

Introduction

Nous avons présenté le fonctionnement de notre méthode, dont le but est de permettre une inférence interactive et automatique des règles, avec ou sans vérité terrain, pour la spécification de systèmes de reconnaissance de documents. Nous présentons maintenant des applications de notre méthode.

Ces travaux nous permettent de valider séparément les différents étapes mises en œuvre dans la méthode EWO ainsi que leur fonctionnement global pour la spécification de système de reconnaissance de documents complet. L'ensemble de ces applications est récapitulé dans le tableau 6.1.

Dans le chapitre 7, nous introduisons la méthode DMOS qui est le système syntaxique de reconnaissance de la structure de documents que nous utilisons pour les expérimentations.

La première application, présentée dans le chapitre 8 se place dans le cadre de la reconnaissance de courriers manuscrits en français. Dans ce contexte, nous présentons les résultats obtenus de deux expérimentations. La première de ces expérimentations porte sur l'évaluation seule de l'inférence automatique des opérateurs de position. La seconde expérimentation vise à valider la méthode complète EWO dans le cadre de l'utilisation d'une vérité terrain annotée par des opérateurs humains.

Dans le chapitre 9, nous présentons les résultats obtenus sur le corpus de la campagne internationale Maurdor, lors de la tâche d'étiquetage logique des documents. Pour cette campagne, la méthode EWO est utilisée dans le cadre de documents avec une vérité terrain annotée disponible et où la segmentation des éléments à reconnaître est connue. Cette application nous permet de valider les fonctionnalités d'inférence des règles (cf. tableau 6.1).

Dans le chapitre 10, nous présentons les expérimentations effectuées avec la méthode EWO sur un corpus sans vérité terrain, les registres de mariages mexicains. Pour cette application, nous avons utilisé une seule source de données pour l'extraction des primitives, des mots-clés extraits avec des modèles de points d'intérêt (POI). Cette application nous permet de valider les fonctionnalités d'acquisition des données sans vérité terrain et d'inférence des règles (cf. tableau 6.1).

Nous attirons l'attention du lecteur sur le fait que l'évaluation de l'apport de la méthode et du coût de l'interaction sont difficiles à quantifier. En effet, il faudrait pouvoir évaluer le temps passé avec la méthode EWO pour concevoir une description grammaticale par rapport au temps passé pour une description manuelle des mêmes documents. Pour la constitution de la pseudo vérité terrain, si nous comptabilisons des

actions abstraites (prise de décision pour un cluster, annotation d'un éléments, etc.), nous allons alors comparer des grandeurs différentes ce qui doit être fait avec précaution. Lorsqu'un système de reconnaissance syntaxique décrit manuellement pré-existait, nous comparons les performances obtenues par notre système avec celles du système existant. Nous nous assurons ainsi que les performances sont au moins comparables en utilisant la méthode EWO pour inférer les règles de la description grammaticale, tout en réduisant le temps nécessaire à la définition des règles.

Corpus	Acquisition des données			Inférence des règles			Intégration dans une description grammaticale	
	Avec vérité terrain		Sans vérité terrain	Variations logiques	Opérateurs de position	Propriétés physiques	Amélioration de la définition des règles	Ordonnancement automatique
	Brute	Augmentation						
RIMES (chapitre 8)		×		×	×	×	×	×
Maurdor (chapitre 9)	×			×	×	×	×	
Mariages mexicains (chapitre 10)			×	×	×			

TAB. 6.1 – Synthèse des expérimentations réalisées par corpus des éléments de EWO validés

Chapitre 7

Méthode DMOS

Pour les expérimentations, nous nous basons sur une méthode syntaxique déjà existante pour laquelle nous utilisons la méthode EWO dans le processus d'écriture de la description grammaticale. Nous proposons de travailler dans le contexte de la méthode DMOS-P (Description et MOdification de la Segmentation avec vision Perceptive), développée dans l'équipe Intuidoc par Coüasnon [Cou06] et Lemaitre [LL08] pour l'introduction de la vision perceptive. Cette méthode est une méthode syntaxique générique permettant la reconnaissance de documents structurés. La méthode DMOS-P repose sur une description spécifique des contenus à extraire, reconnaître et structurer pour un type de documents. Un programme d'interprétation dédié est ensuite généré automatiquement à partir de cette description.

L'architecture globale de la méthode DMOS-P est présentée dans la figure 7.1.

La description symbolique du contenu de la page est réalisée à l'aide d'un langage grammatical déclaratif appelé EPF (*Enhanced Position Formalism*). Le formalisme grammaticale EPF permet d'effectuer une description graphique, syntaxique et sémantique d'un type de documents. Cette description forme le niveau symbolique de la méthode et contient la connaissance spécifique à chaque type de documents. La grammaire définie en EPF utilise pour terminaux des primitives extraites directement dans l'image : les segments et les composantes connexes à différents niveaux de résolution. Les primitives peuvent également être des éléments plus complexes comme des lignes de texte, des traits dans l'image ou tout résultat d'une analyse précédente. Une fois la description réalisée dans le langage EPF, l'analyseur associé est produit automatiquement par une étape de compilation.

La particularité de cette méthode est donc de séparer la connaissance liée à chaque type de document, du noyau. La généricité de cette méthode a été validée sur de nombreux types de documents : partitions musicales, tableaux, formulaires, documents d'archives, et à grande échelle, sur plus de 700 000 documents.

Des extensions de la méthode DMOS ont été proposées :

- l'intégration de mécanismes perceptifs (méthode DMOS-P) pour améliorer la détection de traits effacés [LCC07] et de lignes de texte [LCC09] ;
- la création d'un analyseur dédié aux tableaux complexes et dégradés [MC05] ;

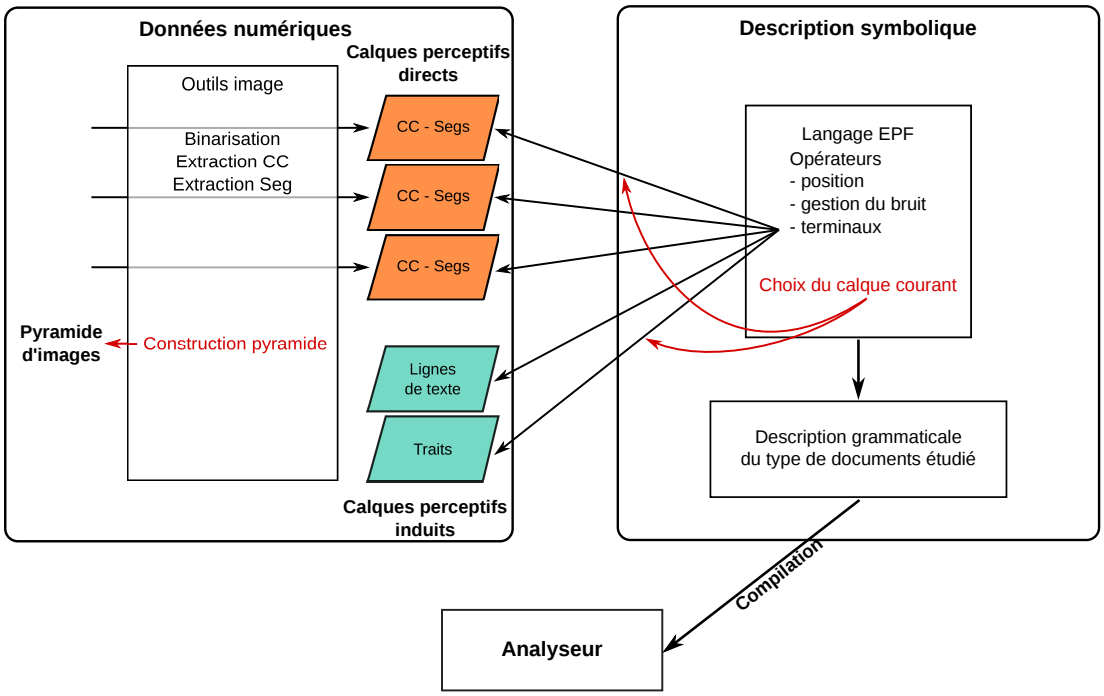


FIG. 7.1 – Architecture globale de la méthode DMOS-P

– l'intégration de mécanisme statistiques pour obtenir une grammaire localement stochastique [MCL11] ;
– la prise en compte du fonds documentaire pour tirer profit des connaissances contextuelles dans le processus de reconnaissance [Cha13].

La méthode EWO présentée dans cette thèse est indépendante de la méthode DMOS-P. La méthode EWO pourrait être utilisée sur un autre système de reconnaissance de structures de documents. DMOS-P sert ici de plateforme de validation. La méthode EWO intervient au niveau de la description grammaticale des documents à étudier. Le concepteur de la grammaire peut alors décrire les documents de manière semi-automatique et interactive, contrairement à la description manuelle qui était faite auparavant.

Chapitre 8

Avec vérité terrain : courriers manuscrits

La première application présentée consiste en l'évaluation de l'efficacité de la méthode EWO pour des corpus avec une vérité terrain. Nous avons pour cela travaillé sur la reconnaissance de courriers manuscrits en français. Les documents utilisés sont ceux du projet RIMES : Recherche et Indexation de données Manuscrites et de facsimilES [GCG+06]. Le corpus de documents est un corpus de documents *homogènes* pour lesquels la *vérité terrain est connue*. Nous comparons notre approche combinant description grammaticale et inférence statistique avec les résultats obtenus par des méthodes purement syntaxiques et des méthodes purement statistiques sur le corpus RIMES.

Dans ce chapitre, nous présentons d'abord le projet RIMES et la tâche de reconnaissance de courriers manuscrits. Puis nous présentons l'évaluation réalisée sur la pertinence de l'inférence des opérateurs de position par la méthode EWO. Nous présentons ensuite les résultats obtenus lors de l'inférence d'une description grammaticale complète pour la reconnaissance des courriers manuscrits.

8.1 Présentation des données

Le projet RIMES porte sur la reconnaissance de courriers manuscrits en français. Le concours organisé dans le cadre de ce projet comportait plusieurs tâches : structuration sémantique, reconnaissance d'écriture manuscrite, reconnaissance de scripteur, reconnaissance de logo et extraction d'informations. Dans la thématique structuration sémantique, nous nous intéressons ici en particulier à la reconnaissance de la structure de courriers manuscrits.

La tâche de reconnaissance de la structure des courriers manuscrits consiste à segmenter et à étiqueter les huit types de blocs suivants (figure 8.1) :

- coordonnées de l'expéditeur (magenta) ;
- coordonnées du destinataire (cyan) ;
- date et lieu (rouge) ;

- objet (vert clair) ;
- ouverture (jaune) ;
- corps de texte (orange) ;
- signature (vert foncé) ;
- PS et pièce jointe (bleu).

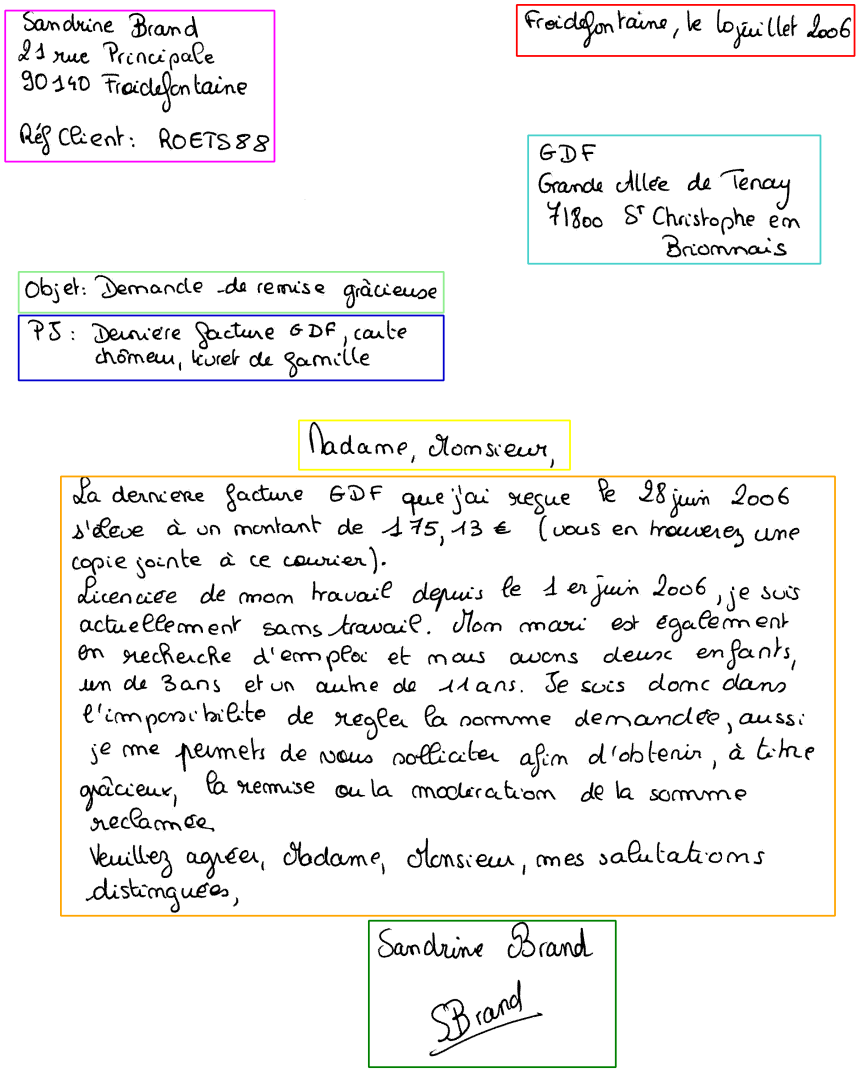


FIG. 8.1 – Blocs à localiser et à étiqueter dans les courriers manuscrits (voir les significations des couleurs dans la partie 8.1)

Le corpus RIMES est composé de 1250 documents. Une vérité terrain produite manuellement est disponible sur les documents du corpus RIMES. Cette vérité terrain contient la boite englobante de chaque élément logique avec son étiquette ainsi que la transcription du texte contenu dans chaque boite englobante.

