







Université François Rabelais  
Tours  
Ecole Doctorale Santé, Sciences et Technologies  
Année Universitaire 2001-2002



---

**THESE POUR OBTENIR LE GRADE DE  
DOCTEUR DE L'UNIVERSITE DE TOURS**

---

Discipline : **Informatique**

Présentée et soutenue publiquement  
par :

**Nathalie FRIBURGER**

**Le 2 décembre 2002**

---

**RECONNAISSANCE AUTOMATIQUE DES NOMS PROPRES**  
*Application à la classification automatique de textes journalistiques*

---

Directeur de thèse : Denis MAUREL

---

**Jury**

<b>El Bèze</b> Marc	Professeur	Rapporteur	Université d'Avignon et des Pays de Vaucluse
<b>Giacometti</b> Arnaud	Maître de conférence	Examineur	Université de Tours
<b>Gueunthner</b> Franz	Professeur	Rapporteur	Université de Munich (Ludwig Maximilians Universität – München)
<b>Maurel</b> Denis	Professeur	Directeur	Université de Tours
<b>Noailly</b> Michèle	Professeur Emérite	Examineur	Université de Bretagne Occidentale
<b>Silberztein</b> Max	Professeur	Examineur	Université de Franche-Comté



# TABLE DES MATIERES

<b>GLOSSAIRE</b>	<b>I</b>
<b>INTRODUCTION</b>	<b>3</b>
<b>LES NOMS PROPRES</b>	<b>5</b>
<b>1 LES NOMS PROPRES EN FRANÇAIS</b>	<b>5</b>
1.1 Qu'est-ce qu'un nom propre ?	5
1.2 La productivité des noms propres	8
1.3 Les dictionnaires de noms propres	10
<b>2 TYPOLOGIES DE NOMS PROPRES</b>	<b>11</b>
2.1 Le typage par un lecteur humain	11
2.2 Typologies morpho-syntaxiques des noms propres	12
2.3 Typologies sémantiques des noms propres	13
2.4 Les ambiguïtés sémantiques des noms propres	15
<b>3 RECONNAISSANCE AUTOMATIQUE DES NOMS PROPRES</b>	<b>15</b>
3.1 Preuve interne et externe	16
3.2 Structure syntaxique des noms propres accompagnés d'une preuve externe	17
<b>4 LES NOMS PROPRES ET LEURS DIFFERENTES PREUVES EN CORPUS</b>	<b>18</b>
4.2 Variation des noms propres	20
<b>5 CONCLUSION</b>	<b>21</b>
<b>LES SYSTEMES D'EXTRACTION D'ENTITES NOMMEES</b>	<b>23</b>
<b>1 LES RECHERCHES DANS LE DOMAINE DE L'EXTRACTION AUTOMATIQUE D'ENTITES NOMMEES</b>	<b>23</b>
1.1 L'avant MUC	23
1.2 La conférence MUC	24
1.3 Les métriques d'évaluation	25
1.4 La tâche d'extraction des entités nommées NE	26
<b>2 LES DIFFERENTES METHODES D'EXTRACTION DES ENTITES NOMMEES</b>	<b>27</b>
2.1 Trois types de systèmes	27
2.2 Les systèmes à base de règles	30
2.3 Les systèmes à apprentissage	31
2.4 Les systèmes hybrides	32
2.5 Le meilleur système selon MUC 7 : LTG system	33
<b>3 L'USAGE DES CONNAISSANCES LINGUISTIQUES DANS LES SYSTEMES D'EXTRACTION DE NOMS PROPRES</b>	<b>34</b>
3.1 L'apport des ressources et leur usage	34
3.2 Utilisation des preuves internes et externes	36
3.3 L'utilisation de la morphologie	37
<b>4 LE TRAITEMENT DES AMBIGUÏTES</b>	<b>37</b>
4.1 Résolution des ambiguïtés structurelles : la délimitation des noms propres	37
4.2 Résolution des ambiguïtés sémantiques	39
4.3 Une heuristique de désambiguïsation : <i>Les mots ont un seul sens par discours</i>	39
<b>5 CONCLUSION</b>	<b>40</b>

---

## **CASSYS, UN SYSTEME DE CASCADE DE TRANSDUCTEURS** **41**

<b>1 LES CASCADES DE TRANSDUCTEURS</b>	<b>41</b>
1.1 Définition d'un transducteur	41
1.2 Définition d'une cascade de transducteurs	41
1.3 Systèmes de cascades existants	42
<b>2 LE SYSTEME CASSYS</b>	<b>44</b>
2.1 Présentation du système Intex	44
2.2 Généralités sur le système CasSys	47
2.3 Exemples de cascade de transducteurs	50
<b>3 CONCLUSION</b>	<b>56</b>

## **PRE-TRAITEMENTS POUR L'EXTRACTION DES NOMS PROPRES** **57**

<b>1 LA SEGMENTATION EN PHRASES</b>	<b>57</b>
1.1 L'ambiguïté du point	60
1.2 La ponctuation : choix de découpage	66
1.3 Les résultats	69
<b>2 ETIQUETAGE DES TEXTES</b>	<b>71</b>
2.1 Dictionnaire de preuves externes ou internes	72
2.2 Les dictionnaires de noms propres	73
2.3 Dictionnaires morphologiques	74
2.4 Comment utiliser des dictionnaires sous Intex	74
<b>3 CONCLUSION</b>	<b>75</b>

## **EXTRACTION DE NOMS PROPRES** **77**

<b>1 L'EXTRACTEUR DE NOMS PROPRES <i>EXTRACNP</i></b>	<b>77</b>
1.1 Architecture du système	77
1.2 L'ordre de passage des règles	78
<b>2 LES NOMS DE PERSONNES</b>	<b>79</b>
2.1 La description des preuves externes et internes des noms de personnes	79
2.2 Morphologie des prénoms et patronymes	81
2.3 Combinatoire de la cascade pour les noms de personnes	84
2.4 Reconnaissance des coordinations de noms de personnes	86
<b>3 LES ORGANISATIONS</b>	<b>88</b>
3.1 Preuves internes de noms d'organisations	88
3.2 Preuves externes de noms d'organisations	93
<b>4 LES LIEUX</b>	<b>94</b>
4.1 Extraction par preuves externes	94
4.2 Utilisation des dictionnaires sans preuves	96
<b>5 RESULTATS</b>	<b>96</b>
5.1 Résultats de l'extraction des noms de personne	98
5.2 Résultats de l'extraction des noms d'organisations	99
5.3 Résultats de l'extraction des noms de lieux	100
<b>6 TOUS LES NOMS PROPRES</b>	<b>100</b>
<b>7 CONCLUSION</b>	<b>101</b>

## **QUEL ROLE POUR LES NOMS PROPRES DANS LA CLASSIFICATION DE TEXTES ?** **103**

---

---

<b>1</b>	<b>REPRESENTATION DES TEXTES</b>	<b>103</b>
1.1	Le modèle vectoriel	103
1.2	Nos représentations des textes	104
<b>2</b>	<b>PONDÉRATION DE TERMES ET MESURES DE SIMILARITÉ</b>	<b>105</b>
2.1	La mesure ensembliste Jaccard	105
2.2	Mesure de similarité avec pondération TF.IDF des termes	105
2.3	Fusion de données	106
2.4	Heuristiques liées aux noms propres pour le calcul de similarité	107
<b>3</b>	<b>COMMENT EVALUER NOS MESURES DE SIMILARITES ?</b>	<b>108</b>
3.1	Corpus de tests	108
3.2	Classification par partitionnement ( <i>clustering</i> )	108
<b>4</b>	<b>COMMENT COMPARER LES CLASSIFICATIONS HIERARCHIQUES OBTENUES ?</b>	<b>111</b>
<b>5</b>	<b>LES RESULTATS</b>	<b>113</b>
5.1	Remarques sur les mesures d'entropie et de pureté	113
5.2	Comparaison des résultats obtenus par différentes mesures de similarités	113
5.3	Comparaison des résultats obtenus par des mesures de similarités fusionnées	115
<b>6</b>	<b>CONCLUSION</b>	<b>116</b>
<b>CONCLUSION ET PERSPECTIVES</b>		<b>117</b>
<hr/>		
	<b>ANNEXE A : FICHE TECHNIQUE DU SYSTEME CASYS</b>	<b>119</b>
	<b>ANNEXE B : METHODOLOGIE DE VERIFICATION DU DECOUPAGE EN PHRASES</b>	<b>123</b>
	<b>ANNEXE C : LEMMATISATION DES TEXTES</b>	<b>125</b>
	<b>ANNEXE D : CORPUS DE TEST DES SIMILARITES ET RESULTATS</b>	<b>129</b>
	<b>PUBLICATIONS</b>	<b>141</b>
	<b>BIBLIOGRAPHIE</b>	<b>143</b>
	<b>INDEX</b>	<b>159</b>

# REMERCIEMENTS

## GLOSSAIRE

---

- Agglutinante** : se dit d'une langue suivant le processus de l'agglutination. Langue qui présentent la caractéristique structurelle de l'agglutination, c'est-à-dire, l'accumulation après le radical d'affixes distincts, pour exprimer les rapports grammaticaux. Ainsi, en turc, à partir de "**ler**" (marque du pluriel) et de "**i**", (marque du possessif), on formera avec le radical **ev**, "*maison*" les mots *evler* "*maisons*" (nominatif pluriel), *evi* "*maison*" (possessif singulier), *evleri* "*maisons*" (possessif pluriel). Les mots d'une langue agglutinante sont ainsi analysables en une suite de morphèmes nettement distincts.
- Antonomase\*** : Figure qui consiste à remplacer, en vue d'une expression plus spécifiante ou plus suggestive, un nom propre par un nom commun (*le Sauveur* pour *Jésus-Christ*) ou un nom commun par un nom propre (*un Tartuffe* pour *un hypocrite*).
- Collocation\*** : Emploi d'un terme relativement à d'autres, toutes variantes morphologiques confondues, et sans égard à la classe grammaticale. *Les noms de fruit comme pomme, orange, poire, pêche (...) se trouvent en collocation fréquente avec dessert, manger, doux, fruit, etc.* (HALLIDAY, *Ét. de ling. appliquée*, t. 1, 1962, p. 22).
- Concordance** : L'élaboration d'une concordance consiste à rechercher dans un texte toutes les occurrences d'un mot ou d'un autre motif linguistique, puis à les présenter, une par ligne, chacune dans son contexte. Les applications relèvent de la lexicographie, de l'apprentissage des langues et de l'exploitation de bases de données littéraires [Laporte, 1997].
- Contexte\*** : Ensemble des unités d'un niveau d'analyse déterminé (phonème, monème ou morphème, unité lexicale, syntagme, phrase...) constituant l'entourage temporel (parole) ou spatial (écriture) d'une unité, d'un segment de discours dégagé par une analyse de même niveau.
- Coréférence\*\*** : en linguistique, fait que deux expressions ou phrases aient le même référent, le même objet commun.
- Gentilé** : nom des habitants d'une ville, d'un pays.
- Graphie\*** : Façon d'écrire un mot. *Clé et clef sont des graphies différentes du même mot, toutes deux en usage* (MOUNIN 1974).
- hyperonymie\*** : Terme dont le sens inclut celui d'un ou de plusieurs autres. Anton. *hyponyme*. *La chaise est un siège (hyperonyme de chaise) dont le propre est d'être avec dossier mais sans bras (différences spécifiques)* (R. MARTIN, *La Déf. verbale*, Université de Metz, Centre d'Analyse Syntaxique, 1978, p. 12).
- Hyponymie\*** : (Mot) dont le signifié est hiérarchiquement plus spécifique que celui d'un autre (d'apr. DDL 1976). Anton. *hyperonyme*. (*La Linguistique*, Paris, Denoël, 1969, p. 193).
- Lemmatisation** : Technique qui permet de présenter un mot sous une forme de référence (exemple : l'infinitif pour les verbes) telles que les entrées des mots dans les dictionnaires.
- Lemme** : produit de la lemmatisation.

**Morphologie\*** : Étude des différentes catégories de mots et des formes qu'ils présentent dans une langue (flexion et dérivation). (MAR. *Lex.* 1933, p.122). *LING. MOD.* Description de la structure interne des mots et étude des règles qui régissent cette structure (d'apr. *Lang.* 1973).

**Onomastique\*** : Discipline ayant pour objet l'étude des noms propres et comprenant diverses branches telles que l'anthroponymie, l'hydronymie et la toponymie:

**Particule\*** : *Particule onomastique*, dite à tort *particule nobiliaire*. Préposition faisant partie intégrante d'un nom patronymique qu'elle précède, mais qui n'atteste pas l'authenticité de cette noblesse.

**Patronyme\*** : Nom de famille, notamment lorsqu'il est transmis par le père (par opposition au prénom).

**Polysémie\*** : Propriété d'un signifiant de renvoyer à plusieurs signifiés présentant des traits sémantiques communs. Anton. *Monosémie*.

**Racine\*\*** : Forme dénuée des désinences et des affixes et qui est porteuse de la signification du mot.

**Sémantique\*** : Étude d'une langue ou des langues considérées du point de vue de la signification; théorie tentant de rendre compte des structures et des phénomènes de la signification dans une langue ou dans le langage.

**Synonymie\*** : Caractère, propriété qui unit deux mots, deux expressions synonymes; relation entre deux ou plusieurs signifiants, telle que ces signifiants sont interchangeable, sans qu'il y ait variation concomitante du signifié (*Media* 1971).

**Syntaxe\*** : Partie de la grammaire traditionnelle qui étudie les relations entre les mots constituant une proposition ou une phrase, leurs combinaisons, et les règles qui président à ces relations, à ces combinaisons.

**Toponyme** : Nom de lieu de localité.

\* définition du TLFi (Le Trésor de la Langue Française Informatisé), <http://zeus.inalf.fr>

\*\* Encyclopædia Universalis

## INTRODUCTION

---

Avec la très grande quantité d'informations disponibles sur Internet ou, de manière plus générale, sur support informatique, créer des outils qui automatisent l'exploration ou l'extraction d'informations pertinentes dans les textes est crucial. Les systèmes d'extraction d'information, de recherche d'information et de fouille de textes sont de plus en plus nombreux et les recherches dans ces domaines deviennent interdépendantes.

Ces systèmes doivent faire face aux difficultés propres à l'écrit :

- les données textuelles contiennent des informations non structurées,
- les constructions langagières sont en partie imprévisibles.

Cette thèse consiste principalement en l'extraction d'information, et plus particulièrement, l'extraction des noms propres dans les textes journalistiques.

En effet, dans ces textes journalistiques, les noms propres sont très fréquents (10% en anglais selon [Coates-Stephens, 1993]). Les noms propres sont porteurs d'une information primordiale sur le sujet des articles de journaux ; ils sont très importants pour une compréhension précise des textes, mais ils sont très pauvrement représentés dans les ressources lexicales électroniques.

Cette thèse s'insère dans le projet **Prolex** initié et coordonné par Denis Maurel au Laboratoire d'Informatique de l'Université de Tours (LI). Prolex s'est d'abord focalisé sur la création d'une base de données de toponymes et de leurs dérivés puis a été étendu à tous les noms propres dans une perspective multilingue. Ce projet rassemble des travaux informatiques et linguistiques pour l'élaboration de ressources électroniques.

Les principaux travaux menés dans le cadre de Prolex portent sur :

*a) Des ressources linguistiques :*

- Création d'un dictionnaire relationnel des toponymes français [Belleil, Maurel, 1997] et internationaux [Piton, Maurel, 2000], d'hydronymes [Maurel, Piton, 1999]
- Et, plus récemment, étude structurelle d'une base de données multilingue de noms propres [Grass *et al.*, 2002].

*b) Des outils :*

- Etude des règles dérivationnelles des toponymes pour la reconstruction des gentilés et ethnonymes [Eggert *et al.*, 1998],
- Repérage et interprétation des noms propres déterminés [Garric, Maurel, 2000].

Cette thèse apporte un outil supplémentaire et essentiel au projet Prolex pour constituer des dictionnaires relationnels de noms propres : le système **extracNP** d'extraction des entités nommées.

Ce travail s'inscrit aussi dans le cadre de la linguistique harissienne, telle qu'elle a été mise en application par Maurice Gross à travers le système Intex [Silberztein, 1993] ;

Nous nous servons des ressources logicielles et linguistiques d'Intex pour développer un système à cascade de transducteurs nommé **CasSys** que nous utilisons dans ExtracNP. Les outils développés dans le cadre de cette thèse seront mis à la disposition des utilisateurs d'Intex.

À titre d'application des résultats de notre extraction, nous avons testé l'utilisation des noms propres dans la classification automatique de textes journalistiques : l'information dont ils sont porteurs les rend particulièrement intéressants pour obtenir une classification de textes.

Cette thèse n'est pas un travail unique mais une somme de travaux qui ont pour but l'extraction de noms propres et leur place dans le processus de classification. Nous présenterons d'abord les différents domaines de notre thèse : la linguistique des noms propres (cf. Chapitre 1), les systèmes d'extraction automatique (cf. Chapitre 2), les cascades de transducteurs et l'utilisation que nous faisons des outils d'Intex (cf. Chapitre 3 et 4). Puis nous décrirons notre système d'extraction des noms propres (cf. Chapitre 5).

Par ailleurs, nous présentons donc différentes mesures de similarité entre textes basées sur les noms propres pour évaluer leur importance et nous expliquons les méthodes d'évaluation de ces mesures ainsi que les résultats obtenus (cf. Chapitre 6).

# Chapitre 1

## LES NOMS PROPRES

---

Longtemps négligé, le statut des noms propres intéresse à nouveau la linguistique française moderne [Kleiber, 1994], [Gary-Prieur, 1994], [Noailly, 1994] etc. ainsi que le relativement jeune domaine du traitement automatique des langues [Maurel, Gueunthner, 2000], [Daille, Morin, 2000], etc.

Dans ce chapitre, nous donnons un aperçu du statut particulier des noms propres dans la langue française en §1. Nous discuterons de la morpho-syntaxe et de la sémantique des noms propres en §2, puis de leur reconnaissance et typage par des moyens automatiques en §3 et §4. Enfin, nous présentons quelques résultats chiffrés d'une étude en corpus afin de confronter nos idées à la réalité des textes journalistiques (§5).

### 1 Les noms propres en français

#### 1.1 Qu'est-ce qu'un nom propre ?

Il existe plusieurs manières de définir un nom propre mais aucune ne fait l'unanimité auprès des linguistes ; citons, par exemple, les définitions du nom propre et du nom commun que donne le *Bon Usage* [Grevisse, Goosse, 1986:751] :

*"Le nom propre n'a pas de signification véritable, de définition ; il se rattache à ce qu'il désigne par un lien qui n'est pas sémantique, mais par une convention qui lui est particulière."*

*"Le nom commun est pourvu d'une signification, d'une définition, et il est utilisé en fonction de cette signification."*

Les noms propres semblent nommer des entités sans les décrire et d'après Grevisse, n'ont pas de définition véritable. Prenons comme exemple deux articles du Petit Larousse, *Carcassonne* et *table*.

**Carcassonne** : chef lieu du département de l'Aude, sur l'Aude et le canal du Midi.

Tandis que le nom commun a une signification, une définition :

**Table** : n.f. meuble fait d'un plateau horizontal posé sur un ou plusieurs pieds / meuble sur lequel on place des mets.

Ces exemples illustrent les propos de [Gary-Prieur, 1994:7] : "alors que l'interprétation d'un nom commun ne met en jeu que la compétence lexicale, celle du nom propre requiert presque toujours une mise en relation avec le référent initial, qui mobilise des connaissances discursives." La définition de *Carcassonne* mobilise des

connaissances autres que la définition d'une *table* : la ville est située à travers sa fonction administrative (chef-lieu dans le département de l'Aude) et de sa position par rapport à des cours d'eau (la rivière et le canal qui passent dans cette ville). Mais cette définition est peu évocatrice pour un lecteur qui ignore où se trouvent l'Aude et le canal du Midi. Par contre, la définition de *table* permet à tous d'imaginer à quoi ressemble une table.

Le nom propre est donc situé dans l'espace et le temps. Il renvoie au domaine de la description dont parle [Molino, 1982] sous le nom de Deixis (Figure 1).

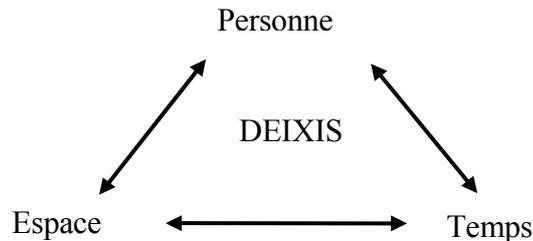


Figure 1: La Deixis

### 1.1.1 Niveau sémantique

Il existe trois grandes approches linguistiques quant à la question très controversée du sens des noms propres [Jonasson, 1994:114]. En voici un résumé :

- Le nom propre est *vide de sens* : il réfère sans désigner,
- Le sens du nom propre est *une description du référent* : soit on considère qu'il a un sens fort et qu'il identifie de manière univoque un référent, soit il a un sens réduit à des traits sémantiques généraux (trait féminin / masculin, humain / non-humain, etc.)
- Le sens du nom propre est *un prédicat de dénomination* : il ne décrit pas l'objet dénoté mais lui donne un nom, par exemple tel homme "est appelé" *Alexandre*.

Les noms propres n'ont donc pas de signification dans le sens où on l'entend pour un nom commun (*Carcassonne* versus *table*). Mais peut-on vraiment parler de deux catégories bien distinctes : est-ce que *Organisation des Nations Unies*, *Jardin des Plantes*, *Parisien*, *Mérovingien*, *Vivendi*, *EDF*, *Renault 5* ou *14 juillet 1789* sont des noms propres ? [Molino, 1982:10] remarque d'ailleurs, qu'en français, "*Tout peut être nom propre*" même une phrase complète et cite comme exemple les noms propres *Trompe-la-mort* ou *N'a-qu'un-œil*.

### 1.1.2 Un continuum nom propre-nom commun

En fait, il semble y avoir une continuité entre les noms propres et les noms communs, que nous allons montrer à travers quelques exemples.

- *Carcassonne* et *Napoléon* sont des noms propres

Selon Grevisse et l'acception commune, les "véritables noms propres" sont les noms de lieux (villes, villages, monuments, régions, pays, îles, montagnes...) et les noms de personnes.

- *Vivendi, EDF* semblent être des noms propres

Grevisse ne met pas les noms de sociétés dans la catégorie des noms propres. Pourtant, *Vivendi* et *EDF* ont les propriétés de noms propres classiques : ils désignent une entité dont nous avons une image mentale bien précise mais qui ne peut être définie comme on le ferait pour un nom commun.

- *Organisation des Nations Unies, Jardin des Plantes et Révolution française* semblent aussi être des noms propres.

*Organisation des Nations Unies* désigne une organisation unique et bien connue, le nom de *Jardin des Plantes* un lieu. Ces noms propres sont composés de noms communs et d'adjectifs ayant tous individuellement une signification qui peut aider à la compréhension de l'entité qu'ils désignent. D'ailleurs même les noms de villes ou de personnes, reconnus comme des noms propres véritables par Grevisse, peuvent être formés par des noms communs (ex : *Bourg-la-Reine, M. Couturier*).

*Révolution française* réfère aussi à un contenu précis. Ce n'est pas seulement une révolution qui aurait eu lieu en France mais un événement important de notre histoire. Le petit Larousse en donne la définition suivante : *(1789-1799) ensemble des mouvements révolutionnaires qui mirent fin, en France, à l'ancien Régime*. Cette définition la situe dans l'espace (*en France*) et le temps (*1789-1799, Ancien Régime*) à la manière d'un nom propre de ville ou de personne.

- *Renault 5* semble être un nom propre

*Renault 5* est le nom d'une marque de voiture reproduite à des milliers d'exemplaires mais ce terme désigne uniquement la voiture *Renault 5* connue pour ces caractéristiques particulières. Alors qu'il est possible de fabriquer une *table*, on ne peut "fabriquer" une *Renault 5* soi-même : ce serait de la contrefaçon. C'est bien sûr une notion extralinguistique mais qui s'applique aux noms propres en général. Il est interdit de se faire passer pour *M. Dupont*. [Rey-Debove, 1994] ajoute qu'un nom de marque désigne une classe engagée dans une hyperonymie mais considère ces noms comme de faux noms propres.

- *Parisien, Mérovingien* sont des dérivés de noms propres

Les dérivés de noms propres (gentilés, ethnonymes, périodes historiques, etc.), bien qu'ils aient une définition (ex : *Parisien* = habitant de Paris, *Mérovingien* = descendant de Mérovée), sont souvent considérés comme des noms propres. Ils ne désignent pas un individu, mais un groupe qui a une certaine individualité.

- *table* est un nom commun

En fait, il semble difficile de délimiter les noms propres des autres noms ; il y a une continuité entre l'ensemble des noms propres et l'ensemble des noms communs (Figure 2).



Figure 2 : Continuum nom propre - nom commun

Les informaticiens qui travaillent dans le domaine de l'extraction d'information, ont abordé le problème de manière pragmatique. Ils ont défini la notion d'**entités nommées**<sup>1</sup> pour regrouper tous les éléments du langage définis par référence : les noms propres au sens classique, les noms propres dans un sens élargi mais aussi les expressions de temps et de quantité.

Dans la suite, nous utiliserons indifféremment les termes de **noms propres** ou **entités nommées** pour désigner les noms propres au sens large du terme.

## 1.2 La productivité des noms propres

Comme les autres mots, les noms propres participent à la création morpho-syntaxique des locuteurs du français. Lexicalisation, détermination et dérivation les rendent particulièrement productifs.

### 1.2.1 La lexicalisation du nom propre

Un *frigorifère* et un *bic* ne sont plus des noms propres : ces noms sont utilisés comme synonyme de *réfrigérateur* et *stylo-bille*. Ce qui est arrivé pour la marque *Frigidaire* ne s'est pas produit pour la marque *Renault* : une *Renault* n'est pas le synonyme d'une voiture quelconque [Rey-Debove, 1994].

On distingue cette lexicalisation de l'antonomase. L'antonomase du nom propre est la lexicalisation d'un nom propre sous une forme métaphorique (ex : un *don juan*, un *eldorado*) : "L'antonomase, comme la métaphore, consiste à rapprocher deux termes, un comparé et un comparant. La différence entre l'antonomase et la métaphore tient à la nature du terme comparant : un membre quelconque d'une classe dans le deuxième cas, un nom propre dans le premier cas. [Flaux, 1991]"

<sup>1</sup> Nous détaillerons ce que sont les entités nommées dans le Chapitre 2.

### 1.2.2 La détermination des noms propres

On distingue les noms propres dont l'emploi est caractérisé par une absence d'article (*César, Tours, Guerlain* sont des noms de personnes, villes, marques, etc.) ou qui s'utilisent avec un article défini, par exemple : *la France, la Seine, la SNCF* (ce sont plutôt des noms de pays<sup>2</sup>, régions, fleuves ou désignations par sigles) [Noailly, 1994]. Ils sont désignés par [Gary-Prieur, 1994] en termes de *noms à article défini lexical*.

La présence d'un article défini devant un nom propre peut être due à deux autres phénomènes :

- Dans le premier cas, les noms propres intègrent un article défini. Ce dernier appartient à leur morphologie, il ne dispose d'aucune autonomie (ex : *Le Corbusier, Le Mans, La Fontaine, Les Seychelles* etc.). L'article est d'ailleurs introduit par une majuscule comme le serait le premier élément d'un nom propre composé (ex : *Jean-Michel, Saint-Dizier*). Seuls les toponymes précédés d'une préposition autorisent la contraction de l'article défini qui leur appartient avec la préposition *de* (ex : *des nouvelles du Caire* versus *une architecture de Le Corbusier*). Les noms propres de pays ou de région se construisent presque systématiquement avec l'article défini (ex : *la France, la Loire, la Bretagne* mais pas *Malte, Israël*).
- Dans le deuxième cas, [Garric, Maurel, 2000] ont identifié trois types d'interprétations du nom propre déterminé à partir des études de [Kleiber, 1991], [Gary-Prieur, 1994] et [Jonasson, 1994] :
  - l'interprétation métaphorique ou exemplaire :  
*Reste qu'un José Bové serait plus crédible que Philippe de Villiers* (Libération)
  - l'interprétation dénomminative :  
*Un deuxième homme, un certain Branko Mlaco, trente-six ans* (Le Monde)
  - l'interprétation fractionnée ou multipliée :  
*Et le dernier Sartre, le Sartre de la fin, celui de l'entretien avec Benny Lévy* (Le Monde)

### 1.2.3 La dérivation du nom propre

Les dérivés de noms propres proviennent principalement de :

- Noms de personne (ex : *chiraquien, pasteuriser, déstaliniser, homérique, gaulliste*)
- Noms de lieu géographique (adjectif dérivé : *italien*, nom d'habitant : *Italien*, nom de la langue : *italien*, préfixe dérivé et mots composés : *italo*, dérivés par suffixation : *italianisant*, verbe : *franciser*) [Eggert et al., 1998].

<sup>2</sup> Exceptés les emplois d'étiquetage (sur les cartes géographiques par exemple) ou dans les contextes syntaxiques imposant la non-détermination (après la préposition *en*, ex : *en France*).

### 1.3 Les dictionnaires de noms propres

Le besoin de la reconnaissance des noms propres pose le problème de la création de dictionnaires électroniques les répertoriant. On doit se demander s'il est possible et utile de créer de telles ressources ?

On constate que les noms propres et leurs dérivés sont souvent absents des dictionnaires papiers [Rey, 1977] ou électroniques [Maurel *et al.*, 1996].

[Rey, 1977:30] déclare : "... il faut mentionner l'important problème que pose l'absence des noms propres dans les dictionnaires de langue. En effet, les noms propres fournissent non seulement des lexicalisations (un harpagon) mais des monèmes productifs (marxiste, marxisme, martien). Certes, on trouve américain dans tous les dictionnaires, mais berrichon, puis limougeot et enfin castelpontain ... ou encore giscardien, aznavourien... ne peuvent pas tous figurer dans le lexique décrit".

[Sampson, 1989] dit que la construction de dictionnaires est héroïque et longue car ils appartiennent à une classe très ouverte de mots. On peut ajouter que les noms propres sont trop nombreux pour être placés dans un dictionnaire. Un grand nombre d'entre eux sont inconnus [Mani *et al.*, 1996] et ils sont plus ou moins fugaces selon leur catégorie. Nous parlerons ici des catégories les plus fréquentes : toponymes, anthroponymes et organisations.

Les **toponymes** sont des noms propres relativement stables i.e. le nom donné à tel ou tel toponyme change rarement [Piton, Maurel, 1997], mais des modifications se font au gré de l'histoire :

- *La République Tchèque* est souvent nommée *Tchéquie* mais ce mot est absent des dictionnaires,
- *Châlons-sur-Marne* est devenu récemment *Châlons-en-Champagne*<sup>3</sup>.

Les toponymes sont, de plus, en nombre assez limité. Un dictionnaire de toponymes est construit depuis quelques années dans le cadre du projet Prolex [Maurel *et al.*, 1996] et sera utilisé dans notre système d'extraction des noms propres.

Les **anthroponymes** (prénoms et patronymes) sont très nombreux. [Liu, Haas, 1988] font remarquer que sur 75 millions de personnes aux Etats-Unis, il existe 1,5 millions de noms de personnes différents et quelques centaines de ces noms de personnes représentent 60% des noms portés outre-atlantique ; il faut répertorier 175 000 noms pour couvrir 88% des noms de personnes américains.

Enfin, les **noms d'organisations**, très nombreux aussi, sont certainement encore plus insaisissables : leur apparition et leur disparition se faisant au gré de l'économie. Pour compliquer le problème, une organisation porte un nom qui possède parfois des variantes (ex : *Organisation des Nations Unies*, *Nations Unies*, *ONU*). Pour [Rau *et al.*, 1991], construire un lexique de noms de compagnies est même impossible.

La grande quantité de noms de personnes et d'organisations pose le problème de la notoriété, car tous ces noms ne peuvent être placés dans un dictionnaire. Un tel dictionnaire devrait être construit sur les bases d'une certaine notoriété des noms propres à y introduire ; la difficulté est alors de distinguer la mode passagère de la célébrité durable ? Comment décider ou non de la notoriété des noms propres ?

<sup>3</sup> Voir à ce propos l'étude de [Guerrin, 1998].

Dans le cadre du traitement automatique des langues, les dictionnaires ne répertorient que peu les noms propres ; les noms propres sont donc repérés comme mots inconnus au même titre que les fautes d'orthographe, les abréviations et les dérivations peu communes. Les dictionnaires de noms propres sont donc d'un très grand intérêt pour éviter ce problème. [Belleil, Maurel, 1997] proposent que les noms propres, étant donné leur particularité, soient répertoriés dans des dictionnaires d'un style nouveau : les **dictionnaires relationnels**. Ce sont plus que de simples listes de mots ; les noms propres et leurs relations y sont conservées sous la forme d'une base de donnée. Par exemple, les relations entre un lieu et son/ses gentilés sont conservées (Ex : *Les habitants de Saint-Dizier* → *les Bragards*).

## 2 Typologies de noms propres

Nous présentons ici un certain nombre de travaux sur la typologie morpho-syntaxique, sémantique ou pragmatique de noms propres.

### 2.1 Le typage par un lecteur humain

Si on ne connaît pas un nom propre, le discours général nous fera comprendre de quel type il est : lieu, personne ou autre ? [Belleil, 1997] prend pour exemple un extrait du récit fantastique *Bilbo le Hobbit* contenant des noms de lieux et de personnes fruits de l'imagination de Tolkien ; pourtant, les noms propres de ce texte sont reconnus et catégorisés par le lecteur en noms de lieux ou de personnes sans aucun problème. Il en est de même pour les noms propres présents dans des textes dont le récit est plus proche du fond commun culturel.

#### *Extrait de Bilbo le Hobbit*

*Au nord de **Carrock**, l'orée de **Mirkwood** se rapprochait des bords de la **Grande Rivière** et, bien qu'à cet endroit les montagnes descendissent plus près, **Beorn** leur avait conseillé de prendre ce chemin ; car, à quelques jours de chevauchée en plein nord du **Carrock**, se trouvait l'entrée d'un sentier qui traversait **Mirkwood** et ...*

Pour le lecteur humain il y a donc deux niveaux de reconnaissance du nom propre, qui ne sont pas exclusifs l'un de l'autre :

- Reconnaissance *a priori* : le nom propre est reconnu parce qu'il est connu, et il appartient à l'univers commun des connaissances (ex : *La Loire, Paris, Sartre*).
- Reconnaissance *a posteriori* : la graphie (majuscule) et la sémantique des prédicats<sup>4</sup> induisent le type du nom propre ou le précisent en cas d'ambiguïté.

<sup>4</sup> Voir, par exemple, les travaux de Gaston Gross sur les classes d'objets [Gross, 1994].

## 2.2 Typologies morpho-syntaxiques des noms propres

Les noms propres font partie de la catégorie syntaxique des noms. Les noms propres ont, en français, certaines caractéristiques qui les distinguent des noms communs "la plupart du temps" : absence d'article, absence de flexion morphologique, présence d'une majuscule. Ces caractéristiques ne sont pas absolues ; comme nous l'avons vu dans les paragraphes précédents, il existe des noms propres employés avec des articles (ex : *la Suisse*), d'autres ont une marque de flexion (ex : *des Allemands, les deux Corées*) et ils ne se résument pas à des mots portant une majuscule initiale.

[Allerton, 1987] propose une classification morpho-syntaxique des noms propres, pour l'anglais, divisée en quatre types :

- Les noms propres purs (ex : *Paris, Michael, Miami*).
- Les noms propres mixtes sont des combinaisons de noms propres et de noms communs (ex : *Hyde Park, the River Thames*).
- Les noms propres basés sur des noms communs sont entièrement composés de noms communs, (ex : *the Black Sea, Central Park*).
- Les noms propres codés : c'est-à-dire les acronymes et les combinaisons de lettres et de nombres (ex : *BBC, M25*).

[Jonasson, 1994] propose pour le français deux types de noms propres :

- Les noms propres purs : ce sont des "noms propres véritables" (ex : *Jean-Pierre Papin, Boulogne-Billancourt*) ; Jonasson remarque qu'ils ne renseignent pas sur les propriétés de l'objet qu'ils désignent. Ce sont des noms de lieux ou de personnes que l'on peut repérer à l'aide de la majuscule.
- Les noms propres mixtes ou à base descriptive : les noms propres mixtes contiennent des noms propres purs et des noms communs (ex : *le Collège de France, la Tour Eiffel, le golfe Juan*) mais aussi des adjectifs (ex : *La Nouvelle-Orléans*). Les noms propres à base descriptive sont composés d'un ou plusieurs noms communs éventuellement accompagnés d'adjectifs ou de prépositions (ex : *le Massif Central, Banque Centrale Européenne, Comité International Olympique, Syndicat National des Pilotes de Ligne, la Grande Barrière de Corail*). Les noms propres à base descriptive ou mixte sont des lieux, rues, places, parcs, bâtiments, organisations de toutes sortes.

Jonasson ajoute que "*Si on considère un trait comme la monoréférentialité, il est bien plus caractéristique des Npr<sup>5</sup> descriptifs ou mixtes que des Npr purs. Les premiers sont en général forgés expressément pour convenir à un seul particulier, qu'ils désignent en le décrivant, et ne sont normalement pas utilisés associés à d'autres particuliers.*"

On peut souligner le fait qu'un nom propre mixte de l'anglais (ex : *Wall Street Journal*) sera considéré comme nom propre pur dans un texte en français.