La reconnaissance des courriers manuscrits doit faire face aux difficultés rencontrées dans les documents manuscrits non contraints. En effet, même si les usages français définissent la manière d'organiser un courrier, cette organisation n'est pas toujours respectée dans les courriers manuscrits. Les coordonnées expéditeur peuvent par exemple être placées en haut à droite du courrier alors que les conventions nous indiquent de les placer en haut à gauche.

La tâche à laquelle nous nous consacrons est la segmentation de la page en blocs de texte avec une seule fonction logique. Il est indispensable de pouvoir prendre en compte la structure logique pour réaliser la segmentation physique.

8.2 Évaluation

8.2.1 Métrique

Pour évaluer nos résultats, nous avons utilisé la métrique du concours RIMES, dans sa deuxième version proposée en 2008 [GCBG09]. Cette métrique permet de produire un taux global d'erreur sur l'étiquetage complet d'un ensemble de documents.

Le taux d'erreur est calculé en comparant les étiquettes attribuées à chacun des pixels de l'image avec celles contenues dans les vérités terrain. Seuls les pixels noirs sont pris en compte, dans une image binarisée. Le taux d'erreur global représente le taux de pixels noirs de l'image ayant été mal étiquetés.

8.2.2 Méthodes existantes

Plusieurs méthodes ont été créées pour répondre à la problématique du concours RIMES, la reconnaissance de courriers manuscrits en français. Nous présentons ici quatre systèmes complets différents, deux systèmes statistiques et deux systèmes syntaxiques :

- un système purement syntaxique décrit avec la méthode DMOS (Description and Modification of Segmentation) proposé par Lemaitre [LCC08]. Ce système à base de règles a été décrit intégralement manuellement ;
- un système utilisant la méthode DMOS combinant une approche structurelle et une approche statistique, proposé par Maroneze [MCL11]. La description structurelle décrite manuellement par Lemaitre [LCC08] est utilisée en la combinant avec une approche stochastique à l'aide de l'opérateur FIND_BEST_FIRST pour la recherche de la meilleure proposition pour l'élément « ouverture » ;
- une approche basée sur les champs aléatoires de Markov en utilisant des caractéristiques structurelles et spatiales, complétée par un post-traitement, proposée par Lemaitre [LGGP07] ;

– une approche basée sur les CRF (*Conditional Random Fields*) combinant un modèle CRF pour diviser le document et un autre modèle de CRF pour assigner une étiquette logique à chaque ligne, proposée par Montreuil [MNGH10].

8.3 Opérateurs de position

Dans cette partie, nous voulons montrer l'efficacité de la méthode EWO pour l'inférence d'opérateurs de position dans une méthode syntaxique. Nous rappelons qu'avant l'introduction de la méthode EWO, les opérateurs de position étaient décrits manuellement par l'utilisateur.

8.3.1 Approche

L'inférence des opérateurs de position a été validée sur les documents du corpus RIMES indépendamment du fonctionnement global de la méthode EWO. Pour ce faire, nous avons remplacé les opérateurs de position définis manuellement par des opérateurs de position inférés de manière automatique dans une grammaire existante. Des exemples d'opérateurs de position ont été donnés dans la section 4.2.2.1. La grammaire utilisée est celle soumise lors du concours RIMES de 2008 par Lemaitre [LCC08]. Pour l'inférence des opérateurs de position, nous nous sommes basés sur la vérité terrain disponible pour le corpus de documents RIMES.

Nous avons utilisé les mêmes bases d'apprentissage et de test que celles de Lemaitre [LCC08]. L'ensemble d'apprentissage est composé de 300 lettres manuscrites, l'ensemble de test est composé de 950 documents.

8.3.2 Résultats

Les résultats obtenus, présentés dans le tableau 8.1, valident l'utilisation de la méthode EWO pour l'inférence des opérateurs de position. En effet, l'utilisation de la méthode EWO pour l'inférence des opérateurs de position permet de diminuer le nombre de paramètres à définir manuellement tout en diminuant le taux d'erreur global. Cette expérimentation montre que l'apprentissage automatique des opérateurs de position permet effectivement d'améliorer le rappel sans impacter la précision.

	Opérateurs de position	
	[LCC08]	[LCC08] + opérateurs inférés
Nombre de paramètres manuels	102	66
Nombre de paramètres automatiques	0	**48**
Taux d'erreur global	11,34	**9,78**

TAB. 8.1 – La méthode EWO permet une diminution du nombre d'opérateurs de position définis manuellement tout en diminuant le taux d'erreur global (résultats obtenus sur 950 documents)

L'ensemble des opérateurs de position n'a pas été appris dans cette approche car nous n'avons pas procédé à une augmentation de la vérité terrain (cf. section 4.1.4). Nous n'avons alors pas pu inférer l'ensemble des opérateurs de position portant sur les lignes de texte, granularité non disponible dans la vérité terrain. Ceci explique la présence de 66 paramètres décrits manuellement dans notre version de la description grammaticale.

Les résultats montrent donc la pertinence et l'efficacité de l'inférence automatique des opérateurs de position proposée par la méthode EWO. Le temps nécessaire à leur description est fortement *réduit* tout en offrant de *meilleures performances* que les opérateurs de position définis manuellement, le taux d'erreur passant de 11,34 à 9,78.

8.4 Grammaire complète

Dans cette partie, nous cherchons à démontrer l'intérêt de la méthode EWO pour la *description d'un système complet*. Nous nous plaçons dans le cadre spécifique de documents avec une *vérité terrain annotée* pour lesquels nous avons procédé à une *augmentation de la vérité terrain* (cf. section 4.1.4).

8.4.1 Approche

Une description grammaticale complète a été générée en utilisant la méthode EWO pour la reconnaissance de courriers manuscrits en français. Nous avons utilisé pour cela la vérité terrain manuellement annotée fournie dans le cadre du projet RIMES. Cette vérité terrain a été augmentée avec des *lignes de texte* et des *composantes connexes* selon le mécanisme présenté dans la section 4.1.2. Ces primitives d'analyse de la grammaire ont notamment été utilisées pour résoudre les problèmes de segmentation.

Chacun des huit types d'éléments logiques décrivant un courrier a été analysé indépendamment avec la méthode EWO. Cette analyse a permis l'inférence des variations logiques ainsi que l'inférence de la structure physique (opérateurs de position et propriétés physiques). La méthode EWO a également été utilisée pour déterminer l'ordre des règles de la description grammaticale comme nous l'avons présenté dans la section 4.3.2.

Comme dans le système présenté par Maroneze [MCL11], nous nous sommes appuyés sur un classifieur pour la reconnaissance de l'élément logique « ouverture ». En effet, l'analyse du vocabulaire dans la méthode EWO nous a confirmé le vocabulaire très restreint utilisé dans le corpus. De plus, les simulations de la règle grammaticale effectuées sous la forme de requêtes dans la méthode EWO nous ont montré que les propriétés physiques seules ne nous permettent pas d'identifier correctement l'ouverture. Cependant, contrairement à Maroneze nous n'avons pas utilisé l'opérateur FIND_BEST_FIRST. En effet les informations inférées sur la structure physique grâce à la méthode EWO limitent naturellement la confusion.

8.4.2 Résultats

Les résultats présentés dans le tableau 8.2 comparent les taux d'erreur du système de reconnaissance produit en utilisant la méthode EWO et des quatre systèmes présentés dans la section 8.2.2 sur la base de test de la compétition RIMES (100 documents).

Système	Taux d'erreur
Approche syntaxique (DMOS) [LCC08]	8,97
Approche syntaxique (DMOS) et FBF [MCL11]	5,53
Approche statistique basée sur un MRF [LGGP07]	8,53
Approche statistique basée sur des CRF [MNGH10]	6,33
Méthode EWO	**5,82**

Tab. 8.2 – Taux d'erreur obtenus sur la base de test de la compétition RIMES (100 documents) pour la tâche d'analyse du document

Le tableau 8.2 montre que le système de reconnaissance construit en utilisant la méthode EWO obtient des résultats comparables à ceux obtenus avec le meilleur système statistique [MNGH10] et à ceux obtenus avec le meilleur système syntaxique décrit manuellement [MCL11]. En comparaison avec une approche purement syntaxique, notre méthode permet de définir plus rapidement la description grammaticale. En comparaison avec la méthode statistique, notre méthode permet d'assurer la cohérence des blocs de texte.

Afin de compléter ces résultats obtenus sur la base de test assez restreinte de la compétition RIMES (100 documents), le tableau 8.3 présente les résultats obtenus sur les ensembles d'apprentissage, de validation et de test. Les résultats sur les ensembles d'apprentissage et de validation ne sont pas disponibles pour les autres méthodes évaluées lors de la compétition. Les performances stables sur ces trois ensembles de documents montrent la fiabilité du système défini en utilisant la méthode EWO. Nous n'avons pas effectué de validation croisée car la méthode EWO nécessite une interaction avec l'utilisateur. Celui-ci acquiert des connaissances à chaque utilisation de la méthode EWO sur le corpus. Nous ne pourrions donc pas considérer la phase d'apprentissage sur un échantillon comme indépendante de celles effectuées sur les autres échantillons.

Base	Nombre de documents	Taux d'erreur
Apprentissage	900	5,7
Validation	250	5,3
Test	100	5,8

Tab. 8.3 – Taux d'erreurs obtenus en utilisant la méthode EWO sur les bases d'apprentissage, validation et test de la compétition RIMES pour la tâche d'analyse de documents

Les résultats prouvent l'efficacité de la méthode EWO à produire de manière semi-automatique et interactive une description grammaticale complète. L'utilisation de la vérité terrain augmentée a permis de résoudre efficacement la tâche de segmentation

des blocs de texte. Le système ainsi produit dépasse les limites des approches actuelles. En effet, contrairement aux deux méthodes syntaxiques, il n'y a pas eu de description manuelle des systèmes et l'utilisateur a eu une vue exhaustive sur les documents à traiter. De plus, contrairement aux deux méthodes statistiques présentées, le système produit avec la méthode EWO assure la cohérence des blocs de texte, permet de gérer les cas rares et d'introduire facilement de la connaissance.

8.5 Apports de notre approche

L'application présentée nous permet de valider la méthode EWO dans le cadre d'une utilisation sur un corpus avec vérité terrain. Nous avons évalué indépendamment l'apport de la méthode EWO pour l'inférence des opérateurs de position. Nous avons également mis en avant les avantages de la méthode EWO pour l'inférence d'une description grammaticale complète. Pour cette tâche, nous avons de plus validé l'intérêt de l'augmentation de la vérité terrain.

La comparaison avec des opérateurs de positions décrits manuellement permet de montrer l'efficacité de l'inférence des opérateurs de position. En effet, les résultats obtenus avec les opérateurs de position inférés sont meilleurs que ceux obtenus avec les opérateurs de position définis manuellement. Dans le même temps, le nombre de paramètres devant être définis manuellement par l'utilisateur est diminué, réduisant le temps nécessaire à la définition de la description grammaticale.

La comparaison des résultats obtenus par le système de reconnaissance produit avec la méthode EWO et les autres systèmes purement syntaxiques et statistiques de la littérature a montré la pertinence de notre approche. En effet, nous avons obtenu des résultats comparables à ceux des meilleurs systèmes de chaque catégorie tout en réduisant le temps nécessaire à la création de la description grammaticale. Le système produit présente de plus certains avantages par rapport aux systèmes statistiques présentés. En effet, il est compréhensible par un être humain, capable de décrire une structure hiérarchique complexe, assure la cohérence des blocs et est capable de gérer les cas rares.

Chapitre 9

Documents hétérogènes : le corpus MAURDOR

La seconde application présentée porte sur un corpus de documents hétérogènes (type de documents, langue des documents, manuscrits et dactylographiés). Pour cela, nous avons participé à la tâche d'extraction de la structure logique de la campagne MAURDOR [BGG+14]. Sur ce corpus, nous avons évalué l'apport de la méthode EWO pour la description de la structure logique et physique de documents et pour l'amélioration de la définition des règles dans le cas d'un corpus avec vérité terrain. Notre méthode a été utilisée pour la participation au deuxième tour de la campagne d'évaluation, en décembre 2013. Une vérité terrain détaillée est connue pour chacun des documents (position spatiale, langue, transcription, etc.).

La particularité de la tâche d'extraction de la structure logique (module 5) dans le cadre de la campagne MAURDOR est que la *segmentation des documents est déjà connue* et est fournie en entrée du système construit. La tâche se rapproche alors d'une tâche de classification. Nous comparons les résultats obtenus par notre approche statistique avec ceux obtenus par l'autre système présenté lors de la campagne. Ce second système se base sur une approche syntaxique avec un classifieur SVM multiclasses et des règles simples décrites manuellement. Cette comparaison nous permet de valider les apports de notre méthode EWO pour la description de règles, apportant une vision exhaustive sur les données.

Dans ce chapitre, nous présentons d'abord le contexte de la campagne Maurdor et la tâche d'extraction de la structure logique. Nous présentons ensuite les résultats obtenus avec la description grammaticale obtenue en utilisant la méthode EWO.

9.1 Présentation du corpus

Nous présentons d'abord le corpus de la campagne Maurdor, puis la tâche d'extraction de la structure sur laquelle nous nous sommes focalisés.

9.1.1 Campagne MAURDOR

Le corpus de la campagne MAURDOR contient des documents mixtes (dactylographiés et manuscrits) très hétérogènes. Les documents sont regroupés en cinq catégories :

- C1 : Formulaire imprimé (rempli en manuscrit) ;
- C2 : Document commercial, privé ou professionnel, imprimé ou photocopié (devis, bon de commande, bordereau de livraison ou facture, page de catalogue, tract politique ou commercial, article de presse, contrat ou document juridique, document administratif ou officiel, note de frais, ticket de caisse, récépissé, chèque, reçu bancaire, carte, plan, dessin, confirmation de réservation de transport ou d'hébergement, etc.) ;
- C3 : Correspondance privée manuscrite sur papier libre ou à en-tête, carte de félicitation, d'invitation, de remerciements, page de cahier ou de bloc-notes, note manuscrite ou « post-it », etc.) ;
- C4 : Correspondance privée ou professionnelle dactylographiée (courrier dactylographié, note de service, ordonnance médicale, impression de courriel, page de garde de fax ;
- C5 : Autres types de documents : plan, schéma, dessin, croquis réalisé « à main levée », tableau de chiffres ou de codes alphanumériques, échec de numérisation ou d'impression/document illisible ou inexploitable.

Le tableau 9.1 présente la répartition des 10 000 documents du corpus dans les différentes catégories.

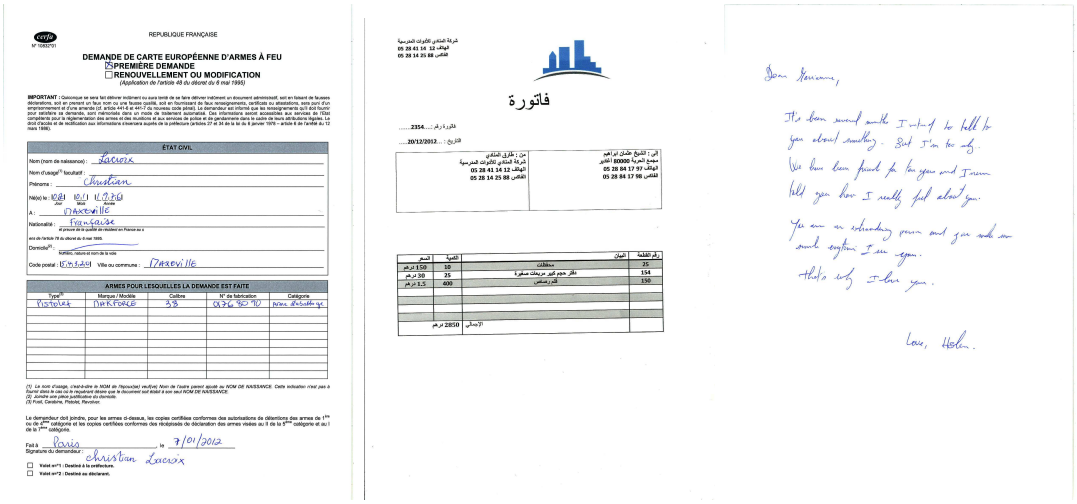
Catégorie	C1	C2	C3	C4	C5
Fréquence	12%	40%	25%	20%	3%

TAB. 9.1 – Répartition des documents du corpus MAURDOR dans les différentes catégories

Les polices et la taille des caractères des documents dactylographiées sont très variables et les documents ont été numérisés selon différentes méthodes. Ces documents peuvent contenir des logos, des tampons d'entreprises ou de particuliers, des tableaux ou des graphiques en plus des zones de texte. La langue principale du document peut être l'anglais, l'arabe ou le français. La répartition entre les différentes langues est présentée dans le tableau 9.2. Cette répartition est respectée autant que possible dans chaque catégorie de documents. Il est possible que les documents contiennent des éléments de texte dans une autre langue.

Langue	Anglais	Arabe	Français
Fréquence	25%	25%	50%

TAB. 9.2 – Répartition des documents du corpus MAURDOR dans les différentes langues



(a) C1 : Formulaire en français (b) C2 : Facture en arabe (c) C3 : Exemple de courrier manuscrit en anglais

(d) C4 : Exemple de courrier dactylographié en arabe (e) C5 : Exemple de dessin

FIG. 9.1 – Exemples de documents du corpus MAURDOR pour chacune des catégories du corpus

Selon les contraintes de la compétition MAURDOR, la reconnaissance complète des documents est décomposée en cinq modules. Les modules sont abordés séquentiellement comme représenté dans la figure 9.2. L'évaluation de chaque module est faite indépendamment des résultats obtenus pour les autres modules à l'exception de l'évaluation « end-to-end », qui consiste en l'évaluation d'un système complet contenant tous les modules. Pour permettre l'évaluation isolée d'un module, les résultats attendus des modules précédents sont communiqués aux participants.

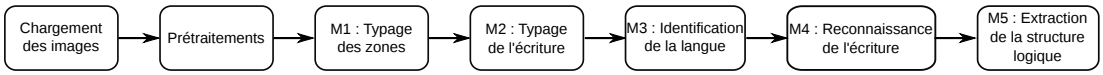


FIG. 9.2 – Chaîne de traitement complète de la reconnaissance des documents pour la campagne MAURDOR

9.1.2 Module 5 : Extraction de la structure logique

L'extraction de la structure logique se décompose en plusieurs opérations : identification des fonctions des zones textuelles, de l'ordre de lecture et des groupes. Pour la participation au module 5 d'extraction de la structure logique, nous possédons en entrée du module les informations suivantes :

- l'orientation de la page ;
- le type de chaque zone (zone textuelle, zone graphique, etc.) ;
- pour les zones textuelles :
  - le type d'écriture (manuscrite ou dactylographiée),
  - la langue du texte,
  - la transcription du texte.

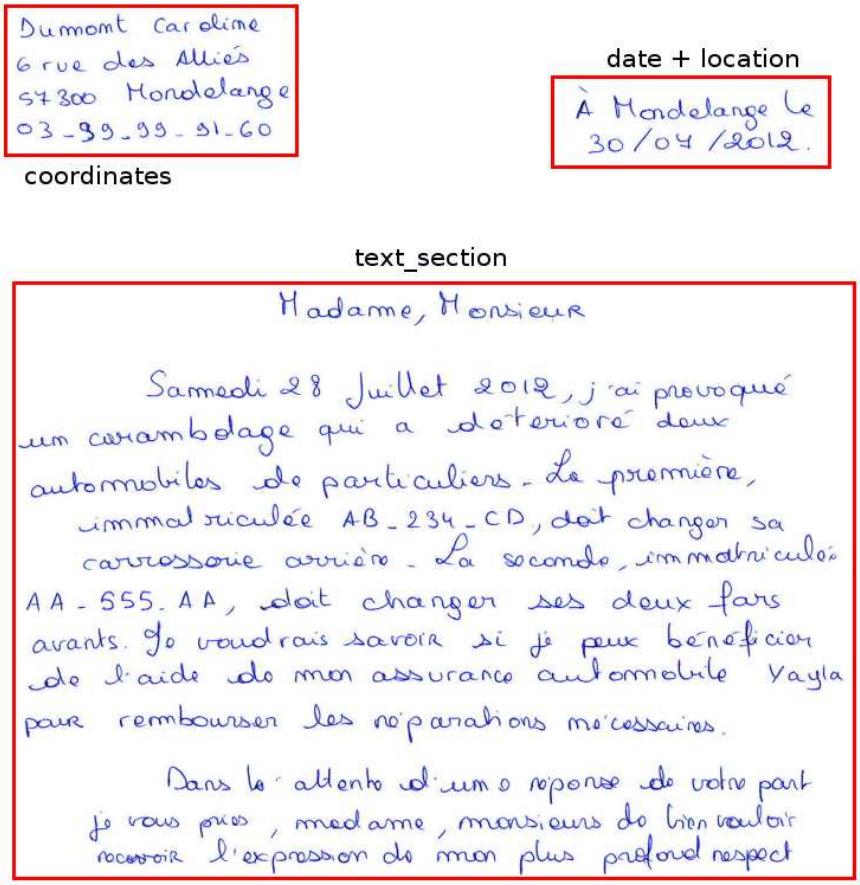
Fonctions des zones textuelles Les zones textuelles peuvent avoir *aucune, une ou plusieurs fonctions logiques* spécifiques que nous cherchons ici à identifier. Les différentes fonctions logiques à identifier sont :

- *title* (s'il s'agit du titre du document) ;
- *reference* (s'il s'agit de la référence du document) ;
- *date* (s'il s'agit de la date à laquelle le document a été rédigé) ;
- *location* (s'il s'agit du lieu où le document a été rédigé) ;
- *object* (si un objet explicite est spécifié) ;
- *header* (s'il s'agit d'un en-tête de page) ;
- *footer* (s'il s'agit d'un pied de page) ;
- *legend* (s'il s'agit de la légende d'un tableau ou d'un graphique ;
- *annotation* (s'il s'agit d'une annotation relative à une partie du document) ;
- *text_ section* (s'il s'agit d'un corps de texte).

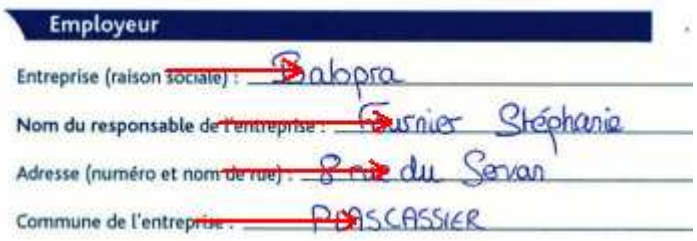
Un exemple d'étiquetage des zones textuelles est présenté dans la figure 9.3(a).

Ordre de lecture L'ordre de lecture doit être retrouvé pour certaines zones. C'est le cas notamment pour les réponses dans un formulaire qui sont ainsi liés à la question à laquelle elles répondent (figure 9.3(b)). Les colonnes d'un journal sont également liées entre elles grâce à l'ordre de lecture.

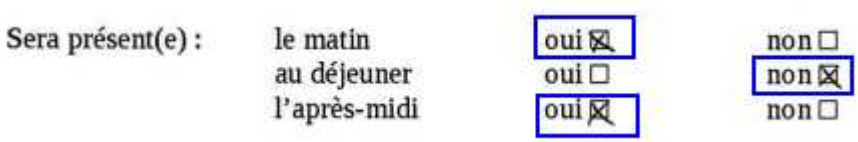
Groupes Certaines zones sont regroupées sans qu'il existe un ordre de lecture entre elles. C'est le cas par exemple d'un graphique et sa légende ou d'une case à cocher et du libellé qui lui est associé (figure 9.3(c)).



(a) Exemple d'étiquetage des fonctions des zones textuelles



(b) Exemple d'ordre de lecture



(c) Exemple de détermination de groupe

FIG. 9.3 – Exemples 9.3(a), 9.3(b) et 9.3(c) illustrant les trois tâches du module 5 d'extraction de la structure logique

9.2 Évaluation

Nous souhaitons ici évaluer l'intérêt de la méthode EWO dans le cadre d'un corpus hétérogène avec une vérité terrain richement détaillée. Nous validons ici l'*inférence de la structure logique* ainsi que la *description de la structure physique*. Nous n'avons pas procédé à une augmentation de la vérité terrain, notamment car les éléments à étiqueter sont les blocs *déjà segmentés*.

9.2.1 Base de documents

Le système de reconnaissance produit grâce à la méthode EWO a été évalué lors du second tour de la campagne MAURDOR. L'évaluation a été réalisée sur 1000 documents respectant les mêmes proportions que les échantillons utilisés pour l'apprentissage.

9.2.2 Métrique utilisée

La métrique de la compétition se base sur un score de zone. Chaque zone est définie par quatre caractéristiques qui peuvent être vides si aucune structure logique n'est rattachée à la zone :

- le sous-type sémantique contient une information du type corps de texte, légende... ;
- la zone qui précède, dans l'ordre de lecture, celle étudiée ;
- l'ensemble E des zones présentes dans un même groupe non-ordonné que la zone étudiée.

Chacune des caractéristiques donne lieu à un score compris entre 0 et 1. Pour les deux premières caractéristiques, 1 point est comptabilisé par réponse correcte. Pour la dernière, la moyenne harmonique (F-mesure) de la précision et du rappel est calculée après avoir ajoutée la zone étudiée aux ensembles hypothèse et référence. Chaque zone se voit attribuer un score de zone correspondant à la moyenne des trois scores précédemment obtenus. La moyenne de l'ensemble des scores de zones est calculée au niveau du document. Puis, la moyenne des scores des documents est calculée au niveau de la collection, ce qui correspond au score brut S_b.

S_b est ensuite normalisé en fonction de S_0. S_0 est le score obtenu par une hypothèse qui considèrerait que chaque zone n'a ni sous-type, ni prédécesseur, ni successeur et un ensemble E vide. Le score final, S, est alors le suivant :

$$\text{Si } S_b \leq S_n, S = 100 \times \frac{S_b - S_n}{S_n}, \text{ sinon } S = 100 \times \frac{S_b - S_0}{1 - S_0}$$

Le score final est donc compris entre -100 et 100. Il est positif si le système ajoute plus d'information qu'il n'ajoute d'erreurs.

9.2.3 Présentation des systèmes évalués

Deux systèmes ont été soumis pour la tâche 5 de la compétition MAURDOR. Notre système est basé sur une unique description grammaticale pour les trois opérations à

effectuer de la tâche 5. Le système 2, fourni par un autre participant, utilise quant à lui un classifieur combiné à des règles décrites manuellement.

9.2.3.1 Système produit avec la méthode EWO

Notre système est basé sur une description grammaticale des documents à reconnaître. Cette description grammaticale a été construite de manière semi-automatique en utilisant la méthode EWO. La méthode EWO est utilisée pour détecter automatiquement les variations logiques pour chacune des étiquettes logiques à attribuer. Par exemple, pour l'étiquette logique « text_section », la méthode EWO produit deux variantes logiques : les corps de texte de lettre et les colonnes d'articles de presse.

Afin de réaliser l'étiquetage des fonctions logiques, nous utilisons notamment le vocabulaire présent dans les zones à étiqueter pour sélectionner des zones candidates pour chaque étiquette logique. Pour cela, une analyse fréquentielle du texte contenu dans chaque zone est proposée au sein de la méthode EWO. Par exemple, la méthode EWO nous permet d'apprendre que les pieds de page sont des zones contenant le texte : « fax », « tel », « phone »ou « www. ».

Les zones candidates sont ensuite conservées ou non grâce à l'utilisation des propriétés physiques de la zone candidate (position dans la page, taille de la zone, langue du texte, etc.). Pour les pieds de page, nous apprenons la structure physique suivante grâce à la méthode EWO :

- dactylographié ;
- moins de 4 lignes ;
- largeur $< 70\%$ largeur page ;
- pas de texte en dessous.