Ces deux typologies de Allerton et Jonasson sont similaires : la seule distinction tient au fait que, en français, les noms propres ne portent pas forcément une majuscule

---

<sup>5</sup> Npr est l'abréviation du mot "nom propre" pour les linguistes.

sur tous les mots qui les composent. [Daille, Morin, 2000] introduisent donc une typologie basée sur des critères graphiques plutôt que sur la présence de noms communs ou non dans le nom propre :

- Les noms propres simples : un seul mot commençant par une majuscule (ex : *Marseille, France*)
- Les noms propres complexes : ceux-ci sont composés de plusieurs unités lexicales pleines comportant toutes une majuscule (ex : *Quai d'Orsay, Grand Palais*) mais ils peuvent contenir indifféremment des noms communs et des noms propres.
- Les noms propres mixtes : ils sont constitués de plusieurs unités lexicales comportant ou non des majuscules comme le *palais de Chaillot* ou le *Front populaire*.

### 2.3 Typologies sémantiques des noms propres

Certaines classes de noms propres regroupent des noms, surtout de lieu ou de personne, qui n'ont pas de signification en eux-mêmes (ex : *Londres, Espagne, Française*) ; d'autres noms propres se décrivent eux-mêmes (ex : *le Mont-Saint-Michel*).

[Zabeeh, 1968], [Bauer, 1985], [Grass, 2000] proposent des classifications propres à l'onomastique.

La typologie de [Zabeeh, 1968] sépare les noms propres en noms de personnes, en périodes de temps et périodes historiques, en artefacts (produits, arts, objets culturels, les nombres, les mythes), en noms de lieux (géopolitique, géographiques, astronomiques, fictions) et en noms d'institutions économiques et politiques.

[Bauer, 1985], quant à lui, propose une classification pragmatique décrite et étendue par [Grass, 2000]. Bauer décrit cinq classes de noms propres :

- anthroponymes : personnes individuelles et groupes
- toponymes : pays, villes, hydronymes, etc.
- ergonymes : objets et produits manufacturés
- praxonymes : faits historiques, maladies, événements
- phénonymes : ouragans, astres, etc.

[Grass *et al.*, 2002] définissent une classification basée sur deux niveaux hiérarchiques comme [Paik *et al.*, 1996]<sup>6</sup> avec une couverture des noms propres très complète. Le premier niveau est celui des hypertypes : un hypertype correspond aux traits sémantiques primitifs (anthroponymes, toponymes, ergonymes et pragmonymes). Le second niveau est celui des types : il comprend des champs lexicaux relativement homogènes, en relation d'hyponymie<sup>7</sup> avec les hypertypes. Les hypertypes et les types sont décrits ci-dessous :

<sup>6</sup> [Paik *et al.*, 1996] présentent, à travers leur travail de TAL (Traitement Automatique des Langues), une taxonomie des noms propres à deux niveaux mais ils oublient les événements, phénomènes naturels, etc.

<sup>7</sup> L'**hyponymie** désigne la relation entre un terme spécifique et le terme générique qui désigne la classe générale à laquelle appartient l'objet ou le concept représenté par le terme spécifique. Le terme spécifique est un **hyponyme** tandis que le terme générique est son **hyperonyme** (ex : la **rose** est une sorte de **fleur**).

- **Anthroponymes** : patronyme, prénom, célébrité, dynastie (ex : *Carolingien*), divinité ou personnage mythique ou fictif (ex : *Zeus, Astérix*), entreprise (ex : *Nestlé, General Electric, BASF*), association ou parti (ex : *parti socialiste, CGT, MRAP*), ensemble artistique ou club sportif, établissement public ou privé, organisation internationale (ex : *Unesco, ONG*), gentilé ou ethnonyme. On remarque que les noms d'organisations sont considérés comme des anthroponymes.
- **Toponymes** : région (dans un pays), ville, groupe de pays (ex : *Otan, Europe de l'Est*), quartier ou voie, bâtiment, hydronyme (ex : *la Marne, le Lac du Der*), géonyme (ex : *la plaine de la Limagne, l'Aiguille du midi, le Gulf Stream*), objet céleste, lieu mythique ou fictif (ex : *l'Eldorado*).
- **Ergonymes** : appellation commerciale (ex : *une Scénic*), entreprise (ce sont aussi des anthroponymes), œuvre, objet mythique ou fictif (ex : *Excalibur*), vaisseau (ex : *A340, la fusée Ariane, le porte-avion Charles-de-Gaulle*).
- **Pragmonymes** : phénomène météorologique (ex : *El Nino*), catastrophe (ex : *Tchernobyl, cyclone Mitch*), manifestation artistique ou sportive (ex : *le Festival des Vieilles Charrues, les Jeux Olympiques*), fête (religieuse ou nationale), événement historique ou politique (guerre, révolution, putsch, génocide, sommet).

Du côté du traitement automatique des langues, les travaux sur l'extraction des noms propres ont conduit les informaticiens à proposer des typologies. [Coates-Stephens, 1993] propose, par exemple, une typologie en sept classes qui est suffisamment simple pour réaliser une classification automatique mais tient compte de la réalité des noms propres. Cette taxonomie est un peu plus complexe que celle proposée par le programme MUC<sup>8</sup> (personnes, lieux, organisations) :

- Noms de personnes
- Noms de lieux
- Noms d'organisations
- Noms d'origine (ex : *Algérien*) qui sont pour la plupart des informations contenues dans le contexte
- Noms de législations (ex : *taxe Tobin, loi 1901, traités...*), indices boursiers (*indice Nikkei ...*)
- Noms de sources d'informations (média, journaux, ...)
- Noms d'événements : guerres, révolutions, désastres, foires...
- Noms d'objets

---

<sup>8</sup> Nous préciserons ce qu'est le programme MUC dans le Chapitre 2 sur l'extraction automatique des noms propres.

On notera cependant qu'une telle taxonomie, si elle est d'usage pratique pour le traitement informatique, est hétéroclite. Au contraire, la partition de [Grass *et al.*, 2002] en hypertype est linguistiquement homogène<sup>9</sup>.

#### 2.4 Les ambiguïtés sémantiques des noms propres

Le nom propre, comme le nom commun, peut avoir des homonymes. [Maurel *et al.*, 1996] en distingue trois types :

- *L'homonymie dans un même type* désigne des objets qui portent le même nom. Par exemple, en France, 17 localités s'appellent *Saint-André*.
- *L'homonymie par élision* désigne des noms propres dont on omet une partie. Ainsi, 23 noms composés de ville sont formés à partir de *Neuilly*. Souvent le lecteur humain saura reconstruire la forme effacée. L'évocation de *Nogent* renverra, par exemple, à *Nogent-sur-Marne* dans le *Val-de-Marne*, à *Nogent-sur-Loir* dans la *Sarthe* ou à *Nogent-sur-Seine* dans l'*Aube*.
- *L'homonymie dans les différents types* désigne un mot qui peut correspondre à des noms propres appartenant à des types différents. Par exemple, *Washington* est un nom de lieu et un nom de personne.

### 3 Reconnaissance automatique des noms propres

La reconnaissance et le typage des noms propres sont deux problèmes croisés. En effet, pour extraire un nom propre, on utilise des indices qui permettent de le repérer, mais aussi de le catégoriser.

Un système d'extraction ne peut faire de distinction entre un nom propre et un nom commun, et surtout le catégoriser avec la seule syntaxe. Comme le dit [McDonald, 1996], les noms propres ont un aspect systématique et une structure qu'on peut décrire à l'aide d'informations souvent plus lexicales que syntaxiques.

Le premier indice naïf pour extraire les noms propres est la majuscule : il est très insuffisant car, comme nous l'avons déjà vu par des exemples, un nom propre peut être composé de plusieurs mots dont certains ne portent pas de majuscules. De plus, la majuscule qui se trouve sur le premier mot d'une phrase est ambiguë : s'agit-il d'un nom propre ou, simplement, d'un mot banal portant une majuscule, par exemple, parce qu'il est au début d'une phrase ?

En fait, les indices les plus sûrs pour détecter et catégoriser les noms propres sont leurs contextes d'apparitions droits ou gauches et/ou leur composition interne.

---

<sup>9</sup> Il est assez traditionnel en linguistique d'associer la classe des humains à celle des humains collectifs qui utilisent les mêmes structures prédicat-argument.

### 3.1 Preuve interne et externe

[McDonald, 1996] propose un outil de reconnaissance et de classification des noms propres fondé sur les notions de **preuve interne** et **preuve externe** que nous présentons ici. La plupart des outils informatiques de reconnaissance de noms propres utilisent ces preuves sans les nommer ainsi.

Les **preuves internes** se trouvent à l'intérieur même du nom propre. Ce sont des mots qui permettent de le repérer à coup sûr et, éventuellement de le typer. Les preuves internes peuvent prendre la forme d'un ou plusieurs mots ou d'une abréviation connue pour faire partie d'un nom propre.

Exemples de preuves internes :

***Organisation** des Nations Unies*

la ***Bourse*** de Paris

le ***Mont*** Blanc

Microsoft ***Inc.***

*Wall Street **Journal***

La preuve interne est prise en compte par tous les systèmes d'extraction de noms propres. De tels mots se trouvent en début ou fin de noms propres (surtout dans les noms d'organisation).

Un prénom peut aussi être utilisé comme preuve interne.

Exemples :

***George** Sand*

***Jean-Jacques** Goldman*

La **preuve externe** est le contexte d'apparition des noms propres dans la phrase. Les noms propres sont une manière de référer à des individus d'un type spécifique. Dans le discours, surtout journalistique, l'auteur donne aux lecteurs des informations complémentaires sur les personnes, lieux, organisations qu'il cite : ces informations peuvent aider, dans un processus automatique, à déterminer le type d'un nom propre. La preuve externe sera aussi appelée **contexte droit** ou **contexte gauche** selon que le contexte se trouve à la droite ou à la gauche du nom propre dans la phrase.

Exemples de preuves externes :

la ***ville de*** Marseille

le ***professeur*** Tournesol

le ***groupe*** Vivendi

*Derrick, **l'inspecteur** allemand*

La preuve externe est nécessaire pour obtenir des performances élevées dans l'extraction automatique des noms propres. Si on ne prend en compte que la preuve interne, on peut aboutir à des erreurs de classification. Par exemple, le nom propre contenu dans l'expression "*la société Hugues Aircraft*" pose problème : *Hugues* est un prénom. La seule preuve interne apportée par ce prénom fait penser que *Hugues Aircraft* est un nom de personne ce qui est contredit par la preuve externe. Ce type d'erreur de catégorisation est très fréquent entre noms de personnes et noms d'organisations.

Ces preuves internes et externes seront largement utilisées dans notre description linguistique des noms propres et de leurs contextes.

### 3.2 Structure syntaxique des noms propres accompagnés d'une preuve externe

[Noailly, 1991], [Gary-Prieur, 1994], [Forsgren, 1994] travaillent sur les constructions dans lesquelles peuvent intervenir des noms propres ; le nom propre peut être épithète, attribut, sujet, objet, en apposition.

Le Tableau 1 décrit les formes élémentaires que peut prendre la preuve externe accompagnée d'un nom propre. Dans ce tableau, <NP> symbolise un nom propre et <CTXT> une preuve externe (ou contexte).

Formes	Exemples
<b>Nom propre épithète ou en apposition</b>	
<CTXT> <NP>	<i>le chorégraphe Maurice Béjart ...</i>
<CTXT>, <NP> ,	<i>le chef d'orchestre, von Karajan, ...</i>
<NP>, <CTXT> ou ..., <NP>, <CTXT>	<i>Helmut Kohl, le chancelier allemand ...</i> <i>Pierre Dupont, 59 ans, ...</i> <i>Et, Tours, ville d'art et d'histoire ...</i>
<b>Nom propre sujet ou attribut</b>	
<NP> <Verbe> <CTXT>	<i>Deneuve est une actrice ...</i>
<CTXT> <Verbe> <NP>	<i>Sa filiale est Vivendi ...</i>
<b>Nom propre avec contextes entre parenthèses</b>	
<NP> (<CTXT>) ce contexte entre parenthèses peut être lui-même un nom propre (appartenance à une organisation, groupe politique, lieu)	<i>Bruce Lee (mort en juillet 1973, à trente-trois ans) ...</i> <i>M. Germain (Tours) ...</i>
<b>Nom propre avec une préposition</b>	
<CTXT> de <NP>	<i>le président de l'ONU</i> <i>le lac du Der</i>
<b>Coordinations de noms propres</b>	
<CTXT_pluriel> <NP>, ... , <NP> et <NP>	<i>Les auteurs antiques Platon, Homère, Pline et Epicure ...</i>
<NP>, <NP>, ... , et <NP>, <CTXT_pluriel>	<i>Hugo, Rimbaud et Lamartine, poètes français ...</i>
<CTXT_pluriel> de <NP>, <NP> et <NP>	<i>Les villes de Paris et Marseille ...</i>

Tableau 1 : Preuves externes (contextes) et noms propres

Les noms propres apparaissent donc dans des constructions complexes de groupes nominaux ou avec des appositions. Leurs contextes peuvent contenir simplement un

adjectif (ex : *l'anglais Tony Blair*), ou prendre une forme plus complexe (ex : *le chef du gouvernement français, Lionel Jospin*). Une incise<sup>10</sup> peut permettre d'exprimer une relation entre noms propres (ex : *Frédéric Mitterrand, neveu de François Mitterrand* ou *Canal Plus, filiale de Vivendi*). Ces structures peuvent être composées pour donner des formes plus complexes (ex : *la société française Canal Plus, filiale de Vivendi* = <CTXT > <NP>, <CTXT> de <NP>).

#### 4 Les noms propres et leurs différentes preuves en corpus

Nous présentons ici une étude que nous avons réalisée en corpus afin de mieux nous rendre compte de la quantité de noms propres que l'on peut catégoriser grâce à des indices linguistiques. Il est assez aisé de créer une grammaire qui va identifier la plupart des noms propres. Par contre, leur attribuer un type est beaucoup plus difficile.

Dans cette étude, nous avons dénombré les preuves externes et internes suivant le type des noms propres qu'elles accompagnent. Ce travail a été réalisé sur un journal *Le Monde* complet, daté du 12 janvier 1999 (545 Ko, 90 056 formes simples).

Un journal complet est long à lire et surtout à étudier à la main. Nous avons donc procédé de la manière suivante : nous avons localisé, à l'aide du système de traitement de textes *Intex*<sup>11</sup>, toutes les formes pouvant être des noms propres.

Nous avons trouvé au total 3 755 noms propres faisant partie des catégories les plus fréquentes : personnes, lieux et gentils, organisations, objets et marques, phénomènes et catastrophes, manifestations et événements. Les résultats<sup>12</sup> sont résumés dans le Tableau 2.

Nous avons étudié deux autres journaux *Le Monde*. Nous avons constaté que les quantités de chacun des types de nom propre varie assez fortement par rapport à l'étude présentée ici (la taille du journal variant d'un jour à l'autre) ; c'est pourquoi il faut prendre les quantités de noms propres du Tableau 2 à titre indicatif. Par contre, les proportions de preuves internes et externes pour chaque type sont les mêmes. Nous avons aussi étudié brièvement un petit corpus du journal *Ouest France* et de la *Nouvelle République* (journal local d'Indre-et-Loire) ; nous avons remarqué que leur qualité rédactionnelle différente s'accompagnait d'une diminution très importante de la présence de preuves externes pour les noms de personnes (heureusement il y a bien souvent un prénom), par contre, dans ces journaux, les noms de lieux sont plus couramment accompagnés d'une preuve externe que dans *Le Monde*.

Les résultats du travail sur *Le Monde* montrent que 50,4% de tous les noms propres de ce journal sont accompagnés de preuves : c'est assez peu. En fait, ce sont les noms de lieux et les gentils qui sont la cause d'une proportion de preuve aussi faible.

Nous avons dénombré, pour les trois principaux types de nom propre, le nombre d'occurrence catégorisable par une preuve interne et/ou par une preuve externe. Nous présentons, dans le Tableau 3, les résultats obtenus pour les catégories de noms propres les plus importantes : personnes, organisations et lieux.

<sup>10</sup> Voir le travail sur les incises de [Fairon, 2000].

<sup>11</sup> *Intex* est un système de traitement automatique des textes dont nous parlerons plus longuement dans le Chapitre 3.

<sup>12</sup> [Day, Palmer, 1997] ont analysé des corpus dans de nombreuses langues (anglais, espagnol, portugais, japonais, chinois et français) et montrent les grandes différences entre les quantités de noms de personnes, lieux et organisations dans les textes de ces différentes langues.

Type d'entités	Nombre total d'occurrences dans le corpus		Noms propres accompagnés d'une preuve interne ou externe permettant de les typer	
	Qté	%	Qté	%
Personnes	1 014	27	945	93,2
Organisations	1 012	27	661	65,3
Lieux	1 323	35,2	267	20,2
Gentilés et ethnonymes	316	8,4	0	0
Objets, marques	39	0,8	24	61,5
Phénomènes et catastrophes	1	0,03	0	0
Manifestations et événements	50	1,3	36	72
Tous les noms propres	3 755	100	1 894	50,4

Tableau 2: Résultats de notre étude selon le type du nom propre

	Présence d'un(e) ...				
	Contexte gauche	Contexte droit	Contexte gauche associé à une preuve interne	Preuve interne	Sans aucune preuve ou trop ambigus
Personnes	19,4	0,3	8,6	60,1	11,6
Organisations	12,7	1,2	1	51,2	33,9
Lieux	16,8	0	1	3	79,2

Tableau 3 : Proportions des différentes preuves selon le type de nom propre (en %)

#### 4.1.1 Noms de personnes

Les noms de personnes représentent 27% des noms propres dans ce journal. Parmi eux, 93,2% peuvent être repérés et surtout catégorisés comme noms de personnes.

19,4% de noms de personnes sont détectables à l'aide d'un contexte gauche (le plus souvent une civilité, une fonction ou un métier), mais la majorité (60,1%) le sont par une preuve interne qui est un prénom ou une particule de patronymes non ambiguë en français (ex : *von*, *di*, etc.). On remarque que 8,6% des noms de personnes sont accompagnés d'une double preuve : un contexte gauche et une preuve interne (ex : *la danseuse Marie-Claude Pietragala*).

#### 4.1.2 Noms d'organisations

Les noms de compagnie ou d'organisation représentent environ 27% des noms propres dans notre corpus (comme les personnes) dont 65,3% sont détectables ce qui est, par contre, beaucoup moins que les noms de personnes.

51,2% des noms d'organisations commencent par une preuve interne : un premier mot capitalisé qui dénote d'un nom d'organisation (ex : *Fonds Monétaire International*).

D'ailleurs, 7% des noms d'organisations contenant une preuve interne sont en fait des noms étrangers (Ex : *Mellon Foundation*, *Bank of America*) et 1% peuvent être trouvés grâce à leur morphologie, par exemple la présence d'une esperluette (Ex : *AT&T*) La preuve externe gauche représente 12,7% des noms d'organisation (Ex : *filiale de Vivendi*) et seulement 1,2% pour le contexte droit.

#### 4.1.3 Noms de lieux

Les noms de lieu représentent 35,2% des noms propres dans ce journal. Les lieux ont rarement un contexte qui permet de les catégoriser (16,8%) et encore moins une preuve interne (1% seulement).

Beaucoup de lieux sont présents accompagnés de la préposition *en* (5,6%) : d'après nos observations ce mot est un très bon indicateur puisque dans notre corpus tous les mots commençant par une majuscule présents derrière *en* étaient des régions ou des noms de pays (Ex : *en Amérique*, *en Lozère*, exception : *en Avignon*). Cependant, il faut rester prudent vis à vis de cet indice (ex : *croire en Dieu*, *être en Deug*).

La preuve interne est surtout fréquente dans les noms de ville (ex : *Chalons-en-Champagne*, *Chaumont-sur-Loire*) ou dans des noms de départements (ex : *Ile-et-Vilaine*) sous la forme de tirets et de prépositions.

#### 4.1.4 Tous les noms propres

Il est possible de repérer 1 894 noms propres grâce à leurs contextes ou à une preuve interne sur un total de 3 553, ce qui représente moins de la moitié.

On se rend compte que le repérage des noms propres est plus ou moins difficile selon leur type : les noms de personnes ont très souvent un contexte d'apparition qui permet de les trouver et de les catégoriser comme personnes. Plus de la moitié des noms d'organisations sont trouvés grâce à des preuves internes. Par contre, les noms de lieux, gentilés et ethnonymes sont rarement accompagnés de contextes et de preuves internes : il faudra, en fait, utiliser des dictionnaires de toponymes pour les repérer. Les objets, phénomènes et événements sont très peu nombreux. Cependant, la moitié des noms d'objets de ce journal avaient un contexte gauche et la moitié des événements une preuve interne (ex : *Festival des vieilles charrues*) ; les événements sont assez proches des noms d'organisations.

Les noms propres, qui ne peuvent être catégorisés, peuvent au moins être repérés à l'aide d'indices syntaxiques et de la présence de lettres capitales. Il faudra utiliser d'autres moyens pour les typer ; par exemple, leur affecter le même type que leurs homonymes trouvés dans un même texte.

## 4.2 Variation des noms propres

Enfin, pour reconnaître les noms propres, il faudra, comme en terminologie [Jacquemin, 2001], prendre en compte leurs variations [Daille, Morin, 2000] :

- Les variantes graphiques (ex : *Parti Socialiste* ou *Parti socialiste*) [Trouilleux, 1997]
- Les variantes tels que les sigles ou abréviations,

- Certaines coordinations (ex : *le grand palais et le petit palais* → *le grand et le petit palais*)
- Les ellipses (ex : *école normale sup* → *normale sup* → *normale*)

## 5 Conclusion

Nous avons présenté, dans ce chapitre, le statut du nom propre dans la langue à travers la difficulté de le définir, et à travers ses constructions morpho-syntaxiques et sémantiques.

L'étude en corpus nous montre que les différents types de noms propres ne sont pas égaux devant leur découverte car leurs contextes d'apparitions et leurs quantités dans les articles de journaux sont très différents. Nous remarquerons principalement que :

- Tous les noms propres ne sont pas accompagnés d'indices suffisants pour les typer (notamment les noms de lieux),
- Catégoriser les noms propres demandera une forte lexicalisation des grammaires d'extraction.

Il faudra donc adapter les moyens d'identifier les noms propres en fonction de ces différences.



## Chapitre 2

### **LES SYSTEMES D'EXTRACTION D'ENTITES NOMMEES**

---

Dans ce chapitre, nous commencerons par un historique des systèmes d'extraction d'entités nommées et par le rôle des conférences MUC (Message Understanding Conference) dédiées à l'extraction d'information dans les textes (cf. §1). Puis nous présenterons les différents types de systèmes d'extraction existants (§2) ainsi que leur emploi de la description des langues et le traitement des ambiguïtés (§3 et §4).

#### **1 Les recherches dans le domaine de l'extraction automatique d'entités nommées**

##### **1.1 L'avant MUC**

Le premier travail informatique sur les noms propres semble être celui de Borkowski en 1967. Il proposait alors un système d'identification des noms de personnes et de leurs titres dans les journaux en utilisant des listes de marqueurs (preuves internes).

À la fin des années 70, des chercheurs de Yale créaient le système SAM [Gershman, 1977] : ce système travaillait sur des articles de journaux et analysait la structure syntaxique des groupes nominaux complexes dans lesquels les noms propres peuvent apparaître.

Le système *Frump* (Fast Reading Understanding and Memory Program) [Dejong, 82] extrayait de l'information sur des désastres naturels à partir d'extraits de journaux. Frump repérait les éléments intéressants du texte tels que les entités nommées à l'aide de mots clefs. Puis [Kuhns, 1988] réalisa un système d'analyse du contenu des textes, nommé *NAS* (News Analysis System), qui reconnaissait les noms de compagnies et de personnes. D'autres systèmes se focalisèrent sur un type de noms propres : par exemple, l'extracteur de noms de compagnies de [Rau, 1991] ou le système Synoname [Siegfried, Bernstein, 1991] qui recherche les noms d'auteurs dans les bibliographies.

Le système le plus complet, réalisé avant que MUC ne mette en évidence l'intérêt de l'extraction des noms propres, s'appelle *Funes*<sup>13</sup> [Coates-Stephens, 1993].

En fait, peu de recherches en informatique portent sur les noms propres avant le début des années 1990. C'est, en 1995, que MUC créa la tâche d'extraction des entités nommées.

---

<sup>13</sup> Cf. §2.1 et §2.2 de ce chapitre.

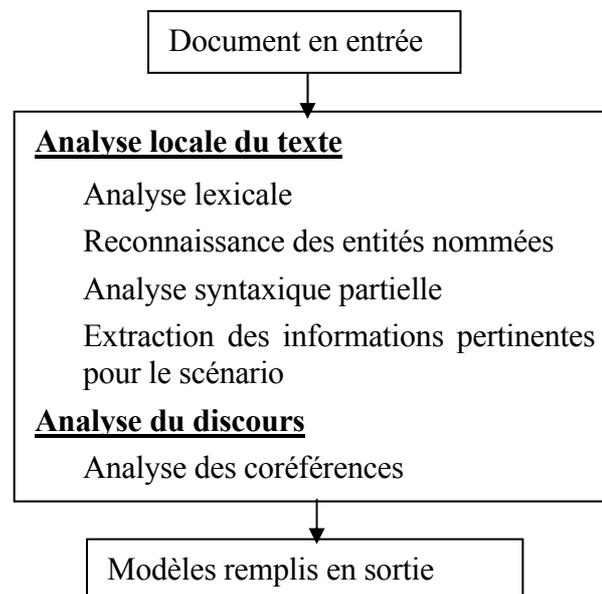
## 1.2 La conférence MUC

La conférence **MUC**<sup>14</sup>, créée par le programme américain Tipster, cherche à promouvoir la recherche en compréhension automatique des textes par l'amélioration de l'extraction d'information et la création de systèmes plus portables. MUC prend la forme d'un concours par lequel les systèmes participants sont évalués.

La tâche principale proposée par MUC est l'extraction d'information ; c'est-à-dire le remplissage de modèles constitués de champs d'information structurés. Ces champs permettent de décrire les événements dont parlent les textes.

Depuis 1987, sept conférences MUC ont été organisées. Au début, MUC proposait un travail d'extraction d'information à réaliser sur des messages de la Navy. Pour MUC 3 en 1991, la tâche fut compliquée par un nombre croissant de champs à remplir dans les formulaires ; les textes portaient sur des actes terroristes en Amérique centrale et du Sud. MUC 5 traitait des dépêches dont le thème était les de joint-ventures et les produits microélectroniques, et MUC 6 des textes du Wall Street Journal. Les premières conférences MUC travaillaient sur des corpus très homogènes de textes : ces corpus se sont compliqués au et à mesure des conférences successives.

À partir de MUC 6, des nouvelles sous-tâches ont été définies : la recherche de coréférences, la désambiguïsation du sens des mots, la recherche des entités nommées, etc. La tâche d'extraction d'information nécessitait déjà la prise en compte de ces sous-tâches, mais MUC 6 propose de les améliorer "individuellement" sans tenir compte de la tâche complète d'extraction d'information.



*Figure 3 : Description générale d'un système d'extraction d'information*

Pour comprendre l'imbrication des tâches proposées par MUC, la Figure 3 présente l'architecture générale d'un système d'extraction d'information [Grishman, 1997]. L'extraction est réalisée en deux étapes :

<sup>14</sup> <http://www.muc.saic.org>

- on procède à une analyse locale permettant de trouver des faits simples,
- on analyse le discours afin d'inférer des faits plus complexes.

La tâche d'extraction des entités nommées a lieu pendant l'analyse locale, après une analyse lexicale du texte, et simplifie le reste des traitements.

### 1.3 Les métriques d'évaluation

Pour mieux comprendre la suite, voici une présentation des métriques d'évaluation utilisées.

Lorsqu'un système doit produire un ensemble de résultats à partir de textes, deux types d'erreurs peuvent se produire comme le rappelle [Popescu-Belis, 1999]. Le système peut omettre des réponses par rapport au résultat idéal : c'est le **silence**. Il peut ajouter des réponses par rapport au résultat idéal : c'est le **bruit**.

Les résultats obtenus par les systèmes d'extraction sont évalués par différentes métriques d'évaluation<sup>15</sup> dont les plus répandues sont le rappel, la précision et la F-mesure.

Le rappel est d'autant plus élevé que le silence est faible ; La précision est d'autant plus élevée que le bruit est faible.

Le rappel et la précision sont intimement liés ; en effet, en tentant par exemple d'améliorer la précision (diminuer le bruit), on risque d'empiéter sur la capacité du système à trouver le plus grand nombre possible de documents pertinents (diminuer le silence). Plus les valeurs de rappel et précision sont élevées<sup>16</sup>, plus les résultats du système évalué sont bons.

- Le **rappel** mesure la quantité de réponses pertinentes d'un système par rapport au nombre de réponses idéales.

$$R = \frac{\text{nombre de réponses pertinentes du système}}{\text{nombre de réponses idéal}}$$

*Formule 1 : Rappel*

- La **précision** est la quantité de réponses pertinentes du système parmi l'ensemble des réponses qu'il a fourni.

$$R = \frac{\text{nombre de réponses pertinentes du système}}{\text{nombre de réponses fournies par le système}}$$

*Formule 2 : Précision*

- La **F-mesure** [van Rijsbergen, 1979] permet de combiner en une seule valeur les mesures de précision et de rappel de manière pour

<sup>15</sup> Ces métriques d'évaluation ne sont pas propres à l'extraction d'information. Elles sont utilisées dans de nombreux autres domaines tels que la recherche documentaire [van Rijsbergen, 1979].

<sup>16</sup> Les valeurs de rappel et de précision varient entre 0 et 1 (on les multiplie souvent par 100 pour obtenir une proportion).

pénaliser les trop grandes inégalités entre ces deux mesures (Formule 3). Le paramètre  $\beta$  permet de régler les influences respectives de la précision P et du rappel R ; il est très souvent fixé à 1. Selon Popescu-Belis, la **F-mesure** favorise les systèmes dont les deux valeurs sont homogènes.

$$\text{F-mesure} = \frac{(\beta+1) \cdot P \cdot R}{\beta \cdot P + R}$$

*Formule 3 : F-mesure, formule générale*

$$\text{F-mesure} = \frac{2 \cdot R \cdot P}{R + P}$$

*Formule 4 : F-mesure*

#### 1.4 La tâche d'extraction des entités nommées NE

En 1995, la 6<sup>ème</sup> édition de MUC propose pour la première fois la tâche d'extraction des entités nommées pour l'anglais, NE<sup>17</sup>, suivie, en 1996, par la création de la Multilingual Entity Task (MET) [Maiorano, Wilson, 1996] qui propose d'évaluer les résultats de systèmes d'extraction d'entités nommées en espagnol, japonais et portugais.

La tâche NE distingue trois types d'entités à reconnaître et à catégoriser : ENAMEX, TIMEX et NUMEX [Chinchor, 1997].

- TIMEX contient les expressions de temps et de dates.
- NUMEX rassemble les nombres et pourcentages, ainsi que les quantités monétaires.
- Les entités de type ENAMEX sont composées par les noms propres ou assimilés et par les sigles.

ENAMEX propose une catégorisation assez grossière mais qui suffit aux applications de recherche d'information. Les entités ENAMEX sont de trois sortes :

- **Organisations** : noms de sociétés, gouvernements et autres entités organisationnelles (ex : *Organisation des Nations Unies* ou *Microsoft*),
- **Personnes** : noms de personnes (ex : *Bill Clinton*) ou de familles (ex : *les Kennedy*),
- **Noms de lieux** : lieux définis politiquement ou géographiquement comme les villes, départements, régions internationales, hydronymes, montagnes (ex : *Allemagne, Paris, Silicon Valley*).

[Sekine, Eriguchi, 2000] précisent que, depuis 1999, les tâches NE et MET proposent de s'intéresser aux artefacts (produits manufacturés, œuvres, prix ...).

---

<sup>17</sup> NE = Named Entity

La tâche NE préconise l'utilisation de la notation de la TEI<sup>18</sup> (Text Encoding Initiative) pour baliser les noms propres trouvés (exemples ci-dessous).

```
<ENAMEX TYPE="ORGANIZATION">Taga Co.</ENAMEX>
<ENAMEX TYPE="ORGANIZATION">Bridgestone Sports
Co.</ENAMEX>
<ENAMEX TYPE="ORGANIZATION">European
Community</ENAMEX>
Mr. <ENAMEX TYPE="PERSON">Harry Schearer</ENAMEX>
the <ENAMEX TYPE="PERSON">Kennedy</ENAMEX> family
Secretary <ENAMEX TYPE="PERSON">Robert
Mosbacher</ENAMEX>
<ENAMEX TYPE="LOCATION">Northern California</ENAMEX>
<ENAMEX TYPE="LOCATION">Taiwan</ENAMEX>
```

L'extraction des entités nommées regroupe 15 participants à MUC 6. La tâche est assez simple et les systèmes obtiennent de très bons résultats dès sa première édition. [Grishman, Sundheim, 1996] notent que la plupart des participants arrivent à plus de 90% de rappel et de précision, le meilleur score étant de 96% avec une précision de 97%. [Sundheim, 1995] dit que le rappel pour la tâche d'extraction des entités nommées réalisée par un humain est de 97% seulement. Les résultats affichés par les différents systèmes à MUC 6 sont très bons mais il faut rappeler qu'ils traitent de textes très homogènes limités à un domaine assez restreint.

Pour MUC 7, l'ensemble de textes d'entraînement fourni aux participants portait sur les accidents d'avions tandis que les tests d'évaluation des systèmes portaient sur des événements de lancement. Le changement de domaine a eu des effets sur les systèmes participants puisque leurs scores finaux ont été bien inférieurs à ce qui était espéré. C'est pourquoi, la tâche NET obtient une F-mesure moyenne de 97% pour MUC 6 alors qu'elle est inférieure à 94% pour MUC-7.

## 2 Les différentes méthodes d'extraction des entités nommées

### 2.1 Trois types de systèmes

Il existe trois types principaux de systèmes pour extraire les noms propres ; ils sont rappelés dans [Poibeau, 1999], [Sekine, Eriguchi, 2000], [Daille, Morin, 2000].

Voici un certain nombre de ces systèmes<sup>19</sup> d'extraction d'entités nommées, classés en trois catégories et selon la langue qu'ils traitent. Dans la suite (§2.2, §2.3, §2.4), nous

---

<sup>18</sup> [www.tei-c.org](http://www.tei-c.org)

ne présenterons pas une analyse exhaustive de chacun d'eux, mais les principales méthodes employées par ces systèmes, puis d'un point de vue "plus linguistique", nous donnerons un aperçu de ce qui a été éprouvé.

a) **Les systèmes à base de règles** : La majorité des systèmes utilisent cette approche. Les systèmes typiques à base de règles utilisent les preuves externes et internes, ainsi que des dictionnaires de noms propres pour les repérer. Les règles sont écrites à la main.

– **En anglais**

- FUNES (Figuring-out Unknown Nouns from English) [Coates-Stephens, 1993]
- FASTUS [Hobbs *et al.*, 1996] obtient une F-mesure de 94% à MUC 6 pour l'extraction des entités nommées.
- PNF (Proper Name Facility) [McDonald, 1996]
- LaSIE (LArge Scale Information Extraction) [Wakao *et al.*, 1996]. Les résultats de rappel sont de 91%, 90%, 88% pour les organisations, personnes, lieux et la précision est resp. de 91%, 95%, 89% sur des textes du Wall Street Journal pour MUC 6 [Stevenson, Gaisauskas, 2000].
- Nominator [Wacholder *et al.*, 1997] extrait 91% des noms propres avec une précision 92%, seuls 79% des noms propres trouvés sont catégorisés par le système.
- NetOwl Extractor (système commercial de Isoquest Inc.) [Krupka, Hausman, 1998] obtient une F-mesure de 96,42% pour MUC 6 et seulement 91,32% pour MUC 7.

– **En français**

- Exoseme [Wolinski *et al.*, 1995] est un système de filtrage avec un module sur les entités nommées en français. Ce système obtient 90% de rappel et en catégorise correctement 85% sur des dépêches AFP.
- ThingFinder [Trouilleux, 1997].

– **Autres langues ou multilingues**

- FACILE (Fast and Accurate Categorisation of Information by Language Engineering) [Black *et al.*, 1998]. *Facile* est un projet européen dont la tâche principale est le filtrage de nouvelles en quatre langues différentes (anglais, allemand, italien, espagnol). Ce système

---

<sup>19</sup> Quatre problèmes empêchent une comparaison des résultats des différents systèmes d'extraction d'entités nommées dont nous parlons ici :

- les résultats obtenus par chacun d'eux ne sont pas donnés de façon homogène (rappel seul ou précision seule ...).
- les sources textuelles sur lesquelles leur extraction est basée ne sont pas forcément les mêmes (sauf participants de MUC).
- l'extraction des entités nommées est une sous-tâche du système et les auteurs ne donnent que les résultats du système dans son entier (extraction d'information par exemple).
- certains systèmes n'ont pas été évalué ou en tout cas, il n'y a pas d'article contenant leur évaluation.