Pour la détection de l'ordre logique, la méthode EWO nous donne une vue exhaustive des cas existants :

- peignes dans les formulaires ;
- colonnes d'articles.

Grâce à la méthode EWO, nous avons une vision très détaillée des différents cas présents, ce qui nous permet d'écrire des règles précises. Ainsi, nous ne recherchons pas simplement le texte le plus proche des champs de formulaires mais celui est qui est une réponse probable du libellé du champs.

Pour la détection des groupes, la vision détaillée des éléments à rechercher nous permet d'obtenir des règles détaillées sur les éléments à rechercher :

- case à cocher : recherche du texte associé avec les bonnes propriétés en terme de position, qui est dactylographié, sur un seul ligne et court ;
- signature : recherche du texte proche ne contenant pas de chiffre, écrit à la main, sur une seule ligne et contenant moins de 4 mots.

9.2.3.2 Système 2

Le système 2 propose différentes méthodes pour les trois opérations de la tâche 5. La détection des fonctions des zones textuelles se fait avec une approche statistique en deux étapes :

1. Rejet des zones textuelles sans fonction logique à l'aide d'un prétraitement et de KNN.
2. Affectation d'une fonction logique à l'aide d'un SVM multiclasses. La transcription n'est pas utilisée comme caractéristique de la zone de texte par le classifieur SVM.

La détection des groupes est fait par une description manuelle de règles :

1. Pour les cases à cocher, recherche des champs de formulaires avec un ratio hauteur/largeur spécifique et contenant un trait. Le texte associé est alors le texte le plus proche.
2. Pour les signatures, recherche du bloc de texte le plus proche de la signature et détection dans ce bloc d'un nom de famille (nombre de mots, lettres majuscules, etc.). La recherche des groupes contenant une signature n'est pas implémentée pour les documents en arabe.

La détection de l'ordre logique est également faite par une description manuelle de règles :

1. Identification des champs de formulaires remplis
2. Identification du libellé associé au champ (bloc de texte le plus proche)
3. Création d'un ordre de lecture entre le libellé et le texte du champ de formulaire

9.3 Résultats

Les résultats obtenus par les deux systèmes en compétition pour la tâche 5 de la campagne MAURDOR sont présentés dans le tableau 9.3. Notre système présente les meilleurs résultats sur la tâche « ordre de lecture » et des résultats similaires au système 2 sur la tâche « groupe ». Pour la tâche de typage des zones textuelles, notre système obtient de moins bons résultats que le système 2.

Système	Type	Ordre	Groupe
EWO	55	**45**	60
Système 2	**69**	28	61

Tab. 9.3 – Performances globales pour la tâche 5 au deuxième tour de la campagne MAURDOR

Les performances obtenues pour la tâche « ordre de lecture » s'expliquent par le fait que tous les cas ne sont pas traités par les règles décrites manuellement dans le système 2. En effet, seuls les champs de formulaires sont affectés avec un ordre de lecture. Or

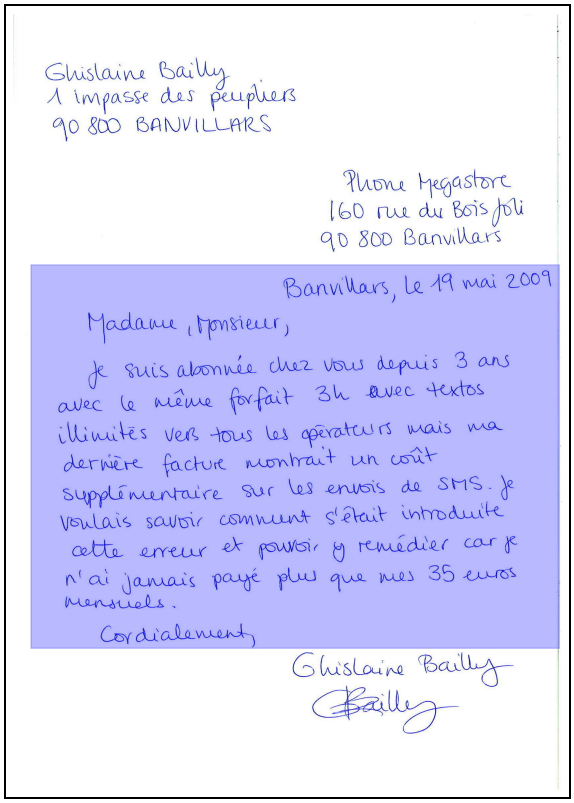


FIG. 9.4 – Exemple d'erreur de vérité de terrain détectée avec la méthode EWO : l'élément date a été mal segmenté

le corpus contient un autre type de zones avec un ordre de lecture : les colonnes de texte dans les pages de journaux. L'utilisation de la méthode EWO nous permet, en nous donnant une *vue exhaustive* sur le corpus, de détecter tous les cas présents dans le corpus. Ce n'est pas possible avec une approche manuelle.

Pour la tâche d'étiquetage des fonctions textuelles des zones, nous obtenons de moins bons résultats que le système 2. En effet, les données étant déjà segmentées cette tâche est en fait une tâche de *classification*. Les meilleures performances du système 2, qui utilise un algorithme de classification (SVM multi-classes), sont donc logiques dans ce contexte.

La vue exhaustive sur les données donnée par la méthode EWO nous a également permis de détecter des erreurs dans la vérité terrain grâce à la détection automatique des outliers (cf. section 4.2.1). Nous avons ainsi participé à la phase d'*adjudication* des données en détectant 164 erreurs qui ont été validées par les organisateurs du concours sur un sous-ensemble du corpus complet de 1 000 documents. La figure 9.4 présente un exemple d'erreur de vérité terrain détectée grâce à la méthode EWO. L'élément *date* a été mal segmenté et est détecté comme étant un élément atypique par la méthode EWO en raison de la hauteur de sa boite englobante.

9.4 Discussion

L'expérimentation présentée nous a permis de valider l'apport de la méthode EWO par rapport à une description manuelle des règles. En effet, pour la reconnaissance des groupes et pour l'ordre de lecture, qui se basent sur une description manuelle de règles dans le cas du système 2, nous obtenons des résultats similaires (groupe) ou meilleurs (ordre de lecture) que le système 2. Cela s'explique notamment par la vision exhaustive du corpus permise par la méthode EWO.

La tâche d'extraction de la structure logique de la campagne MAURDOR se place dans un contexte particulier où la segmentation des zones est déjà connue. La tâche d'extraction de la structure logique devient alors une tâche de classification et non plus une tâche de reconnaissance comme lorsque la segmentation et l'étiquetage doivent être faits conjointement. Dans ce contexte, notre méthode qui se base sur la description des propriétés propres des éléments est moins efficace qu'une méthode de classification. Ces résultats moins performants qu'une méthode de classification étaient prévisibles. Cependant, ce contexte de test n'est pas réaliste puisque les données sont toutes correctement segmentées. Dans le cadre d'une application réelle, certaines zones ne seraient pas correctement segmentées, ce qui compliquerait la tâche pour un classifieur.

Cette application démontre la capacité de la méthode EWO à donner une *vue exhaustive sur les documents* à analyser. Cette visibilité permet à un utilisateur humain de décrire rapidement et efficacement un système de meilleure qualité que s'il l'avait fait manuellement.

Chapitre 10

Sans vérité terrain : actes de mariages mexicains

Afin de montrer l'efficacité de notre approche lorsqu'il n'y a pas de vérité terrain connue sur les documents, nous proposons d'utiliser notre méthode pour l'analyse de registres de mariages mexicains. Les données sont issues du corpus de la compétition HIP2013 FamilySearch.

Nous comparons les résultats obtenus en inférant la description grammaticale sans vérité terrain avec ceux obtenus avec la description grammaticale définie manuellement et soumise au concours en 2013.

Nous présentons dans un premier temps le corpus HIP2013 FamilySearch. Puis nous détaillons la tâche que nous nous sommes fixée, qui diffère de la tâche du concours. Nous présentons ensuite la phase de création de la pseudo vérité terrain ainsi que la construction de la description grammaticale. Les résultats obtenus avec cette description grammaticale sont évalués par rapport à ceux de la description grammaticale manuelle. Enfin, nous concluons sur les apports de notre méthode sur ce corpus et proposons une évaluation du coût de construction de la pseudo vérité terrain.

10.1 Présentation des données

10.1.1 Concours HIP2013 FamilySearch

La tâche du concours HIP2013 FamilySearch consiste à détecter quatre régions d'intérêt dans des actes de mariages mexicains d'archives (XX^{ème} siècle). Ces régions d'intérêt sont des champs manuscrits dans un texte pré-imprimé et correspondent aux informations suivantes (figure 10.1) :

1. Mois de mariage ;
2. Année de mariage ;
3. Ville d'origine de l'époux ;
4. Ville d'origine de l'épouse.

Après avoir détecté les régions d'intérêt, les participants doivent regrouper les images selon le contenu textuel des régions d'intérêt. Chaque image est regroupée quatre fois, une fois pour chacune des régions d'intérêt. Il n'est par contre pas nécessaire de reconnaitre le contenu manuscrit des régions d'intérêt.

Pour la compétition, un jeu de données d'apprentissage de 10 000 documents était fourni aux participants. Une vérité terrain était fournie pour chaque document sous la forme du contenu textuel des régions d'intérêt. La vérité terrain ne contient pas la localisation des régions d'intérêt Par exemple, pour le registre de la figure 10.1, la vérité terrain est la suivante :

```
mayo
cuarenta y seis
Ixtacalco, Distrito Federal
Ixtacalco, Distrito Federal
```

Le jeu de données de test est composé de 20 000 images.

10.1.2 Sous-tâche du concours

Dans notre expérimentation, nous nous sommes focalisés sur une sous-tâche de la compétition : la localisation des régions d'intérêt *mois* et *année*. En effet, nous désirons évaluer notre méthode indépendamment des performances de la méthode de clustering sur l'écriture manuscrite qui vient en post-traitement du système de reconnaissance de structure de documents. Nous cherchons donc à localiser les zones d'intérêt, sans évaluer la reconnaissance de leur contenu. La zone est détectée qu'il y ait un écrit présent ou non dans le champ. Pour cette tâche, il n'y a pas de vérité terrain disponible sur le corpus du concours HIP2013 FamilySearch. L'apprentissage a été réalisé sur 7 000 documents du jeu de données d'apprentissage sans vérité terrain du concours.

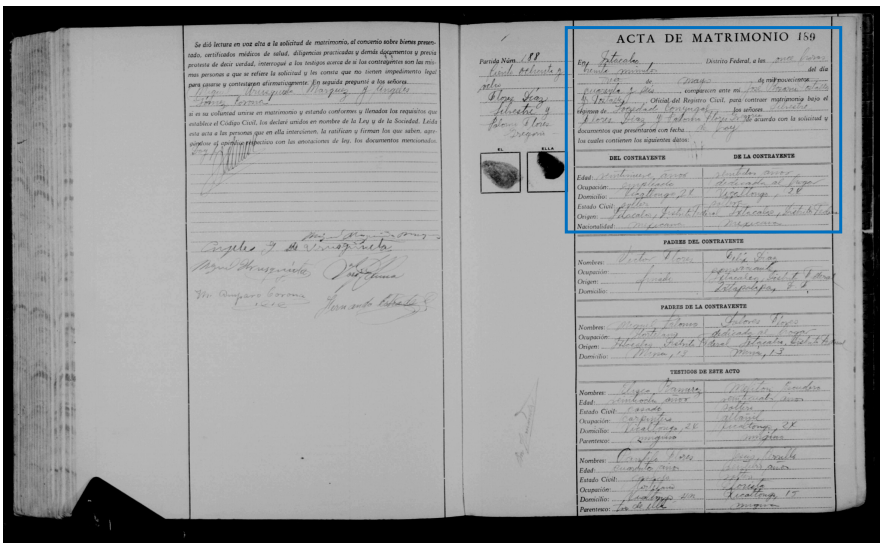
10.2 Création de la pseudo vérité terrain

Les documents utilisés dans la base d'apprentissage ne possèdent pas de vérité terrain annotée manuellement. Pour permettre l'inférence automatique et interactive de la description grammaticale, nous devons construire une pseudo vérité terrain comme nous l'avons décrit dans la section 4.1.4. Pour cela, nous procédons en deux étapes que nous allons détailler ci-après :

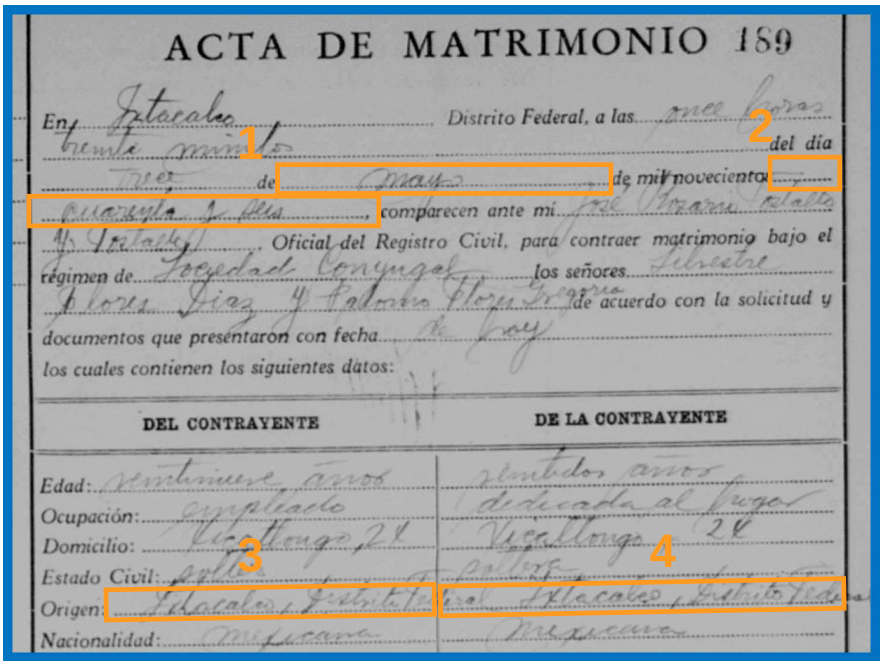
1. Extraction des primitives ;
2. Fiabilisation des primitives.

10.2.1 Extraction des primitives

Les primitives utilisées sont des mots-clés du paragraphe contenant les régions d'intérêt *mois* et *année*. Nous utilisons huit mots-clés différents (représentés en orange dans la figure 10.2) : Distrito (district), Federal (fédéral), día (jour), de (de), de mil novecientos (de mille neuf cents), comparecen (comparaissent), Oficial (officier) et Registro



(a) Exemple d'acte de mariage mexicain à analyser. La zone entourée en bleu est la zone contenant les 4 régions d'intérêt



(b) Représentation sur un exemple des quatres zones d'intérêt à détecter

FIG. 10.1 – Exemple d'acte de mariage mexicain du corpus HIP2013 FamilySearch

(bureau de l'état civil). Nous cherchons à obtenir pour chaque document une unique occurrence de chaque mot-clé. Les mots-clés ont été sélectionnés car ils sont présents dans le paragraphe contenant les champs *mois* et *année* que nous cherchons à localiser (dont la boite englobante est représentée en rouge dans la figure 10.2). Ces mots-clés nous permettront de délimiter la position des deux champs recherchés.

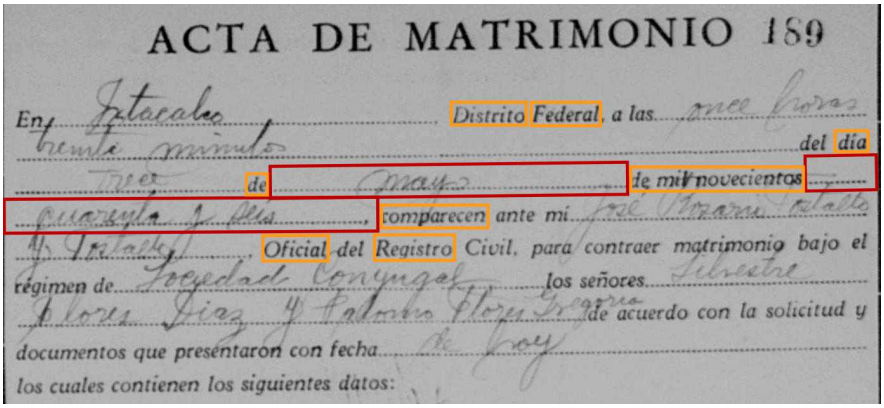


FIG. 10.2 – Représentation des mots-clés utilisés comme primitives d'analyse pour la construction de la pseudo vérité terrain (en orange) et des champs recherchés (en rouge)

Les mots-clés sont détectés selon une méthode basée sur la disposition de descripteurs locaux appelés Points d'Intérêt (POI) [Cam12]. Nous représentons ici succinctement les grands principes de cette approche. Nous n'avons pas utilisé d'OCR pour la recherche des mots-clés car la dégradation des documents ainsi que les interactions manuscrits/imprimés ne permettaient pas une reconnaissance suffisante des mots-clés.

Les POI sont des points de l'image qui présentent des variations locales de luminosité. Cette sélection doit être stable : les mêmes points sont sélectionnés dans toutes les images pour représenter le même objet. De plus, ces points doivent être discriminants : dans une image, il doit y avoir peu de confusion entre les descripteurs locaux. Pour chaque point d'intérêt sélectionné, nous calculons un descripteur local. Nous utilisons le descripteur introduit par Lowe [Low04]. Ce descripteur calcul des statistiques sur la direction du gradient dans un petit voisinage du point.

10.2.2 Fiabilisation des primitives

Les mots-clés obtenus lors de l'extraction des primitives sont bruités. Pour un document, nous n'obtenons pas une unique occurrence de chaque mot-clé comme nous le désirons (tableau 10.1). Par exemple pour les mots-clés courts « día » et « de », nous trouvons en moyenne respectivement 6,5 et 9,1 occurrences par document. Afin d'obtenir une pseudo vérité terrain pour l'inférence des règles grammaticales, il est indispensable de fiabiliser les primitives, c'est-à-dire de sélectionner les mots-clés détectés qui correspondent à la présence réelle du texte cherché.

La fiabilisation des primitives est effectuée (cf. section 4.1.4). Les clusters sont construits à partir des positions des mots-clés dans la zone de référence qu'est le para-

Mot-clé	Nombre d'occurrences par document
comparecen	4,0
de	9,1
de mil novecientos	1,1
día	6,5
Distrito	2,7
Federal	2,0
Oficial	3,7
Registro	1,0

TAB. 10.1 – Nombre moyen d'occurrences des mots-clés par document détectés par la méthode des points d'intérêt (POI) sur la base d'apprentissage de 7 000 documents

graphe accompagné du titre « Acta de matrimonio ». La fiabilisation se fait indépendamment pour chacun des huit types de mots-clés. Nous obtenons alors l'ensemble de données étiquetées utilisable pour l'inférence des règles.

10.3 Construction de la description grammaticale

Une fois la pseudo vérité terrain constituée, nous pouvons procéder à l'extraction de connaissance afin de construire la description grammaticale permettant de décrire les registres de mariages mexicains. Nous cherchons ici à apprendre les différents types de pré-imprimés existant dans le corpus pour les registres de mariage. Lorsqu'un registre est analysé, nous pouvons alors chercher dans cet ensemble de modèles pré-imprimés connus celui qui a été utilisé pour constituer le document.

10.3.1 Inférence des modèles de mots-clés

Un clustering des positions des mots-clés dans la page selon l'axe des abscisses et l'axe des ordonnées sur la base d'apprentissage de 7 000 documents est effectué sur les données de la pseudo vérité terrain. Le clustering nous permet de détecter les modèles de position par mot-clé.

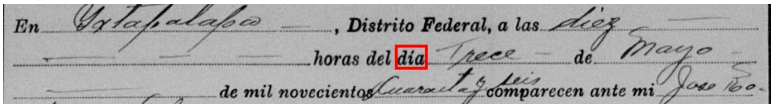
Les clusters ainsi formés sont visualisés par l'utilisateur qui apporte du sens dans l'analyse en proposant des libellés de cluster porteurs d'une sémantique. Il valide ainsi la pertinence du clustering produit automatiquement. Le nombre de modèles de position pour chaque mot-clé est présenté dans le tableau 10.2. La figure 10.3 présente trois exemples de modèles de position différents détectés pour le mot-clé « día ».

10.3.2 Inférence des modèles de documents

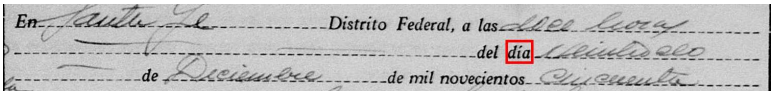
Nous construisons pour chaque document de l'ensemble d'apprentissage ainsi constitué une signature du document. Nous utilisons pour cela les modèles de position par mot-clé. L'occurrence du mot-clé contenu dans le document est affecté au modèle de

Mot-clé	Nombre de modèles de position
comparecen	9
de	5
de mil novecientos	6
día	7
Distrito	5
Federal	4
Oficial	7
Registro	4

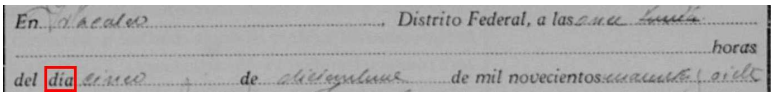
TAB. 10.2 – Nombre de modèles de position détectés pour chaque mot-clé durant la phase d'extraction de connaissance sur la base d'apprentissage de 7 000 documents



(a) Modèle 1



(b) Modèle 2



(c) Modèle 3

FIG. 10.3 – Exemples de trois modèles de position différents pour le mot-clé « día » détectés dans le corpus d'apprentissage HIP2013 FamilySearch

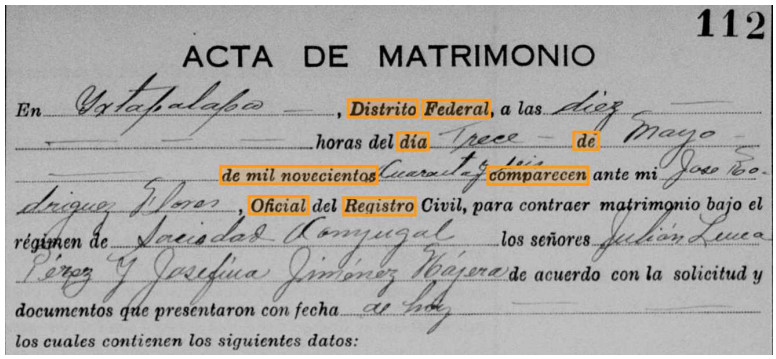
position correspondant et la signature est la concaténation de chacun des modèles de position correspondants. Un exemple est présenté dans le tableau 10.3.

Fichier	Modèle de position du mot-clé dia	de	de mil nov	comparecen	signature
00001	3	4	7	6	3#4#7#6

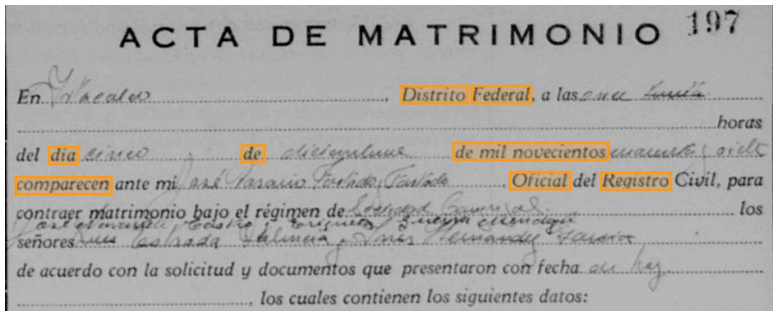
TAB. 10.3 – Exemple de création de la signature pour un registre de mariage

Pour construire les modèles de documents, nous utilisons les registres pour lesquels une unique occurrence de chacun des huit mots-clés est présente. Ce sont les documents pour lesquels il n'existe pas d'ambiguïté sur les modèles de position par mot-clé à utiliser. L'apprentissage des modèles de documents a pu être effectuée sur 5406 documents sur les 7000 documents de l'ensemble d'apprentissage.

Une analyse des fréquences des signatures des documents est ensuite effectuée. 11 modèles de pré-imprimés différents sont alors détectés dans l'ensemble d'apprentissage. La figure 10.4 présente deux exemples de modèles de pré-imprimés différents détectés dans le corpus d'apprentissage.



(a) Modèle 1



(b) Modèle 2

FIG. 10.4 – Exemples de deux modèles de pré-imprimés différents détectés dans le corpus d'apprentissage HIP2013 FamilySearch

La répartition des 11 modèles de documents dans le corpus d'apprentissage est présentée dans le tableau 10.4. Leur répartition inégale montre l'intérêt d'une analyse automatique et exhaustive de l'ensemble d'apprentissage. En effet, il aurait été très difficile avec une analyse manuelle de détecter l'ensemble de ces modèles. L'analyse aurait alors été effectuée sur un petit ensemble d'apprentissage qui n'aurait pas pu être représentatif.

Model	1	2	3	4	5	6	7	8	9	10	11
Count	1448	822	740	652	566	470	359	123	92	33	25

TAB. 10.4 – Les modèles de documents sont inégalement répartis dans le corpus d'apprentissage

10.3.3 Intégration dans la description grammaticale

Les modèles de pré-imprimés détectés et validés en interaction avec l'utilisateur nous permettent de générer automatiquement la description grammaticale des registres de mariage. Pour cela, nous avons besoin de générer les opérateurs de position de chacun des mots-clés par modèle de pré-imprimé. Nous inférons donc automatiquement : 6 paramètres × 8 opérateurs de position × 11 modèles de pré-imprimés = 528 paramètres.

Afin de déterminer pour un document à analyser quel modèle de pré-imprimé a été analysé, nous utilisons l'opérateur FIND_BEST_FIRST introduit par Maroneze [MCL11]. Cet opérateur permet de construire une grammaire stochastique localement. Lorsque nous analysons un document, chaque modèle de pré-imprimé est testé. Une pénalité est alors calculé représentant la non-adéquation du modèle de pré-imprimé au document analysé. L'opérateur FIND_BEST_FIRST nous permet alors de sélectionner le modèle de pré-imprimé avec la pénalité la plus faible, c'est-à-dire celui qui correspond le mieux au document.

Calcul de la pénalité Lorsque nous testons l'adéquation d'un modèle, chacun des mots-clés est recherché à sa position supposée, apprise sur l'ensemble d'apprentissage. Si le mot-clé n'est pas trouvé à cette position, la pénalité du modèle est augmentée de 1. Si le mot-clé est trouvé alors la pénalité du modèle est augmenté de :

$$1 - \frac{\text{aire de l'intersection}}{\text{aire du mot-clé}}$$

La figure 10.5 montre un exemple de calcul de pénalité pour un mot-clé. Le mot-clé « de mil novecientos » est recherché dans la zone rouge. Une occurrence est trouvée qui n'est pas totalement incluse dans la zone de recherche. La pénalité calculée pour ce mot-clé est 0,477444.

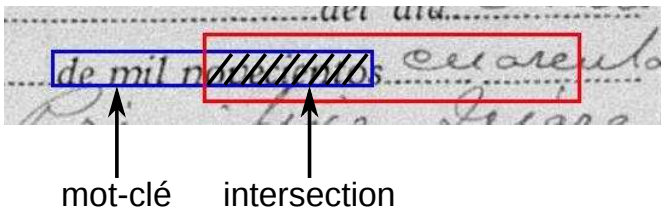


FIG. 10.5 – La pénalité du mot clé « de mil novecientos » est 0,477444 car il n'est pas totalement inclus dans la zone de recherche

Cette approche nous permet de détecter les bons mots-clés parmi les mots-clés contenant du bruit détectés avec les modèles de POI. De plus, cela nous permet également de synthétiser le mot-clé s'il n'y a pas de primitives correspondant dans le document.

10.4 Évaluation

L'évaluation est faite sur 2 000 documents du jeu de données de test du concours qui ont été annotés manuellement. Dans chaque document, la position des champs « mois »

et « année » est annotée. Nous ne tenons pas compte pour cela de la présence ou non de texte dans les champs.