Nous donnons leurs résultats sur la tâche NE lorsqu'ils ont été communiqués !

traite les entités nommées et a obtenu, à MUC 7, un rappel de 92% et une précision de 93% pour cette tâche.

- NERC [Karkaletsis *et al.*, 1999] pour le grec
- b) **Les systèmes à apprentissage** : ils construisent leurs connaissances automatiquement grâce à un apprentissage sur un corpus d'entraînement [Collins, Singer, 1999].
  - **En anglais**
    - Alembic [Aberdeen *et al.*, 1995].
    - BBN Identifinder [Miller *et al.*, 1999], [Bikel, 1997].
    - MENE (Maximum Entropy Named Entity) [Borthwick *et al.*, 1998].
    - [Collins, Singer, 1999] obtiennent 91% de rappel et 83% de précision sur le New York Times.
    - Answer extraction [Abney *et al.*, 2000].
  - **Autres langues**
    - [Cucerzan, Yarowski, 1999] en turc, roumain, etc. Ils obtiennent une F-mesure de 70,47% pour le roumain, 54,30% pour l'anglais, 55,32% pour le grec, 53,04% pour le turc et 41,70% pour l'Hindi.
  - **Indépendant du langage**
    - [Gallippi, 1996] obtient une F-mesure de 94,0% sur l'anglais, 89,2% pour l'espagnol.
- c) **Les systèmes hybrides** : Ils utilisent des règles écrites à la main mais construisent aussi une partie de leurs règles à l'aide d'informations syntaxiques et d'informations sur le discours tirées de données d'entraînement grâce à des algorithmes d'apprentissage, des arbres de décisions.
  - **En anglais**
    - LTG system (Language Technology Group) [Mikheev *et al.*, 1998] est le meilleur à MUC 7, nous lui réservons une section afin d'expliquer comment il procède pour extraire les entités nommées.
    - [Lin, 1998] obtient une F-mesure de 86,37%.
  - **En français**
    - [Senellart, 1998] en français et en anglais.
    - SemTex [Poibeau, 1999] en français et anglais, il annonce 80% de rappel sur le journal *Le Monde*.
    - [Fourour, 2002] obtient, avec Nemesis, 90% de rappel et 95% de précision.
  - **Autres langues**
    - [Cucchiarelli *et al.*, 1998] en italien avec 84% de précision et 85% de rappel sur le journal *Il Sole 24 Ore*.
    - SweNam [Dalianas, Aström, 1998] en suédois.

Nous allons maintenant donner un aperçu des méthodes utilisées par les trois types de systèmes présentés ci-dessus.

## 2.2 Les systèmes à base de règles

Ces systèmes décrivent la structure des entités nommées grâce à des règles qui utilisent un étiquetage syntaxique, des preuves internes et externes ou des dictionnaires de noms propres<sup>20</sup>.

Le système *FUNES* [Coates-Stephens, 1993] utilise un grand nombre de règles syntaxiques et sémantiques pour décrire et structurer, à partir d'expressions régulières, l'environnement de chaque type de noms propres dans la langue anglaise. Les règles écrites par Coates-Stephens ont inspiré d'autres systèmes tels que *PNF* (module du système *SPARSER*) [McDonald, 1996] ou *Facile*. *PNF* commence par délimiter les noms propres avec un automate à nombre fini d'états et revoie ensuite la délimitation des noms propres en classant ceux-ci d'après leurs preuves internes et externes.

*Nominator* [Wacholder *et al.* 1997] utilise un ensemble d'heuristiques basées sur des motifs de mots en majuscules, des mots clefs, la ponctuation et la localisation des noms propres dans la phrase et dans le texte, mais ce système ne fait pas usage de dictionnaires. Il construit une liste de candidats noms propres puis recherche les variantes (abréviations, ellipses) de ces noms propres.

Après étiquetage syntaxique du texte, *Nerc* [Karkaletsis *et al.*, 1999] fait usage de listes de mots et de règles pour extraire les noms propres sur des textes en langue grecque.

*FASTUS* [Hobbs *et al.*, 1996] est un système d'extraction d'information à base de cascade de transducteurs. Une partie de ces règles d'extraction concerne les entités nommées.

*LaSIE*, décrit dans [Gaisauskas *et al.*, 1995] et [Wakao, Gaisauskas, 1996], est un système basé sur un étiquetage syntaxique du texte et une grammaire locale des noms propres décrite par des règles Prolog. La grammaire utilise une approche hétérogène et contient des informations graphologiques, syntaxiques, sémantiques. Finalement, ce système révisé la catégorie des noms propres par une analyse du discours<sup>21</sup>.

*Facile*, présenté par [Black *et al.*, 1998], applique un poids aux règles ; si deux règles détectent une même partie du texte, c'est celle qui a le plus fort poids qui est préférée. Pour *Facile*, la catégorie qui donne les plus mauvais résultats est celle des organisations : ceci est partiellement expliqué par des règles qui dépendent trop du domaine ainsi qu'une base de données de preuves mal adaptées. *NetOwl Extractor* [Krupka, Hausman, 1998] propose, lui aussi, une phase de compétition des règles : cette phase permet de sélectionner l'interprétation la plus probable pour un texte. Le poids d'une règle est fonction de son numéro d'ordre, de sa longueur et du type de nom propre qu'elle repère.

Les systèmes français proposent des solutions équivalentes.

[Trouilleux, 1997] utilise des transducteurs et ne repère que les noms propres contenant au moins une majuscule. Il réalise une analyse grammaticale locale avec deux grammaires concurrentes : une grammaire spécifique écrite à la main et une générale qui a été automatiquement extraite d'un corpus de référence où les entités nommées étaient connues.

---

<sup>20</sup> Nous décrivons l'usage fait de ce type de ressources dans la section 3 de ce chapitre.

<sup>21</sup> Si deux variantes d'un nom propre sont trouvées et que l'une est catégorisée mais pas l'autre, on attribue la même catégorie à la seconde.

*Exoseme* [Wolinski *et al.*, 1995] commence par segmenter les noms propres par une analyse morphologique puis catégorise les noms propres grâce à leur contexte.

On note que les stratégies à base de règles ne sont pas idéales pour des corpus composés de textes ne répondant pas à des critères rédactionnels stricts [Kosseim, Poibeau, 2001], par exemple, les e-mails.

### 2.3 Les systèmes à apprentissage

[Collins, Singer, 1999] ont créé un classifieur basé sur un apprentissage non supervisé travaillant sur des textes anglais. Les seules règles sont au nombre de 7 et sont extrêmement simples :

*New York est un lieu, California est un lieu, U.S. est un lieu,  
Tout nom qui contient Mr. est un nom de personne, Tout nom  
qui contient Incorporated est un nom d'organisation, I.B.M. est  
une organisation, Microsoft est une organisation*

Ces 7 règles de base (*seed rules*) permettent à l'algorithme d'apprentissage de s'amorcer et de déduire de nouvelles règles. Par exemple, avec la phrase *Mr. Cooper, a president of ...*, le système infère que *president* prédit un nom de personne car *Mr.* prédit un nom de personne.

Le système *Answer extraction* [Abney *et al.*, 2000] comporte une sous-tâche d'extraction des entités nommées. Il découvre d'abord des candidats noms propres ; la classification en lieux, organisation et personnes est réalisée ensuite en utilisant la méthode proposée par [Collins, Singer, 1999] et présentée ci-dessus.

[Cucerzan, Yarowski, 1999] ont créé un algorithme de repérage des noms propres basé sur un apprentissage itératif des motifs contextuels et morphologiques. Ce système utilise des probabilités et un ensemble de données d'entraînement très petit ; l'utilisateur du système doit fournir des exemples de noms de personnes, de prénoms, ainsi que de noms de lieux (les noms d'organisations ne sont pas gérés). L'algorithme d'apprentissage par amorçage<sup>22</sup> apprend, à partir de ces exemples et travaille d'un texte non annoté, en se basant sur les préfixes et suffixes de mots qui sont de bons indicateurs pour certaines classes d'entités nommées dans des langues comme le roumain et le turc. Le système prend en compte les contextes des noms propres lorsqu'il n'y a pas de règles morphologiques.

Le système *MENE* proposé par [Borthwick *et al.*, 1998] réalise une reconnaissance statistique des entités nommées en étiquetant les parties d'entités nommées avec quatre états possibles : "*start*", "*unique*", "*end*", "*continue*" et "*other*". La phrase *Jerry Lee Lewis flew to Paris* est étiquetée de la manière suivante *Person\_start person\_continue person\_end other other location\_unique*.

*BBN IdentiFinder* (anciennement nommé *Nymble*) [Bikel *et al.*, 1997], [Boisen *et al.*, 2000] utilise une méthode d'apprentissage basée sur les modèles de Markov cachés pour reconnaître les noms propres, sans aucun prétraitement et indépendamment de la langue. L'apprentissage est réalisé sur des corpus annotés.

---

<sup>22</sup> Algorithme d'Expectation - Maximization Bootstrapping ou EM Bootstrapping.

[Béchet *et al.*, 2000] proposent un système utilisant des arbres de décisions ("Semantic Classification Trees") qui permettent de typer les mots préalablement étiquetés<sup>23</sup> d'un texte.

*Alembic* [Aberdeen *et al.*, 1995], à MUC 6, est un système basé sur l'algorithme d'apprentissage conduit par les erreurs proposé par [Brill, 1995].

[Gallipi, 1996] crée une stratégie d'acquisition à base d'arbres de décision. Il utilise des heuristiques indépendantes des langues (aidées par la morphologie, le lexique ou la syntaxe) et tente d'en acquérir automatiquement de nouvelles. Les résultats sur l'espagnol et le japonais sont moins bons que sur l'anglais, car Gallippi applique les arbres de décisions de l'anglais directement sur les textes espagnols et japonais (les arbres spécifiques à l'espagnol et au japonais sont créés ensuite grâce aux résultats obtenus avec les arbres de décisions anglais : les arbres de décisions sur l'espagnol et le japonais ne sont donc pas écrits spécifiquement pour ces langues).

## 2.4 Les systèmes hybrides

À la fin des années 80, [Church, 1988] proposait déjà une procédure d'étiquetage stochastique (à l'aide de chaînes de Markov cachées) et travaillait principalement sur les groupes nominaux, et notamment ceux qui contiennent des noms propres.

[Cucchiarelli *et al.*, 1998] réalisent un système en langue italienne avec un apprentissage statistique. Il utilise un corpus d'apprentissage, un analyseur syntaxique, un dictionnaire de synonymes, et environ 250 règles de base pour avoir un modèle initial de contextes typiques de noms propres.

*SweNam* [Dalianas, Aström, 1998] extrait des noms propres en suédois. Ce système combine des techniques d'apprentissage (sur 10 000 articles) et des règles pour construire un étiqueteur d'entités nommées. Le suédois est une langue agglutinante : *SweNam* recherche donc des suffixes connus pour extraire et typer les noms propres (ex : *Parken* pour un parc, *Fabriken* pour une usine ...).

[Lin, 1998] a créé une base de données de collocation de mots. Cette base est utilisée pour créer automatiquement des règles d'extraction de noms propres en utilisant un classifieur de type "réseau Bayésien".

Pour ce qui est des systèmes français, [Senellart, 1998] semi-automatise la construction des transducteurs pour la reconnaissance des noms de personnes. Les transducteurs sont construits à l'aide d'une concordance<sup>24</sup> dont les parties pertinentes sont choisies à la main et vont être intégrées automatiquement dans des transducteurs qui vont reconnaître les noms propres. Les groupes nominaux décrits dans les transducteurs sont composés de noms de personnes et de leurs contextes (titres, fonctions, métiers etc.). Les transducteurs sont passés les uns après les autres sur le texte mais il ne s'agit pas d'une cascade de transducteurs (dont nous verrons les principes au Chapitre 3) car ceux-ci n'utilisent pas les transformations de leurs prédécesseurs.

<sup>23</sup> [Béchet *et al.*, 2000] étiquette les mots des texte à l'aide des tri-grammes ; l'étiquetage est morpho-syntaxique et sémantique (prénom, nom de famille, pays, villes, organisations).

<sup>24</sup> Une concordance est obtenue en appliquant un transducteur sur un texte : il s'agit de la liste des séquences de mots repérées par ce transducteur.

[Poibeau, 1999] utilise lui aussi des transducteurs dans le module *SemTex* du projet ECRAN<sup>25</sup>: ce système repère principalement les noms d'entreprises et les noms de leurs dirigeants afin de faire de la veille économique. *SemTex* repère les noms propres avec des patrons prédéfinis en anglais et français sur les journaux *Le Monde* et *Herald Tribune*. Un seul transducteur, fortement récursif, contient des appels à d'autres transducteurs qui décrivent les grammaires des entités nommées suivantes : email, URL, date, lieu, personne, compagnie. Le problème d'un transducteur contenant toutes les grammaires est le suivant : si deux règles de même longueur de ce transducteur peuvent s'appliquer sur une séquence du texte, le résultat sera aléatoire car il dépend de l'ordre dans lequel les règles du transducteur ont été compilées ; or, on ne peut ordonner les règles qui se trouvent dans un même transducteur. *SemTex* utilise les noms propres qu'il a "appris" avec ces grammaires pour extraire les mots inconnus qui leur sont identiques ou en partie égaux et les étiqueter avec le même type.

[Fourour, 2002] propose tout récemment le système *Nemesis* ; il réalise une première reconnaissance de noms propres à l'aide de lexiques de preuves externes et internes associées à des étiquettes sémantiques ; puis des expressions régulières s'appuyant sur la preuve interne sont appliquées aux textes pour extraire les noms propres. De nouveaux lexiques sont créés automatiquement afin de compléter la reconnaissance.

LTG, proposé par [Mikheev et al, 1998], est le système à base de règles qui obtient les meilleurs résultats à MUC ; c'est pourquoi nous lui réservons une place particulière et le présentons dans la partie suivante.

## 2.5 Le meilleur système selon MUC 7 : LTG system

Le système *LTG* est le meilleur pour l'anglais à MUC 7. *LTG* obtient un rappel de 93,6% pour une précision de 95% sur les entités ENAMEX.

*LTG* (Language Technology Group) est composé d'un système *ltdok* qui permet de trouver des noms propres candidats. Ensuite, *FsgMatch* utilise les résultats de *ltdok* ainsi que des règles de transduction pour extraire les entités nommées et les typer.

Le processus d'extraction des entités ENAMEX par *LTG* est le suivant :

- Etape 1 : Passage des règles les plus sûres (*sure-fire rules*). Après un étiquetage en parties du discours du texte, cette étape applique des règles qui contiennent preuves internes et externes. Cette phase obtient une précision de 96 à 98%. Les noms de lieux sont reconnus grâce à un dictionnaire et à leur contexte, par exemple *in* suivi de *Washington*.
- Etape 2 : Reconnaissance partielle (*partial match*). Cette étape est réalisée grâce à l'interaction de deux outils. Le premier collecte les entités déjà identifiées dans le document. Ensuite le système génère des variantes d'entités nommées en changeant l'ordre des mots ou en supprimant. Le second outil utilise cette liste de noms propres

---

<sup>25</sup> Ecran (Extraction of Content Research At Near-market) est un projet européen d'extraction d'information qui travaille sur le français, l'anglais et l'italien. Ce projet propose d'offrir un accès filtré à la masse d'information textuelle délivrée par la télévision et les ordinateurs personnels, <http://www.dcs.shef.ac.uk/research/ilash/Ecran/>.

annotés et un algorithme probabiliste pour finaliser l'étiquetage des noms propres.

- Etape 3 : Règles relaxées (*rule relaxation*). Ce sont des règles plus souples en terme de contraintes contextuelles. Par exemple, un prénom (connu d'un dictionnaire de prénoms) suivi d'un ou plusieurs mots capitalisés inconnus sera suffisant pour catégoriser ce nom propre comme nom de personne. Cette étape résout aussi le problème des conjonctions et celui des entités en début de phrases.
- Etape 4 : Nouvelle reconnaissance partielle. À partir de ce moment, toutes les ressources du système ont déjà été utilisées. Cette reconnaissance partielle annote les noms propres de manière probabiliste.
- Etape 5 : traitement du titre des articles (*title assignment*) : En anglais, tous les mots des titres portent une majuscule sur leur première lettre, c'est pourquoi ils sont traités par LTG comme un cas particulier. Des règles et un algorithme probabiliste leur sont appliqués pour trouver les entités qu'ils contiennent.

Le système *LTG* dépasse les résultats des autres concurrents grâce à ces scores très importants dans la catégorie des noms d'organisations : il obtient une précision de 91% et un rappel de 95% pour cette catégorie. Le meilleur système sur la catégorie des organisations après *LTG* n'obtient que 87% de précision et 89% de rappel. *LTG* traite les variantes de noms d'organisations à l'étape 2 : si le nom est composé de plusieurs mots, le système essaie de trouver un ordre partiel par rapport aux noms d'organisations déjà trouvés (Ex : *Grupo Televisa = grupo televisa SA*). Pour ce qui est des autres entités, Mikheev remarque que les noms de personnes sont les plus simples à trouver. Par contre, les lieux sont difficiles à trouver car de nombreux contextes suggèrent des lieux mais n'en sont pas (ex : *un satellite à 13 km de Columbia (la navette)*).

### 3 L'usage des connaissances linguistiques dans les systèmes d'extraction de noms propres

Les extracteurs de noms propres à base de règles utilisent des indices linguistiques variés (structure interne, contextes de noms propres, dictionnaires). Voici les principales idées ressortant de l'ensemble des travaux évoqués en §2.

#### 3.1 L'apport des ressources et leur usage

##### 3.1.1 Les listes de mots déclencheurs (preuves externes ou internes)

Les systèmes d'extraction d'entités nommées utilisent des listes contenant des mots, souvent appelés **mots déclencheurs**, appartenant aux preuves externes ou internes des noms propres (ex : *le désert du Sahara, British Airways*).

[Wakao *et al.*, 1996], [Hayes, 1994], [Coates-Stephens, 1993], [Poibeau, 1999], [Paik *et al.*, 1996] et d'autres font usage de telles listes.

Le système *FUNES* [Coates-Stephens, 1993], par exemple, utilise une base de connaissance et des lexiques (2 000 racines de mots communs et 500 verbes avec leur flexion). [Trouilleux, 1997] se sert de listes de mots déclencheurs pour rechercher non seulement les contextes gauches des noms propres mais aussi pour déterminer leur frontière droite. [Stevenson, Gaisauskas, 2000] utilisent seulement des listes de preuves internes générées automatiquement à partir d'un texte annoté.

Le système *Nominator* [Ravin, Wacholder, 1997] contient lui aussi quelques ressources mais il suit la tendance de [Cowie, Lehnert, 1996] qui est d'en minimiser l'utilisation pour augmenter la robustesse et la rapidité d'exécution des systèmes. *Nominator* dispose seulement d'un petit dictionnaire sémantique mais n'utilise pas d'informations syntaxiques.

### 3.1.2 Les dictionnaires de noms propres

Nous discutons, ici, de l'usage de dictionnaires de noms propres car leur intérêt semble contestable.

La plupart des systèmes d'extraction de noms propres dépendent de dictionnaires de noms propres connus. Les applications commerciales comme, par exemple, *NameFinder* [Hayes, 1994] font un grand usage de ce genre de données car de tels dictionnaires permettent d'obtenir des résultats très rapidement avec un minimum d'investissement. *NetOwl Extractor* utilise de très gros dictionnaires ce qui lui a permis d'obtenir de très bons résultats à MUC 6, cependant, il a subi les conséquences de cet usage trop intensif lors de MUC 7 dont la tâche NE était plus compliquée. [Bikel *et al.*, 1997] proposent un système d'apprentissage qui utilise aussi de très grands dictionnaires de noms propres. L'approche par les dictionnaires est contestée par Mikheev.

En effet, [Mikheev *et al.*, 1999] ont étudié l'usage et l'impact de très gros dictionnaires dans l'extraction des entités nommées. Il tente de répondre à deux questions :

- Est-ce que les dictionnaires/listes sont importants pour trouver les noms propres?
- Quelle doit être leur taille pour obtenir des résultats intéressants ?

Pour y répondre, ces chercheurs ont essayé d'extraire les noms propres avec ou sans dictionnaires. Voici comment ils ont procédé :

Des dictionnaires d'entités nommées ont été créés grâce à l'ensemble d'entraînement de MUC 7 ; ensuite, un système très simple qui n'utilise que les dictionnaires pour repérer les noms propres est implémenté. Ce système simpliste obtient un rappel de 49% pour les organisations, 26% pour les personnes, et 76% pour les lieux, avec une précision de 90% environ. Ce sont les noms de personnes qui sont le moins adaptés à un système uniquement basé sur des dictionnaires.

Un deuxième essai a été mené en combinant ces dictionnaires avec un petit ensemble d'entraînement, ce qui a légèrement amélioré les résultats par rapport au premier système.

Ensuite, [Mikheev *et al.*, 1999] ont tenté d'améliorer leur système LTG en y ajoutant des dictionnaires (4 900 noms de lieux, 10 000 prénoms, 30 000 noms de compagnies). L'utilisation de tous ces dictionnaires a donné une amélioration légère des résultats due

aux noms de lieux. Les 200 noms de lieux les plus fréquents suffisent, en fait, à améliorer les résultats. Il semble donc intéressant d'utiliser des dictionnaires limités contenant les noms propres les plus communs. [Wakao *et al.*, 1996] et [Mikheev *et al.*, 1999] s'accordent pour dire que les noms de lieux nécessitent des dictionnaires pour être localisés. De plus, ils sont rarement accompagnés d'une preuve permettant leur repérage (ce qui semble confirmé par notre étude en corpus du Chapitre 1).

Plutôt que de dictionnaires, certains se servent d'une base de connaissance. [Wolinski *et al.*, 1995] ont créé une base de connaissance de noms propres dans leur système *Exoseme*. La représentation est faite par des graphes conceptuels qui autorisent la mémorisation de données hétérogènes (Figure 4).

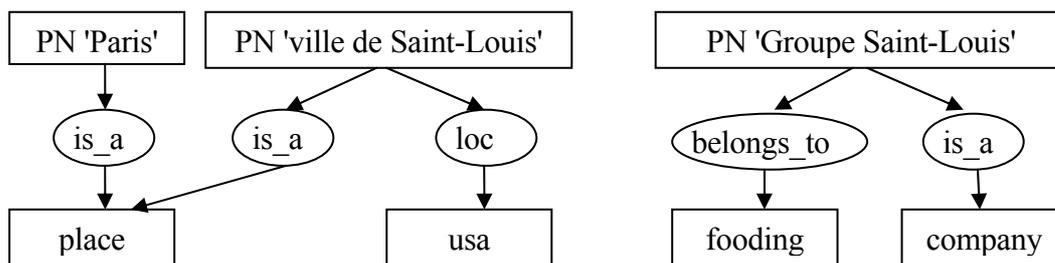


Figure 4 : Représentation de la base de connaissances d'Exoseme

Dans leur système, les mots équivalents (et synonymes ex : *Marseille = la cité phocéenne*) sont mémorisés (y compris les abréviations et les fautes) et enrichissent au fur et à mesure la base de connaissance. Cette base est utilisée pour désambigüiser le type du nom propre en prenant en compte son contexte.

### 3.2 Utilisation des preuves internes et externes

Les preuves internes et/ou externes sont unanimement utilisées par les systèmes d'extraction des entités nommées basés sur des règles.

[Rau, 1991] a créé un extracteur de noms de compagnies en anglais basé sur la preuve interne. Ce système teste la présence de suffixes *Co*, *Inc*, etc. Rau profite du fait que la suffixation est un phénomène très fréquent dans les textes en anglais "américain". Après avoir trouvé un suffixe, elle réalise un retour arrière jusqu'au premier mot en minuscule en prenant en compte les conjonctions de coordinations et en utilisant une liste de mots vides (*stopwords*). Puis Rau génère des variantes (sigles ou formes raccourcies). Elle extrait ainsi 97,5% des noms d'organisation qui ont un suffixe et 40% d'autres noms d'organisation grâce à ceux déjà trouvés.

De la même manière, [Stevenson, Gaisauskas, 2000] présentent un système qui n'utilise que les preuves internes contenues dans des listes de noms propres générées automatiquement à partir de textes annotés en SGML tirés de MUC 6.

La preuve interne ne suffit pas à lever l'ambiguïté du type : certains noms de compagnie peuvent être confondus avec des noms de personnes par exemple. La preuve externe est donc très importante [Wakao *et al.*, 1996].

Le système PNF de [McDonald, 1996] type les entités nommées en deux étapes. Lorsque la preuve interne échoue, il utilise la preuve externe.

- La première étape permet de délimiter le nom propre à l'aide de la preuve interne. Il utilise une grammaire close de mots fonctionnels permettant d'éliminer les mots qui ne sont pas dans le nom propre. Puis la preuve interne est utilisée pour catégoriser le nom propre.
- Lorsque la première étape n'a pas pu assigner une catégorie au nom propre, la preuve externe est utilisée à travers une grammaire du contexte des entités.

### 3.3 L'utilisation de la morphologie

La morphologie est assez peu utilisée pour le repérage des noms propres français ; on remarque le travail sur les noms de lieux de [Belleil, 1997].

[Coates-Stephens, 1993] utilise des heuristiques morphologiques pour l'anglais afin de catégoriser certains noms propres. Par exemple :

*Si un mot se termine par **ese** / **ians** / **ian** alors c'est un nom d'origine,  
Sinon si ce mot se termine par **shire** c'est un nom de lieu.*

Ces règles sont appelées uniquement si la racine du mot est déjà un nom de lieu connu

[Cucerzan, Yarowski, 1999] se servent des préfixes et suffixes de mots pour trouver certaines classes d'entités nommées. Par exemple, les suffixes de noms tels que *escu* en roumain, *ovic* en croate, ou *son* en anglais, indiquent un nom de personne. Se baser sur les suffixes permet d'obtenir des résultats assez élevés sur des langues comme le roumain, le grec ou le turque).

[Taghva, Gilbreth, 1995] utilisent la morphologie pour trouver les noms de compagnies sous forme de sigles dans les textes. Un candidat sigle est composé de 3 à 10 lettres de longueur. Les sigles de deux lettres existent mais causent beaucoup de problème d'ambiguïté avec d'autres mots. Ce genre de règles est à prendre avec précaution : en français, *RTL (Radio Télé Luxembourg)* est bien un acronyme de noms d'organisation, par contre *PDG (Président Directeur Général)* n'en est pas un.

## 4 Le traitement des ambiguïtés

### 4.1 Résolution des ambiguïtés structurelles : la délimitation des noms propres

[Jacquemin, Bush, 2000] travaillent sur les entités nommées afin de créer des ressources pour le traitement de la parole ; à cette occasion, ils définissent les problèmes d'extraction partielle (les erreurs liées aux mauvaises délimitations des entités nommées) comme suit :

- La sur-reconnaissance : la séquence reconnue contient l'entité nommée mais est trop longue.

- La sous-reconnaissance décrit le fait que l'entité reconnue est contenue dans l'entité initiale. Par exemple, dans la phrase *L'ancien président Valéry Giscard d'Estaing a visité Vulcania*, si on ne repère que *Valéry Giscard*, l'entité est sous-reconnue car on aurait dû trouver *Valéry Giscard d'Estaing*.

La sur-reconnaissance et la sous-reconnaissance se manifestent surtout à la droite des noms propres.

On trouve assez simplement le début d'un nom propre (présence d'une majuscule) mais les mots qui suivent n'en portent pas forcément (des conventions typographiques existent<sup>26</sup> mais les auteurs ne les connaissent pas et placent la majuscule où ils le veulent). Par conséquent, la limite droite est difficile à trouver (ex : *La Fédération nationale de la Mutualité française*).

*Exoseme* [Wolinski *et al.*, 1995] segmente les noms propres grâce aux marqueurs grammaticaux (prépositions, conjonctions, virgules, points), mais cette segmentation est insuffisante puisque des noms propres peuvent contenir des conjonctions ou des prépositions.

Le système PNF [McDonald, 1996] utilise une heuristique plus complexe. Il groupe toute séquence contiguë de mots capitalisés et s'arrête au premier mot non capitalisé ou à une virgule ; les autres ponctuations sont gérées cas par cas.

*Nominator* [Wacholder *et al.*, 1997] procède de façon similaire et se base sur les motifs de mots en majuscules, les ponctuations et la localisation des mots dans la phrase. *Nominator* gère aussi le problème posé par les conjonctions (ex : *Victoria and Albert Museum, IBM and Bell Laboratories*) et les prépositions. Une étendue est définie selon le type des opérateurs (conjonctions, prépositions) et le type des noms propres. Par exemple, *of* est à l'intérieur de l'étendue du nom *Museum* car *Museum* peut contenir de telles modifications prépositionnelles (ex : *Museum of Natural History*).

[Trouilleux, 1997] décrit une grammaire du contexte droit<sup>27</sup> pour le français. L'extension à droite d'un nom propre peut contenir des adjectifs, noms, prépositions, déterminants, coordinations qui dépendent du type du nom propre, et se termine nécessairement par un nom ou un adjectif. Les possibilités d'extension à droite dépendent du type de nom propre considéré. Voici un résumé de la proposition de Trouilleux :

- Extension après un nom de personne : pas d'extension à droite traitée.
- Extension après un lieu : rattachement autorisé sur les adjectifs, rattachement sur les syntagmes prépositionnels introduits par *de* et un point cardinal (ex : *Pyrénées Orientales, Allemagne de l'Ouest*).
- Extension après nom d'organisation ou d'événement : avec préposition : *pour, sur, dans, à, au, aux, contre, entre* (ex : *le Haut Commissariat aux Réfugiés*) et sans préposition (ex : *Fonds monétaire international*).

<sup>26</sup> Par exemple, Code de rédaction interinstitutionnel de l'Office des publications de la Communauté européenne, <http://eur-op.eu.int/code/fr/fr-4100201.htm>.

<sup>27</sup> Une description plus complexe a été présentée aux journées de l'ATALA [Trouilleux, 1999] mais ce que nous présentons ici suffit à donner un aperçu de ces idées.

- Tous les noms propres peuvent admettre des extensions avec préposition *sans*, *de*, ou une coordination *et* (ex : les *Eaux et Forêts*, *Médecins sans Frontières*).

L'idée de décrire une grammaire des extensions possibles des noms propres selon leurs types est intéressante. Néanmoins, même si un adjectif est autorisé après tel ou tel type de noms propres, cela pose problème : *l'Europe centrale* désigne bien une entité, mais dans *l'Europe riche* seule *Europe* est un nom propre, *riche* n'en fait pas partie. [Trouilleux, 1997] ne précise pas le rappel obtenu par *ThingFinder* mais précise que, sur l'ensemble des chaînes typées par son système, 55% le sont correctement.

En anglais, il y a beaucoup moins de problèmes de limites droites ; en effet, les noms propres portent sur tous les mots qui les composent une majuscule et se terminent souvent par un mot indiquant leur catégorie (ex : *Central Park*, *National Security Agency*).

Une autre ambiguïté structurelle tient à la majuscule des débuts de phrase. [Wacholder *et al.*, 1997] traitent l'ambiguïté du premier mot d'une phrase de la manière suivante : si un mot commençant une phrase est trouvé, avec une majuscule sur sa première lettre, à un endroit du texte autre que le début d'une phrase, ce mot est un nom propre.

## 4.2 Résolution des ambiguïtés sémantiques

[Wacholder *et al.*, 1997] étudient les ambiguïtés sémantiques des noms propres dans les textes. Ils résolvent cette ambiguïté sémantique en calculant la probabilité qu'un nom propre soit d'un type ou d'un autre. L'ensemble des composants obligatoires ou optionnels pour chaque type est listé : par exemple une personne peut avoir une profession, un titre, un prénom, un nom. Si on rencontre *Justice Departement* dans un texte, *Justice* est un prénom accompagné de *Departement* qui est un nom d'organisation, alors la probabilité d'avoir un nom de personne est plus faible que si le nom propre repéré avait été *Justice Johnson*.

[Wolinski *et al.*, 1995] proposent de désambiguïser le type d'un nom propre par un contexte local et une base de connaissance : si on rencontre *Saint-Louis* et *Etats-Unis* dans le même texte, on parle certainement de la capitale du Missouri. Le contexte global permet, lui aussi, la désambiguïstation d'un nom propre, si une partie de nom propre, déjà trouvé et catégorisé, se retrouve quelque part dans le texte.

## 4.3 Une heuristique de désambiguïstation : *Les mots ont un seul sens par discours*

Les entités nommées sont introduites dans les textes une première fois avec leur forme la plus explicite ou la plus complète. Ensuite on y réfère par des raccourcis et variantes plus informelles.

Cette heuristique est utilisée par presque tous les systèmes d'extraction d'entités nommées, qu'ils le disent explicitement [Wacholder *et al.*, 1997] ou non.

[Gale *et al.*, 1992], à travers un travail sur la désambiguïsation du sens, observent qu'en anglais, si un nom polysémique apparaît deux fois ou plus dans un discours<sup>28</sup>, le plus souvent toutes ces occurrences partagent le même sens. Cette tendance est très forte puisque, selon eux, 98% des noms polysémiques respectent cette loi dans les discours "bien écrits".

On peut donc suspecter des résultats très similaires pour le français.

## 5 Conclusion

Nous avons vu ici que trois grands types d'extracteurs d'entités nommées existent : à base de règles écrites à la main, par apprentissage ou par une approche hybride.

Les plus communs sont les systèmes à base de règles écrites à la main. Ces systèmes obtiennent de très bons résultats mais ils demandent un investissement humain conséquent. Les tendances générales des systèmes à base de règles sont :

- L'utilisation de listes de preuves internes et externes,
- La description de règles avec des expressions régulières.

Les systèmes d'apprentissage, à l'inverse, minimisent le travail de description pour des résultats plus faibles. Les systèmes hybrides sont intéressants et c'est d'ailleurs un de ces systèmes, LTG, qui obtient les meilleurs résultats.

L'expérience française n'est pas aussi grande que celle des systèmes travaillant sur l'anglais [Senellart, 1998] restreint son travail aux noms de personnes, et [Poibeau, 1999] aux noms d'organisations et de dirigeants d'entreprises. Exoseme [Wolinski *et al.*, 1995] est limité aux textes des dépêches AFP. Thingfinder [Trouilleux] prend en compte tous les types y compris les événements, objets, etc. mais ne type correctement que 55% d'entre eux. [Fourour, 2002] obtient de très bons résultats (90% de rappel et 95% de précision) sur *Le Monde*.

Nous présentons notre propre système d'extraction d'entités nommées et les résultats obtenus au Chapitre 5. Nous verrons que notre système obtient les meilleurs résultats pour l'instant sur le français.

---

<sup>28</sup> Les discours testés sont des articles de l'encyclopédie américaine Grolier, du thésaurus Roget et du corpus Brown (Brown Corpus of Standard American English).

## Chapitre 3

# CASSYS, UN SYSTEME DE CASCADE DE TRANSDUCTEURS

---

La description linguistique de règles d'extraction des noms propres nécessite un formalisme adéquat. Nous avons choisi les transducteurs et le système Intex pour les raisons que nous évoquerons dans ce chapitre. Ce travail nous a conduit à implémenter le système **CasSys** qui permet de réaliser des cascades de transducteurs en utilisant les outils fournis par Intex.

Nous passerons en revue, ici, les systèmes utilisant les cascades de transducteurs et leurs utilisations diverses dans le traitement automatique des langues (cf. §1), puis nous décrirons notre propre système en §2.

### 1 Les cascades de transducteurs

Les automates à nombre fini d'états, et particulièrement, les transducteurs sont très utilisés dans le traitement automatique des langues [Roche, Schabes, 1995].

#### 1.1 Définition d'un transducteur

Un automate à nombre fini d'états permet de représenter des séquences de symboles. Un automate est composé de nœuds et de transitions qui portent les symboles à reconnaître. Une séquence est *reconnue* par un automate si elle appartient au langage représenté par cet automate.

Un transducteur est un automate dont les transitions sont étiquetées par un couple de symbole : un symbole reconnu en entrée et un symbole produit en sortie. Un transducteur permet donc de reconnaître une chaîne de en entrée et produit, en sortie, une autre suite de caractères.

Les automates et les transducteurs peuvent subir des opérations de factorisation, déterminisation et minimisation, ce qui les rend plus efficaces et plus compacts.

Le formalisme des transducteurs à nombre fini d'états excelle dans la description des phénomènes complexes de la langue.

#### 1.2 Définition d'une cascade de transducteurs

Une cascade de transducteurs est une succession de transducteurs appliqués<sup>29</sup> sur un texte, dans un ordre précis, pour le transformer ou en extraire des motifs. [Abney, 1996]

---

<sup>29</sup> Voici ce que nous entendons lorsque nous disons qu'un transducteur est appliqué sur un texte. Le transducteur est comparé au texte de la manière suivante :

définit une cascade de transducteurs comme une séquence de couches ("*sequence of strata*") qui décrivent des grammaires locales.

De manière plus formalisée, une cascade est la répétition des actions suivantes sur un texte  $T$  :

- On applique un transducteur (ou grammaire locale)  $G_i$  sur le texte  $T_i$  qui est transformé en un texte  $T_{i+1}$ ,
- puis un transducteur  $G_{i+1}$  transformera le texte  $T_{i+1}$  en  $T_{i+2}$ ,
- et ainsi de suite.

Chaque transducteur utilise les résultats des transducteurs précédents.