10.4.1 Métrique

Pour évaluer la sous-tâche de détection de la position des champs « mois » et « année », nous devons évaluer la correspondance spatiale entre les champs détectés avec notre système de reconnaissance et ceux annotés dans la vérité terrain. Pour cela, nous utilisons la métrique introduite par Garris [Gar95] qui nous permet d'évaluer le recouvrement entre la zone attendue et la zone obtenue. Nous calculons l'intersection entre la zone attendue et la zone effectivement obtenue. La largeur de cette intersection doit être proche de la largeur de la zone attendue. La hauteur de l'intersection doit être également suffisamment grande pour pouvoir contenir le texte du champ s'il est présent.

Nous définissons deux seuils pour l'évaluation des résultats :

- un champ est considéré comme *complètement reconnu* si au moins 95% de la largeur et 75% de la hauteur a été reconnu
- un champ est considéré comme *partiellement reconnu* si 1) il n'est pas totalement reconnu, 2) au moins 80% de sa largeur est reconnu ainsi que 75% de sa hauteur.

Dans les autres cas, le champ est considéré comme manquant.

Le document est considéré comme reconnu lorsque tous les champs du document sont complètement ou partiellement reconnus et qu'il n'y a pas de zone détectée en trop dans le document.

10.4.2 Résultats

Lors de l'évaluation, nous comparons les résultats obtenus par notre méthode à la vérité terrain. Les résultats présentés dans le tableau 10.5 montrent que la description grammaticale construite à partir des modèles de pré-imprimés inférés permet de localiser efficacement les champs « mois » et « année ». Seulement 2,4% des champs ne sont pas reconnus et 89,8% des documents sont correctement reconnus. Cela montre que la description grammaticale inférée sans vérité terrain est efficace.

		Modèles	
		Inférés	Manuels
Zone	Reconnaissance complète	91.4%	89.7%
	Reconnaissance partielle	6.2%	4.0%
	Manquant	**2.4%**	6.3%
Taux de reconnaissance du document		**89.8%**	78.9%

Tab. 10.5 – Comparaison des résultats obtenus sur 2 000 documents avec des modèles inférés automatiquement et des modèles définis manuellement

Nous comparons également les résultats obtenus avec ceux obtenus par une mé-

thode où les modèles de documents sont définis manuellement. Cette méthode est celle soumise par Lemaitre [LC13] lors de la compétition HIP2013 FamilySearch. Dans cette description, quatre modèles ont été manuellement décrits. Cette méthode a été classée deuxième lors de la compétition HIP2013 FamilySearch.

Lorsque nous comparons les résultats obtenus par les modèles inférés aux modèles définis manuellement, nous pouvons remarquer que notre méthode obtient de meilleurs résultats. En effet, il y a moins de zones manquantes avec notre méthode (131 zones manquantes contre 343 zones avec les modèles définis manuellement). De plus, le taux de documents bien reconnus est fortement amélioré en utilisant les modèles de pré-imprimés inférés, avec une augmentation de 11% de documents bien reconnus (soit 217 documents sur 2 000).

10.5 Évaluation du coût de construction de la pseudo vérité terrain

Nous cherchons ici à comparer le coût de construction de la pseudo vérité terrain par le processus de fiabilisation des primitives au coût d'annotation manuelle d'une vérité terrain. Pour cela nous comparons le nombre d'actions à effectuer dans chacun des cas. Une action élémentaire correspond à :

- l'annotation d'un mot-clé dans le cadre de l'annotation manuelle des documents ;
- la prise de décision de conserver ou supprimer toutes les occurrences d'un cluster dans le cadre de la fiabilisation des primitives.

La base d'apprentissage est constitué de 7 000 documents. Dans chacun de ces documents, nous voulons obtenir 8 mots-clés. Il y a donc 56 000 zones à annoter manuellement si l'on voulait créer une vérité terrain sur cet ensemble. Dans le cadre de la fiabilisation des primitives, nous avons effectué 276 actions élémentaires qui ont abouti à la création d'un ensemble d'apprentissage contenant 54 141 zones. Il a fallu *203 fois moins* d'actions élémentaires qui si nous avions dû produire la vérité terrain manuellement. Si nous avions annoté manuellement 276 zones, alors nous aurions obtenu un corpus d'apprentissage constitué de 34,5 documents. Le coût de construction de la pseudo vérité terrain est donc très largement inférieur a celui de la construction d'une vérité terrain annotée manuellement.

Le nombre de modèles détectés dans le corpus d'apprentissage de 7000 documents est important, 11 modèles différents sont détectés. De plus, ces modèles ne sont pas équitablement représentés dans le corpus d'apprentissage (tableau 10.4). Un ensemble d'apprentissage restreint n'aurait alors pas pu aboutir à la détection de l'ensemble de ces modèles.

10.6 Conclusion

Cette application nous permet de mettre en avant l'intérêt de la méthode EWO pour les corpus sans vérité terrain. Grâce à la méthode EWO, nous avons pu inférer efficacement des règles à partir d'un corpus sans vérité terrain. Les résultats obtenus

sont meilleurs que ceux obtenus par des modèles définis manuellement dans le système soumis lors du concours HIP2013 FamilySearch.

Les bons résultats obtenus sur la reconnaissance des documents démontrent que la pseudo vérité terrain constituée semi-automatiquement en interaction avec l'utilisateur est de bonne qualité. De plus, la constitution de cette pseudo vérité terrain a un coût bien plus faible que l'annotation manuelle d'un corpus de documents. Ainsi, nous avons montré qu'alors que nous obtenons une pseudo vérité terrain sur 7000 documents, le même nombre d'actions auraient seulement permis l'annotation d'une vérité terrain de 34,5 documents. Ce faible corpus ne permettrait pas une bonne représentativité des documents au vue de la grande variabilité des pré-imprimés dans le corpus et de leur distribution non uniforme.

Conclusion de la troisième partie

Dans cette troisième partie, nous avons présenté des applications variées de la méthode EWO pour l'inférence semi-automatique et interactive de la structure des documents. Ces applications nous ont permis de valider séparément les différentes étapes mises en œuvre dans la méthode EWO ainsi que son fonctionnement global pour la description de systèmes de reconnaissance de la structure de documents complets.

Les travaux sur les courriers manuscrits ont permis de valider l'efficacité de la méthode EWO pour l'inférence de la description grammaticale dans le cadre d'un corpus homogène avec une vérité terrain. Une description grammaticale complète a été produite et a permis de montrer l'intérêt de l'enrichissement de la vérité terrain pour la segmentation des éléments. La comparaison à des systèmes purement syntaxiques et statistiques a montré que le système obtenu avec la méthode EWO a des performances équivalentes aux meilleures systèmes de chaque approche.

L'analyse des documents du corpus Maurdor a permis de valider l'intérêt d'une analyse exhaustive des corpus pour l'inférence des règles. En effet, le corpus Maurdor est un exemple de corpus fortement hétérogène disposant d'une vérité terrain annotée. Il est donc difficile dans ce cadre d'obtenir des règles de bonne qualité avec une description manuelle. Le nombre de documents analysés manuellement pour produire les règles est limité et ne permet pas de détecter la variabilité existant dans l'ensemble du corpus au contraire des règles décrites grâce à la méthode EWO.

Les expérimentations menées sur le corpus d'actes de mariages mexicains de la compétition FamilySearch HIP2013 ont permis de valider l'acquisition des données dans le cadre d'un corpus sans vérité terrain. Les bons résultats obtenus sur les documents montrent que la pseudo vérité terrain constituée semi-automatiquement est de bonne qualité. De plus, elle présente un coût de construction très faible en comparaison de celui de construction d'une vérité terrain annotée manuellement.

Les résultats obtenus ont donné lieu à plusieurs publications. L'inférence des opérateurs de position a été décrite dans [2]. L'inférence de règles sans vérité terrain sur le corpus des actes de mariages mexicains a été présenté dans [3].

Chapitre 11

Conclusion générale

Dans cette thèse, nous avons présenté EWO, une nouvelle méthodologie de construction des systèmes syntaxiques pour la reconnaissance de la structure de documents. Cette méthodologie se base sur l'introduction d'une phase d'apprentissage semi-automatique et interactive dans le processus de création de la description grammaticale.

11.1 Rappel des objectifs

Notre objectif a été de faciliter l'adaptation à un nouveau type de documents des systèmes syntaxiques de reconnaissance de la structure de documents. En effet, les documents à traiter dans le domaine de l'analyse de la structure de documents sont de plus en plus complexes et les corpus de plus en plus hétérogènes. Dans ce contexte, il est difficile pour l'utilisateur de définir correctement les règles de construction du document à partir d'un échantillon restreint analysé manuellement comme c'était fait jusqu'à présent dans le contexte des méthodes syntaxiques.

Pour faciliter l'adaptation à un nouveau type de documents, nous proposons d'introduire une étape d'apprentissage dans la construction de la description grammaticale. Cette approche nous permet de générer des règles apprises dans un contexte complexe pour lequel les méthodes purement statistiques ne sont pas adaptées. Nous disposons alors du grand pouvoir d'expression des méthodes syntaxiques tout en ayant une adaptabilité à un nouveau corpus facilitée, comme c'est le cas dans le cadre des méthodes statistiques.

11.2 Points forts de notre approche

Nous avons proposé une méthode reposant sur une décomposition de la reconnaissance du document complet en sous-problèmes et une résolution de chaque sous-problème selon deux approches :

– manuellement par l'utilisateur, s'il a de fortes connaissances *a priori* sur les documents ;

– semi-automatiquement à l'aide de la méthode EWO, si l'utilisateur a peu de connaissances *a priori* sur le sous-problème ou si la variabilité est importante.

Nous avons défini trois objectifs majeurs pour notre méthode :

1. Généricité : notre méthode est utilisable sur n'importe quel type de documents ;
2. Intégration de connaissance *a priori* : nous pouvons intégrer les nombreuses connaissances de l'utilisateur sur les documents à utiliser ;
3. Possible absence de vérité terrain : une vérité terrain annotée manuellement est onéreuse à produire. Par conséquent, de nombreux corpus de documents ne possèdent pas de vérité terrain et lorsque celle-ci existe, elle porte forcément sur un nombre restreint de documents. L'échantillon n'est alors pas forcément représentatif. Notre méthode nous permet de traiter des corpus sans vérité terrain ce qui répond également à la contrainte de généricité. De plus, la méthode EWO permet d'avoir une vue exhaustive sur le corpus de documents.

Ces objectifs théoriques nous ont mené à la proposition de la méthode EWO reposant sur trois points forts que nous allons maintenant rappeler :

– inférence des règles, pour la structure logique et la structure physique ;
– gestion de l'absence de vérité terrain ;
– vision exhaustive des données.

11.2.1 Inférence de règles

La méthode EWO permet à l'utilisateur une inférence des règles à la fois du point de vue de la structure logique et du point de vue de la structure physique du document. Les règles sont construites progressivement. L'inférence conjointe des structures logique et physique des documents n'avait pas été proposée jusqu'alors dans les méthodes de la littérature.

En effet, la structure logique est complexe à déterminer automatiquement car elle exprime souvent des connaissances de l'utilisateur sur la constitution du document. Nous avons proposé une méthodologie basée sur une détection automatique des structures existant dans l'ensemble de données et une interaction avec l'utilisateur pour l'étiquetage des structures afin d'apporter du sens aux données. Cette méthode nous permet d'inférer les variantes logiques des règles.

Lorsque l'inférence logique est effectuée, la méthode EWO permet de compléter l'inférence de la règle en cours de construction en inférant la structure physique. L'apprentissage semi-automatique de la structure physique nous permet d'avoir une vision exhaustive des données ce qui réduit le temps nécessaire à la description des règles. Pour l'inférence de la structure physique nous avons proposé l'apprentissage semi-automatique des variations physiques des règles. Cela consiste en l'apprentissage des positionnements des éléments ainsi que leur agencement les uns par rapport aux autres. Ces positionnements sont utilisés pour définir l'orientation et l'ordre de l'analyse. Nous avons également proposé l'apprentissage des propriétés physiques des éléments. Ces propriétés sont utilisées à la fois lors de la segmentation et lors de la reconnaissance des éléments dans le document.

11.2.2 Gestion de l'absence de vérité terrain

Nous avons proposé une méthodologie d'apprentissage capable de fonctionner sans vérité terrain annotée sur les documents à analyser. Les méthodes statistiques qui proposent un apprentissage automatique des modèles pour la reconnaissance de la structure de documents ne sont en général pas capables de faire cet apprentissage sans vérité terrain.

Pour gérer l'absence de vérité terrain, nous nous appuyons sur les redondances d'éléments de la structure pouvant émerger dans un grand corpus de documents. Nous avons proposé une méthode d'acquisition des données reposant sur deux étapes : une extraction automatique de primitives dans un grand volume de documents et une fiabilisation de ces primitives dans un processus interactif avec l'utilisateur humain. Pour proposer un processus de fiabilisation efficace des données, les primitives sont regroupées automatiquement à l'aide d'un algorithme de clustering. Pour chacun des clusters formés, un petit nombre d'exemples représentatifs du cluster est présenté à l'utilisateur. C'est à partir de ces quelques exemples que l'utilisateur prend la décision de conserver ou de rejeter toutes les primitives contenues dans le cluster. Les primitives conservées à l'issue de ce processus de fiabilisation viennent alors constituer une pseudo vérité terrain qui est utilisée par la méthode EWO pour l'inférence de la description grammaticale des documents.

11.2.3 Vision exhaustive des données

Lorsqu'une approche manuelle est utilisée pour la constitution de la description grammaticale, seule un petit échantillon de documents est analysé. L'hétérogénéité de plus en plus importante des corpus et la complexité des documents à analyser ne permettent pas de garantir la bonne représentativité du corpus dans ce petit échantillon de documents.

Au contraire, notre approche permet d'analyser un grand nombre de documents et de donner à l'utilisateur une vue à la fois synthétique et exhaustive des documents à traiter en s'appuyant sur un clustering des données. De plus, les cas rares sont également détectés lors de l'inférence des règles. Cela permet à l'utilisateur de décider de modéliser ou non les règles qui leur correspondent pour la description grammaticale globale.

Cette approche nous permet alors d'inférer la structure logique des documents en détectant les variantes logiques des règles ainsi que la structure physique des documents en détectant les variantes physiques des positionnements.

11.3 Validation de la méthode EWO

Des expérimentations ont été conduites sur des corpus variés. Ces expérimentations ont permis de valider chaque élément de la méthode EWO, avec des évaluations par parties et globales. Les systèmes obtenus en utilisant la méthode EWO ont permis de valider le gain en temps et en performances de l'introduction d'une phase d'apprentissage semi-automatique et interactive pour l'inférence de règles dans la description de

systèmes syntaxiques.

L'inférence des règles a été réalisée pour des corpus de documents homogènes (courriers manuscrits en français du corpus RIMES et registres de mariages mexicains d'archive du corpus de la compétition FamilySearch HIP2013) ainsi que pour un corpus de documents complexes fortement hétérogène (corpus de la compétition MAURDOR). Pour ces trois corpus, le système syntaxique décrit grâce à la méthode EWO a obtenu des résultats comparables ou meilleurs que ceux des systèmes pré-existants.

La gestion de l'absence de vérité terrain a été validée dans le cadre de la reconnaissance de registres de mariages mexicains d'archive. La méthode EWO nous a permis de construire une pseudo vérité terrain sur 7 000 documents à un coût très faible en comparaison du coût de production d'une vérité terrain annotée manuellement. En effet, la réalisation de la pseudo vérité terrain a nécessité la réalisation de 200 fois moins d'actions pour l'utilisateur que s'il avait annoté une vérité terrain pour le même nombre de documents.

La nécessité d'une vision exhaustive sur les documents pour obtenir une description grammaticale efficace a également été validée. Pour cela, nous avons comparé les descriptions grammaticales obtenues grâce à la méthode EWO à des descriptions grammaticales pré-existantes décrites manuellement. Les expérimentations ont prouvé que les systèmes décrits avec la méthode EWO sont plus performants que ceux décrits manuellement et sont décrits plus rapidement. Par exemple, pour la reconnaissance des actes de mariages mexicains, nous avons amélioré le taux de reconnaissance des documents de 11%, passant de 78,9% à 89,8% de documents bien reconnus (soit 217 documents sur 2 000).

11.4 Perspectives

Nous présentons maintenant quelques axes pour les travaux futurs.

11.4.1 Extension des cadres applicatifs

La méthode EWO est une méthode générique d'inférence de règles pour la reconnaissance de la structure de documents. Nous avons présenté trois exemples d'applications différents dans la partie III. Il nous semble intéressant d'appliquer la méthode EWO sur d'autres corpus et notamment de combiner plusieurs types de primitives.

11.4.1.1 Les séparateurs d'articles dans la presse ancienne

Nous avons mené quelques expérimentations sur un corpus de documents de presse ancienne. Ces expérimentations se placent dans le contexte d'une grammaire déjà existante, pour laquelle nous souhaitons améliorer la détection des séparateurs horizontaux d'articles. La méthode EWO vient ici s'intégrer au cœur d'un système syntaxique existant décrit manuellement pour y injecter une phase d'apprentissage afin de résoudre un sous-problème complexe. En effet, la grande variabilité présente dans les documents pour les séparateurs horizontaux rend difficile leur description manuelle.

Cette expérimentation se place de plus dans le contexte de documents pour lesquels il n'y a pas de vérité terrain disponible. Afin de réaliser l'acquisition des données dans ce contexte, plusieurs types de primitives sont disponibles : lignes de texte issues de l'OCR ainsi que les mots reconnus, lignes de texte issues d'une grammaire existante, séparateurs actuellement détectés, etc. Les primitives sont ensuite combinées afin de déterminer un ensemble d'éléments horizontaux pouvant être des séparateurs d'articles. Le système syntaxique existant décrivant les pages de presse ancienne nous permet également de limiter le nombre de candidats à analyser puisque certains éléments logiques sont déjà reconnus dans les pages. Les éléments horizontaux candidats sont ensuite fiabilisés afin d'éliminer le bruit présent et de créer ainsi un ensemble de données annotées utilisable pour l'inférence des règles décrivant les séparateurs d'articles.

À l'heure actuelle, nous n'avons pas eu le temps de mener suffisamment d'expérimentations pour les présenter ici mais il nous semble que la combinaison de primitives pour la création de la pseudo vérité terrain est une piste importante pour l'acquisition de données dans des documents à la structure complexe et/ou dégradés.

11.4.1.2 L'analyse des PDF

Les entreprises traitent de plus en plus de documents numériques et notamment de documents au format PDF. Ces documents peuvent contenir plus d'informations qu'une image de document numérisé. Par exemple, ils peuvent contenir le texte, des informations sur la typographie, les positionnements des différents éléments, les images, etc. Cependant leur exploitation complète nécessite la reconnaissance de la structure des documents qui n'est pas contenue dans le PDF.

De manière générale, nous allons retrouver toutes les problématiques des images de documents numérisés dans le contexte des documents au format PDF. La méthode EWO peut donc être appliquée dans ce contexte afin de permettre l'extraction de connaissances et l'inférence semi-automatique de descriptions grammaticales pour la reconnaissance de la structure des documents.

11.4.2 Passage à l'échelle

Un prolongement possible de nos travaux consiste à consolider ces derniers en permettant un passage à l'échelle afin de traiter des corpus de documents plus volumineux, notamment dans un contexte industriel. L'augmentation du volume de documents traités permettra également d'apprendre des structures de documents plus complexes.

Dans ce contexte, il nous semble essentiel de conserver la méthodologie globale de la méthode EWO telle que nous l'avons présentée. Nous conservons donc la décomposition en sous-problèmes que nous avons proposée et les trois étapes de résolution des sous-problèmes : acquisition des données, inférence des règles et assistance à l'intégration dans la description grammaticale. En effet, cette approche nous permet de limiter le volume de données à traiter pour l'inférence des règles de chaque sous-problème ce qui nous permet notamment de réaliser une acquisition des données performante lorsqu'il n'y a pas de vérité terrain annotée disponible.

Pour permettre le passage à l'échelle, il serait intéressant de modifier l'implémentation du clustering EAC pour permettre son application à de plus grand volume de données. Le clustering EAC a pour l'instant été appliqué à des corpus contenant plusieurs milliers de documents (7 000 documents dans le cas du corpus FamilySearch HIP2013). Les méthodes utilisées pour l'apprentissage de la structure physique peuvent également être changées. Nous pourrions par exemple permettre l'apprentissage de classifieurs pour la définition des propriétés physiques d'un élément.

11.4.3 Extension du champ d'application

11.4.3.1 Application de la méthode EWO à d'autres méthodes de reconnaissance de la structure de documents

La méthode EWO est une méthode générique pour l'acquisition de connaissances dans le contexte de la reconnaissance de la structure de documents. Il serait intéressant d'appliquer la méthode EWO à d'autres méthodes syntaxiques que celle utilisée pour les expérimentations mais également dans le contexte de systèmes dédiés, non spécifiquement à base de règles. Nous pourrions également envisager son application dans le cadre de méthodes purement statistiques où la décomposition en sous-problèmes permettrait d'intégrer une composante structurelle.

11.4.3.2 Construction rapide de bases d'apprentissage étiquetées

De nombreuses méthodes nécessitent un échantillon étiqueté pour l'étape d'apprentissage. Nous avons proposé une méthode de construction semi-automatique et interactive des données étiquetées utiles pour l'apprentissage. Le mécanisme en deux étapes proposé (génération de candidats puis fiabilisation des candidats) est générique. Il nous semble donc intéressant de tester cette méthode d'acquisition des données dans d'autres contextes différents de l'inférence de règles en reconnaissance de structure de documents

L'avantage de notre méthode est qu'elle permet de construire rapidement et semi-automatiquement une vérité terrain complexe. En effet, la décomposition en sous-problème que nous avons proposée permet de construire progressivement l'ensemble de données étiquetées. Cela permet d'affiner l'ensemble de données étiquetées au fur et à mesure de l'acquisition de connaissances et de constituer ainsi une base d'apprentissage étiquetée complexe.

Bibliographie

[ABKS99] Mihael Ankerst, Markus M. Breunig, Hans-Peter Kriegel, and Jörg Sander. Optics : Ordering points to identify the clustering structure. *SIGMOD Rec.*, 28(2) :49–60, June 1999.

[Bez81] James C. Bezdek. *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA, 1981.

[BGG+14] Sylvie Brunessaux, Patrick Giroux, Bruno Grilheres, Mathieu Manta, Maylis Bodin, Khalid Choukri, Olivier Galibert, and Juliette Kahn. The maurdor project : Improving automatic processing of digital documents. In *11th IAPR International Workshop on Document Analysis Systems, DAS 2014, Tours, France, April 7-10, 2014*, pages 349–354, 2014.

[BH67] Geoffrey H. Ball and David J. Hall. A clustering technique for summarizing multivariate data. *Behavioral Science*, 12(2) :153–155, 1967.

[BR08] Abdel Belaïd and Yves Rangoni. Structure Extraction in Printed Documents Using Neural Approaches. In Simone Marinai and Hiromichi Fujisawa, editors, *Machine Learning in Document Analysis and Recognition*, volume 90 of *Studies in Computational Intelligence*, pages 21–43. Springer, 2008.

[Bra69] Walter S. Brainerd. Tree generating regular systems. *Information and Control*, 14(2) :217 – 231, 1969.

[Cam12] Jean Camillerapp. Utilisation des points d'intérêt pour rechercher des mots imprimés ou manuscrits dans des documents anciens. In *Conférence Internationale sur l'Écrit et le Document (CIFED'12)*, pages 163–178, 2012.

[CF02] Adrian E. Raftery Chris Fraley. Model-based clustering, discriminant analysis, and density estimation. *Journal of the American Statistical Association*, 97(458) :611–631, 2002.

[CFGS03] F. Cesarini, E. Francesconi, M. Gori, and G. Soda. Analysis and understanding of multi-class invoices. *Document Analysis and Recognition*, 6(2) :102–114, 2003.

[Cha13] Joseph Chazalon. *Contextual and assisted interpretation of digitized fonds : application to sales registers from the 18th century.* Theses, INSA de Rennes, January 2013.

[CJDR09] Santanu Chaudhury, Megha Jindal, and Sumantra Dutta Roy. Model-guided segmentation and layout labelling of document images using a hierarchical conditional random field. In Santanu Chaudhury, Sushmita Mitra, C.A. Murthy, P.S. Sastry, and SankarK. Pal, editors, *Pattern Recognition and Machine Intelligence*, volume 5909 of *Lecture Notes in Computer Science*, pages 375–380. 2009.

[Con93] A. Conway. Page grammars and page parsing. a syntactic approach to document layout recognition. In *Document Analysis and Recognition, 1993., Proceedings of the Second International Conference on*, pages 761–764, Oct 1993.

[Cou06] Bertrand Couasnon. Dmos, a generic document recognition method : application to table structure analysis in a general and in a specific way. *International Journal of Document Analysis and Recognition (IJDAR)*, 8(2-3) :111–122, 2006.

[DB79] David L. Davies and Donald W. Bouldin. A cluster separation measure. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-1(2) :224–227, April 1979.

[dlH05] Colin de la Higuera. A bibliographical study of grammatical inference. *Pattern Recogn.*, 38(9) :1332–1348, September 2005.

[DP91] D. Derrien Peden. Frame-based system for macro-typographical structure analysis in scientific paper. In *Proc. 1st Int. Conf. on Document Analysis and Recognition*, pages 311–319, 1991.

[DS14] Andreas Dengel and Faisal Shafait. Analysis of the logical layout of documents. In David Doermann and Karl Tombre, editors, *Handbook of Document Image Processing and Recognition*, pages 177–222. Springer London, 2014.

[Dun73] J.C. Dunn. A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters. *J. Cybern.*, 3 :32–57, 1973.

[Fed71] Jerome Feder. Plex languages. *Information Sciences*, 3(3) :225 – 241, 1971.

[Fis91] J.L. Fisher. Logical structure descriptions of segmented document images. In *Proc. 1st Int. Conf. on Document Analysis and Recognition*, pages 302–310, 1991.

[FJ02] Ana Fred and Anil K. Jain. Evidence accumulation clustering based on the k-means algorithm. In *Structural, Syntactic, and Statistical Pattern Recognition, LNCS 2396 :442 ?451*, pages 442–451. Springer-Verlag, 2002.

[FRMS12] Chris Fraley, Adrian E. Raftery, Thomas Brendan Murphy, and Luca Scrucca. *mclust Version 4 for R : Normal Mixture Modeling for Model-Based Clustering, Classification, and Density Estimation*, 2012.

[FSJ+01] Le Bourgeois F., Souafi-Bensafi S., Duong J., Parizeau M., Coté M., and Emptoz H. Using statistical models in document images understanding. In *Workshop on Document Layout Interpretation and its Applications, DLIA*, 2001.

[Gar95] M.D. Garris. Evaluating spatial correspondence of zones in document recognition systems. In *Image Processing, 1995. Proceedings., International Conference on*, volume 3, pages 304–307 vol.3, Oct 1995.

[GB95] A. Grbavec and D. Blostein. Mathematics recognition using graph rewriting. In *Document Analysis and Recognition, 1995., Proceedings of the Third International Conference on*, volume 1, pages 417–421 vol.1, Aug 1995.

[GCBG09] E. Grosicki, M. Carree, J.-M. Brodin, and E. Geoffrois. Results of the rimes evaluation campaign for handwritten mail processing. In *Document Analysis and Recognition, 2009. ICDAR '09. 10th International Conference on*, pages 941–945, July 2009.

[GCG+06] Emmanuèle Grosicki, Matthieu Carré, Edouard Geoffrois, Emmanuel Augustin, and Francoise Preteux. La campagne d'évaluation RIMES pour la reconnaissance de courriers manuscrits. In *Actes Colloque International Francophone sur l'Ecrit et le Document (CIFED'06)*, pages 61–66, Fribourg, Switzerland, September 2006.