L'ordre de passage des transducteurs dépend de leur degré de "certitude". En effet, on commence par appliquer les transducteurs qui permettent de découvrir des îlots de certitudes ("*islands of certainty*") : ce sont les motifs les plus évidents, les moins ambigus. La reconnaissance de ces clauses simples permet de réduire l'espace de recherche dans la suite de la cascade.

Un transducteur seul ne permet pas de décrire des phénomènes linguistiques complets mais chaque transducteur participe à cette description. On reconnaît uniquement les phénomènes certains et, au fil des transducteurs passés, la couverture des phénomènes de la langue augmente ce qui offre une haute précision au système.

Un certain nombre de travaux en traitement automatique des langues ont développé des cascades de transducteurs profitant de leurs avantages en termes de robustesse, précision et rapidité pour des applications telles que l'extraction d'information, l'analyse syntaxique, etc.

### 1.3 Systèmes de cascades existants

De nombreux systèmes de cascades ont été développés, prouvant le grand intérêt qu'ils suscitent en traitement automatique des langues. Nous présentons, dans cette section, des systèmes de cascade utilisés dans des tâches d'analyse syntaxique, d'extraction d'information et de traduction.

#### 1.3.1 Pour l'analyse syntaxique

L'analyse syntaxique peut être effectuée par la description des phénomènes syntaxiques dans des transducteurs. Dans cette optique, [Abney, 1996] a créé le système Cass (Cascaded Analysis of Syntactic Structure) qui réalise l'analyse syntaxique de textes en anglais et en allemand. Son système reconnaît d'abord comme groupes syntaxiques de base tout ce qui est délimité par des mots vides (*et, avec, en, dans, pour, etc.*) : ces groupes sont appelés des chunks [Abney, 1991]. Ensuite les motifs syntaxiques sont décrits par des expressions régulières traduites en un automate à états finis. Plusieurs couches d'automates sont appliquées sur le texte pour obtenir l'analyse

- 
- Si la séquence décrite dans le transducteur ne correspond à la suite de mots du texte, on passe au mot suivant du texte. Et on recommence à appliquer le transducteur sur le texte à partir de ce mot.
  - Si la séquence décrite dans le transducteur correspond aux mots du texte, on recommence à passer le transducteur à partir du mot suivant le dernier mot reconnu.

syntactique finale. Reprenant ces idées, un analyseur syntactique pour le suédois a aussi été créé par [Kokkinakis *et al.*, 1999].

Le Système *IFSP* (Incremental Finite State Parser), développé au *Xerox Research Center* [Aït-Mokthar, Chanod, 1997], permet, lui aussi, de réaliser des analyses syntactiques. L'analyse syntactique d'un texte par *IFSP* peut emprunter les deux approches suivantes :

- L'approche constructiviste est basée sur des contraintes ajoutées, pendant l'analyse, aux textes [Abney, 1991], [Appelt *et al.*, 1993], [Greffenstette, 1996]. On ajoute, aux segments de phrases découverts par les transducteurs, des informations (syntactiques, etc.)
- L'approche réductionniste commence avec un ensemble d'analyses alternatives connues. Au fur et à mesure, des contraintes permettent d'éliminer celles qui ne sont pas satisfaisantes [Chanod, Tapainen, 1996].

*IFSP* permet de passer une séquence de transducteurs sur un texte en utilisant des opérateurs de remplacement. Chaque transducteur permet d'ajouter des informations syntactiques. En entrée, le système utilise un texte étiqueté (partie du discours, genre, nombre). L'analyseur donne comme résultat une séquence de la forme ci-dessous :

[VC [VC Lorsqu'[NP on NP] tourne VC] [NP le commutateur NP]

[Gala Pavia, 1999] exploite le système *IFSP* dans le but de réaliser une analyse syntactique de l'espagnol.

### 1.3.2 Pour la traduction

Les systèmes à mémoire de traduction sont des traducteurs basés sur l'exemple. [Vogel, Ney, 2001] proposent un système à mémoire de traduction pour l'allemand et l'anglais. La mémoire de traduction est entraînée par un corpus bilingue. Ainsi les morceaux de phrases d'une langue ont une traduction dans le corpus de l'autre langue. Les transducteurs portent des groupes de mots et leur traduction en sortie. Une phrase en langue source se trouve sur les transitions du transducteur ; la phrase en langue cible est émise par l'état final. Plusieurs traductions sont autorisées pour une même phrase avec un certain score.

### 1.3.3 Pour l'extraction d'information

Le système *FASTUS* (*Finite State Automata-based Text Understanding*) est développé depuis 1992 et sponsorisé par la DARPA<sup>30</sup>.

*FASTUS* [Appelt *et al.*, 1993] est un système d'extraction d'information depuis des textes anglais et japonais. Ce système utilise une cascade de transducteurs pour analyser les textes en des phrases de plus en plus larges dans le but d'extraire, pour une question donnée, des informations pertinentes. Il commence par repérer les mots complexes : noms composés, dates et noms propres. Ensuite, *FASTUS* reconnaît les groupes nominaux et verbaux simples, puis les groupes nominaux et verbaux plus complexes.

---

<sup>30</sup> DARPA = Defense Advanced Research Projects Agency

Les motifs repérés sont utilisés pour découvrir tous les événements et relations qui les lient (à travers leurs coréférences par exemple) et des formulaires qui répondent aux questions *qui fait quoi ? à qui ? quand ? et où ?* sont remplis. Ce système a été présenté aux évaluations MUC 6 de la tâche d'extraction d'information mais aussi à la tâche NE où il obtient des scores de 92% de rappel et 96% de précision.

[Ciravegna, Lavelli, 1999] implémentent, dans *FACILE*, une cascade de transducteurs pour faire une extraction d'information en trois cascades consécutives : la première cascade contient des règles empiriques, la seconde partie décrit des cas réguliers de la grammaire et la troisième cascade applique des règles par défaut, utilisées uniquement si aucune autre règle n'a marché.

## 2 Le système CasSys

CasSys utilise les outils fournis par le système Intex. Nous commençons par le présenter.

### 2.1 Présentation du système Intex

Le système Intex, créé au sein du LADL<sup>31</sup> [Silberztein, 1993], est un environnement de développement qui permet de construire des descriptions formalisées de grammaires et d'utiliser des ressources telles que des dictionnaires de la langue à large couverture. Tous les objets traités par Intex sont ou peuvent être transformés en des transducteurs à nombre fini d'états. Toutes les opérations sur les textes, grammaires et dictionnaires se ramènent ainsi à des opérations sur des transducteurs.

- La Figure 5 représente une **phrase** étiquetée par les dictionnaires fournis avec Intex. Le premier mot "*parallèlement*" est étiqueté adverbe (*ADV+zI*), le second mot "*le*" est un pronom (*PRO*) ou un déterminant (*DET*), "*vert*" est un adjectif (*A*) ou un nom (*N*), etc.
- La Figure 6 décrit la **grammaire locale** des négations devant un verbe. Dans la phrase "*Tu ne m'en veux pas*", cette grammaire reconnaît la séquence "*ne m'en veux*" (suivre le chemin indiquer par les flèches sur la figure)
- La Figure 7 est un graphe décrivant le **dictionnaire** des nombres de 2 à 99 ; ce graphe est fourni avec les autres dictionnaires d'Intex.

---

<sup>31</sup> Laboratoire d'Automatique Documentaire et Linguistique, <http://ladl.univ-mlv.fr/French/>

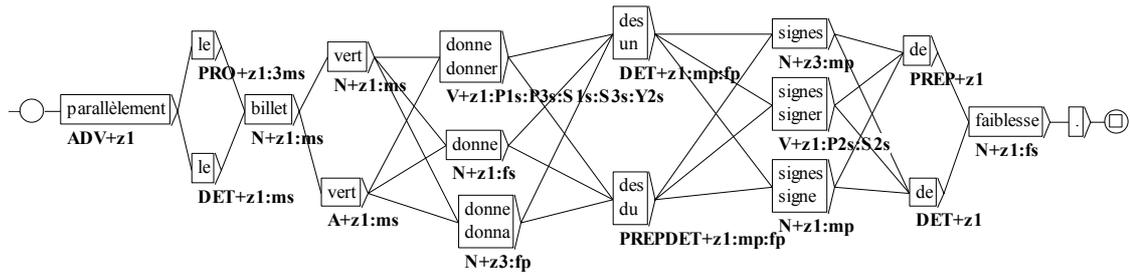


Figure 5 : Représentation Intex de la phrase "Parallèlement le billet vert donne des signes de faiblesses."

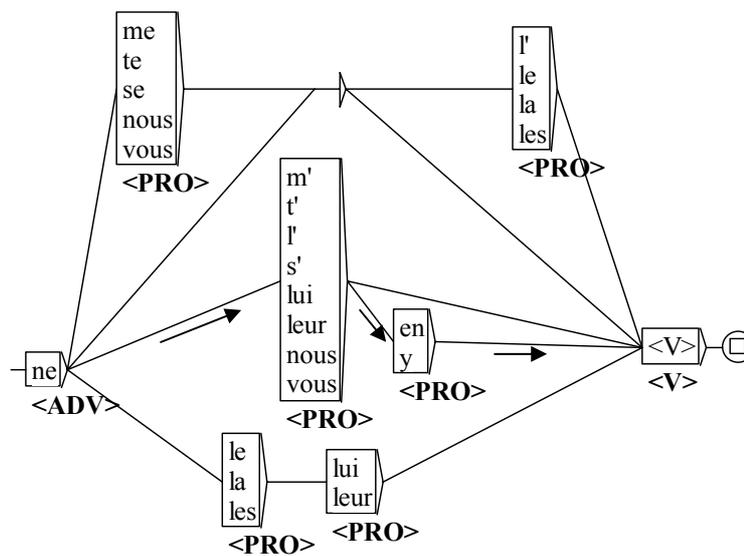


Figure 6 : Représentation d'une grammaire de la négation pré-verbale

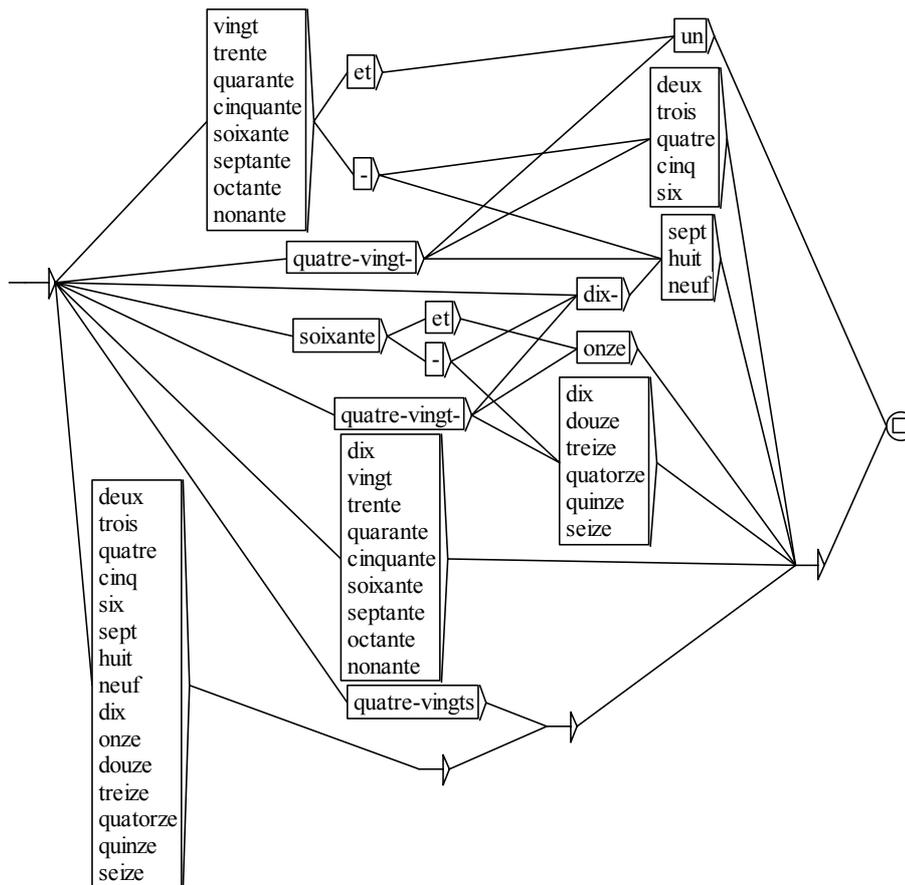


Figure 7 : Dictionnaire des nombres de 2 à 99

Nous voyons, à travers ces exemples, que les transducteurs d'Intex permettent de représenter les grammaires et dictionnaires sous une forme compacte : ainsi les nombres de 2 à 99 nécessiterait un dictionnaire de 98 entrées, résumées ici par une description simple et lisible.

### Le formalisme des transducteurs sous Intex

Sous Intex, les transducteurs sont représentés par des **graphes**. Un graphe est un ensemble de nœuds connectés qui possède un nœud initial et un nœud terminal. Nous utiliserons désormais l'appellation **graphe** pour parler des transducteurs sous Intex.

- Les nœuds des graphes sont des boîtes qui contiennent les étiquettes en entrée du transducteur,
- Une étiquette sous un nœud est une sortie produite (ex : dans la Figure 8, la boîte contenant *une* a une sortie *:fp* pour indiquer qu'un nombre, compris entre 2 et 99, terminé par *une*, est au féminin pluriel),
- Le nœud initial est représenté par une flèche gauche, le nœud terminal est un carré dans un rond,
- Les nœuds qui apparaissent grisés contiennent des sous-graphes. Par exemple, la boîte *Dnum[2,99]* du graphe de la Figure 8 fait appel au graphe décrivant les nombres de 2 à 99 de la Figure 7. Ainsi, la

représentation de la grammaire des nombres de 100 à 999 est plus compacte et le graphe des nombres de 2 à 99 sert à plusieurs fins.

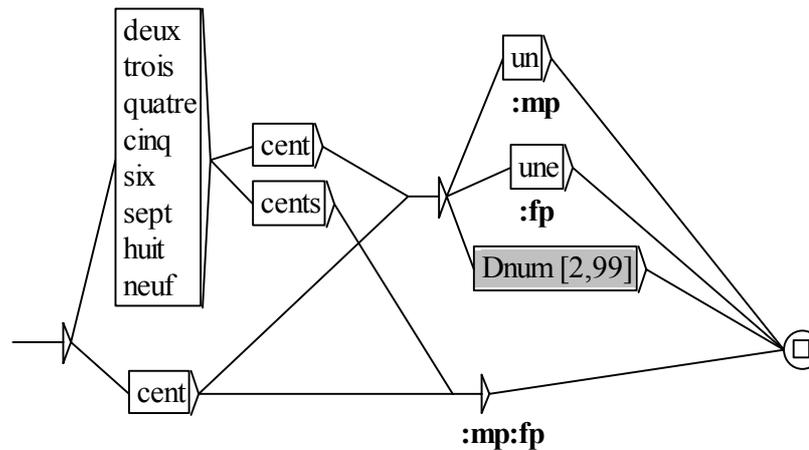


Figure 8 : Graphe Intex représentant la grammaire des nombres de 100 à 999

Les transducteurs permettent d'associer à des séquences reconnues des informations en sortie présentes dans le graphe. Ces informations peuvent :

- être fusionnées à la séquence trouvée dans le texte. La fusion de cette séquence et de sa sortie apparaît dans le texte (mode *merge* d'Intex).
- remplacer la séquence reconnue dans le texte (mode *replace* d'Intex).

Ces deux opérations permettent de transformer un texte pour y ajouter des informations, ou remplacer certains motifs par d'autres.

## 2.2 Généralités sur le système CasSys

CasSys<sup>32</sup> peut être utilisé dans le but de l'extraction de noms propres mais aussi pour d'autres usages d'une cascade de transducteurs (analyse syntaxique par exemple).

Le système CasSys permet d'appliquer à un texte une liste de transducteurs avec un certain nombre d'options. Il est possible de lancer un certain nombre de prétraitements sur le texte (découpage en phrases, passage de dictionnaires) avant de générer la cascade, puis les transducteurs sont passés sur le texte avec des options telles que les modes *merge* ou *replace* d'Intex. L'annexe A (p. 119) contient des détails techniques sur l'implémentation de CasSys, et notamment son utilisation des programmes d'Intex et sa configuration.

Lorsque nous appliquons un graphe sur un texte, nous avons choisi d'utiliser l'heuristique de la séquence la plus longue (longest pattern matching). Qu'entendons nous par là ? L'analyse d'un texte par un graphe (transducteur) se passe de la manière suivante. Si le graphe est appliqué à un texte :

<sup>32</sup> CasSys sera mis à la disposition des utilisateurs d'Intex pour leur permettre de générer des cascades de transducteurs.

- soit un seul chemin de ce graphe reconnaît une séquence d'un texte,
- soit plusieurs sont corrects.

Si le transducteur arrive à son état final par plusieurs chemins alors c'est le chemin qui reconnaît la séquence la plus longue qui est choisi, car plus la séquence à reconnaître est longue, moins elle est ambiguë. Si le transducteur n'arrive pas sur l'état final alors la reconnaissance recommence sur le mot suivant du texte.

Cette heuristique est intéressante sauf si plusieurs chemins de même longueur reconnaissent la même séquence malgré une sortie différente. Alors, un des chemins sera choisi sans qu'on puisse avoir un contrôle sur ce choix : le plus mauvais chemin peut donc être choisi ! La solution passe par la cascade de transducteur : un graphe qui comporterait deux chemins ambigus sera découpé en deux graphes. Le premier contiendra celui qui est le plus sûr, le second celui qui est le moins sûr. Le programme Intex permet de choisir cette heuristique largement utilisée par les systèmes de cascade de transducteurs.

Aux possibilités offertes par Intex, nous avons ajouté des spécificités au système CasSys (décrites dans la suite).

### 2.2.1 Une option de CasSys : mémorisation de séquence dans un index

Les modes *replace* ou *merge* permettent de remplacer une séquence reconnue ou de la fusionner avec les sorties du graphe. Le fonctionnement interne d'Intex est le suivant : Intex mémorise dans un fichier toutes ces séquences transformées et leur emplacement puis transforme le texte en remplaçant les séquences transformées au bon endroit dans le texte.

CasSys propose d'éliminer du texte la séquence reconnue par un graphe<sup>33</sup> et de la placer dans un fichier index. Le fichier, dans lequel Intex a mémorisé toutes les séquences reconnues, est utilisé pour créer un fichier *index* et les séquences sont remplacées dans le texte par une étiquette. L'intérêt d'une telle manipulation est le suivant : si on supprime au fur et à mesure les séquences déjà trouvées, elles ne pourront plus être ambiguës avec une séquence décrite dans un autre graphe. Le format de l'étiquette indique quel est le graphe qui a reconnu cette séquence et permet de retrouver la séquence correspondante dans le fichier index. La syntaxe de l'étiquette est la suivante :

```
<$nom_du_graphe:position_dans_le_fichier_index$>
```

Voici un exemple : <\$person14:2653\$> est l'étiquette d'une séquence reconnue par le graphe *person14* et cette séquence est conservée à la position 2653 dans l'index<sup>34</sup>.

Pour qu'une séquence soit extraite par cette option de CasSys, il faut qu'elle réponde à une condition dont nous verrons tout l'intérêt à travers des exemples. Il faut que le graphe produise un balisage de type HTML autour de la séquence reconnue :

- <nom\_de\_la\_balise> pour la balise ouvrante, et

<sup>33</sup> Ce système n'est utilisable que par le mode *merge* d'Intex.

<sup>34</sup> Le fichier d'index mémorisant les séquences reconnues par une cascade de transducteurs porte le nom du texte sur lequel est appliquée la cascade avec une l'extension *idx*.

- `</nom_de_la_balise>` pour la balise fermante.

Le nom des balises est libre et on peut créer autant de niveaux d'emboîtement que l'on souhaite.

Les balises doivent suivre un "*bon parenthésage*<sup>35</sup>" et être placées dans les sorties du graphe.

Une séquence reconnue par un transducteur mais qui ne se trouve pas à l'intérieur d'un balisage ne sera pas extraite du texte. Les parties entre balises seront par contre placées dans le fichier index si on le souhaite.

#### Exemple 1:

Le graphe *exemple1* de la Figure 9 reconnaît exclusivement la phrase *sous la direction de von Karajan*. On applique ce transducteur sur le texte suivant :

*Le concert a eu lieu, sous la direction de von Karajan, en Bavière.*

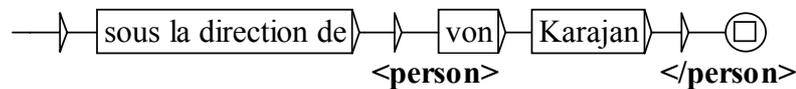


Figure 9: Transducteur exemple1

Le texte devient :

*Le concert a eu lieu, sous la direction de <\$exemple1:0\$>, en Bavière*

La séquence `<person> von Karajan </person>` est extraite du texte et placée dans un index à la position 0. Seule la partie de séquence reconnue entre les balises `<person>` et `</person>` est extraite du texte. De plus, l'étiquette précise que la séquence a été reconnue par le graphe *exemple1*.

#### Exemple 2 :

Voici maintenant un exemple plus complexe. Le graphe *exemple2* de la Figure 10 reconnaît une coordination de mots commençant par une majuscule désignés par l'étiquette `<PRE>` d'Intex.

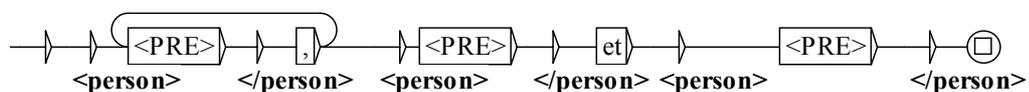


Figure 10 : Transducteur exemple2

<sup>35</sup> Balisages corrects :

```
<ba11> <ba12> </ba12> </ba11>
<ba11> <ba12> </ba12> <ba13> </ba13> </ba11>
<ba11> <ba12> <ba13> </ba13> </ba12> </ba11>
```

Balisages incorrects :

```
<ba11> </ba12>
<ba11> <ba12> </ba11>
<ba11> </ba12> <ba12> </ba11>
```

On applique ce transducteur sur le texte suivant :

*Le mois prochain, Donald, Mickey, Minnie et Daisy sont à Disneyland Paris.*

Ce texte devient :

*Le mois prochain, <\$person:0>, <\$person:24\$>, <\$person:48\$> et <\$person:72\$> sont à Disneyland Paris*

Et le fichier index contient les quatre séquences extraites :

*<person>Donald</person>*

*<person>Mickey</person>*

*<person>Minnie</person>*

*<person>Daisy</person>*

Un même graphe peut extraire plusieurs entités différentes.

Il est aussi possible d'extraire dans un même graphe des séquences n'ayant pas le même balisage

## 2.3 Exemples de cascade de transducteurs

Nous voulons donner ici un aperçu des possibilités offertes par une cascade de transducteurs avec les outils Intex et le système CasSys.

### 2.3.1 Normalisation des sigles

Nous avons créé une cascade pour normaliser les textes contenant des sigles écrits en suivant l'ancienne convention (présence de points entre chaque lettre du sigle). Les sigles reconnus vont être transformés en éliminant les points de ces sigles.

La cascade consiste en deux graphes :

1. *repSigPoint.grf* / en mode fusion
2. *replaceSig.grf* / en mode remplacement

Le graphe *RepSigPoint* (Figure 11) est appliqué en mode fusion et ajoute les balises *<Sigle>* et *</Sigle>* autour d'un sigle. Ce graphe prend en compte le fait que le point final d'un sigle peut être aussi la fin d'une phrase<sup>36</sup>. Dans ce cas, on ne met pas le point final à l'intérieur des balises.

Exemples :

La phrase "*Des O.V.N.I. attaquent.*" devient "*Des <Sigle> O.V.N.I. </Sigle> attaquent.*"

Et, "*On a vu des O.V.N.I.*" devient "*On a vu des <Sigle> O.V.N.I </Sigle>.*"

---

<sup>36</sup> Voir le chapitre 4, section sur le découpage d'un texte en phrases.

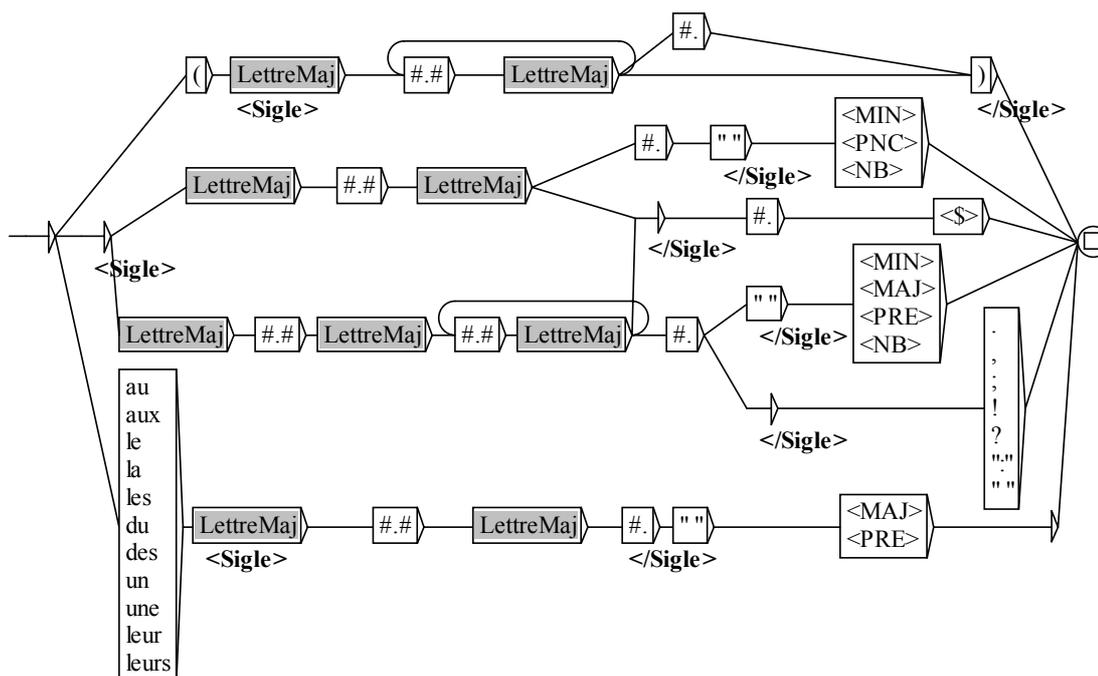


Figure 11: Transducteur repSigPoint

Le graphe *replaceSig* (Figure 12) permet de remplacer un sigle par son équivalent sans point : *replaceSig* retire les points qui se trouvent dans les sigles entre les balises *<Sigle>* et *</Sigle>*. *ReplaceSig*<sup>37</sup> ne possède aucune sortie : les balises *<Sigle>*, *</Sigle>* et les points du sigle sont remplacés par les sorties vides du transducteur : ils sont éliminés du texte. Pour que les lettres du sigle ne soient pas éliminées elles-aussi, le sous-graphe *transLettres* (Figure 13) permet de remplacer chaque lettre par elle-même.

Le texte devient :

*Les OVNI sont là.*

*On a vu des OVNI.*

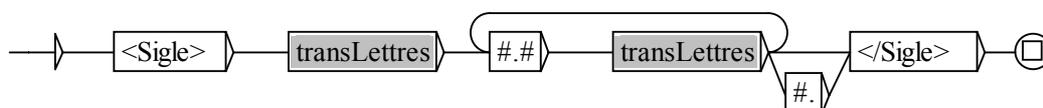


Figure 12 : Transducteur replaceSig

<sup>37</sup> Le symbole dièse (#) est un caractère réservé d'Intex qui interdit la présence d'un espace dans la séquence à repérer.

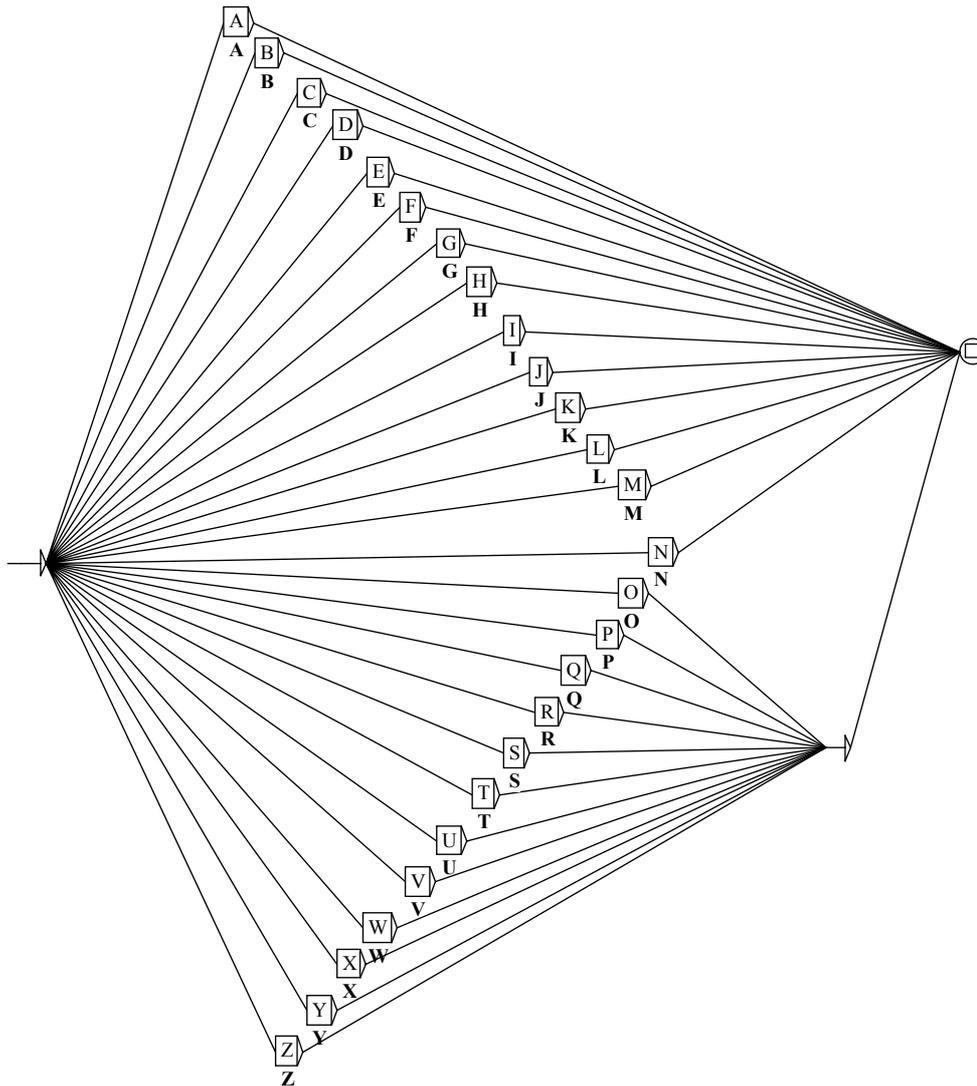


Figure 13 : Transducteur translettres

### 2.3.2 Extraire des entités nommées : un bref exemple.

Voici les transformations successives d'un texte lorsqu'on lui applique une cascade de transducteurs permettant de reconnaître les entités nommées dans un texte et de les extraire avec leur contexte éventuel.

Le texte original  $T_0$  est le suivant :

<b>T E X T E</b>	<i>Le mystère de Pleine-Fougères</i> <i>UN PORTRAIT-ROBOT et un signalement précis de l'homme suspecté d'avoir violé et tué Caroline Dickinson, le 18 juillet 1996, dans l'auberge de jeunesse de Pleine-Fougères (Ille-et-Vilaine), devaient être diffusés vendredi 13 février. L'enquête, reprise de zéro par le juge Renaud Van Ruymbeke, a permis de surmonter l'accumulation de négligences et d'erreurs commises par son prédécesseur. Elle s'oriente vers un violeur en série.</i>
----------------------------------	--

On obtient  $T_{10}$  après passage de transducteurs<sup>38</sup> qui repèrent des contextes de noms de personnes. Le fichier Index contient maintenant un contexte : `<ctxt:titre+metier>juge</ctxt>`. Le texte  $T_{10}$  et le fichier index obtenus sont représentés ci-dessous.

<b>T E X T E</b>	<i>{S} Le mystère de Pleine-Fougères</i> <i>{S} UN PORTRAIT-ROBOT et un signalement précis de l'homme suspecté d'avoir violé et tué Caroline Dickinson, le 18 juillet 1996, dans l'auberge de jeunesse de Pleine-Fougères (Ille-et-Vilaine), devaient être diffusés vendredi 13 février. {S} L'enquête, reprise de zéro par le &lt;Sctxt_tit_metier:0\$&gt; Renaud Van Ruymbeke, a permis de surmonter l'accumulation de négligences et d'erreurs commises par son prédécesseur. {S} Elle s'oriente vers un violeur en série.</i>
<b>I N D E X</b>	<code>&lt;ctxt:titre+metier&gt;juge&lt;/ctxt&gt;</code>

Puis on obtient le texte intermédiaire  $T_{27}$  ci-dessous. Les graphes *personl4* et *personl* ont repéré deux noms de personnes. Le graphe *personl* a repéré un contexte de noms de personne (`<Sctxt_tit_metier:0$>`) suivi d'un prénom et d'un nom.

<sup>38</sup> Le symbole {S} qui apparaît dans le texte est la marque de séparation des phrases sous Intex (voir chapitre 4).

<b>T E X T E</b>	<p><i>{S} Le mystère de Pleine-Fougères</i></p> <p><i>{S} UN PORTRAIT-ROBOT et un signalement précis de l'homme suspecté d'avoir violé et tué &lt;\$person14:117\$&gt;, le 18 juillet 1996, dans l'auberge de jeunesse de Pleine-Fougères (Ille-et-Vilaine), devaient être diffusés .{S} L'enquête, reprise de zéro par le &lt;\$person1:31\$&gt;, a permis de surmonter l'accumulation de négligences et d'erreurs commises par son prédécesseur.{S} Elle s'oriente vers un violeur en série.</i></p>
<b>I N D E X</b>	<pre>&lt;ctxt:titre+metier&gt;juge&lt;/ctxt&gt; &lt;person&gt;&lt;\$ctxt_tit_metier:0\$&gt;&lt;prenom&gt;Renaud&lt;/prenom&gt; &lt;nom&gt;Van Ruymbeke&lt;/nom&gt;&lt;/person&gt; &lt;person&gt;&lt;prenom&gt;Caroline&lt;/prenom&gt;&lt;nom&gt;Dickinson &lt;/nom&gt;&lt;/person&gt;</pre>

À la fin de la cascade, on obtient le texte suivant : une date a été extraite par un graphe nommé *dates*. Trois noms de lieux ont été trouvés par les graphes *loc1* et *loc2*.

<b>T E X T E</b>	<p><i>{S} Le mystère de &lt;\$loc1:355\$&gt;</i></p> <p><i>{S} UN PORTRAIT-ROBOT et un signalement précis de l'homme suspecté d'avoir violé et tué &lt;\$person14:117\$&gt;, le &lt;\$dates:259\$&gt;, dans l'&lt;\$loc2:211\$&gt; (&lt;\$loc1:303\$&gt;), devaient être diffusés .{S} L'enquête, reprise de zéro par le &lt;\$person1:31\$&gt;, a permis de surmonter l'accumulation de négligences et d'erreurs commises par son prédécesseur.{S} Elle s'oriente vers un violeur en série.</i></p>
<b>I N D E X</b>	<pre>&lt;ctxt:titre+metier&gt;juge&lt;/ctxt&gt; &lt;person&gt;&lt;\$ctxt_tit_metier:0\$&gt;&lt;prenom&gt;Renaud&lt;/prenom&gt; &lt;nom&gt;Van Ruymbeke&lt;/nom&gt;&lt;/person&gt; &lt;person&gt;&lt;prenom&gt;Caroline&lt;/prenom&gt;&lt;nom&gt;Dickinson &lt;/nom&gt;&lt;/person&gt; &lt;lieu:social&gt;&lt;ctxt&gt;auberge de jeunesse&lt;/ctxt&gt; de &lt;nom&gt;Pleine-Fougères&lt;/nom&gt; &lt;/lieu&gt; &lt;date&gt;&lt;j&gt;18&lt;/j&gt;&lt;m&gt;juillet&lt;/m&gt;&lt;a&gt;1996&lt;/a&gt;&lt;/date&gt; &lt;lieu:top&gt;&lt;nom&gt;Pleine-Fougères&lt;/nom&gt;&lt;/lieu&gt; &lt;lieu:top&gt;&lt;nom&gt;Ille-et-Vilaine&lt;/nom&gt;&lt;/lieu&gt;</pre>

Si on le souhaite, CasSys conserve tous les textes transformés au fur et à mesure de la cascade. Ceci permet de vérifier les résultats de la cascade : en facilitant la correction d'éventuelles erreurs dans l'ordre de passage des graphes ou en pouvant repérer des erreurs dans les graphes.

Comme les étiquettes placées dans le texte contiennent l'adresse dans le fichier index de l'information qui en a été extraite, on peut replacer cette information dans le texte comme suit.

<b>T E X T E</b>	<p><i>{S} Le mystère de &lt;lieu:top&gt;&lt;nom&gt;<b>Pleine-Fougères</b>&lt;/nom&gt;&lt;/lieu&gt;</i></p> <p><i>{S} UN PORTRAIT-ROBOT et un signalement précis de l'homme suspecté d'avoir violé et tué &lt;person&gt;&lt;prenom&gt; <b>Caroline</b>&lt;/prenom&gt;&lt;nom&gt;<b>Dickinson</b>&lt;/nom&gt;&lt;/person&gt;, le &lt;date&gt;&lt;j&gt;<b>18</b>&lt;/j&gt;&lt;m&gt; <b>juillet</b> &lt;/m&gt;&lt;a&gt;<b>1996</b>&lt;/a&gt; &lt;/date&gt;, dans l'&lt;lieu:social&gt; &lt;ctxt&gt; <b>auberge de jeunesse</b> &lt;/ctxt&gt; de &lt;nom&gt; <b>Pleine-Fougères</b> &lt;/nom&gt; &lt;/lieu&gt; (&lt;lieu:top&gt;&lt;nom&gt; <b>Ille-et-Vilaine</b> &lt;/nom&gt;&lt;/lieu&gt;), devaient être diffusés .{S}</i></p> <p><i>L'enquête, reprise de zéro par le &lt;person&gt;&lt;ctxt:titre+metier&gt; <b>juge</b> &lt;/ctxt&gt;&lt;prenom&gt; <b>Renaud</b> &lt;/prenom&gt;&lt;nom&gt; <b>Van Ruymbeke</b> &lt;/nom&gt;&lt;/person&gt;, a permis de surmonter l'accumulation de négligences et d'erreurs commises par son prédécesseur.{S} Elle s'oriente vers un violeur en série.</i></p>
----------------------------------	---

Pour notre travail sur les entités nommées, nous n'utilisons pas le balisage proposé par la TEI [Burnard, Sperberg-McQueen, 1995]. En effet, ce balisage compliquerait largement l'écriture de nos grammaires. Nous avons opté pour une version simplifiée<sup>39</sup>. Notre balisage est le suivant :

- Une balise dite **primaire** délimite une entité et lui attribue un type (ex : <person>, <lieu>, <org>, <date> etc.)
- À l'intérieur de cette balise primaire, nous aurons des balises **secondaires** de trois types :
  - o Les balises de contexte <ctxt>, elles délimitent un contexte ou preuve externe trouvé par le graphe.
  - o Les balises de nom <nom> délimitent les entités nommées proprement dites.
  - o Les balises de prénom <prenom> permettent de délimiter le prénom d'une personne lorsqu'il est connu d'un dictionnaire.
- On peut ajouter des informations à l'intérieur des balises en respectant la syntaxe suivante <nom\_balise:info1+info2+...>.

Exemple :

```
<person><ctxt:titre+metier>juge</ctxt><prenom>Renaud
</prenom> <nom>Van Ruymbeke</nom></person>
```

<sup>39</sup> Il suffit d'écrire un petit programme pour réaliser une conversion de nos balises en XML par exemple.

### 3 Conclusion

Les systèmes à cascades de transducteurs sont très utilisés pour réaliser des tâches de traitement automatique des langues naturelles. Intex et ses graphes nous ont permis de créer l'outil de cascade de transducteur **CasSys**. Les options offertes par CasSys ajoutent, au système Intex, des fonctionnalités intéressantes pour qui veut créer une cascade.

Dans le chapitre suivant, nous exposons les pré-traitements que nous réalisons sur les textes avant d'extraire les noms propres (section §1 et §2) : découpage en phrases et étiquetage. Puis nous expliquerons l'architecture de notre outil d'extraction de noms propres et son utilisation de CasSys.

## Chapitre 4

# PRE-TRAITEMENTS POUR L'EXTRACTION DES NOMS PROPRES

---

La plupart des systèmes linguistiques d'extraction d'information préparent un texte avant de l'analyser. Cette phase de prétraitements peut comprendre la segmentation en phrases<sup>40</sup> ou en unités plus petites (les *chunks*<sup>41</sup> par exemple), l'étiquetage du texte par une analyse morpho-syntaxique etc. Pour l'extraction des noms propres, nous verrons combien un traitement de qualité de la ponctuation sera bénéfique pour la suite (cf. §1). Nous étiquetterons nos textes grâce aux dictionnaires décrits en §2.

### 1 La segmentation en phrases

L'intérêt principal de la segmentation en phrases dans le processus d'extraction des noms propres est la désambiguïsation du point : dans un texte, les points marquent principalement une fin de phrase, une abréviation<sup>42</sup> et peuvent même cumuler les deux fonctions [Silberztein, 1993], [Dister, 1997]. Parallèlement, la présence d'une majuscule après un point est, elle aussi, ambiguë : marque-t-elle un début de phrase ? un nom propre ? les deux ? ou ni l'un ni l'autre<sup>43</sup> ?

Le découpage en phrases n'est pas une tâche simple mais les graphes du système Intex, présentés dans le chapitre précédent, permettent de décrire ce qui caractérise les débuts ou fins de phrases de manière élégante. En effet, Intex propose un graphe de découpage en phrase nommé *sentence* (Figure 14) ; ce graphe, appliqué en mode fusion (*merge*) à un texte, ajoute l'étiquette {S} au texte. Elle symbolise la séparation entre deux phrases (exemple ci-dessous).

*{S}M. Jacques Delors a confirmé, vendredi 1 janvier, au journal du soir de France 2, qu'il réunira dans deux semaines, à Paris, les dirigeants des partis socialistes et sociaux-démocrates d'Europe.{S} Le président de la Commission européenne avait annoncé cette réunion après la première assemblée générale du club Témoin, créé en octobre dernier et dont il est l'inspirateur.{S} Il souhaite, a-t-il expliqué, " redonner aux socialistes français le goût de s'affirmer socialistes, ce qui ne les empêche pas, ensuite, de conclure des alliances avec d'autres ".{S}*

---

<sup>40</sup> Voir l'article de [Jones, 1994] sur l'importance de la ponctuation dans l'analyse syntaxique.

<sup>41</sup> Voir [Abney, 1991].

<sup>42</sup> Aussi dans les numéros de téléphones.

<sup>43</sup> Parties de textes intégralement en majuscules, titres d'articles dans certains journaux (avec une majuscule sur chacun des mots).

Voici, en résumé, ce que fait ce graphe<sup>44</sup> :

- La règle 1 (en haut du graphe) permet de placer le séparateur de phrase {S} si on rencontre un point, un point d'interrogation, un point d'exclamation ou deux points suivis d'un mot capitalisé ou d'un nombre. Le symbole <^> permet de placer un point sur les phrases en début de texte.
- La règle 2 traite une exception à la règle 1 : si on rencontre *M.*, *MM.*, *Dr.*, *Prof.* suivis d'un mot capitalisé, il ne faut pas placer de fin de phrase (ex : *M. Dupont*, *C. Duras et le Prof. F. Javert*).
- La règle 3 prend en compte une exception à la règle 2 : si on rencontre la phrase *Luc utilise la vitamine C. Duras préfère le pain*, il faut reconnaître la séquence *vitamine C* avant *Duras* pour que la phrase soit correctement découpée *Luc utilise la vitamine C.*{S} *Duras préfère le pain*. Le sous-graphe *MotsComposésAvecMaj* contient une liste de mots composés possédant une lettre majuscule (ex : *vitamine C*, *virage en S*, etc.).
- Finalement, quelques cas particuliers sont pris en compte : *cf.* et *P.S.* : ne sont pas suivis d'une fin de phrase. Le point-virgule est suivi d'une fin de phrase.

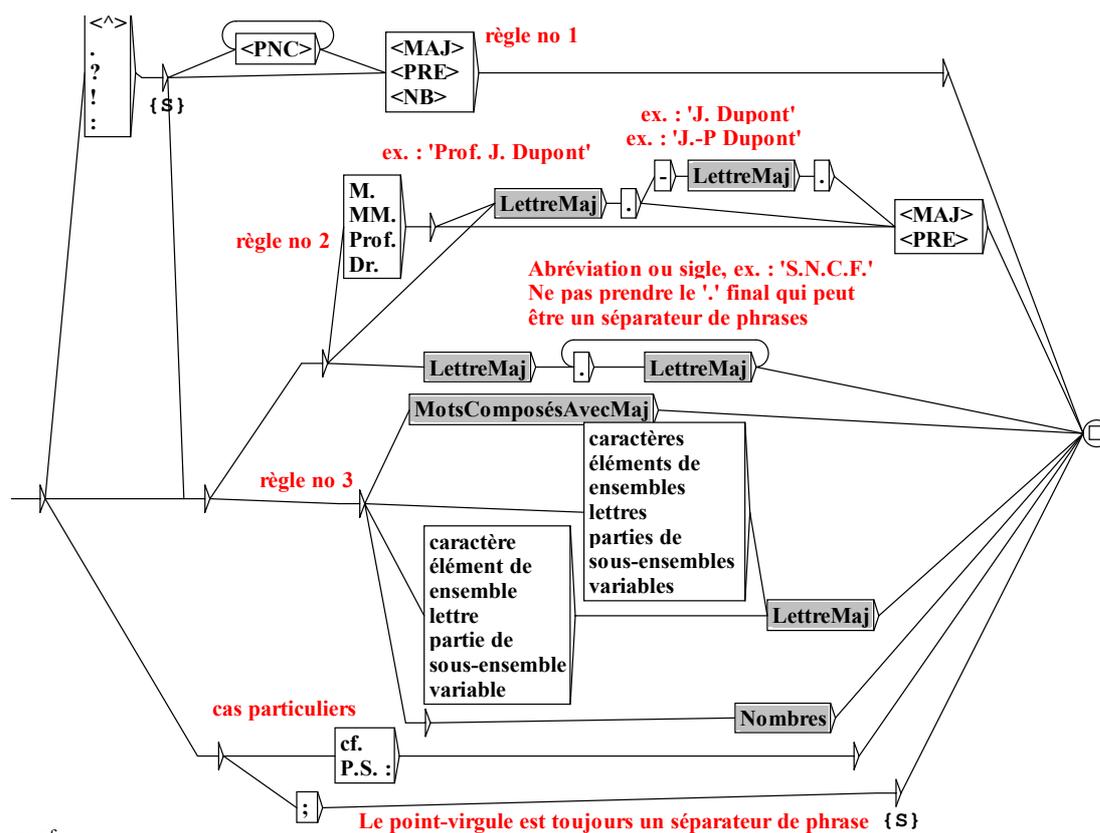


Figure 14: Graphe Sentence

Nous avons constaté que le graphe de découpage en phrases actuel pouvait être amélioré. On note, par exemple, des erreurs dues :

<sup>44</sup> Voir le manuel d'utilisation d'Intex, p. 94.

- à l'application réitérée de la règle 1
 

*... que Broeck refuse de cautionner.{S} " M.{S} Gailhaguet n'a pas autorité pour désigner ou récuser l*

*... à 20 heures.{S} Tél.{S} : 01-40-20-84-00. ...*
- à l'application de la règle 2 puis de la règle 1
 

*... les importations autorisées par l'accord C.E.E.{S}-U.{S}S.A. provoqueront des excédents pendant quatre ans ...*

*... et des activités tertiaires (la M.A.I.F.{S}, E.{S}D.F., les Télécoms, deux hôtels, des banques.{S}).*

*{S}Un nouveau président pour les A.O.C.{S}, M.{S} Pierre Piron, en compagnie de l'ancien président et de responsables du Val de Loire.*

*... à la S.N.C.F. {S} (Société Nationale des Chemins de Fer).*

Lorsqu'on automatise le découpage en phrases, on peut craindre deux types d'erreurs :

- les bruits (trop d'étiquettes {S}, et donc trop de phrases),
- et les silences (des étiquettes manquantes, et donc, des phrases non repérées).

Dans notre travail, la ligne de conduite générale a été d'éviter le bruit. En effet, dans les phases de traitement ultérieures, il est plus gênant, dans l'utilisation d'Intex<sup>45</sup>, d'avoir des découpages en phrases supplémentaires que l'inverse.

Nous utilisons des étiquettes morphologiques : <MAJ>, <MIN>, <PRE>, <NB><sup>46</sup>, etc. La possibilité de formaliser l'absence de blanc entre les points et les lettres (grâce au symbole #) sera très utile pour décrire des motifs comme les sigles.

Les points d'exclamation et d'interrogation, le point-virgule , les deux points et la virgule ne sont pas ambigus. Nous avons amélioré le découpage des phrases en élargissant, entre autres, les motifs présents dans les contextes gauche et droit d'un point.

---

<sup>45</sup> pour la localisation de motifs par exemple.

<sup>46</sup> Dans l'automate *sentence*, nous utilisons des étiquettes morphologiques propres à Intex :

- <PNC> correspond aux séparateurs,
- <MAJ> est un mot en majuscule,
- <PRE> est un mot en minuscule commençant par une majuscule,
- <MIX> est mot comprenant des majuscules et des minuscules mélangées (ex : *McDonald*),
- <MIN> est un mot en minuscule,
- <NB> est un nombre (écrit en chiffres).

Voici notre nouveau graphe *sentence* qui va permettre la segmentation en phrases (Figure 15). On distingue le cas général (en haut du graphe) et trois sous-graphes décrivent les cas particuliers (*cas2*, *cas3*, *cas4*).

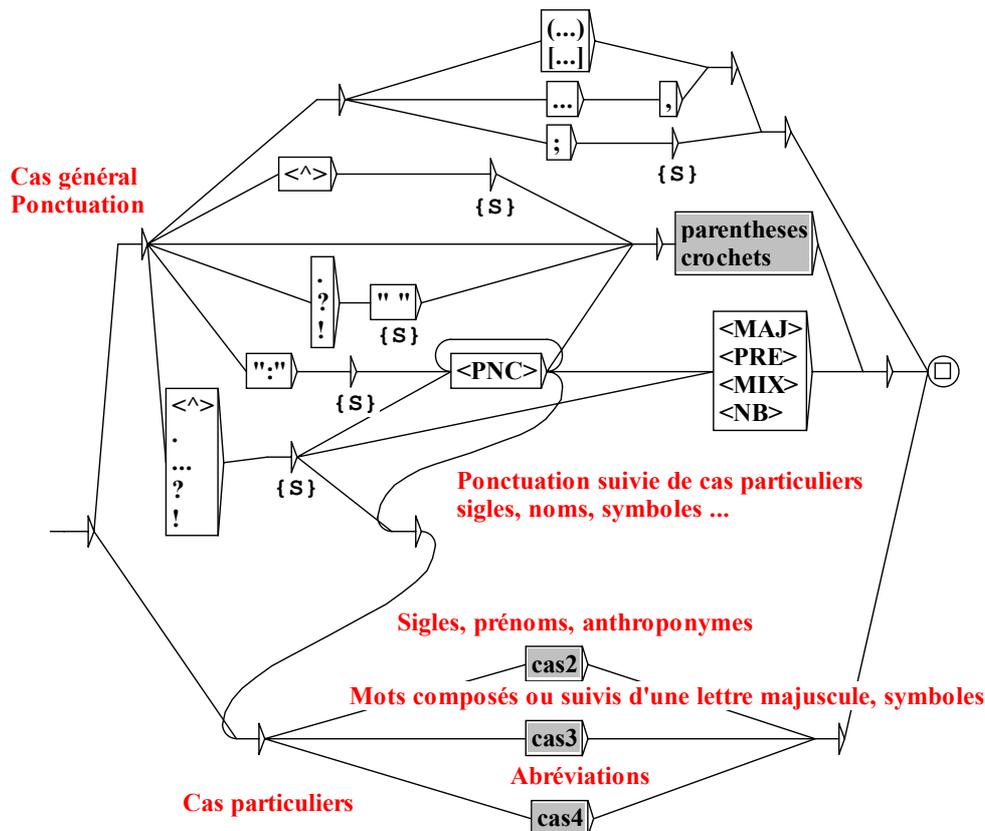


Figure 15 : Notre graphe Sentence

### 1.1 L'ambiguïté du point

Le point est ambigu en présence de majuscules ou de chiffres. Hormis les débuts de phrase, les motifs contenant à la fois des majuscules et des points sont de quatre types :

- Les noms de personnes lorsqu'ils sont précédés de titres ou civilités abrégés (*M.* = *Monsieur*, *MM.* = *Messieurs*, *Prof.* = *Professeur*, etc. dans *M. Dupont*, *MM. Dupont et Durand*), ou lorsqu'ils sont précédés d'un prénom abrégé (ex : *J. Dupont*), ou les deux à la fois (ex : *M. J. Dupont*).
- Les sigles lorsque l'ancienne notation, avec des points, est utilisée (ex : *La S.N.C.F. gère les chemins de fer.*).
- Les mots composés se terminant par une lettre majuscule et les symboles :

*Ce timbre coûte 20 F. Il a été acheté chez un philatéliste.*

*Ces aliments contiennent de la **vitamine A, B et C**. Durant me l'a confirmé.*

- Les abréviations diverses

*éd. Gallimard*

**Chap. 4**

*Cf. France-Italie en juin 2000.*

Ces différents cas sont maintenant traités respectivement dans les sous-graphes :

- *cas2* pour les noms de personnes et les sigles,
- *cas3* pour les symboles et mots composés avec une majuscule,
- *cas4* pour les abréviations.

### 1.1.1 Le point, les sigles et les noms de personnes

Nous traitons ensemble les sigles et les noms de personnes. La raison est qu'un sigle composé de deux lettres et qui se trouve en fin de phrase est ambiguë avec un nom de personne (ex : *Ils ont innocenté O.J. Simpson du meurtre de sa femme*).

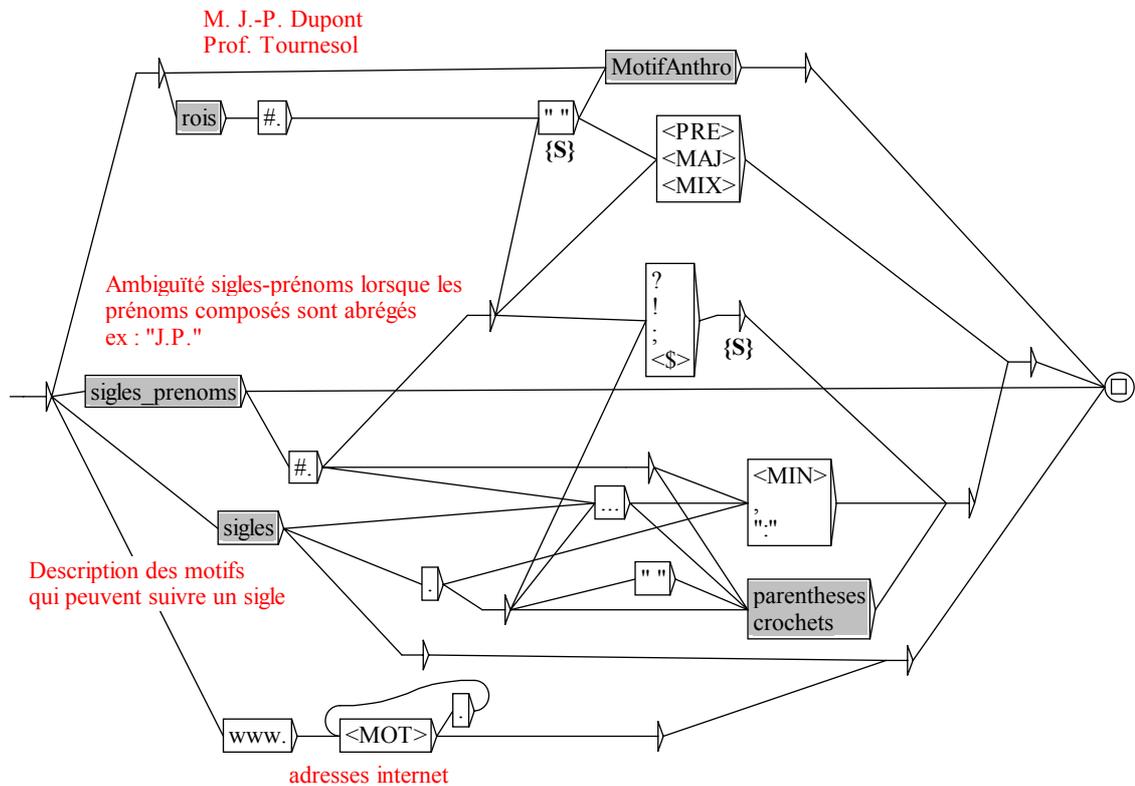


Figure 16 : Graphe cas2

Le graphe *cas2* (Figure 16) présente l'automate qui évite l'insertion d'un séparateur de phrases dans les sigles et les noms de personnes<sup>47</sup>.

Le sous-graphe *MotifAnthro* (Figure 17) décrit les motifs des noms de personnes non ambiguës avec des sigles. Les points qui suivent *M*, *MM* ou *Prof* ne sont pas reconnus comme points finaux s'ils sont suivis d'un prénom abrégé ou d'un mot

<sup>47</sup> <\$> désigne une fin de paragraphe.



L'automate *sigles* décrit aussi les compositions de sigles telles que *C.G.T.-C.F.D.T.*

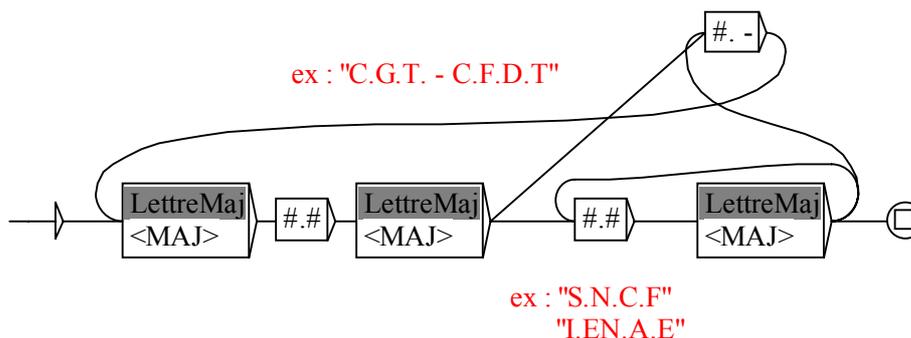


Figure 18 : Graphe sigles

Nous éviterons aussi de placer une fin de phrase entre un sigle et une information entre parenthèses qui le suit comme dans :

*S.N.C.F.* (*Société Nationale des Chemins de Fer*).

### 1.1.2 Le point et les symboles

Les symboles composés d'une seule lettre majuscule : abréviations d'unités de mesure (ex : *V* = volt, *W* = watt, etc.) et symboles monétaires (ex : *F* = Franc, etc.) ne posent pas de problème à l'intérieur d'une phrase, car ils ne sont pas suivis d'un point (au contraire des sigles).

*Un Magritte de 14 000 F a été volé au Centre Pompidou.*

*La centrale hydraulique génère que 5 600 W par minute.*

Mais lorsqu'ils sont en fin de phrase, les symboles sont suivis d'un point qui rend la séquence ambiguë : un symbole suivi d'un point pourrait être analysé comme l'initiale d'un prénom, et le mot en majuscules qui commence la phrase suivante comme un nom de famille. D'après notre graphe de reconnaissance des noms de personnes *MotifAnthro*, aucune étiquette de fin de phrase ne serait insérée entre l'initiale et le nom erronément reconnus comme tels :

*C'est un Magritte de 14 000 F. Volé au Centre Pompidou, il ne sera sans doute jamais retrouvé.*

Le traitement du cas des prénoms abrégés entre en conflit avec celui des symboles en fin de phrases. Pour pallier ce problème, nous avons recensé dans le graphe *cas3* (Figure 19) tous les cas où une lettre majuscule provoque une ambiguïté en fin de phrase.

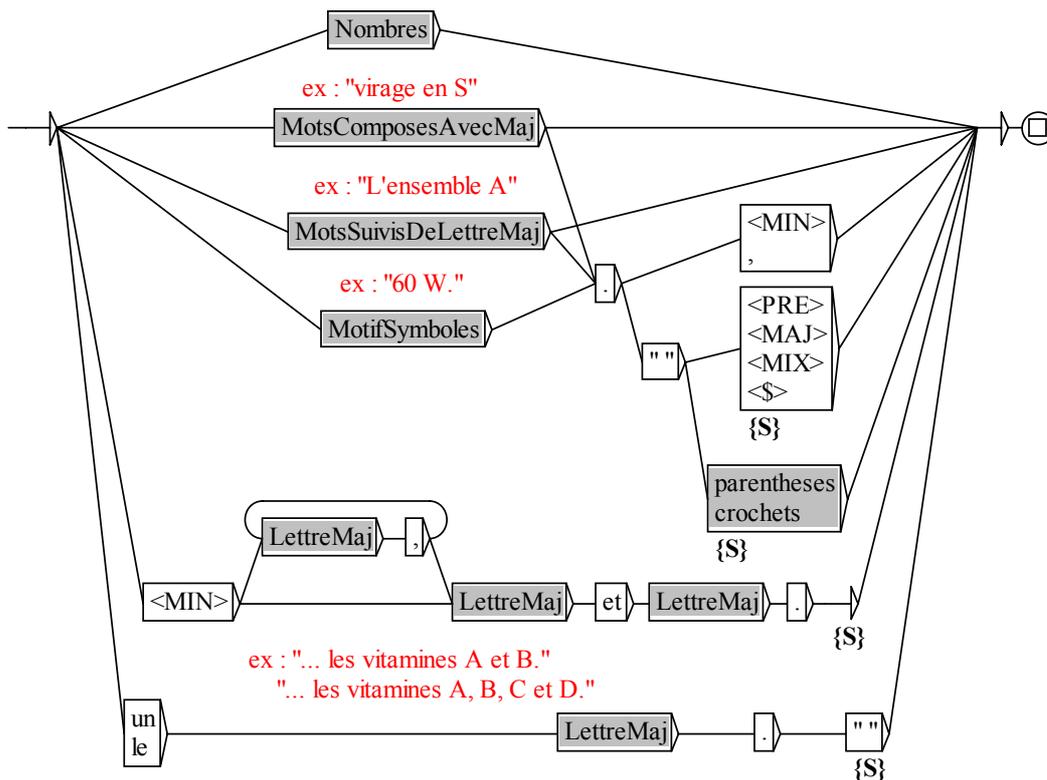


Figure 19 : Graphe cas3

Le sous-graphe *MotifSymboles* (Figure 20) reprend les motifs que l'on peut trouver autour d'un symbole. Nous y décrivons l'emploi du symbole monétaire *F* et des autres symboles. *Symboles1Maj* contient la liste des symboles de volume, quantités et autres ; c'est la présence d'un nombre avant le symbole qui va permettre de désambiguïser ce cas avec le cas d'un nom de personne, et d'insérer correctement le symbole de fin de phrase.

Les phrases suivantes sont correctement découpées.

*C'était un Matisse de 1 350 000 F. {S} Volé au Centre Pompidou, il ne sera sans doute jamais retrouvé.*

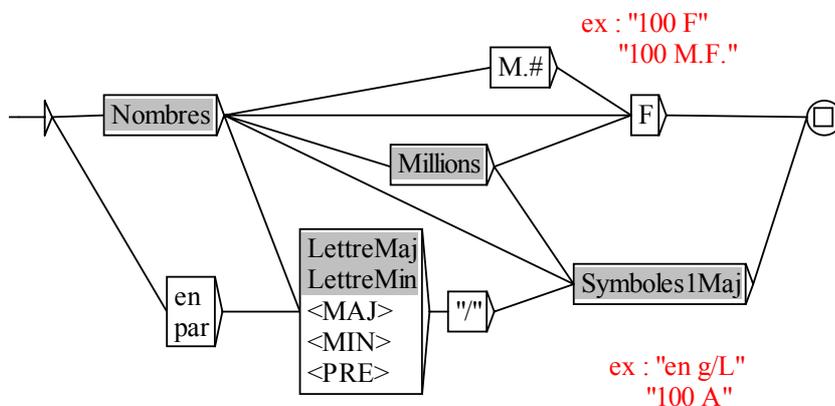


Figure 20 : Graphe MotifSymboles

Si, dans un texte, on rencontre une unité telle que *g/L* (concentration en gramme par litre) ; ce motif composé d'une lettre, de la barre oblique /, d'une lettre majuscule et d'un point est lui aussi ambigu.

*La concentration en sel est de 110 g/L. {S} Les résultats sont étonnants.*

Les séquences ci-dessous sont des motifs également pris en compte.

*vitamines A, B et C.*

*un A.*

*le B.*

Le sous-graphe *MotsComposésAvecMaj* (dans *cas3*) rassemble une liste de mots composés qui contiennent une majuscule (ex : *linéaire B, film X, série Z*, etc.) tandis que le sous-graphe *MotsSuivisDeLettreMaj* contient des listes de mots qui ne sont pas des mots composés mais qui peuvent être suivis d'une majuscule (*variable, ensemble, équipe, pool, bac*, etc.).

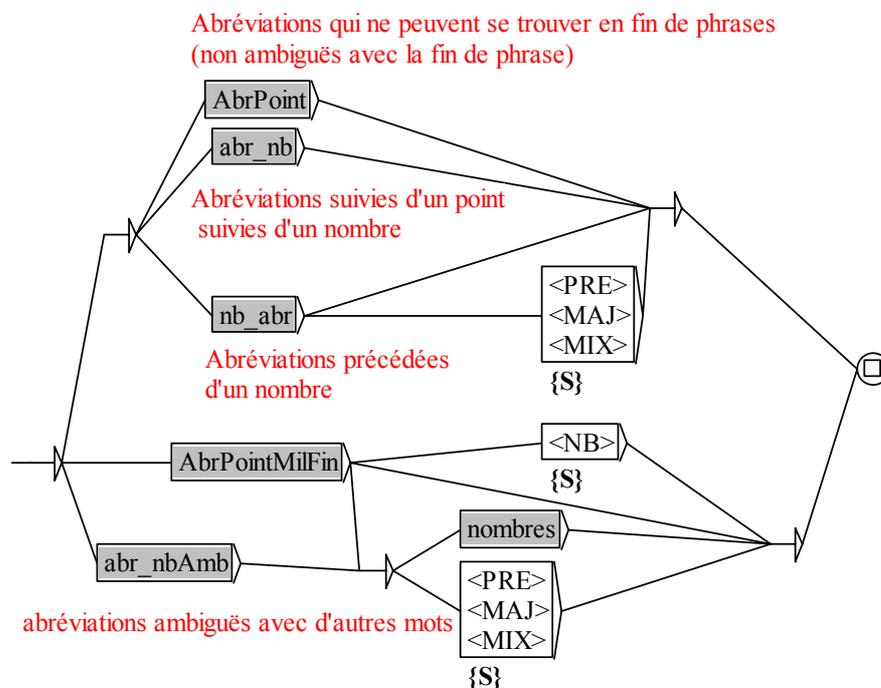


Figure 21 : Graphe Cas4

### 1.1.3 Le point et les abréviations

Le point abrégatif est ambigu puisqu'en fin de phrase il absorbe le point final : la fin de phrase n'est donc pas formellement marquée.

On distinguera deux types d'abréviations :

- celles qui ne peuvent jamais se trouver à la fin d'une phrase et dont le point ne peut être ambigu avec le point final. *cf.*, *i.e.* ou *c.-à.-d.* sont nécessairement suivies d'une précision ou d'une indication et ne peuvent donc être la fin d'une phrase

- celles qui peuvent éventuellement se trouver à la fin d'une phrase (ex : *etc.*, *ap. J.-C.*). Si une telle abréviation est suivie d'un mot en minuscule, d'une virgule, d'un double point ou d'une parenthèse, alors le point n'est pas une fin de phrase. Par contre, si le mot qui suit est un nombre ou s'il commence par une majuscule, on ne peut choisir automatiquement si l'on est en fin de phrase.

Nous avons préféré éviter de mettre une marque de fin de phrase lorsque cela risque de créer trop fréquemment du bruit. Nous avons aussi pris en compte des abréviations qui doivent ou peuvent être suivies d'un nombre afin d'empêcher une fin de phrase avant ce nombre.

Le recensement de toutes les abréviations graphiques (dans le graphe *cas4*, Figure 21) pose problème car on ne peut en faire une description exhaustive. Beaucoup sont par ailleurs non codifiées, et relèvent de la créativité des scripteurs. Nous n'avons recensé que les plus fréquentes. Dans le graphe *cas4*, un certain nombre de sous-graphes liste les abréviations : le sous-graphe *abr\_nb* recense des abréviations finissant par un point et suivies d'un nombre (les numéros de téléphones *Tel. 02 47 36 14 35*, mais aussi *R.N.7*, *B.P. 256* etc.), le sous-graphe *nb\_abr* contient des nombres suivis d'un mot abrégés (ex : *275 p.*, *12 p.m.*), le sous-graphe *abrPoint* liste des abréviations qui ne peuvent être en fin de phrase (ex : *cf.*, *i.e.*, *c.-à.-d.*), et le sous-graphe recense les expressions abrégées qui peuvent éventuellement se trouver à la fin d'une phrase (ex : *etc.*, *av. J.C.*, *et al.*).

## 1.2 La ponctuation : choix de découpage

Le découpage des phrases, lorsqu'il n'est pas ambigu, peut être effectué par le cas général décrit dans la partie supérieure du graphe *Sentence* (Figure 15). Si on découpe les phrases en n'utilisant que cette partie du graphe, on obtient un très mauvais découpage, car il ne traite pas les ambiguïtés.

*...installé par les militaires, M.{S} Joseph Nérette, l'assurance qu'il ...*

*Pour le syndicat de la F.{S}E.{S}N.{S}, c'est le premier acte d'une politique ...*

*chez M. J.{S}P.{S} Denier (Ouest-France)*

Les signes de ponctuation non terminaux sont les suivants : la virgule, les deux points, les guillemets, les parenthèses et crochets. Les signes de ponctuation terminaux sont le point, le point d'interrogation, le point d'exclamation, le point virgule et le saut de paragraphe.

### 1.2.1 Parenthèses et crochets

Nous n'insérons pas de signe de séparation de phrases dans les parenthèses et les crochets, car cela coupe la phrase dans laquelle ces parenthèses ou ces crochets apparaissent (voir Figure 22).

Exemple :

*" La réaction de panique rétrospective qu'a déterminée la crise de 68, révolution symbolique qui a secoué tous les petits porteurs de capital culturel, a créé (avec, en renfort,*

***l'effondrement \_ inespéré ! \_ des régimes de type soviétique)**  
les conditions favorables à la restauration culturelle aux termes  
de laquelle la " pensée Sciences Po " a remplacé la " pensée  
Mao ". (Le Monde)*

*Se conformant aux règles qui régissent les rapports entre  
maisons de disques, Deutsche Grammophon a pris les devants  
en demandant aux différents éditeurs concernés l'autorisation de  
faire porter ses couleurs à des chefs d'orchestre (**Schuricht,  
Klemperer, artistes EMI et Decca ; Walter, artiste CBS,  
Kleiber, artiste Decca**) présents dans ce coffret commémoratif  
d'un intérêt exceptionnel. (Le Monde)*

Dans les exemples précédents, on a empêché (à l'aide du graphe *parentheses*) l'insertion du symbole de fin de phrase {S}, entre des parenthèses, s'il y a un signe de ponctuation terminal dans ces parenthèses.

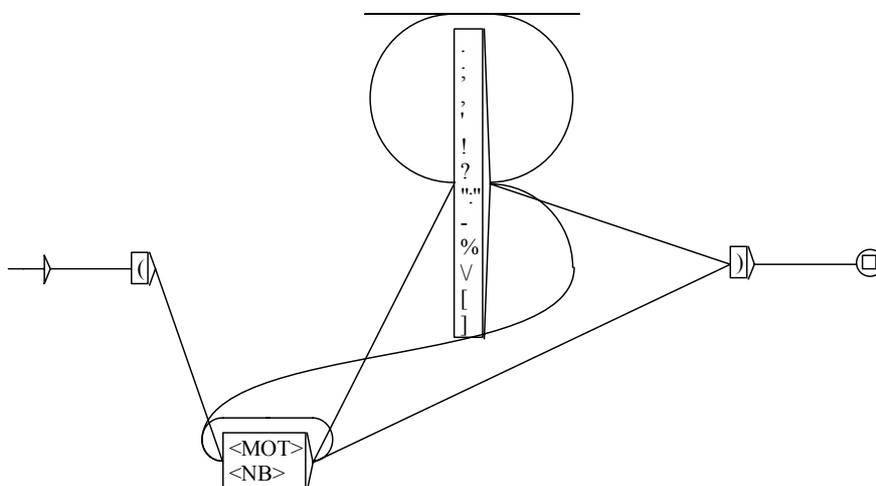


Figure 22 : Graphe Parentheses

Le revers de ce choix est que l'on rencontre parfois, entre parenthèses ou crochets, d'assez longs textes contenant de nombreuses phrases. Les phrases internes à ces parenthèses ou crochets ne sont donc pas découpées.

Exemple :

*Il y a ballottage. {S} [Six candidats postulaient à la succession de Jean Duroisel (RPR), récemment décédé. L'absence d'un candidat du Front national, et la présence d'un candidat unique pour la majorité départementale RPR-UDF, M. Guy Verin, ont rendu les choix relativement clairs à droite. En 1985, Jean Duroisel avait été réélu au second tour avec 2 341 voix (59,44 %) contre 1 597 (40,55 %) à Mme Dumant sur 3 938 suffrages exprimés, 4 093 votants (soit 21,55 % d'abstention) et 5 218 inscrits. Au premier tour, les résultats avaient été les suivants : inscr., 5 219 ; vot., 3 982 ; abst., 23,70 % ; suffr. expr., 3 813 ; Mme Dumant, 1 100 voix (28,84 %) ; Jean Duroisel, 1 070 (28,06 %) ; M. Verrier, div. d., 1 027 (26,93 %) ; Mme Pinchon,*

*FN, 186 (4,87 %) ; MM. Basquin, PC, 152 (3,98 %) ; Rasson, PCI, 150 (3,93 %) ; Kuc, div. d., 128 (3,35 %).] (Le Monde)*

### 1.2.2 Les tirets

Actuellement, notre graphe Sentence ne prend pas en considération les tirets. Ceux-ci peuvent être utilisés comme des parenthèses et contenir des signes de ponctuation terminaux, comme au point précédent. Lorsqu'ils sont utilisés pour marquer une énumération, on placera un signe de fin de phrase à chaque nouveau paragraphe.