[GRS00] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Rock : A robust clustering algorithm for categorical attributes. *Information Systems*, 25(5) :345 – 366, 2000.

[GRS01] Sudipto Guha, Rajeev Rastogi, and Kyuseok Shim. Cure : an efficient clustering algorithm for large databases. *Information Systems*, 26(1) :35 – 58, 2001.

[HK99] Alexander Hinneburg and Daniel A. Keim. Clustering methods for large databases : From the past to the future. *SIGMOD Rec.*, 28(2) :509–, June 1999.

[IA91] R. Ingold and D. Armangil. A top-down document analysis method for logical structure recognition. In *Proc. 1st Int. Conf. on Document Analysis and Recognition*, pages 302–310, 1991.

[Jai10] Anil K. Jain. Data clustering : 50 years beyond k-means. *Pattern Recogn. Lett.*, 31(8) :651–666, June 2010.

[JD88] Anil K. Jain and Richard C. Dubes. *Algorithms for Clustering Data.* Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[KHK99] George Karypis, Eui-Hong (Sam) Han, and Vipin Kumar. Chameleon : Hierarchical clustering using dynamic modeling. *Computer*, 32(8) :68–75, August 1999.

[KNSV93] M. Krishnamoorthy, G. Nagy, S. Seth, and M. Viswanathan. Syntactic segmentation and labeling of digitized pages from technical journals. *IEEE Trans. Pattern Anal. Mach. Intell.*, 15(7) :737–747, July 1993.

[KR87] Leonard Kaufman and Peter Rousseeuw. Clustering by means of medoids. *Statistical Data Analysis Based on the L1-Norm and Related Methods*, pages North–Holland, 1987.

[KR90] Leonard Kaufman and Peter J. Rousseeuw. *Introduction*, pages 1–67. John Wiley & Sons, Inc., 1990.

[LC13] Aurélie Lemaitre and Jean Camillerapp. HIP 2013 FamilySearch Competition - Contribution of IRISA. In *HIP - ICDAR Historical Image Processing Workshop*, Washington, United States, August 2013.

[LCC07] Aurélie Lemaitre, Jean Camillerapp, and Bertrand Coüasnon. Contribution of Multiresolution Description for Archive Document Structure Recognition. In *ICDAR 2007*, volume 1 of *Ninth International Conference on Document Analysis and Recognition, 2007*, Curitiba, Brazil, September 2007.

[LCC08] Aurélie Lemaitre, Jean Camillerapp, and Bertrand Coüasnon. A generic method for structure recognition of handwritten mail documents. In *Document Recognition and Retrieval DRR XV*, San Jose, États-Unis, 2008.

[LCC09] Aurélie Lemaitre, Jean Camillerapp, and Bertrand Coüasnon. Multi-script Baseline Detection Using Perceptive Vision. In *14th Biennial Conference of the International Graphonomics Society (IGS 2009)*, pages –, Dijon, France, September 2009.

[LFJ10] A. Lourenço, A. L. N. Fred, and A. K. Jain. On the scalability of evidence accumulation clustering. In *International Conf. on Pattern Recognition - ICPR*, volume ., pages 782 – 785, August 2010.

[LGGP07] M. Lemaitre, E. Grosicki, E. Geoffrois, and F. Preteux. Preliminary experiments in layout analysis of handwritten letters based on textural and spatial information and a 2d markovian approach. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, volume 2, pages 1023–1027, Sept 2007.

[LJ12] T. Lerddararadsamee and Y. Jiraraksopakun. Local maximum detection for fully automatic classification of em algorithm. In *Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON), 2012 9th International Conference on*, pages 1–4, 2012.

[LL08] Aurélie Lemaitre Legargeant. *Use of perceptive vision for document structure recognition.* Theses, INSA de Rennes, December 2008.

[LNN97] ChunChen Lin, Y. Niwa, and S. Narita. Logical structure analysis of book document images using contents information. In *Document Analysis and Recognition, 1997., Proceedings of the Fourth International Conference on*, volume 2, pages 1048–1054 vol.2, Aug 1997.

[Low04] David G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision*, 60(2) :91–110, November 2004.

[MBD11] Eric Medvet, Alberto Bartoli, and Giorgio Davanzo. A probabilistic approach to printed document understanding. *International Journal on Document Analysis and Recognition (IJDAR)*, 14(4) :335–347, 2011.

[MC05] Isaac Martinat and Bertrand Coüasnon. A minimal and sufficient way of introducing external knowledge for table recognition in archival documents. In *Graphics Recognition. Ten Years Review and Future Perspectives, 6th Internation Workshop, GREC 2005, Hong Kong, China, August 25-26, 2005, Revised Selected Papers*, pages 206–217, 2005.

[MCL11] André O. Maroneze, Bertrand Coüasnon, and Aurélie Lemaitre. Introduction of statistical information in a syntactic analyzer for document image recognition. In *Document Recognition and Retrieval XVIII - DRR 2011, 18th Document Recognition and Retrieval Conference, part of the IS&T-SPIE Electronic Imaging Symposium, San Jose, CA, USA, January 24-29, 2011, Proceedings*, pages 1–10, 2011.

[MK01] Song Mao and Tapas Kanungo. Stochastic language models for automatic acquisition of lexicons from printed bilingual dictionaries. In *Workshop on Document Layout Interpretation and its Applications*, 2001.

[MNGH10] F. Montreuil, S. Nicolas, E. Grosicki, and L. Heutte. A new hierarchical handwritten document layout extraction based on conditional random field modeling. In *Frontiers in Handwriting Recognition (ICFHR), 2010 International Conference on*, pages 31–36, Nov 2010.

[NS95] Debashish Niyogi and Sargur N. Srihari. Knowledge-based derivation of document logical structure. In *in Proceedings of International Conference on Document Analysis and Recognition*, pages 472–475, 1995.

[PB95] N. R. Pal and J. C. Bezdek. On cluster validity for the fuzzy c-means model. *Trans. Fuz Sys.*, 3(3) :370–379, August 1995.

[PR69] John L. Pfaltz and Azriel Rosenfeld. Web grammars. In *Proceedings of the 1st International Joint Conference on Artificial Intelligence*, IJCAI'69, pages 609–619, San Francisco, CA, USA, 1969. Morgan Kaufmann Publishers Inc.

[Ram06] Jean-Yves Ramel. *Propositions pour la représentation et l'analyse de documents numériques*. PhD thesis, Université François Rabelais, Tours, November 2006.

[RC96] M. Armon Rahgozar and Robert Cooperman. Graph-based table recognition system. volume 2660, pages 192–203, 1996.

[Rou87] Peter Rousseeuw. Silhouettes : A graphical aid to the interpretation and validation of cluster analysis. *J. Comput. Appl. Math.*, 20(1) :53–65, November 1987.

[SEKX98] Jörg Sander, Martin Ester, Hans-Peter Kriegel, and Xiaowei Xu. Density-based clustering in spatial databases : The algorithm gdbscan and its applications. *Data Min. Knowl. Discov.*, 2(2) :169–194, June 1998.

[SGJ03] Catherine A. Sugar, Gareth, and M. James. Finding the number of clusters in a data set : An information theoretic approach. *Journal of the American Statistical Association*, 98 :750–763, 2003.

[SLV05] M. Shilman, P. Liang, and P. Viola. Learning nongenerative grammatical models for document analysis. In *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on*, volume 2, pages 962–969 Vol. 2, Oct 2005.

[SSS+07] Shravya Shetty, Harish Srinivasan, Sargur Srihari, Shravya Shetty, Harish Srinivasan, Matthew Beal, and Sargur Srihari. Segmentation and labeling of documents using conditional random fields. In *Document Recognition and Retrieval DRR XIV*, 2007.

[TA90] S. Tsujimoto and H. Asada. Understanding multi-articled documents. In *Pattern Recognition, 1990. Proceedings., 10th International Conference on*, volume i, pages 551–556 vol.1, Jun 1990.

[TGH01] Robert Tibshirani, W. Guenther, and T. Hastie. Estimating the number of clusters in a data set via the gap statistic. *Journal of the Royal Statistical Society Series B*, 2001.

[TI94] Y. Tateisi and N. Itoh. Using stochastic syntactic analysis for extracting a logical structure from a document image. In *Pattern Recognition, 1994. Vol. 2 - Conference B : Computer Vision amp ; Image Processing., Proceedings of the 12th IAPR International. Conference on*, volume 2, pages 391–394 vol.2, Oct 1994.

[vD00] Stijn van Dongen. *Graph Clustering by Flow Simulation.* PhD thesis, University of Utrecht, 2000.

[VR11] Sandro Vega-Pons and José Ruiz-Shulcloper. A survey of clustering ensemble algorithms. *IJPRAI*, 25(3) :337–372, 2011.

[XW05] Rui Xu and II Wunsch, D. Survey of clustering algorithms. *Neural Networks, IEEE Transactions on*, 16(3) :645–678, May 2005.

[YATT91] A. Yamashita, T. Amano, I. Takahashi, and K. Toyokawa. A model based layout understanding method for the document recognition system. In *Proc. 1st Int. Conf. on Document Analysis and Recognition*, pages 130–138, 1991.

[ZRL96] Tian Zhang, Raghu Ramakrishnan, and Miron Livny. Birch : An efficient data clustering method for very large databases. *SIGMOD Rec.*, 25(2) :103–114, June 1996.

Publications de l'auteur

[1] Carton, C., Lemaitre, A., and Couasnon, B. Fusion of statistical and structural information for flowchart recognition. In *ICDAR - International Conference on Document Analysis and Recognition* (Washington, United States, 2013), pp. 1242–1246.

[2] Carton, C., Lemaitre, A., and Coüasnon, B. Learnpos : a new tool for interactive learning positioning. In *Document Recognition and Retrieval DRR XXI* (2014).

[3] Carton, C., Lemaitre, A., and Coüasnon, B. Automatic and interactive rule inference without ground truth. In *International Conference on Document Analysis and Recognition (ICDAR)* (Nancy, France, Aug. 2015).

[4] Carton, C., Lemaitre, A., and Coüasnon, B. B. LearnPos : un nouvel outil pour l'apprentissage interactif de positionnement. In *Conférence Internationale Francophone sur l'Ecrit et le Document* (Nancy, France, 2014), pp. 325–340.

[5] Carton, C., Lemaitre, A., and Coüasnon, B. B. Inférence semi-automatique et interactive de règles sans vérité terrain. In *Conférence Internationale Francophone sur l'Ecrit et le Document* (2016).

Table des figures

Liste des tableaux

Résumé

Les documents à traiter dans le domaine de l'analyse de la structure de documents sont de plus en plus complexes et les corpus de plus en plus hétérogènes. Nous proposons une nouvelle méthode, la méthode Eyes Wide Open (EWO) pour introduire une phase d'apprentissage semi-automatique et interactive dans la construction de descriptions grammaticales. Grâce à la méthode EWO, il est possible de disposer du grand pouvoir d'expression des méthodes syntaxiques tout en ayant l'adaptabilité des méthodes statistiques.

La méthode EWO permet d'inférer des règles afin de construire de manière progressive la description grammaticale complète des documents. L'inférence des règles concerne à la fois la structure logique et la structure physique des documents. La méthode EWO repose sur deux éléments majeurs : l'émergence automatique de structures grâce à un algorithme de clustering et une interaction avec l'utilisateur pour donner un sens aux structures détectées automatiquement.

Notre méthode permet de plus l'inférence des règles sans vérité terrain annotée disponible sur les documents. Pour ce faire, la méthode EWO repose sur l'analyse de redondances dans de grand volume de documents non annotés. La détection des redondances est faite automatiquement grâce à un algorithme de clustering. Les éléments détectés automatiquement sont ensuite fiabilisés par l'utilisateur afin d'obtenir les données étiquetées d'apprentissage.

La méthode EWO apporte une vision exhaustive et synthétique des données à analyser. Cela permet une meilleure exploitation du corpus que pour les méthodes syntaxiques décrites manuellement. Cela permet de plus une meilleure gestion des cas rares que ce qui est possible pour les méthodes statistiques.

Nous avons validé l'efficacité cette approche sur des documents à structure variée (courriers manuscrits, registres d'archives, formulaires...). Pour chaque corpus de documents, des descriptions grammaticales ont été générées avec à la méthode EWO, obtenant des performances comparables ou meilleures que celles de systèmes pré-existants décrits manuellement. La méthode a également été appliquée avec succès sur un large corpus sans vérité terrain.

Abstract

The documents to analyze in the document structure analysis are getting more and more complex and the corpora are more and more heterogeneous. We propose a new method, the Eyes Wide Open method (EWO) to introduce a semi-automatic and interactive learning step in the building of grammatical descriptions. With the EWO method, it is possible to benefit from the expressiveness of the syntactical methods while having the adaptability of the statistical methods.

The EWO method allows the rules inference to build progressively the full grammatical description of the documents. The rules inference concerns both the logical and the physical structure of the documents. The EWO method relies on two major elements: the automatic discovering of structures with clustering algorithm and an interaction with the user to give sense to the automatically detected structures.

Our method allows the rules inference without annotated ground truth on the documents. To do so, the EWO method relies on the analysis of redundancies on big volume of non annotated documents. The redundancy detection is performed automatically with a clustering algorithm. A data reliability enhancement step is performed in interaction with the user on the automatically detected elements to obtain the training labeled data.

The EWO method allows an exhaustive and concise view of the data to analyze. It allows a better use of the corpus than for the manually described syntactical method. Furthermore, it allows a better management of the rare cases than what is possible with the statistical method.

We validated the efficiency of this method on documents with various structures (handwritten business letters, marriage records, forms...). For each corpus, a grammatical description was generated using the EWO method, obtaining at least similar results to the pre-existing manually described systems. The method was also successfully applied to a large non annotated corpus.

N° d'ordre :

Institut National des Sciences Appliquées de Rennes

20, Avenue des Buttes de Coësmes . CS 14315 . F-35043 Rennes Cedex

Tel : 02 23 23 82 00 – Fax : 02 23 23 83 96