### 1.2.3 Les points de suspension

On trouve les points de suspension lorsqu'on ne veut pas allonger une énumération, dans une phrase volontairement inachevée ou lorsqu'on laisse un temps en suspens avant de terminer la phrase.

Si le point de suspension est suivi d'un mot en minuscule ou d'une virgule, ce point de suspension n'est pas la fin d'une phrase.

*Tout a commencé lorsque l'ANPE lui a proposé une place au Normandy... pour arroser les fleurs. (Ouest France)*

*De Mel Gibson, à Roger Hanin, en passant par Alain Delon, Bernard Kouchner, le Professeur Schwartzenberg, Enrico Macias, Thierry Lhermitte..., les personnalités se pressent nombreuses au journal de la petite chaîne qui meurt. (Le Monde)*

Un problème apparaît lorsque le mot qui suit le point de suspension commence par une majuscule. Si tel est le cas, ce mot peut être :

- un nom propre qui appartient à la phrase en cours,

*Un colloque organisé par l'Université d'Angers à la fin de l'année dernière a bien montré l'étonnante richesse et diversité de cette littérature vendéenne souvent signée des plus grands noms : Hugo, Balzac, Barbey d'Aureyvilly, Chateaubriand, Nerval **et même... Jules Verne !** (Ouest France)*

- un mot ou nom propre au début d'une nouvelle phrase.

*Après un premier acte sur des chapeaux de roue, fait surtout de répliques pas légères mais drôles, la pièce s'étire, **déraille ... Régis Santon** n'arrange rien en laissant Marc de Jonge jouer un général de convention pure, un braillard machinal. (Le Monde)*

Ces exemples ne sont pas rares dans les textes, et nous avons pu observer que les points de suspension étaient dans leur très grande majorité<sup>49</sup> à la fin de la phrase. C'est pourquoi, contrairement à notre démarche qui préfère le silence au bruit, nous avons choisi de placer une fin de phrase après le point de suspension s'il est suivi d'un mot commençant par une majuscule.

<sup>49</sup> Une vérification manuelle des points de suspension en corpus fait apparaître que, dans plus de 90 % des cas, ils se trouvent en fin de phrase.

#### 1.2.4 Les guillemets

Sous INTEX, le symbole guillemet est le guillemet anglais " ". Ce symbole est le même, qu'il soit ouvert ou fermé. On ne peut donc pas empêcher que la phrase dans laquelle on a une citation entre guillemets soit découpée en morceaux car on ne peut distinguer le cas où l'on a une citation dans une citation de celui où l'on a deux citations consécutives très proches.

Exemples :

*Cet homme politique a dit : "Je pense à ce que l'on a appelé le "retour de l'individualisme", sorte de prophétie auto-réalisante qui tend à détruire les fondements philosophiques la notion de responsabilité collective." (Le Monde)*

*Ce qui ne l'empêche pas de bénéficier du soutien des États-Unis, qui le présentent comme "un homme politique modéré" et "un ancien communiste". (Le Monde)*

La seule possibilité qui s'offre à nous est donc de placer les fins de phrases sans tenir compte de la présence des guillemets.

#### 1.2.5 Les deux points

Nous avons choisi de placer une fin de phrase après les deux points lorsqu'ils sont suivis de guillemets :

*{S} Un intérêt qu'un aveugle d'Ecommoy, dans la Sarthe, fidèle abonné de la bibliothèque segréenne a exprimé en ces termes à l'assemblée générale de l'association :{S} " Ça me permet de passer des journées agréables.{S} Des journées qui seraient bien tristes sans cela ".{S} (Ouest France)*

Dans les autres cas, nous ne plaçons pas de fin de phrase car il peut être intéressant, pour des études linguistiques réalisées avec Intex, de ne pas découper la phrase lorsque :

- On a une énumération, ex : *Les trois autres quarts ont été payés par d'autres contributeurs : Arabie saoudite, Allemagne, etc. (Le Monde)*
- On donne une explication, une information supplémentaire, ex : *Le bilan de ces affrontements est très lourd : officiellement 159 morts, surtout des jeunes, et des milliers d'arrestations, officieusement, plus de 500 victimes. (Le Monde)*

### 1.3 Les résultats

Afin de vérifier que le nouveau graphe sentence apporte une amélioration significative par rapport au graphe fourni avec Intex, nous avons comparé les résultats obtenus sur 4,5 Mo de textes composés d'articles du journal *Le Monde* (2,2 Mo et 17 043 phrases), du journal *Ouest-France* (1,9 Mo et 19 572 phrases) et du roman de Balzac *La Femme de trente ans* (387 Ko et 3 664 phrases) (voir l'Annexe B p. 123 pour de plus amples informations sur notre manière de procéder à la vérification des résultats).

Comme les erreurs de découpage de phrases sont principalement dues aux ambiguïtés avec le point, il nous a semblé intéressant de calculer les fréquences des différentes sortes de points afin de comparer plus justement les résultats. Il ressort de cette étude (Tableau 4) que le roman de Balzac contient très peu de points qui puissent être ambigus avec le point final (2% de points appartiennent à des anthroponymes) ce qui va expliquer la faible différence de résultats entre l'ancien automate d'Intex et le nouveau.

Dans le journal *Le Monde*, il y a 8 % de points qui appartiennent à des noms de personnes, 4,5 % à des abréviations et 0 % de sigles (les articles du *Monde* étudiés ici ont une dizaine d'années comme ceux de *Ouest France*, mais *Le Monde* avait alors déjà adopté la norme sans points pour l'écriture des sigles). Le journal *Ouest France* contient environ 20 % de points qui ne sont pas des points de fin de phrases (sigles, noms de personnes, abréviations) et qui peuvent donc être ambigus avec ceux-ci.

	<i>Ouest-France</i>	<i>Le Monde</i>	<i>La Femme de trente ans</i>
Points de fin de phrases	69 %	79 %	86 %
Points de suspension	9 %	8 %	12 %
Points dans les sigles	13 %	0 %	0 %
Points dans les noms de personnes	5 %	8 %	2 %
Points dans les abréviations	1 %	4,5 %	0 %
Autres points	3 %	0.5 %	0 %

Tableau 4 : Les proportions des différentes sortes de points en corpus

On compte dans les deux textes les silences et les bruits qu'ont provoqués les graphes. On calcule ensuite le rappel et la précision<sup>50</sup>.

Le rappel est le nombre de phrases correctes trouvées par le graphe par le nombre de phrases qui auraient dû être trouvées ; c'est-à-dire la proportion de phrases correctement trouvées.

La précision représente le nombre de phrases correctement trouvées par notre système sur le nombre total de phrases trouvées qu'il a trouvées.

Comme nous l'attendions, la précision et le rappel sont améliorés, surtout dans le journal *Ouest France*. Ceci s'explique par la très grande quantité de points non finaux présents dans ce journal et désormais correctement traités. Les résultats du découpage du journal *Le Monde* sont eux aussi améliorés de manière significative. Sur le roman de *Balzac*, on obtient également une amélioration, même si elle est moins nette.

<sup>50</sup> Ces deux mesures sont les mêmes que celles expliquées au Chapitre 2 § 1.3 (p. 25)

<b>Précision</b>	<b>Ouest France</b>	<b>Le Monde</b>	<b>La Femme de trente ans</b>	<b>Totaux</b>
graphe <i>sentence</i> livré avec Intex	96,66%	98,15%	99,63%	98,07%
notre graphe <i>sentence</i>	99,58%	99,91%	99,92%	99,76%
<b>Rappel</b>	<b>Ouest France</b>	<b>Le Monde</b>	<b>La Femme de trente ans</b>	<b>Totaux</b>
graphe <i>sentence</i> livré avec Intex	96,93%	99,04%	99,47%	98,44%
notre graphe <i>sentence</i>	99,69%	99,96%	99,95%	99,83%
<b>F-mesure</b>	<b>Ouest France</b>	<b>Le Monde</b>	<b>La Femme de trente ans</b>	<b>Totaux</b>
graphe <i>sentence</i> livré avec Intex	96,79%	98,59%	99,54%	98,25%
notre graphe <i>sentence</i>	99,63%	99,93%	99,93%	99,79%

Tableau 5 : Précision, rappel et F-mesure des résultats sur les différents corpus

Par ailleurs, la manière dont les graphes sont passés sur le texte pose un problème : si plusieurs cas sont imbriqués entre eux, il peut se produire une erreur. L'exemple suivant illustre ce problème :

*Ce produit est vendu 20 F. {S} Tél.{S} : 03.25.00.01.02.03*

Le motif *20 F. Tél* est reconnu dans le graphe *cas3* par le chemin *motifSymbole* suivi d'un point, d'un espace et d'un mot commençant par une majuscule. Cette séquence ayant été reconnue, l'application du graphe *sentence* redémarre après le mot *Tél*. Le motif *Tél. : 03.25.01.02.03* (décrit dans le graphe *TelFax*) n'est alors pas reconnu et la fin de phrase est mal placée. Cette imbrication ne peut être prise en compte car elle complique énormément le graphe et risque d'induire d'autres erreurs (plusieurs chemins peuvent avoir la même longueur auquel cas l'algorithme d'Intex ne choisit pas forcément le chemin que l'on souhaite). Peut-être serait-il intéressant de réaliser un découpage en phrases à l'aide d'une cascade de transducteurs ?

S'il n'est pas possible de rendre infaillible le découpage automatique des phrases, on peut encore l'améliorer sur un certain nombre de points, notamment par le recensement systématique, dans des domaines précis, des abréviations d'usage ou des mots suivis d'une lettre majuscule

## 2 Etiquetage des textes

Nous utilisons l'étiquetage non désambiguïsé d'Intex (i.e. chaque mot porte toutes les étiquettes possibles trouvées grâce au dictionnaire). Les étiquettes syntaxiques exactes des mots ne sont pas nécessaires, car nous décrivons, dans nos grammaires, un contexte lexical.

Intex permet d'utiliser de nombreuses ressources : ce système est fourni avec des dictionnaires à très large couverture de la langue française. Les dictionnaires Delaf [Courtois, Silberztein, 1990] contiennent les mots simples et leurs formes fléchies, et les dictionnaires Delacq, les mots composés et leurs formes fléchies. Précisons la syntaxe des dictionnaires sous Intex à travers quelques exemples :

*acheté, acheter.V:Kms*

*acheter, acheter.V:W*

*provençale, provençal.A+Top:fs*

Le premier mot est la forme fléchiée (*acheté, acheter* ou *provençale*) et le second la forme canonique ou lemme (*acheter* ou *provençal*). Suivent, après un point, la catégorie<sup>51</sup> syntaxique (*N* = nom, *V* = verbe, *A* = adjectif, etc.), des traits sémantiques (*Top* = toponyme, *Conc* = concret, *Hum* = humain etc.) et des traits morphologiques (*ici*, *Kms* pour participe passé au masculin singulier, *W* pour infinitif et *fs* pour féminin singulier).

Intex permet de créer et d'utiliser nos propres dictionnaires, il suffit de respecter le format ci-dessus.

## 2.1 Dictionnaire de preuves externes ou internes

Le **dictionnaire des noms de professions** [Fairon, 2000] nous sera d'une grande utilité pour l'extraction des noms de personnes. Ce dictionnaire nous servira de preuve externe pour les noms de personnes. Les noms de profession ont l'étiquette <N+Profession>.

Exemples :

*avocat, avocat.N+Profession:ms*

*banquiers, banquier.N+Profession:mp*

*batteur, batteur.N+Profession:ms*

Dans le cadre du projet Prolex, un **dictionnaire des prénoms français et étrangers**<sup>52</sup> (nommé *prénom-prolex*) a été élaboré puis complété tout au long de cette thèse : ce dictionnaire nous servira de preuve interne pour trouver les noms de personnes. Ce dictionnaire contient plus de 6 600 entrées de la forme : *Caroline, Caroline.N+PR+Hum+Prénom:fs*

Ce dictionnaire contient des formes simples de prénoms et quelques formes composées. Il ne contient pas les compositions de prénoms telles que *Jean-Pierre*, car *Jean* et *Pierre* sont des formes simples connues. Pour reconnaître ce prénom composé, il suffit d'écrire une grammaire locale<sup>53</sup> (Figure 23).

<sup>51</sup> Toutes ces catégories sont décrites dans le Manuel d'utilisation d'Intex et dans [Silberztein, 1993].

<sup>52</sup> La couverture actuelle de notre dictionnaire des noms propres sur le journal *Le Monde* représente 96% des prénoms.

<sup>53</sup> Dans cette grammaire locale, <N+Prénom> permet de reconnaître les mots d'un texte qui ont été étiquetés par le dictionnaire des prénoms.

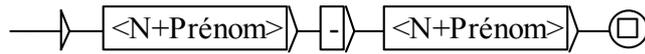


Figure 23 : Un prénom composé

Cependant il existe quelques formes complexes de prénoms que nous avons mises dans le dictionnaire, car toutes ou certaines des parties du prénom composé n'existent pas sous une forme simple. Par exemple :

*Abd-El-Kader, Abd-El-Kader.N+PR+Hum+Prénom:ms*

*Kader* est un prénom simple bien connu mais *Abd* et *El* ne le sont pas<sup>54</sup>.

## 2.2 Les dictionnaires de noms propres

Nous utilisons aussi le **dictionnaire *Prolintex de toponymes*** [Maurel, Piton, 1999] réalisé dans le cadre du projet Prolex. Ce dictionnaire contient plus de 100 000 entrées.

*allemand, allemand.A+Top+PPays+IsoDe:ms*

*Bordeaux, Bordeaux.N+PR+Top+PVil+IsoFr:ms:fs*

*Marseillaise, Marseillais.N+PR+Hum+Top+PVil+IsoFr:fs*

*Méditerranée, Méditerranée.N+PR+Hyd:fs*

Ce dictionnaire permettra de reconnaître de nombreux noms de lieux car ceux ci sont difficiles à repérer par manque de preuve. Les toponymes sont étiquetés N+Top et les adjectifs toponymiques A+Top par ce dictionnaire.

Un **dictionnaire de sigles et d'abréviations** (nommé *sigle-prolex*) a été créé au cours de cette thèse. Il contient environ 3 300 entrées. L'extension du sigle est précisée (ex : *Ena, Ena.N+PR+Sigle+Org:fs/Ecole nationale d'administration*). Il existe de nombreux sigles qui ne sont pas des noms propres :

Exemples :

*DTP, DTP.N+Sigle+Nc:ms /Dyphtérie Tétanos Polio*

*VTC, VTC.N+Sigle+Nc:ms/Vélo Tout Chemin*

Ces sigles ont un statut de noms communs ou d'expression figées : ils appartiennent souvent à la terminologie d'un sous-langage technique.

Il existe aussi beaucoup de sigles homonymes, par exemple :

*CIO, CIO.N+PR+Sigle+Org:fs/Central Intelligence Organization*

*CIO, CIO.N+PR+Sigle+Org:ms/Centre d'information et d'orientation*

*CIO, CIO.N+PR+Sigle+Org:ms/Comité international olympique*

<sup>54</sup> Où plutôt nous ne savons pas s'ils le sont.

De plus, les sigles peuvent représenter des personnes ou des lieux. Le dictionnaire de sigles contient donc l'information sémantique du type du sigle (Org, Person, Top).

Exemples :

*DSK, DSK.N+PR+Sigle+Person:ms/Dominique Strauss-Kahn*

*USA, USA.N+PR+Sigle+Top:mp/United States of America*

*ENA, ENA.N+PR+Sigle+Org:fs/Ecole nationale d'administration*

Ce dictionnaire sera utilisé avec l'aide d'une grammaire locale pour repérer des sigles et les catégoriser.

### 2.3 Dictionnaires morphologiques

Nous réalisons aussi un étiquetage au niveau de la graphie des mots à l'aide de transducteurs. Nous étiquetons les chiffres romains *CR* avec le graphe *ChiffresRomains* fourni par Intex.

Les candidats noms propres sont étiquetés *CNP* (ils sont composés d'au moins 2 lettres, la première lettre est une majuscule et les suivantes sont minuscules ou majuscules).

Nous voulions aussi étiqueter les candidats sigles composés d'au moins deux consonnes en majuscules, mais nous y avons renoncé à cause du nombre trop grand d'ambiguïtés avec des mots qui ne sont pas des noms propres.

Nous étiquetons un certain nombre de formes composées. Les noms de personne tels que *Mac Donald* ou *O'Reilly* sont de bons candidats noms propres composés que nous appelons *CNPCA*. Nous avons donc créé un graphe décrivant les formes de patronymes avec particules étrangères (ex : *Al, al, El, el, Do, do, Van, van, etc.*).

Grâce à ces étiquetages supplémentaires, un mot dont la morphologie est celle d'un nom propre et qui ne se trouve pas dans un dictionnaire de noms propres sera étiqueté comme un nom propre possible.

### 2.4 Comment utiliser des dictionnaires sous Intex

Intex permet d'utiliser les étiquettes placées par les dictionnaires sur le texte dans les graphes.

On peut vouloir rechercher toutes les formes fléchies d'un mot dans un texte. Si on recherche toutes les formes du verbe *avoir*, il suffit de placer l'étiquette *<avoir>* dans une des boîtes du graphe. Cette boîte permettra de reconnaître les formes : *a, aviez, eu, avoires*, etc. du verbe *avoir*.

D'autre part, on peut vouloir rechercher des mots d'une catégorie syntaxique particulière (nom, verbe, pronom, etc.) ou appartenant à une catégorie sémantique (humain, prénom, toponyme, etc.). Ainsi, l'étiquette *<V>* permet de rechercher tous les mots qui sont étiquetés *verbes*<sup>55</sup>, *<N:ms>* repère tous les mots qui sont des *noms au masculin singulier*, *<A+Top>* localise tous les mots étiquetés *adjectifs toponymiques*.

<sup>55</sup> Dans cette recherche, les formes ambiguës *avions, aura, avoir, etc.* seront trouvées même si, dans le texte, il ne s'agit pas du verbe *avoir*.

### **3 Conclusion**

Nous avons vu l'intérêt d'une bonne segmentation des textes à travers l'amélioration du graphe sentence proposé sous Intex. Les résultats obtenus sont supérieurs avec un rappel global de 98,5% et une précision de 98,1% sur les trois types de textes (F-mesure de 98,3%). Pendant notre phrase d'extraction des noms propres, nous n'aurons plus à nous soucier de l'ambiguïté des points finaux de phrases avec ceux qui sont contenus dans les noms propres.

Nous utiliserons un certain nombre de dictionnaires existants ou créés par nos soins pour étiqueter les textes et réaliser notre extraction des noms propres.



## Chapitre 5

# EXTRACTION DE NOMS PROPRES

Ce chapitre décrit notre système d'extraction de noms propres nommé *ExtracNP* (§1) ; il utilise les cascades de transducteurs du système CasSys. Nous expliquons ensuite comment nous procédons pour extraire les différents types de noms propres (§2, 3, 4). Finalement, nous présentons les résultats obtenus par ce système (§5).

## 1 L'extracteur de noms propres *extracNP*

### 1.1 Architecture du système

Le système *extracNP*, dont l'architecture est représentée par la Figure 24, consiste en quatre étapes principales.

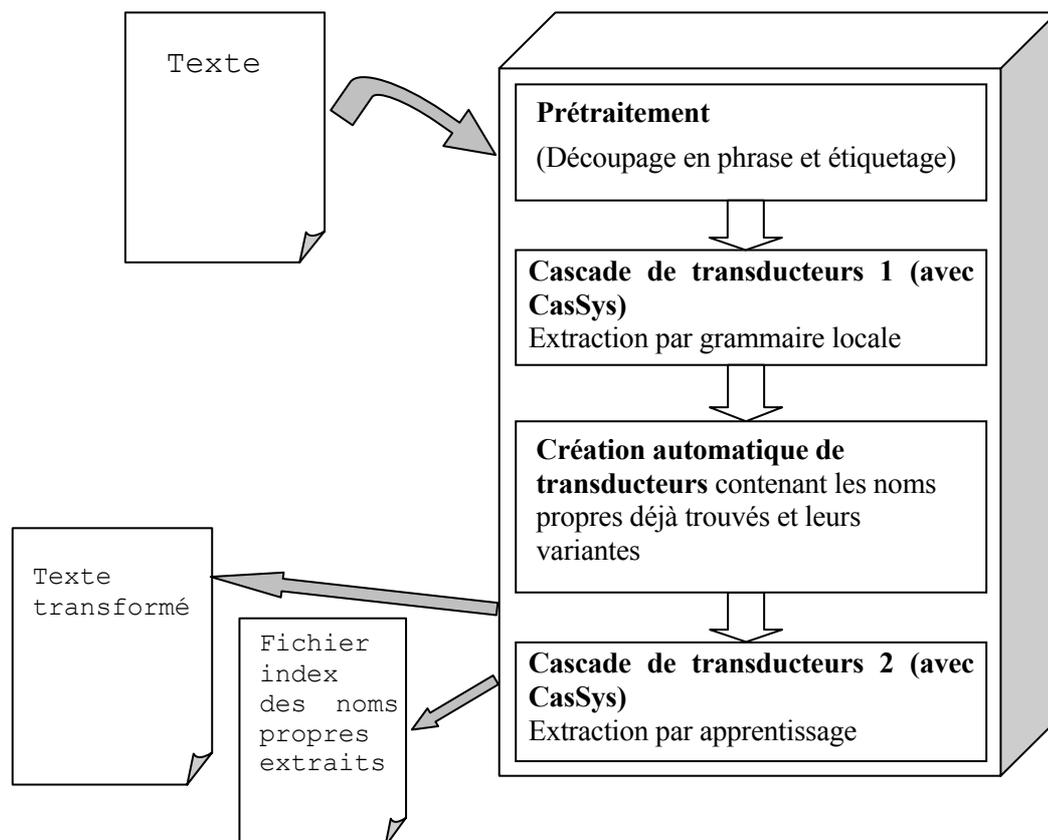


Figure 24 : Architecture de *ExtracNP*

Après le découpage en phrases et l'application des dictionnaires pour étiqueter le texte, la détection des noms propres est réalisée de la manière suivante :

- La **première cascade de transducteurs** permet d'extraire les noms propres par la description de leurs grammaires locales grâce aux transducteurs (graphes d'Intex). Ces grammaires sont fortement lexicalisées afin de catégoriser les noms propres : ce sont les mots trouvés dans le contexte des noms propres qui permettent de leur affecter une catégorie. Nous décrivons les preuves externes (contextes droits et gauches des noms propres) et les preuves internes des noms propres, en les accompagnant d'indices morphologiques et syntaxiques pour délimiter le nom propre.

Deux types d'informations sont extraites par nos grammaires : les noms propres et leurs contextes. Les contextes servent principalement à trouver les noms propres mais, ils fournissent aussi de l'information (ex : *président, société boursière, américain*) qui pourra par la suite être placée dans une base de données, en relation avec les noms propres concernés.

- Une **deuxième cascade de transducteurs** est générée automatiquement à partir des noms propres<sup>56</sup> trouvés par la première cascade passée sur le texte. Ceux-ci sont utilisés comme un dictionnaire : bien souvent, dans un même texte, un nom propre n'a qu'un seul référent (vu au Chapitre 2 § 4.3 p.39). Cette deuxième cascade permet de trouver des noms propres qu'aucun indice dans le texte ne permet de repérer. Dans les rares cas où une erreur de catégorisation survient à la première étape, cette erreur est répercutée à la deuxième étape.

## 1.2 L'ordre de passage des règles

Nous commençons par les règles les plus longues indépendamment du type des noms propres qu'elles reconnaissent. Voici le schéma global de l'ordre de passage des transducteurs :

1. On commence par une grammaire des adresses<sup>57</sup> (assez nombreuses dans les journaux) et peu ambiguës grâce à leurs motifs.
2. Les règles reconnaissant les coordinations de noms de personnes, de noms de société ou de lieux avec un contexte droit ou gauche sont appelées (ex : *MM. Fontanet, Haby, Beullac, Savary ...*).

---

<sup>56</sup> Et de leurs variantes :

- Pour les noms de personnes : le patronyme de personne seul ou accompagné d'un prénom abrégé,
- Pour les noms d'organisations : nous générons quelques formes courtes par élimination des suffixes tels que *Inc, Co*, etc. mais ces variantes sont limitées en difficilement prévisibles.

<sup>57</sup> La grammaire des adresses est très simple à décrire et on ne peut pas confondre un motif d'adresse avec un autre motif. Exemples :

*15, avenue Montaigne, Paris 8<sup>e</sup>*

*1, place Paul-Claudé, 75 010 Paris.*

*2, rue Ravenstein, à Bruxelles*

3. On applique ensuite les règles avec contextes droits ou gauches et éventuellement preuves internes reconnaissant un nom de personne, d'organisation ou de lieu (ex : *Elisabeth Guigou*, *garde des sceaux*), ainsi que les règles concernant les noms propres étrangers pour lesquels une preuve interne suffit car ils ne sont pas ambigus avec des mots français (ex : *International Data Corp.*, *Deutsche Bank*, *Miami Beach*).
4. On applique les règles contenant au moins une preuve interne : prénoms, mots classificateurs d'organisation commençant par une majuscule (ex : *Organisation*, *Compagnie*, *Association*) et l'esperluette & (ex : *AT&T*).
5. Enfin, nous repérons les lieux en dernier.

Les graphes vont permettre une description puissante et lisible des grammaires locales de noms propres que les expressions régulières utilisées par la plupart des systèmes.

Nous avons déjà présenté, de manière générale, au Chapitre 1, la manière d'extraire les noms propres avec l'aide des indices qu'apportent le texte. Dans la suite, nous décrivons, pour chacun des types de noms propres étudiés (noms de personnes, d'organisations et de lieux), comment nous avons procédé pour les reconnaître et les catégoriser (§2, 3 et 4).

## 2 Les noms de personnes

Les noms de personnes sont les entités les plus faciles à extraire : les indices pour les repérer sont très nombreux. Comme l'ont déjà remarqué [Kim, Evens, 1996], l'auteur d'un article de journal donne en général une première fois la forme complète du nom de personne accompagnée d'informations, puis des formes abrégées.

À travers cette section, nous nous rendons compte que la grammaire de reconnaissance des noms de personnes est extrêmement structurée, avec le souci de minimiser le coût de l'écriture de ces grammaires. Nous verrons plus tard que les autres entités nommées ne permettent pas une telle structuration.

Les graphes appelés par la cascade pour extraire les noms de personnes sont au nombre de 43.

### 2.1 La description des preuves externes et internes des noms de personnes

Dans les journaux *Le Monde* que nous avons étudiés, les noms de personnes sont accompagnés à hauteur de 90% de preuves externes et internes. Voici comment se répartissent les preuves externes et internes des noms de personnes :

- a) 45% des noms de personnes<sup>58</sup> sont précédés d'un contexte contenant une civilité, un titre ou un nom de profession, suivis du patronyme. S'y ajoute éventuellement un prénom (preuve interne grâce à notre dictionnaire des prénoms).

Exemples :

*M. Jean-Pierre Soisson déclarait : ...*

*Ce regain de violence a coïncidé avec la visite officielle du **président péruvien Alberto Fujimori** en Equateur, qui a pris fin samedi.*

*... et qui qualifiait la démission du **président Chadli** d' " " événement important et lourd de conséquences " , ...*

- b) 45% des noms de personnes<sup>59</sup> n'ont pas de contextes descriptibles mais contiennent la preuve interne apportée par un prénom connu de notre dictionnaire et suivi du patronyme.

Exemples :

*Adrien Friez estime ces chiffres ...*

*Désormais **Cyrille Bonnand** n'y croit plus trop ...*

- c) 5% des noms de personnes sont trouvés par :

- un contexte droit (beaucoup plus rare qu'un contexte gauche). On peut rencontrer un nom de fonction par exemple (ex : "... **De Gaulle, président** ...").
- la présence d'un verbe utilisé pour désigner une action mettant en jeu une personne (dire, expliquer, etc.) comme dans "*Wieviorka est décédé, le 28 décembre, à Paris*" ou "*Jelev a dit*". Cependant des verbes comme *dire, expliquer, etc.* peuvent être employés avec un sujet non humain (nom d'organisation) : cet indice est donc difficilement exploitable.

- d) Les derniers noms de personnes (5%) n'ont aucun contexte, même complexe qui puisse les distinguer à coup sûr d'autres noms propres. Ces noms de personnes sans contextes sont principalement ceux de personnes très connues pour lesquels l'auteur du texte estime qu'il n'est pas nécessaire de préciser le prénom, ni le titre ou la profession, ou ceux de personnes déjà citées dans l'article (Ex : *Picasso n'est pas le premier à passer à la postérité commerciale*). Nous envisageons pour les traiter de créer un dictionnaire de célébrités.

---

<sup>58</sup> Remarquons que cette proportion tombe à 33% dans le journal *Ouest France* : les journalistes de ce journal ajoutent moins de détails sur la fonction des personnes citées.

<sup>59</sup> Dans le journal *Ouest France*, 59% des noms de personnes sont accompagnés d'un prénom seul.

Les contextes gauche et droit de noms de personne sont, pour la plupart :

- les civilités (ex : *Mme, Monsieur*, etc.),
- les titres de toutes sortes : politiques (ex : *président, ministre, député*, etc.), militaires (ex : *général, lieutenant*, etc.), religieux (ex : *cardinal, évêque*, etc.), juridiques ... Par exemple, les titres "ministres" sont décrits Figure 25.
- les noms de professions (ex : *le juge, l'architecte*, etc.)
- Le dictionnaire des toponymes permet de repérer les adjectifs de nationalité dans des expressions telles que *le président américain Clinton, le chancelier allemand Helmut Kohl*. Mais une nationalité seule ne pourra être utilisée pour catégoriser un nom propre car elle indique soit un nom de personne (ex : *le japonais Takao Saito*), soit un nom de société (ex : *l'allemand Volkswagen*), s'il n'y a pas d'autres indices.

Le contexte va permettre la catégorisation du nom de personne ; nous étudions maintenant les formes que peuvent prendre les patronymes et les prénoms.

## 2.2 Morphologie des prénoms et patronymes

La reconnaissance des prénoms se fait sur la base d'indices morphologiques et grâce à notre dictionnaire. Nous attachons une importance toute particulière à l'extraction des prénoms car générer une variante de nom de personne sera plus simple si on sait différencier le prénom du patronyme.

Les formes de prénom à reconnaître sont les suivantes :

- prénoms simples (ex : *Danièle, Louis*),
- prénoms abrégés simples (ex : *E.* pour *Emmanuel*, *Th.* pour *Thierry*),
- prénoms composés (ex : *Jean-Pierre, Charles Edouard*), ou en partie abrégés (ex : *Pierre-J.*). Ces prénoms sont décrits par la Figure 26. L'étiquette <N+Prénom> permet de découvrir un mot étiqueté par le dictionnaire prénom-prolex.
- prénoms composés en partie inconnus<sup>60</sup> : prénom composé dont une des parties est dans le dictionnaire des prénoms et l'autre est inconnue (Figure 27).
- prénoms composés abrégés, ex : *J.P., J.-P., J-P, J-P.*<sup>61</sup>

<sup>60</sup> Rappelons que l'étiquette <CNP> désigne un candidat nom propre.

<sup>61</sup> Voici un nouvel exemple d'ambiguïté : ce dernier point (dans *J-P.*) est à la fois le point final et le point d'abréviation du prénom commençant par la lettre P !

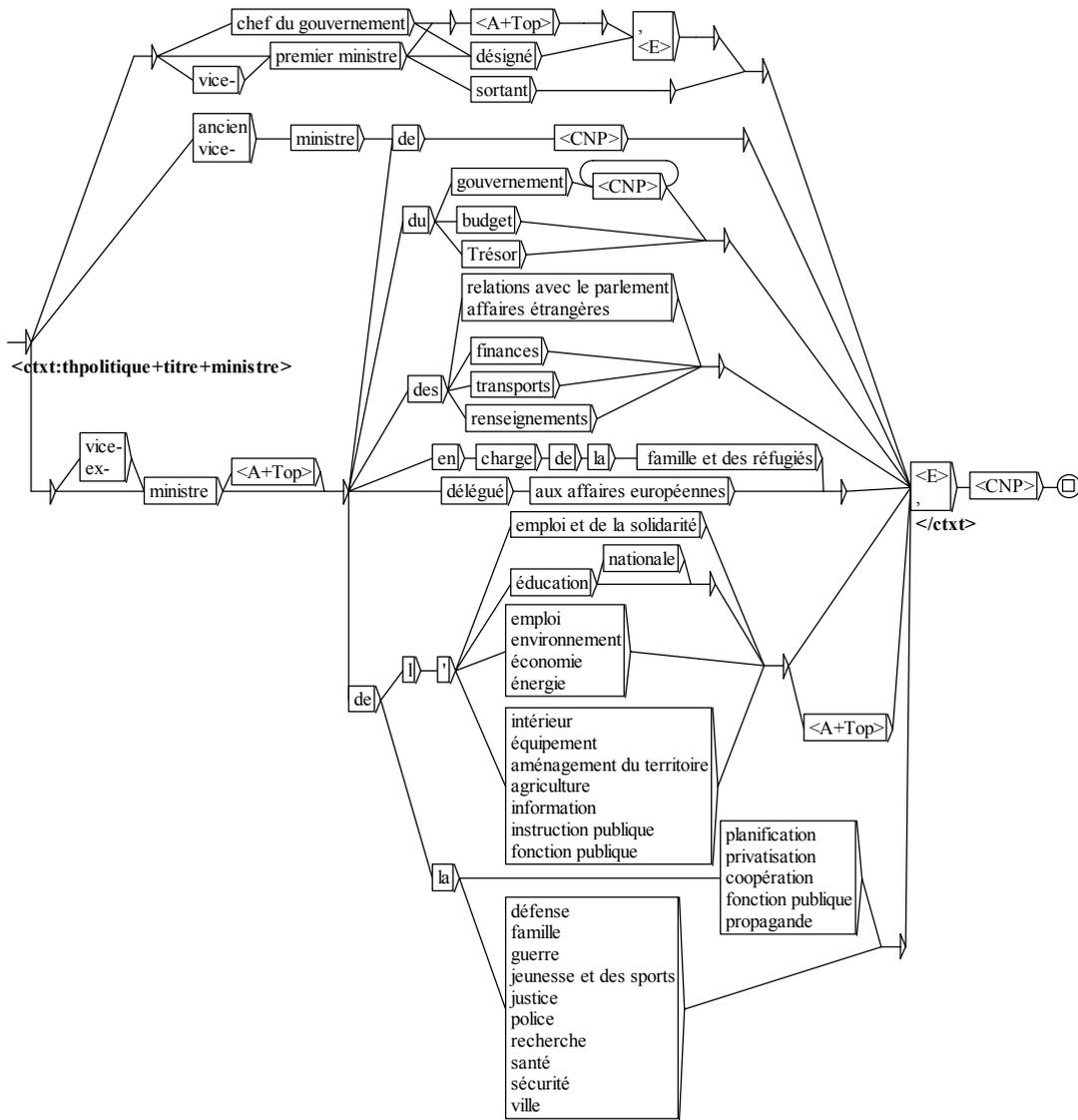


Figure 25 : Contexte "ministre"

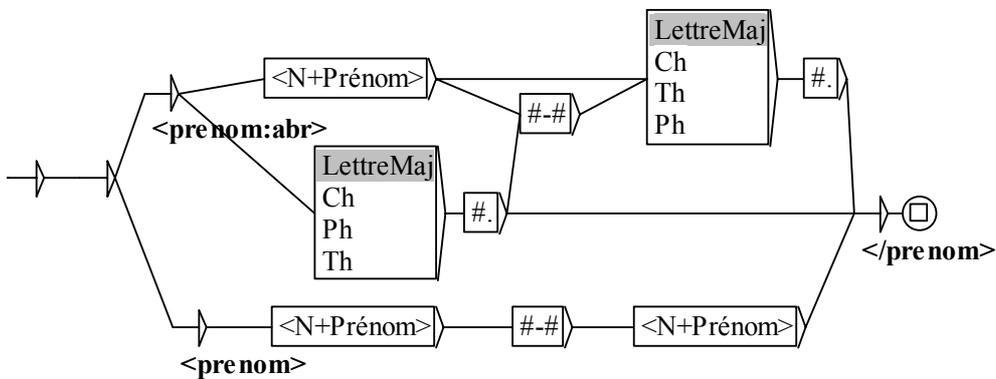


Figure 26 : Graphe décrivant les prénoms composés

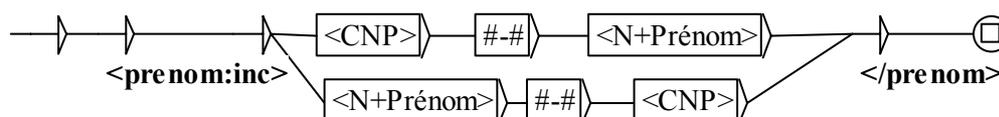


Figure 27 : Graphe décrivant les prénoms en partie inconnus

Nous avons fait l'hypothèse que les personnes sont plus souvent citées par les journalistes en donnant d'abord le prénom puis le nom. Cette règle n'est évidemment pas absolue, puisque, si dans le corpus étudié, aucune forme "nom suivi d'un prénom" n'a été détectée, nous avons observé l'ordre "nom prénom" dans 3% des noms de personnes des articles étudiés dans le journal *Ouest France*.

Les patronymes sont identifiés par des graphes qui reflètent les formes suivantes :

- Patronymes "simples" : ils sont composés d'un ou plusieurs mots commençant par une majuscule (ex : *Dupont*, *Durand-Pérec*) (Figure 28)

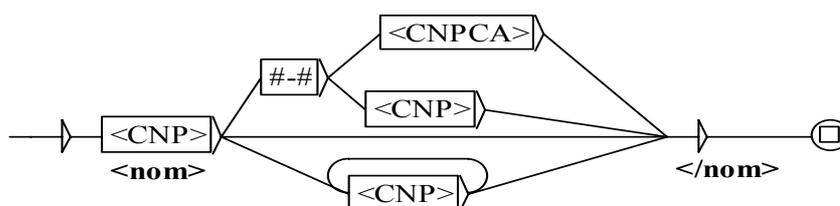


Figure 28 : Graphe représentant les patronymes simples

- Patronymes "composés" d'une particule ; ce sont surtout des noms d'origine étrangère (ex : *Mac Donnell-Douglas*, *O'Ryan*, *von Bulow*, *El Amra*, *Da Silva*, *Do Macedo*, *Ben Driss*) mais aussi des formes françaises (ex : *L'Huillier*, *Le Falch'un*).
- Patronymes français à particules, ex : *Dupont de Nemours*, *de Neuville*, *de la Fontaine*.

Nous avons distingué les patronymes français à particules des patronymes composés car la particule française (*de*, *du*) est très ambiguë en français (avec la préposition *de*) ce qui n'est pas le cas pour les particules étrangères.

Exemples :

*le Carmen de Bizet*

*la Duchesse de Windsor*

*le maire de Paris*

*de Bizet*, *de Windsor* et *de Paris* ne sont pas des patronymes à particules.

*Carmen* est présent dans notre dictionnaire de prénom et va donc être la preuve interne de la présence d'un nom de personne ; si on n'interdit pas la présence de déterminants ou de noms communs devant un prénom suivi de la préposition *de*, on obtient l'interprétation erronée suivante : *le <person> <pre nom> Carmen <\pre nom> <nom> de Bizet <\nom> <\person>*. Comme nous l'avions expliqué au chapitre 1, la

présence d'un déterminant devant un nom relève d'un emploi métaphorique, dénomiatif ou fractionné et nous ne traitons pas ces cas dans notre système d'extraction.

*la Duchesse* et *le maire* sont des contextes de noms de personnes (resp. un titre nobiliaire et une fonction). Nous interdisons donc les noms de personnes à particule *de* derrière un titre ou un nom de profession car ce sont rarement des noms de personnes. Par contre, une civilité suivie d'une préposition *de* (ex : *M. de Neuville*) ne sera probablement jamais ambiguë.

### 2.3 Combinatoire de la cascade pour les noms de personnes

Les noms de personnes seront donc repérés par des combinaisons de contextes gauche ou droit, de prénoms et de patronymes.

L'ordre de passage des transducteurs dans la cascade est fondamental. Illustrons ce point par un exemple. Si nous passons un graphe qui reconnaît *M.* suivi d'un *patronyme* (Figure 29) avant le transducteur qui reconnaît *M.* suivi d'un *prénom* puis d'un *patronyme* (Figure 30), un texte contenant la séquence *M. Jean Dupont* sera mal analysé. Le motif extrait sera :

`<person> M. <nom> Jean </nom> </person>`

au lieu du motif :

`<person><prénom> Jean </prénom><nom> Dupont </nom></person>.`

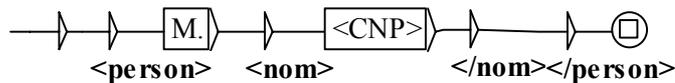


Figure 29 : Graphe reconnaissant *M.* suivi d'un patronyme

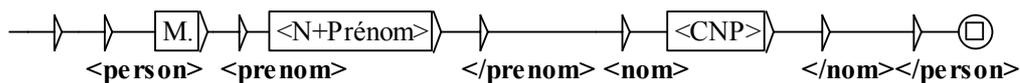


Figure 30 : Graphe reconnaissant *M.* suivi d'un prénom puis d'un patronyme

**Description de l'ordre de passage de la cascade de transducteurs**

	contextes gauches (oui/non)	Prénoms	Patronymes	Exemples
O R D R E  D E  P A S S A G E  D E S  T R A N S D U C T E U R S	Oui	Prénoms composés et abrégés Prénoms simples	Patronymes composés	<i>le président Richard von Weizsöcker</i>
	Oui	Prénoms composés et abrégés	Patronymes simples Patronymes à particules	<i>M. J.-P. de Fonsac</i>
	Oui	Prénoms composés en partie inconnus	Patronymes composés	<i>Pas d'exemple</i>
	Oui	Prénoms composés en partie inconnus	Patronymes simples	<i>Roger-Pol Droit</i> ( <i>Pol n'était pas dans notre dictionnaire de prénoms</i> )
	Oui	Prénoms simples	Patronymes simples	<i>M. Guy Fleury</i>
	Oui	Pas de prénom	Patronymes composés	<i>M. Strauss-Kahn</i>
	Oui sans les professions et les titres.	Pas de prénom	Patronymes à particules	<i>M. de Neuville</i>
	Oui	<CNP> <CNP>	<CNP> - <CNP> <sup>62</sup> <CNP> <CNP>	<i>M. Amnon Lipkin Shahak</i> ( <i>Amnon n'était pas dans notre dictionnaire de prénoms</i> )
	Oui	<CNP>	<CNP>	<i>le général Veljko Kadijevic</i>
	Oui	Pas de prénom	Patronymes simples	<i>Mme Bouchardeau</i>
	Non Interdit les déterminants avant le nom de personne	Prénoms composés et abrégés Prénoms simples	Patronymes composés	<i>Philippine Leroy-Beaulieu</i> <i>Maria da Graca Meneghel</i>
	Non Interdit les déterminants avant le nom de personne	Prénoms composés et abrégés Prénoms simples	Patronymes simples	<i>P. Bourdieu</i>
	Non Interdit les déterminants avant le nom de personne	Prénoms composés en partie inconnus	Patronymes composés	<i>Pas d'exemple rencontré</i>
	Non Interdit les déterminants avant le nom de personne	Prénoms composés en partie inconnus	Patronymes simples	<i>Pas d'exemple rencontré</i>

Tableau 6 : Description d'une partie des transducteurs reconnaissant les noms de personnes

<sup>62</sup> <CNP>-<CNP> désigne 2 candidats noms propres séparés d'un tiret.

Le Tableau 6 décrit la partie de cascade de transducteurs portant sur les noms de personnes. Comment lire ce tableau ? La première ligne se lit, par exemple, de la manière suivante :

*Le premier transducteur reconnaît des noms de personnes composés par :*

- *un prénom simple, composé ou abrégé,*
- *suivi d'un patronyme composé ou à particule (ex : O'Reilly, de La Fontaine),*
- *et précédé d'un contexte déclencheur.*

Cet ordre de passage des transducteurs a été obtenu de manière empirique en essayant de minimiser les erreurs dues au passage d'un graphe avant un autre.

Remarquons qu'avant de passer les transducteurs qui repèrent les noms de personnes qui ne sont accompagnés que d'un prénom, il faut appliquer les transducteurs qui détectent les noms d'organisations afin d'éviter des problèmes d'ambiguïtés entre ces deux types de noms propres (ex : *la société Hugues Aircraft* où *Hugues* est un prénom).

## 2.4 Reconnaissance des coordinations de noms de personnes

Nous avons tenu compte des coordinations de noms de personnes, assez fréquentes dans les textes journalistiques :

*C'est également au nom de ce réalisme que MM. Jacques Delors et Raymond Barre ont souhaité une approche plus " humble " que celle d'Alain Minc.*

*Dix-huit prix de 1 000 F récompenseront aussi des créateurs venus de toute la région : MM. Alyas (Le Faouët), Allain (Lorient), Bouet (Rennes) ...*

*MM. Maurice Girard, soixante-neuf ans, et Alain Mercadé, quarante-huit ans, anciens dirigeants de la SGF, devront verser chacun 80 000 francs.*

*MM.*, *Messieurs* ou *Mesdames*, ou un titre ou nom de profession au pluriel (plus rare) indiquent des noms de personnes coordonnés. La plupart du temps, les auteurs donnent les noms de personnes de manière homogène : s'ils donnent, en premier, le prénom et le nom d'un individu, l'individu suivant sera cité de la même façon. Cependant, cette habitude n'est pas toujours respectée et nous avons estimé qu'il faudrait plus d'une centaine de graphes pour décrire tous les motifs de noms de personnes coordonnés avec toutes leurs variantes possibles, d'autant plus que ces motifs peuvent contenir des insertions d'informations tels qu'un âge, un lieu, un parti politique, etc.

Voici la solution que nous avons mise en œuvre pour diminuer cette complexité : transformer le texte avant d'appliquer simplement la cascade précédente. Les motifs *MM.*, *Messieurs*, etc. vont être supprimés du texte et remplacés par l'insertion du motif *M.* devant chaque élément de la coordination. Une dizaine de graphes différents vont décrire "morphologiquement" les coordinations.



### 3 Les organisations

L'extraction des noms d'organisations est moins structurée que celle des noms de personnes car les formes à reconnaître sont plus variées et plus complexes. Les grammaires sont écrites au cas par cas (Figure 34).

Les graphes appelés par la cascade pour extraire les noms d'organisations sont au nombre de 35.

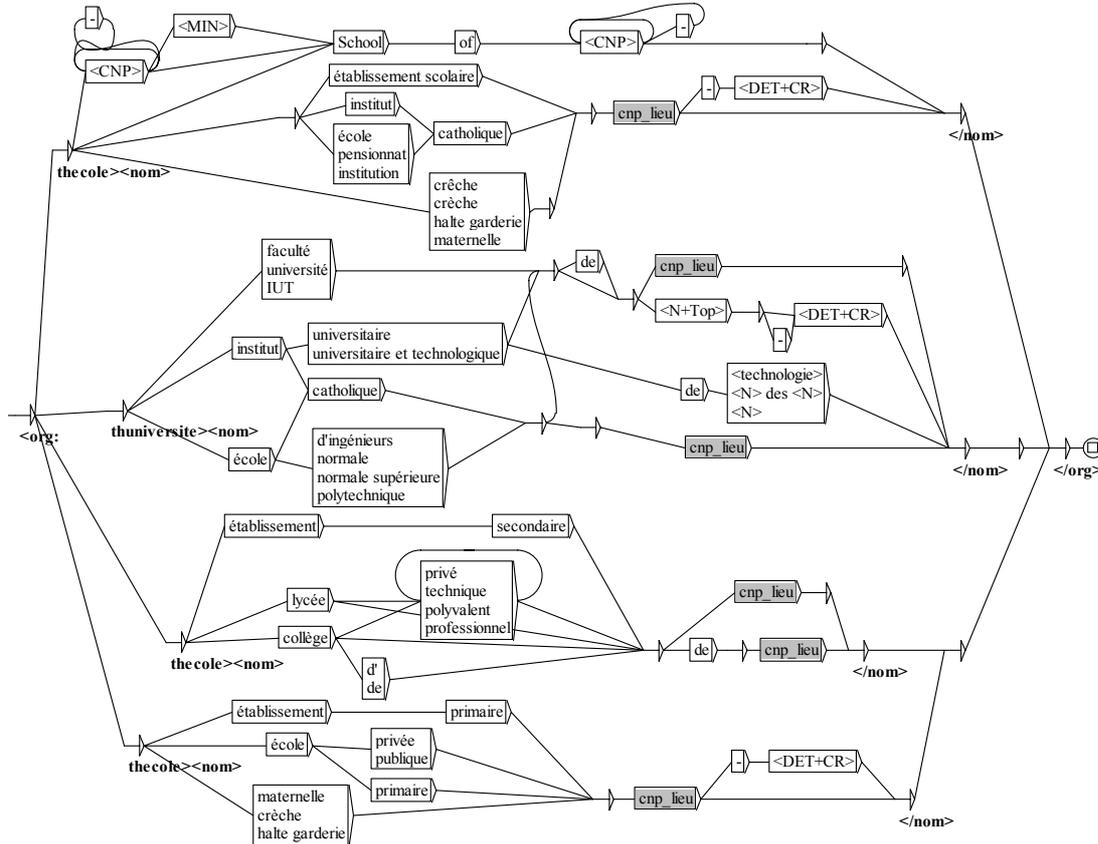


Figure 34 : Grammaire des établissements scolaires

Nous avons remarqué qu'un même mot pouvait être une preuve interne (ex : *Banque Centrale du Mexique*) ou une preuve externe (ex : *la banque Paribas*). Dans le second cas, le nom propre est *Paribas*, car on pourrait aussi rencontrer les expressions suivantes : *le groupe Paribas*, *la société Paribas* ou *Paribas* seul. Nous allons donc, pour les organisations, commencer par la recherche des preuves internes.

Ce phénomène peut aussi se produire sur des noms de lieux : par exemple, on dit *la Mer Rouge* mais on ne dira jamais *la Rouge* alors que l'on peut dire indifféremment *la mer Méditerranée* ou *la Méditerranée*.

#### 3.1 Preuves internes de noms d'organisations

50% des noms d'organisations sont accompagnés d'une preuve interne, car ce sont, pour la plupart, des noms propres à base descriptive. Ils sont formés de noms communs qui permettent de deviner qu'il s'agit d'une entreprise (ex : *Compagnie générale des eaux*, *Société européenne des satellites*), d'une organisation (ex : *Organisation mondiale*

de la santé), d'une banque (ex : *Banque de France*) ou autres (ex : *Front populaire biélorusse*, *Union cycliste internationale*). De tels noms propres sont rarement accompagnés d'un contexte gauche car ils contiennent suffisamment d'information sur leur signification en interne<sup>63</sup>.

Les noms d'organisations à base descriptive ou mixte peuvent porter des majuscules sur tous les mots qui les composent (Ex : *Banque Rotschild*). Par contre, il arrive souvent que ce type de noms d'organisations ne portent pas la majuscule sur chacun des mots qui les composent<sup>64</sup> (Ex : *Union cycliste internationale*). Comment alors savoir où se terminent de tels noms propres ?

[Trouilleux, 1997] propose une grammaire syntaxique de la limite droite, mais nous pensons que la syntaxe n'est pas suffisante pour trouver cette limite droite. Par exemple, en fin de nom d'organisation, on peut autoriser un nom suivi d'un adjectif pourtant dans *Organisation mondiale du commerce performante*, l'adjectif *performante* n'appartient pas au nom d'organisation *Organisation mondiale du commerce*. Par contre, dans *Fédération des industries forestières* ou *Institut national de la recherche agronomique*, le nom et l'adjectif finaux appartiennent tous les deux au nom d'organisation. Un nom d'organisation à base descriptive se termine surtout par un nom commun ou un adjectif. Plutôt que l'utilisation d'aspects syntaxiques seuls, nous avons créé un dictionnaire d'adjectifs et de noms pouvant se trouver dans les noms d'organisations (ces adjectifs sont étiquetés *A+Pr* et les noms *N+Pr*).

---

<sup>63</sup> On ne dirait pas *la société Compagnie générale des eaux*

<sup>64</sup> Les noms propres à base descriptive ou mixte, étant composés de noms communs, sont sujet au bon vouloir du scripteur qui mettra ou non une majuscule sur ces noms propres. Les exemples que nous donnons ici ont été trouvés sans majuscules dans *Le Monde*.

En voici un petit échantillon :

*agricole, agricole.A+Pr:ms*  
*agriculture, agriculture.N+Pr :fs*  
*autonome, autonome.A+Pr:ms*  
*bancaire, bancaire.A+Pr:ms*  
*budgétaire, budgétaire.A+Pr:ms*  
*central, central.A+Pr:ms*  
*civil, civil.A+Pr:ms*  
*communiste, communiste.A+Pr:fs*  
*contemporain, contemporain.A+Pr:ms*  
*démocrate, démocrate.A+Pr:fs*  
*démocratique, démocratique.A+Pr:fs*  
*départemental, départemental.A+Pr:ms*  
*économique, économique.A+Pr:fs*  
*football, football.N+Pr :ms*  
*municipal, municipall.A+Pr:ms*  
*uni, uni.A+Pr:ms*  
*universitaire, universitaire.A+Pr:fs*  
*urbain, urbain.A+Pr:fs*

Les noms et adjectifs toponymiques (A+Top) listés dans le dictionnaire Prolintex serviront aussi à trouver la limite droite des noms d'organisation.

La Figure 35 montre un graphe permettant de trouver des noms comme :

... un membre de son cabinet à l'Assemblée nationale avec des fonds ...  
 ... , l'Assemblée permanente des chambres d'agriculture donne ...  
 ... La remise à neuf du Comité économique et social européen ...  
 .... La Commission européenne est saisie de l'affaire. ...  
 ... auprès de la Fédération camerounaise de football accusée d'avoir ...

Le sous-graphe *lst\_org* contient une liste de début de noms d'organisations (ex : *Banque, Société, Groupe* etc.). Le sous-graphe *par\_org* reconnaît un nom de société entre parenthèses.

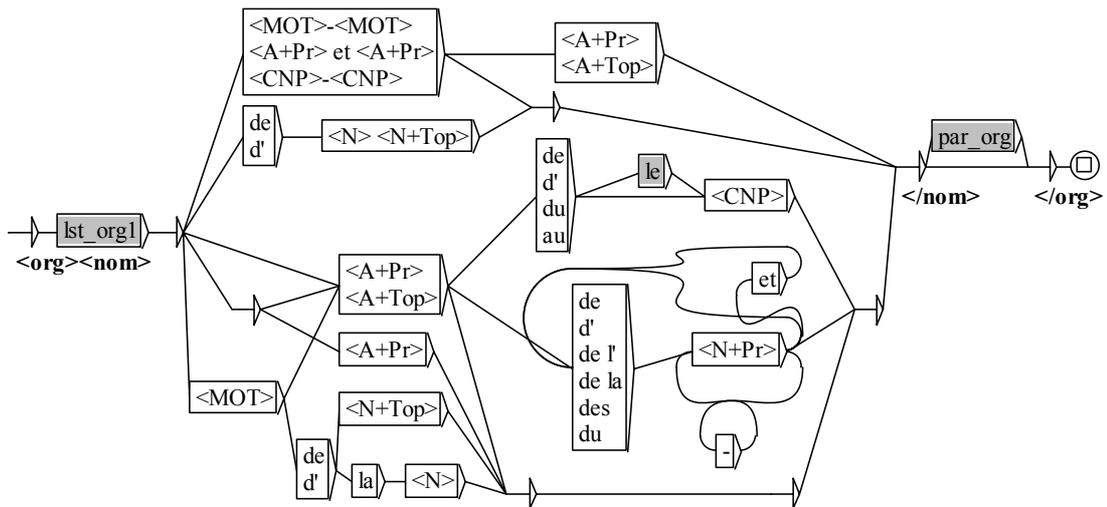


Figure 35 : Graphe simplifié reconnaissant des formes de noms d'organisations à base descriptive

Les noms d'organisations étrangers (Figure 37) sont, par contre, assez simples à trouver : ils portent des majuscules au début de chacun de leurs mots, ils peuvent aussi contenir des mots tels que *of*, *and* (ces mots ne sont absolument pas ambigus avec des mots français).

Exemples :

***Bank of China***

***Banco Santander***

À leur extrémité droite, ces noms propres ont assez souvent une preuve interne du type *Ltd*, *Research* etc. Les noms propres étrangers ne posent pas de problème de limite droite.

Exemples :

***Defense Intelligence Agency***

***WorldCom Inc.***

***Walt Disney Co.***

***Liechtenstein Global Trust,***

***European Language Resources Association***



### 3.2 Preuves externes de noms d'organisations

Les noms d'organisation annoncés par une preuve externe prennent surtout la forme de noms propres purs : ce sont principalement des noms d'entreprise. Leur preuve externe est en général constituée d'un mot tel que *groupe*, *agence*, *société*, etc. et éventuellement d'un adjectif toponymique. Il y environ 15% de noms d'organisation accompagnés d'une telle preuve externe (les contextes droits sont quasi inexistant).

Exemples :

*groupe Amaury*  
*groupe britannique Cable & Wireless*  
*filiale du groupe Saint-Gobain*  
*société Suez-Lyonnaise des eaux*  
*compagnie Airbus*  
*agence Bloomberg*  
*les éditions Hommell*

Les noms d'organisations, précédés de preuve externe, peuvent aussi être à base descriptive (ex : *groupe Démocratie libérale*). Pour améliorer la découverte de la limite droite, nous utilisons aussi le dictionnaire des mots pouvant être trouvés dans les noms d'organisations.

Les sigles sont parfois précédés d'une preuve (ex : *le groupe allemand SPD*) mais le plus souvent aucune preuve ne les accompagne. Un sigle est généralement introduit dans un texte avec sa forme complète donnée entre parenthèse ou vice-versa (Ex : *LDK (Ligue démocratique du Kosovo) ...*)

Une première idée serait de décrire la morphologie des sigles (en français, par exemple, une suite de consonnes en majuscule est forcément un sigle). Cette solution ne convient pas, car les sigles ne sont pas forcément des noms propres ; ils appartiennent au langage courant (ex : *PDG*) ou à la terminologie (ex : *Capes*). Les sigles qui décrivent les noms propres (ex : *SNCF* = Société Nationale des Chemins de Fer, *BHL* = Bernard Henry Lévy, *NY* = New York) sont très divers. Nous utilisons donc, dans un contexte syntaxique très limité, les étiquettes placées par le dictionnaire Sigle-Prolex (décrit au chapitre 4). 16% des noms d'organisations seront finalement découverts ainsi. Les graphes de sigles seront passés après les graphes de noms de personnes, d'organisations et de lieux car il peut y avoir de nombreuses ambiguïtés. S'il n'est pas passé à la fin, ce graphe peut, par exemple, reconnaître *Anne-Marie Comparini* et *UDF* dans *l'élection d'Anne-Marie Comparini (UDF)*<sup>65</sup> comme un nom d'organisation et son sigle : il faut donc reconnaître d'abord les noms de personnes pour éviter ces erreurs.

---

<sup>65</sup> *UDF* n'est pas un sigle désignant *Anne-Marie Comparini*

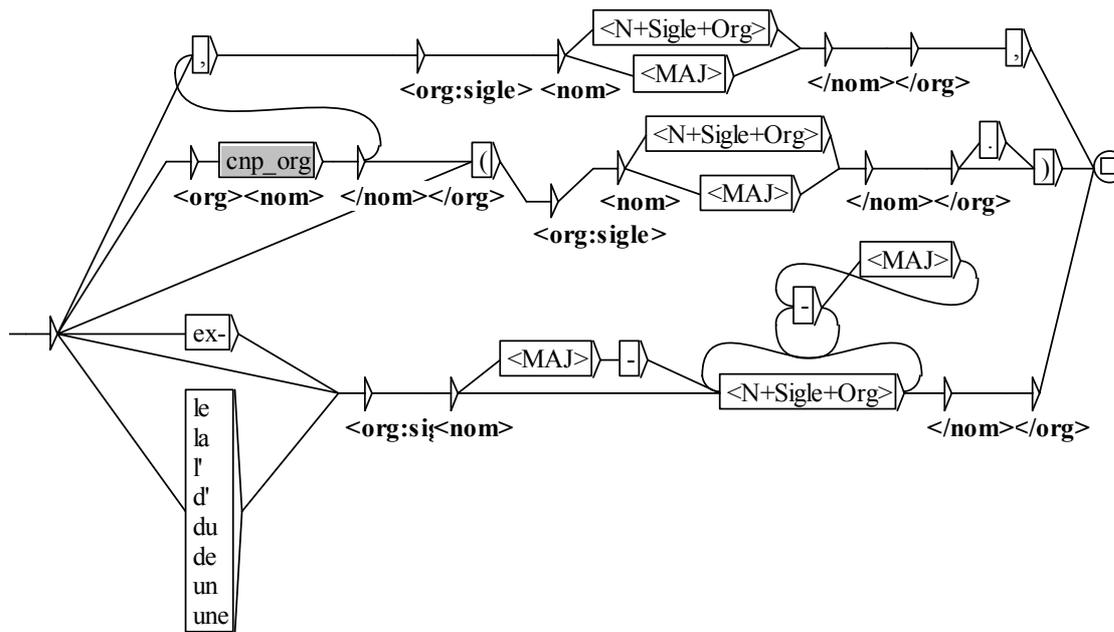


Figure 38 : Graphe reconnaissant les sigles et leur développement s'il est présent

Les coordinations de noms d'organisation sont traitées comme celles des noms de personne avec des amorces au pluriel telles que *sociétés*, *groupes*, etc (voir §2.4 p. 86).

## 4 Les lieux

Comme le font [Grass, Maurel, 2002], nous choisissons de classer comme lieux : les villes, pays, fleuves, montagnes etc. Seuls 20% des noms de lieux ont un contexte gauche et quelques uns une preuve interne. Ceux qui ont une preuve interne sont surtout des noms de ville ou de département (Ex : *Chaumont-sur-Loire* = les tirets et "sur" sont typiques d'un nom de ville) ou des noms de lieux étrangers (Ex : *Trafalgar Square*, *Main Street*, *Yosemite National Park*).

Les graphes appelés par la cascade pour extraire les noms de lieux sont au nombre de 31.

### 4.1 Extraction par preuves externes

Nous avons écrit un certain nombre de graphes permettant de repérer les preuves externes les plus courantes. On peut donner comme exemple, les noms de lieux du graphe de la Figure 39 ou ceux qui sont accompagnés d'un point cardinal à la Figure 40.

La Figure 39 reconnaît par exemple :

*Le département de l'Essonne*

*l'estuaire de la Seine*

*la mer Baltique*

*le mont Sainte-Odile*

*l'océan Arctique*

*la rivière de Tréguier*  
*la vallée de la Vienne*

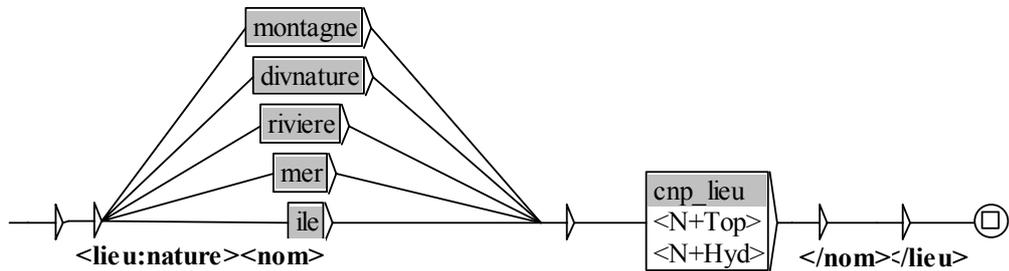


Figure 39 : Reconnaissance des noms géographiques de lieux

Les noms de lieu trouvés par la Figure 40 sont par exemple :

*Afrique du Sud*  
*Asie du Sud-Est*  
*au sud du Sahara*  
*au sud-est de Taipei*  
*Palmerston North*

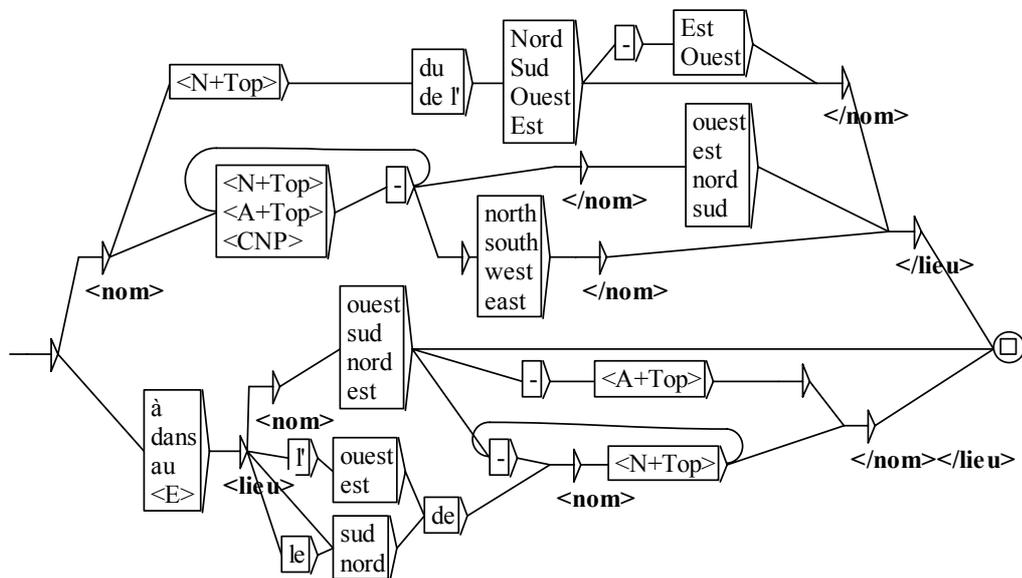


Figure 40 : noms de lieux accompagnés d'un point cardinal

L'écriture de tels graphes est longue pour une quantité de noms propres extraits finalement très faible.

## 4.2 Utilisation des dictionnaires sans preuves

Pour trouver la plus grande partie des noms de lieux mais aussi des gentilés, nous allons principalement utiliser le dictionnaire *Prolintex* de toponymes. Ces graphes seront passés en dernier (après les noms de personnes et d'organisations) pour éviter les ambiguïtés (exemple Figure 41).

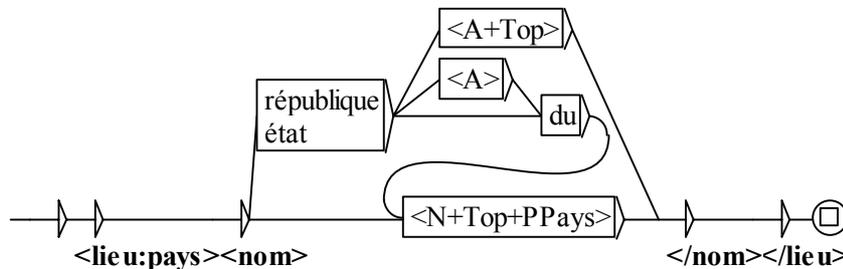


Figure 41 : Graphe reconnaissant des noms de pays

## 5 Résultats

Nous avons vérifié les résultats de notre système sur deux journaux de taille similaire :

- *Le Monde* : 2 numéros de ce journal soit 142 000 mots environ (893 Ko).
- *Ouest France* : une série d'articles correspondant à 152 000 mots environ (915 Ko)

Si ces deux corpus sont similaires en terme de taille, ils sont très différents en terme de contenu puisque *Le Monde* est un journal d'audience nationale voir internationale, alors que *Ouest France* est un grand journal régional. Il faut en outre préciser que la rédaction des articles du *Monde* obéit à des normes de présentation très strictes, ce qui n'est pas le cas de *Ouest France*.

Pour 1 Mo de textes, l'extraction des noms propres prend 801 s (13,33 mn) : ce temps se répartit entre la segmentation en phrases (32 s), l'indexation et le passage des dictionnaires (13,2 s), l'extraction des noms propres (755,2 s) sur un PC de type *Athlon 900 Mhz* doté de *256 Mo de Ram*.

Rappelons que :

- Le rappel est le nombre de noms propres correctement trouvés (et correctement catégorisés) par la cascade de transducteurs divisé par le nombre de noms propres réellement présents dans le texte<sup>66</sup>.
- La précision représente le nombre de noms propres correctement trouvés (et correctement catégorisés) divisé par le nombre de noms propres corrects et incorrects trouvés par la cascade.

Le Tableau 7 et le Tableau 8 résument les nombres d'occurrences des trois entités étudiées dans nos deux corpus (dernière ligne des tableaux).

Le système ExtracNP détecte les noms propres à l'aide de deux cascades de transducteurs successives. Rappelons que la cascade 1 reconnaît les noms propres à l'aide de grammaires locales et que la cascade 2 retrouve des noms propres qui n'ont pas de contextes ou dont le contexte n'est pas décrit par nos grammaires mais qui ont été trouvés dans d'autres contextes par les grammaires de la cascade 1.

Nous présentons les nombres d'occurrences de noms propres qui ont été correctement trouvés par la cascade 1 et la cascade 2 ainsi que la quantité d'erreurs faites par ces deux cascades. Ces valeurs vont nous permettre de calculer le rappel et la précision de notre système.

Bien que le corpus *Ouest France* soit légèrement plus gros que celui du *Monde*, on note que les noms de personnes et noms d'organisations cités sont plus abondants dans *Le Monde*, tandis que les noms de lieux sont plus nombreux dans *Ouest France* : les raisons sont certainement une différence de contenu que nous ne savons expliquer mais en rapport avec les sujets différents traités par les deux journaux.

<i>Le Monde</i>	<b>Personnes</b>	<b>Organisations</b>	<b>Lieux</b>
Quantité d'entités correctement trouvées par la cascade 1	1639	1238	2101
Erreurs de la cascade 1	42	69	111
Quantité d'entités correctement trouvées par la cascade 2	175	160	28
Erreurs de la cascade 2	32	47	14
<b>Quantité total d'entités présentes dans le texte</b>	<b>1936</b>	<b>1588</b>	<b>2208</b>

*Tableau 7 : Résultat de l'extraction en nombre d'occurrences de chaque type de noms propres dans le journal Le Monde*

<sup>66</sup> Les noms propres présents dans le texte sont dénombrés en comptant leurs occurrences à la main et avec l'aide d'Intex.

<i>Ouest France</i>	<b>Personnes</b>	<b>Organisations</b>	<b>Lieux</b>
Quantité d'entités correctement trouvées par la cascade 1	1434	844	3496
Erreurs de la cascade 1	40	36	268
Quantité d'entités trouvées correctement par la cascade 2	107	26	5
Erreurs de la cascade 2	42	3	12
<b>Quantité total d'entités présentes dans le texte</b>	<b>1671</b>	<b>1040</b>	<b>3627</b>

*Tableau 8 : Résultat de l'extraction en nombre d'occurrences de chaque type de noms propres dans le journal Ouest France*

Dans le Tableau 9 et le Tableau 10, nous présentons le rappel et la précision obtenus pour chaque type de noms propres et pour les deux étapes de notre système d'extraction, ainsi que les résultats cumulés par les deux cascades.

Le Tableau 11 donne le résultat global du système sur les journaux *Le Monde* et sur *Ouest France*.

## 5.1 Résultats de l'extraction des noms de personne

Sur *Le Monde*, nous obtenons 84,6% de rappel avec une précision de 99,4% grâce à la première cascade de transducteurs. Cela signifie que nos grammaires et notre dictionnaire de prénoms couvrent très largement les occurrences de noms de personnes présents dans les journaux (les résultats sont équivalents sur le journal *Ouest France*).

La seconde cascade de transducteurs permet de retrouver 6,4% de noms de personnes pour *Ouest France* et 9,0% pour *Le Monde* avec une précision beaucoup plus faible (resp. 71,8% et 84,5%). Nous avons appliqué notre cascade sur l'ensemble des textes d'un journal, au lieu de l'appliquer sur les articles un à un, cela aurait conduit à une précision plus forte de la deuxième cascade et un rappel plus faible.

<i>Le Monde</i>	<b>Personnes</b>	<b>Organisations</b>	<b>Lieux</b>
<b>Cascade 1</b>			
Rappel	84,7	78	95,2
Précision	97,5	94,7	95,0
<b>Cascade 2</b>			
Rappel	9,0	10,1	1,3
Précision	84,5	77,3	66,7
<b>Résultats cumulés des deux cascades</b>			
Rappel	93,7	88,0	96,4
Précision	96,1	92,3	94,5

*Tableau 9 : Rappel et précision pour le journal Le Monde (en %)*

<i>Ouest France</i>	Personnes	Organisations	Lieux
<b>Cascade 1</b>			
Rappel	85,8	81,2	96,4
Précision	97,3	95,9	92,9
<b>Cascade 2</b>			
Rappel	6,4	2,5	0,1
Précision	71,8	89,7	29,4
<b>Résultats cumulés des deux cascades</b>			
Rappel	92,2	83,7	96,5
Précision	94,9	95,7	92,6

Tableau 10 : Rappel et précision pour le journal *Ouest France* (en %)

	<i>Le Monde</i>	<i>Ouest France</i>
<b>Avec la cascade 1 seule</b>		
F-mesure	91,1	92,7
<b>Avec les deux cascades</b>		
Rappel global	93,2	93,3
Précision globale	94,4	93,6
F-mesure	93,8	93,5

Tableau 11 : Rappel, précision et F-mesure globaux pour les deux journaux (en %)

Dans ces journaux, les noms de personnes restants sont pour la plupart cités sans leur prénom ni une quelconque preuve externe (ex : *Car ce qui rend furieux Santini, c'est le fait que des gens puissent être abusés ...* (*Ouest France*)).

Le rappel total des noms de personnes est 93,7% sur *Le Monde* et 92,2% sur *Ouest France* pour des précisions resp. de 96,1% et 94,9%.

## 5.2 Résultats de l'extraction des noms d'organisations

Le résultat obtenu sur les noms d'organisations est beaucoup moins bon : nos grammaires et notre dictionnaire de sigles ne permettent d'en repérer qu'environ 80% dans la cascade 1. En fait, quasiment un quart des noms d'organisations trouvés le sont grâce à notre dictionnaire des sigles. La précision est d'environ 95% pour cette étape, elle chute lors de la seconde cascade (77,3% pour *Le Monde*, 89,7% pour *Ouest France*).

Lors de la seconde cascade, on retrouve 10,1% d'organisations dans *Le Monde* mais seulement 2,5% pour *Ouest France*.

Finalement, l'extraction des organisations dans le journal *Le Monde* obtient un rappel de 88,0% tandis qu'on obtient 83,6% dans *Ouest France*. Comme les autres systèmes d'extraction, nous nous heurtons aux difficultés d'extraction des noms d'organisations (à cause de leur limite droite ou par manque de preuve).

### 5.3 Résultats de l'extraction des noms de lieux

Les noms de lieux sont les noms propres sur lesquels nous obtenons les meilleurs résultats ; pourtant ce sont eux qui ont le moins de preuves internes ou externes. C'est grâce au dictionnaire Prolintex extrêmement complet que nous réussissons à les extraire à hauteur de 95,15% pour *Le Monde* et 96,39% pour *Ouest France* dans la première étape de l'extraction. Dans *Ouest France*, les noms de lieux accompagnés d'un contexte sont beaucoup plus nombreux que dans *Le Monde*.

La seconde cascade de transducteurs ne retrouve quasiment aucun nom de lieu (1,27% pour *Le Monde* et 0,14% dans *Ouest France*) et avec une précision très faible.

Le rappel total sur *Ouest France* et *Le Monde* est de 96,5% pour une précision de 94,45% pour *Le Monde* et 92,59% pour *Ouest France*. La précision est un peu faible car la recherche des noms de lieux est réalisée aux trois quarts par des dictionnaires seuls. Par conséquent, un certain nombre de mots ambigus avec un nom propre sont trouvés par le dictionnaire (ex : *Trente* = ville italienne ou chiffre).

Contrairement aux résultats obtenus sur les noms de personnes et d'organisations (§5.1 et §5.2), les bons résultats de l'extraction des noms de lieux tiennent essentiellement aux travaux antérieurs (à ceux présentés ici) du projet Prolex.

## 6 Tous les noms propres

Finalement, notre système ExtracNP permet d'extraire 93,2% des noms propres dans *Le Monde* et 93,3% dans *Ouest France* avec respectivement une précision de 94,4% et 93,6%. La F-mesure est de 91,1% pour *Le Monde* et 92,7% pour *Ouest France* avec la première cascade seule. La seconde cascade apporte une bonne amélioration aux résultats de l'extraction : la F-mesure est de 93,8% pour *Le Monde* et 93,5% pour *Ouest France*.

Par rapport à l'ensemble des résultats affichés par les systèmes anglais et français présentés au Chapitre 2, notre système obtient les meilleurs résultats actuels sur le français. Ces résultats pourraient être améliorés par un complément aux grammaires locales car, en fait, la majeure partie des noms propres qui n'ont pas été trouvés ne peuvent être trouvés grâce à une méthode linguistique. Bien sûr, nous pouvons extraire plus de noms propres grâce à la morphologie et même la syntaxe mais leur catégorisation n'est pas possible : nous pouvons seulement dire que tel ou tel nom propre est un nom de personne ou d'organisation plutôt qu'un lieu.

La deuxième étape de la cascade permet de retrouver 6% de noms propres dans *Le Monde* et 2% dans *Ouest France* mais la précision s'en ressent.

Pour ce qui est des autres types de noms propres (pragmonymes, ergonymes [Grass, Maurel, 2002], voir p. 13 et 14), leur quantité est tellement faible dans les journaux qu'il ne semble pas raisonnable d'écrire des grammaires permettant de les catégoriser. Un système automatisé serait plus adapté pour eux.

Nous avons, par contre, écrit une grammaire des événements sportifs (ex : *Jeux Olympiques*, *Tournoi*, etc.) et des festivals (ex : *Festival des Vieilles Charrues*, *Salon de l'Automobile*, etc.), et une grammaire reconnaissant les prix ou indices boursiers très courants dans les journaux (ex : *prix Goncourt*, *indice Nikkei*).

Les erreurs de notre système sont principalement :

- des erreurs de catégorisation dues à des mots fortement ambigus ou à des séquences ambiguës :

`<person> <prenom> France </prenom> <nom> Télévision </nom>`  
`</person> a démenti ...`

v.s.

`... <person> <prenom> France </prenom> <nom> Galle </nom>`  
`</person> à Lyon`

- des erreurs de sous-reconnaissance :

`<person> <prenom> Valéry </prenom> <nom> Giscard </nom>`  
`</person> d'Estaing est élu par ses administrés.`

v.s.

`<person> <prenom> André </prenom> <nom> Wiltzer </nom>`  
`</person> d'Haironville désigne le coupable ...`

ou

`<org> <nom> Commission nationale de prévention des nuisances </nom>`  
`</org> sonores`

v.s.

`une <org> <nom> Organisation mondiale du commerce </nom> </org>`  
`performante`

Un étiquetage syntaxique juste du texte semble inutile pour extraire les noms propres car c'est le lexique qui permet de trouver et de catégoriser les noms propres avec une très grande précision.

L'inévitable incomplétude de la grammaire explique aussi une partie des erreurs ou des réponses manquantes.

## 7 Conclusion

Le principe de la cascade de transducteurs est assez simple et efficace ; par contre, les combinaisons et interactions possibles sont nombreuses. La grammaire des noms de personnes est très structurée ; les autres noms propres sont plus difficiles à repérer. Les preuves externes et internes des noms d'organisation sont beaucoup plus variées. Tandis que les noms de lieux nécessitent un dictionnaire pour les repérer car ils sont trop peu fréquemment accompagnés d'une preuve dans les textes.

Le rappel dépasse 93% et la précision 94% malgré la deuxième cascade de transducteurs et son manque de précision.

Pour reconnaître les noms propres, les méthodes linguistiques ont leurs limites :

- Incomplétude des grammaires locales,
- Ambiguïtés,
- Absence de contextes.

On peut remarquer que les grammaires d'extraction des noms propres respectent la loi de Zipf : en effet, un petit nombre de règles s'appliquent fréquemment et assurent une

bonne couverture mais de nombreuses règles supplémentaires sont nécessaires pour améliorer le rappel.

Les noms propres sont très importants pour les systèmes d'extraction d'information ; ils apportent une information importante sur le sens et le contenu des textes. Un texte parlant, par exemple, de *Bill Clinton* et de *Monica Lewinski* évoque rapidement une affaire qui a défrayé la chronique il y a quelques années. Dans la suite de notre travail, nous voulons tenter d'évaluer le pouvoir classifieur des noms propres par rapport aux autres mots.

## Chapitre 6

# QUEL ROLE POUR LES NOMS PROPRES DANS LA CLASSIFICATION DE TEXTES ?

---

Le domaine de l'extraction d'information fait un grand usage des techniques de traitement automatique de textes ; par contre, la recherche d'information qui obtient de bons résultats grâce à des méthodes non linguistiques ne s'intéresse à ces aspects que depuis une décennie. Différents travaux tentent d'améliorer l'efficacité des systèmes en proposant des représentations plus complexes des textes.

Les noms propres ont été largement étudiés dans le domaine de l'extraction d'information et nous pensons qu'ils peuvent aussi jouer un rôle dans les systèmes de recherche d'information. Leur quantité et leur qualité informationnelle dans les journaux semblent les rendre pertinents pour faire de la classification non supervisée [Smeaton *et al.*, 1998].

Le but de ce travail n'est pas de créer un système de classification ou de recherche documentaire<sup>67</sup> mais d'évaluer la qualité des noms propres, par rapport aux autres mots, comme classificateurs de textes. Les noms propres pourraient alors être associés à un processus de classification utilisant d'autres aspects "linguistiques".

Dans ce chapitre, nous présentons les mesures de similarité que nous avons choisies pour tester la pertinence des noms propres par rapport aux autres mots (cf. §1 et §2). Puis, nous expliquons comment nous avons évalué les résultats des classifications obtenues avec ces mesures (cf. §3, §2 et §5).

## 1 Représentation des textes

La classification non supervisée (*clustering*) crée des groupes de textes en utilisant des similarités entre chaque texte. Pour calculer ces similarités, il faut, avant tout, choisir les "unités textuelles" sur lesquelles se basera la représentation du texte.

### 1.1 Le modèle vectoriel

Nous avons choisi de travailler sur le modèle vectoriel<sup>68</sup> [Salton, 1975] classique dans le domaine de la classification. Dans ce modèle, un document est représenté par un

---

<sup>67</sup> C'est le projet international TREC (Text Retrieval Conference) lancé au début des années 90 par le NIST aux USA qui permet d'évaluer les systèmes de recherche documentaire.

<sup>68</sup> D'autres modèles de représentation existent ; par exemple, le modèle LSI (Latent Semantic Indexing) qui est une variante du modèle vectoriel et qui prend en compte la sémantique des unités linguistiques représentées par leurs dépendances cachées.

ensemble<sup>69</sup> d'unités linguistiques qui peuvent être, par exemple, des mots, des lemmes (ex : *vendre, bourse*) ou des séquences de mots (ex : *président français*).

Les termes contenus dans les vecteurs sont associés à leurs fréquences d'apparition dans les textes : cette fréquence est souvent utilisée pour la définition d'une mesure de similarité. [Riloff, Lehnert, 1994] soulignent que les problèmes soulevés par cette représentation proviennent de la synonymie et de la polysémie qui ne sont pas prises en compte. Deux mots peuvent être comptés ensemble alors qu'ils n'ont pas forcément le même sens. Pour des textes longs, cet amalgame est moins significatif car plus de mots sont en commun.

Un autre problème est que les groupes de mots (les textes) n'ont pas le même sens que les mots du groupe pris séparément, autrement dit "*Le sens du tout est-il calculable à partir du sens des parties ?*" [Besançon, 2001]. Il y a donc réduction de l'ambiguïté des mots du groupe lorsque l'unité linguistique d'un vecteur est le groupe de mots [Krovetz, Croft, 1992]. Ainsi, plusieurs études linguistiques se développent récemment sur la notion de collocation [Daille, Williams, 2001].

Les avis sont variés quand on parle de créer une représentation plus linguistique. Pour [Riloff, Lehnert, 1994], utiliser l'extraction d'information pour classer des textes est une voie intéressante. En extrayant des informations spécifiques, on évite les portions de textes qui ne sont pas pertinentes. Par exemple, dans le domaine du terrorisme, on peut extraire les auteurs des actes terroristes, les victimes, les armes utilisées, le lieu de l'attentat etc. Ainsi, un texte qui a instancié un formulaire d'extraction d'information sur le terrorisme est pertinent pour ce sujet, sinon il ne l'est pas. Ces techniques obtiennent des scores de rappel relativement bas car les documents ne contiennent pas tous assez d'informations pertinentes et ne sont donc pas correctement classés alors qu'ils appartiennent à une catégorie bien définie. [Voorhees, 1999] dit que, jusqu'alors, les traitements linguistiques n'étaient pas très appropriés à la recherche documentaire. La faible amélioration viendrait du fait que les méthodes non-linguistiques exploitent implicitement la même information que les méthodes linguistiques qui le font explicitement. Les systèmes de recherche d'information non linguistiques obtiennent de très bons résultats, mais leur amélioration pourrait passer, semble-t-il, par l'introduction de notions linguistiques.

## 1.2 Nos représentations des textes

Nous avons créé trois représentations des vecteurs des textes pour nos expériences :

- a. les mots lemmatisés (y compris les noms propres),
- b. les noms propres<sup>70</sup> associés à leurs types,
- c. les mots lemmatisés sans les noms propres.

Nous allons utiliser ces différentes représentations pour mesurer des similarités entre textes et pour tester, grâce à ces mesures, les différentes classifications obtenues (en ce qui concerne la lemmatisation des mots et la préparation des vecteurs, voir l'Annexe C p. 125).

---

<sup>69</sup> Souvent appelé "sac de mot".

<sup>70</sup> Nous pensons que la quantité de noms propres extraits par notre système d'extraction extracNP est suffisante (94% de rappel) pour pouvoir être utilisée.

## 2 Pondération de termes et mesures de similarité

Il existe trois principaux types de mesures de similarité pour le modèle vectoriel : des mesures ensemblistes (Dice, Jaccard, etc.), des mesures géométriques (Cosine, Euclidienne, etc.), des mesures de type distributionnel (mesure du  $\chi_2$ , etc.).

Nous présentons ici la mesure de Jaccard et la mesure Cosine associée à une pondération TF.IDF.

### 2.1 La mesure ensembliste Jaccard

Les mesures ensemblistes utilisent l'information de l'absence ou de la présence d'un terme dans un texte. Nous avons choisi de tester la mesure de Jaccard qui, intuitivement nous semble une mesure intéressante pour les noms propres. Nous pensons en effet qu'étant donné la qualité informationnelle des noms propres, la mesure Jaccard qui ne prend en compte que le nombre de noms communs à deux textes sera peut être suffisante pour faire de la classification avec les noms propres.

Jaccard est simplement le nombre de mots communs contenus dans deux textes  $d_i$  et  $d_j$  divisé par le nombre de mots contenus dans leur union (Formule 5).

$$Jaccard(d_i, d_j) = \frac{|d_i \cap d_j|}{|d_i \cup d_j|}$$

*Formule 5 : Mesure Jaccard*

### 2.2 Mesure de similarité avec pondération TF.IDF des termes

Pour vérifier que les noms propres peuvent améliorer la qualité des mesures de similarité, nous avons choisi la mesure TF.IDF très utilisée et reconnue [Voorhees, 1999], décrite dans [Salton, Buckley, 1988].

La mesure TF.IDF est composée de deux parties :

- Une pondération locale :  $TF_{ik}$  (Term Frequency) est la fréquence du terme  $T_k$  dans le texte  $d_i$ . Autrement dit, il s'agit du nombre de fois qu'un terme apparaît dans un document. Plus un terme apparaît dans un document, plus il semble devoir être pris en compte. Cette fréquence, utilisée seule, favorise les longs documents.
- Une pondération globale :  $IDF_k$  (Inverse Document Frequency) est la fréquence inverse du terme  $T_k$  dans la collection  $C$  où  $N$  est le nombre de textes dans la collection  $C$  et  $n_k$  est le nombre de documents contenant au moins une occurrence du terme  $T_k$  (Formule 6).  $IDF$  permet de favoriser les termes concentrés dans quelques documents. La pondération globale dépend de la collection entière des documents.

$$idf_k = \log\left(\frac{N}{n_k}\right)$$

*Formule 6 : Inverse Document Frequency*

Le poids  $w_{ik}$  d'un terme est donné par la Formule 7.

$$w_{ik} = tf_{ik} \cdot idf_k$$

*Formule 7 : Poids  $w_{ik}$  d'un terme  $k$  dans le document  $i$  par TF.IDF*

Une valeur élevée de TF.IDF indique qu'un mot apparaît fréquemment dans un document et peu dans les autres.

Avant normalisation, les pondérations locales et globales ne prennent pas en compte la longueur du document, ce qui entraîne deux problèmes [Besançon, 2001] :

- Les documents les plus longs utilisent les mêmes mots de façon répétée donc les similarités sont plus grandes entre des documents plus longs.
- Les documents les plus longs ont une thématique plus variée et utilisent plus de termes distincts ce qui peut augmenter le nombre de correspondance de termes entre deux textes.

Finalement, la similarité normalisée entre deux textes est donnée par la Formule 8.

$$\text{sim}(d_i, d_j) = \text{cosine}(d_i, d_j) = \frac{\sum_{k \in i \cap j} w_{ik} \cdot w_{jk}}{\sqrt{\left(\sum_{k \in i} tf_{ik} \cdot idf_k^2\right) \cdot \left(\sum_{k \in j} tf_{jk} \cdot idf_k^2\right)}}$$

*Formule 8: Similarité entre deux textes  $d_i$  et  $d_j$*

### 2.3 Fusion de données

[Fox, 1983] représente plusieurs aspects différents des documents d'une collection : le vecteur du texte est alors un ensemble de sous-vecteurs correspondant chacun à un aspect différent des documents de la collection. La similarité de deux textes est une somme pondérée des similarités de chaque sous-vecteur (Formule 9).

$$\text{sim}(d, d') = \sum_i \alpha_i \cdot \text{sim}_i(d, d')$$

*Formule 9: Similarités fusionnées*

[Fox, 1995] expérimente une fusion à trois sous-vecteurs : les mots non trouvés par Wordnet ou non désambiguïsés, les ensembles de mots désambiguïsés, les noms synonymes.

[Shaw, Fox, 1995] crée des combinaisons de mesures (par sommation pondérée de trois sous-vecteurs : les mots non trouvés par Wordnet ou non désambiguïsés, les

ensembles de mots désambiguïsés, les noms synonymes), ce qui donne de meilleurs résultats que la sélection d'une des similarités calculées.

[Bellot, 2000] propose une combinaison linéaire de mesures de similarité (Cosine, Okapi et le nombre de mots communs) mais il semble que l'amélioration apportée ne soit ni évidente ni significative.

Nous avons testé des combinaisons linéaires de mesures de similarité avec nos trois représentations de textes différentes ; chacune devenant un sous-vecteur. Les combinaisons que nous avons testées sont :

- a. soit le sous-vecteur des mots (dont la mesure de similarité est  $sim_{mots}$ ) avec le sous-vecteur des noms propres (similarité  $sim_{np}$ ) d'un même texte,
- b. soit le sous-vecteur des mots seuls (mesure de similarité  $sim_{mots\_seuls}$ ) avec le sous-vecteur des noms propres.

La fusion de similarités de ces deux sous-vecteurs est donnée par la Formule 10 où  $\alpha$  et  $\beta$  sont les pondérations des deux sous vecteurs (au cours de nos expériences nous faisons varier les valeurs de  $\alpha$  et  $\beta$ ).

$$sim(d, d') = \alpha \cdot sim_{mots}(d, d') + \beta \cdot sim_{np}(d, d')$$

*Formule 10 : Mesure de similarité fusionnée entre sous-vecteurs de mots et de noms propres*

## 2.4 Heuristiques liées aux noms propres pour le calcul de similarité

Nous avons testé des heuristiques qui permettent de prendre en compte les phénomènes de coréférence des noms propres dans le calcul des mesures de similarité. Ces heuristiques concernent les noms de personnes et les noms d'organisations.

Nous utilisons les noms propres extraits par notre système ExtracNP (voir Chapitre 5).

Dans un article de journal, les personnes sont en général nommées au moins une fois avec leur prénom puis ce prénom n'est plus précisé dans la suite de l'article. Nous considérons que, dans un même article, pour un même nom de personne, si le prénom n'est plus précisé il s'agit de la même personne : un journaliste ne citerait pas deux personnes différentes dans le même article sans les distinguer par leur prénom.

De plus, si deux personnes qui ont le même nom de famille mais deux prénoms différents sont citées dans un même texte (ex : *Jacques Chirac* et *Bernadette Chirac*), la fréquence associée au mot *Chirac*, trouvé sans le prénom, est partagée pour la raison suivante : si dans un texte *Jacques Chirac* et *Bernadette Chirac* sont cités ensemble, on ne peut savoir, sans connaissances pragmatiques, si *président Chirac* réfère à *Jacques* ou à *Bernadette*.

Pour ce qui est de la mesure de similarité entre deux textes, elle est calculée de la manière suivante lorsqu'on est en présence des noms de personnes : on compare les noms de famille puis les prénoms (s'ils sont connus), si les prénoms sont différents, ce ne sont pas les deux mêmes personnes.

Notre système d'extraction reconnaît les noms d'organisations, leurs variantes et leurs formes abrégées si elles sont précisées dans un même texte. Par exemple, le terme

*Organisation des Nations Unies* est équivalent à *ONU* alors dans le vecteur de termes *Organisation des Nations Unies* et *ONU* ne sont qu'un seul et même terme.

Nous avons pu remarquer que ces heuristiques, sur des jeux de tests spécialement créés pour contenir des noms de personnes homonymes avec des prénoms différents, les résultats obtenus avec l'heuristique sont bien meilleurs que sans. Par contre, sur des ensembles de textes ne contenant pas de personnes homonymes, ces heuristiques n'apportent rien au résultat de la classification.

### 3 Comment évaluer nos mesures de similarités ?

#### 3.1 Corpus de tests

L'évaluation nécessite l'étude d'un corpus avec une classification connue.

C'est le projet **Amaryllis** qui va nous permettre d'obtenir un tel corpus. Amaryllis est un programme français de recherche d'information équivalent à TREC et lancé en 1996. Son objectif est de "*promouvoir l'élaboration de corpus et de procédures d'évaluation concernant le français, pour permettre à la recherche de progresser et au domaine de se doter d'instruments de mesure rendant possible une comparaison objective des différentes approches.*" [Savoy, 2000].

Chaque campagne d'évaluation propose un certain nombre de requêtes et des corpus de textes (*Le Monde*, *INIST*, etc.). Les systèmes participants doivent trouver les textes du corpus qui répondent aux requêtes proposées. Afin d'évaluer les résultats des différents systèmes, les réponses de chaque requête sont construites par des experts humains ; les résultats proposés par les systèmes participants sont ensuite comparés aux réponses construites à la main et permettent de les réviser.

Grâce à ces textes et ces requêtes, nous avons pu créer un corpus classé en utilisant l'hypothèse des groupes ("*cluster hypothesis*") formulée par [van Rijsbergen, 1979] :

*"Closely associated documents tend to be relevant to the same requests".*

En utilisant cette hypothèse, nous avons construit des classes de textes avec les résultats pertinents aux thèmes des requêtes d'Amaryllis sur les corpus OFIL1 et OFIL2 (articles du journal *Le Monde*).

Nous prenons et mélangeons ces classes au hasard pour obtenir plusieurs collections de textes à grouper<sup>71</sup>.

#### 3.2 Classification par partitionnement (*clustering*)

Il existe deux types de classification :

- c. Le *clustering* ou *partitionnement* : il s'agit de trouver une partition, c'est-à-dire des regroupements, d'un ensemble de documents.
- d. La *catégorisation* : il s'agit de répartir un ensemble de documents dans des catégories (ou classes) prédéfinies.

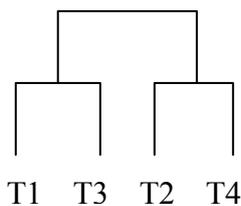
---

<sup>71</sup> Nous avons écrit un programme qui automatise la création de groupes de textes mélangés et retourne un fichier contenant la classification idéale de chacun de ces groupes.

Le partitionnement groupe des documents similaires sans connaissance de classes *a priori*. Nous avons réalisé cette classification avec l'algorithme de classification ascendante<sup>72</sup> hiérarchique (CAH). Cette méthode construit un arbre (ou dendrogramme) dans lequel les textes les plus proches sont placés dans les mêmes branches. Cette méthode de classification très populaire et très robuste a fait ses preuves depuis longtemps et fournit en général une sortie de très grande qualité [Voorhees, 1986], [Willet, 1988]; elle est très utilisée en recherche documentaire. Cependant les algorithmes de CAH sont lents lorsqu'ils sont appliqués sur une très grande collection de textes. Leur complexité en temps et en mémoire [Bellot, El-Bèze, 2001] est en  $O(n^2)$  (par exemple, un ensemble de 500 000 documents donnent une matrice  $(n^2-n)/2$  à calculer et demande 500 Go de place). Cette lenteur n'est pas un problème puisque nous voulons avant tout évaluer des mesures de similarités.

Le principe de la classification hiérarchique ascendante est de regrouper à chaque itération les deux textes qui sont les plus proches. En voici l'algorithme :

1. Calculer toutes les similarités entre documents.
2. Au départ, chaque document est un groupe (aussi appelé cluster).
3. Fusionner les paires de groupes les plus proches.
4. Mettre à jour la matrice en éliminant les colonnes et lignes fusionnées.
5. Retourner à 3 si il reste plus d'un cluster.



Exemple : soit les textes T1, T2, T3, T4, la matrice de similarité correspondante est donnée par la Figure 42. Par exemple les textes T1 et T3 (ligne 1, colonne 3) ont une similarité de 0,8. Un texte et lui-même a une similarité de 1.

T1 et T3 sont fusionnés car ce sont les deux documents les plus proches de la matrice ; la matrice est mise à jour (Figure 43) et les colonnes 1 et 3 sont fusionnées en une seule. Puis T2 et T4 sont fusionnés car ils ont la plus forte similarité. Enfin, les groupes {T1, T3} et {T2, T4} sont fusionnés.

	T1	T2	T3	T4
T1	1	0.2	0.8	0.3
T2	0.2	1	0.4	0.7
T3	0.8	0.4	1	0.1
T4	0.3	0.7	0.1	1

Figure 42 : Matrice des similarités des 4 textes

<sup>72</sup> Des méthodes non-hiérarchiques existent : K-Means, K-Medoids, etc. Elles sont beaucoup plus rapides que les CAH mais le résultat final dépend de la partition initiale choisie. Des combinaisons de méthodes hiérarchiques et non-hiérarchiques existent, par exemple, Buckshot [ Cutting et al., 1992]

	T1,T3	T2	T4
T1,T3	1	0.2	0.1
T2	0.2	1	0.7
T4	0.1	0.7	1

*Figure 43 : Matrice des similarités après la première fusion*

Les différentes stratégies de fusion des textes dont les plus connues sont :

- a. *Single Link* (lien minimal) : La similarité du groupe fusionné est égale à la similarité des deux membres les plus similaires de ces groupes (Figure 45).
- b. *Group Average Link* : La distance entre deux groupes est la moyenne arithmétique de toutes les distances séparant un document du premier groupe d'un document du second. Selon [Willett, 1988], c'est la meilleure méthode pour donner une hiérarchie de bonne qualité.
- c. *Méthode de Ward* : À chaque étape, on regroupe les classes dont le regroupement minimise l'inertie intra-classe [Ward, 1963]. Cette méthode conduit à des classes homogènes et à une hiérarchie symétrique [Rasmussen, 1992].
- d. *Complete Link* (lien maximal) : La similarité du groupe fusionné est égale à la similarité des deux membres les moins similaires de ces groupes (Formule 11). Ce critère crée des groupes très étroits et limités (Figure 44), assez compacts et faciles à distinguer [Zamir *et al.*, 1997].

Nous avons choisi d'utiliser le critère du *lien maximal* ou *Complete Link* pour fusionner les groupes de textes entre eux.

$$\text{sim}(c_i, c_j) = \min_{x \in c_i, y \in c_j} \text{sim}(x, y)$$

*Formule 11 : Critère Complete Link*

[Voorhees, 1986] a étudié les trois méthodes *a*, *b*, et *c*, et conclut que les résultats sont similaires sur de petites collections. Par contre, sur de plus grosses collections de textes, la méthode *Complete Link* est la meilleure mais la plus chère en temps, tandis que la méthode *Group Average* est meilleure en temps. [Smeaton *et al.*, 1998] préfèrent aussi la méthode *Complete Link* qui produit des groupes étroits ce qui semble adapté aux textes tels que des articles de journaux dont les sujets sont très nombreux. [Zamir, Etzioni, 1999] disent aussi que *Complete Link* produit des groupes dans lesquels tous les documents sont très proches les uns des autres. Alors que les méthodes *Single Link* et *Group Average* produisent des groupes sans fin : en effet, avec ces critères (surtout *Single Link*), le texte le plus similaire aux textes déjà placés dans la hiérarchie est ajouté à l'arbre ce qui produit un effet d'échelle (voir la Figure 45) ; de plus, avec ces deux dernières méthodes, deux documents peuvent être dans le même cluster alors qu'ils ont une similarité très faible.

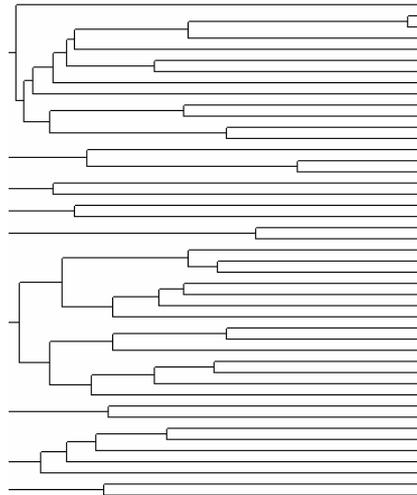


Figure 44: Arbre obtenu avec le critère Complete Link

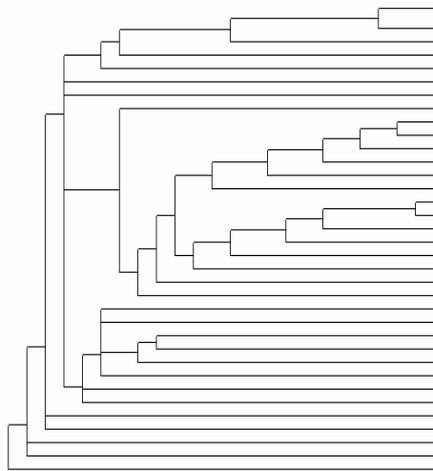


Figure 45 : Arbre obtenu avec le critère Single Link

Nous allons comparer les résultats obtenus par nos différentes mesures de similarités avec la classification hiérarchique ascendante utilisant le critère *Complete Link*.

#### 4 Comment comparer les classifications hiérarchiques obtenues ?

Il est difficile de trouver une mesure objective de la qualité des groupes obtenus. [Moore *et al.*, 1997], [Boley, 1998], [Steinbach *et al.*, 2000], [Strehl *et al.*, 2000], [Zhao, Karypis, 2001] ont choisi, à travers leurs travaux de classification, d'utiliser la mesure d'entropie et la mesure de pureté<sup>73</sup> pour évaluer la qualité de leurs résultats.

L'entropie est une mesure du désordre empruntée aux physiciens. Nous utilisons la mesure de l'entropie d'une classification proposée dans [Strehl *et al.*, 2000] et [Zhao, Karypis, 2001]. Chaque texte est étiqueté avec le numéro de la classe de référence à laquelle il appartient. Nous comparons cette référence aux résultats que nous obtenons.

<sup>73</sup> Ces mesures sont calculées à nombre de classe égal.

L'entropie  $e_c$  d'un groupe de textes  $c$  ( $k$  est le nombre de classes de références) est calculée par la Formule 12.  $card(i,c)$  est le nombre d'occurrences de la classe  $i$  dans le cluster  $c$ . L'entropie  $e_c$  est nulle quand tous les textes appartenant à  $C$  sont dans la même classe de référence, sinon l'entropie est supérieure à 0 ( $0 \leq e_c \leq 1$ ). L'entropie grandit avec le désordre des textes, donc plus l'entropie est basse, plus la qualité des regroupements est grande.

$$e_c = -\frac{1}{\log k} \sum_{i=1}^k \left( \frac{card(i,c)}{\sum_{i=1}^k card(i,c)} \log \left( \frac{card(i,c)}{\sum_{i=1}^k card(i,c)} \right) \right)$$

*Formule 12 : Entropie d'un groupe de textes*

L'entropie totale  $e_T$  des groupes de textes est la somme des entropies des différents groupes pondérée par le nombre de textes de chaque groupe sur le nombre de textes total (Formule 13) Dans cette formule,  $q$  est le nombre de groupes trouvés,  $n$  est le nombre total de documents et  $n_c$  le nombre de textes dans le groupe  $c$ .

$$e_T = \sum_{c=1}^q \frac{n_c \cdot e_c}{n}$$

*Formule 13 : Entropie totale d'un ensemble de groupes*

La pureté des clusters est une sorte de pendant de l'entropie. La pureté mesure si un groupe contient plutôt des documents qui viennent d'une même classe [Zhao, Karypis, 2001]. La pureté est donnée par la Formule 14.

$$P_c = \frac{1}{\sum_{i=1}^k c(i,c)} \max_i c(i,c)$$

*Formule 14 : Pureté d'un groupe de textes*

Plus la pureté est élevée (proche de 1), plus la classification est de bonne qualité.

La pureté totale est la somme pondérée des puretés des différents groupes.

$$P_T = \sum_{c=1}^q \frac{n_c \cdot P_c}{n}$$

*Formule 15 : Pureté totale*

Pour [Strehl *et al.*, 2000], l'entropie est une mesure plus compréhensible que la pureté car elle considère la distribution entière plutôt que les textes qui sont ou ne sont pas dans les classes les plus fréquentes.

Pour réaliser les calculs d'entropie et de pureté, il faut découper les arbres hiérarchiques en groupes ; en général, on coupe un arbre en choisissant un seuil de similarité ou un nombre de classes. Nous préférons choisir la deuxième solution qui va nous permettre d'évaluer la qualité des classifications pour un même nombre de classes.

Les arbres obtenus par une classification sur le critère maximal permettent d'obtenir des groupes très étroits sans liens les uns avec les autres ; nous entendons par-là que l'arbre hiérarchique est constitué de plusieurs arbres déconnectés les uns des autres car les similarités entre ces arbres valent 0 lorsqu'on se rapproche de la racine : il y a en fait plusieurs racines qui sont chacune un groupe de textes (revoir la Figure 44).

Lorsque nous comparons les classifications obtenues, nous le faisons à nombre égal de classes mais ce nombre de classes peut varier suivant les groupes de textes et suivant le type de similarité utilisé. Par exemple, les similarités utilisant les noms propres créent énormément de classes car de nombreux textes ont des similarités qui valent 0 (il y a beaucoup moins de noms propres par textes que de mots, c'est ce qui explique ce phénomène). Pour environ 200 textes classés par les noms propres, notre classification obtient une quarantaine d'arbres déconnectés alors qu'avec tous les mots, on en obtient une dizaine. Nous comparerons donc les résultats des différentes similarités en fonction du nombre d'arbres obtenus par la similarité dont le nombre d'arbres est maximum. Le nombre d'arbres est en fait le nombre de groupes formés par la classification automatique<sup>74</sup>.

## 5 Les résultats

Tous les détails concernant les corpus de tests ainsi que les résultats sur les mesures d'entropie et de pureté sont donnés dans l'annexe D.

En moyenne, il y a 141 mots dans les vecteurs portant sur tous les mots du texte (les vecteurs ont une taille qui varie de 20 à 450 mots) et 27 noms propres par vecteur de noms propres (ces derniers ont une taille qui varie de 0 nom propre à 120).

Pour évaluer les mesures, nous avons comparé leurs résultats sur chaque texte et nous leur avons associé un score par texte égal à 1 pour la meilleure et 0 pour les autres. Il ne nous restait plus qu'à additionner les scores.

### 5.1 Remarques sur les mesures d'entropie et de pureté

Nous utilisons, pour nos évaluations, les mesures d'entropie et de pureté pour mesurer la qualité des groupes. Cependant, nous avons remarqué que ces deux mesures sont biaisées lorsqu'on les calcule avec un nombre de groupes élevé. En effet, plus le nombre de groupes est grand (plus les groupes sont petits), plus l'entropie et la pureté sont bonnes ; ce phénomène est normal puisqu'un groupe qui ne contiendrait qu'un seul texte obtiendrait forcément une entropie de 0 et une pureté de 1.

### 5.2 Comparaison des résultats obtenus par différentes mesures de similarités

Voici les premières mesures de similarité que nous avons testées :

1. mesure Jaccard sur le vecteur des mots seuls
2. mesure TF.IDF sur le vecteur des mots seuls

---

<sup>74</sup> Nous avons remarqué que la méthode *Complete Link* donne un nombre d'arbres (ou groupes) toujours supérieur à la classification de référence.

3. mesure Jaccard sur le vecteur des noms propres
4. mesure TF.IDF sur le vecteur des noms propres
5. mesure Jaccard sur le vecteur de tous les mots
6. mesure TF.IDF sur le vecteur de tous les mots

Commençons par analyser le cas le plus classique, c'est-à-dire un vecteur contenant tous les mots du texte (5 et 6). Sur nos 29 jeux d'essais, la mesure TF.IDF sur ce vecteur est largement meilleure que la mesure de Jaccard (l'entropie est la meilleure 26 fois sur 29 et la pureté 25 fois sur 29).

Pour le vecteur de mots seuls (mesure 1 et 2), c'est la mesure TF.IDF qui donne aussi les meilleurs résultats (27 résultats meilleurs sur 29 pour l'entropie comme pour la pureté).

Regardons maintenant ce qu'il se passe pour les vecteurs de noms propres (mesures 3 et 4). Sur l'ensemble des groupes de textes à classer, la différence entre Jaccard et TF.IDF est moins nette.

Si nous observons maintenant les mesures 3 et 4, l'entropie de TF.IDF est plus élevée que Jaccard sur 19 des tests, parmi les 10 restants, 5 donnent de meilleurs résultats avec Jaccard et 5 ont des résultats égaux avec la mesure de Jaccard ou TF.IDF. La différence de résultats sur les noms propres est moins forte qu'avec les similarités TF.IDF et Jaccard calculées sur des vecteurs de mots. Nous avons ensuite étudié la distribution des résultats pour les groupes de 200 textes et les groupes de 50 textes. En fait, sur les 22 tests effectués avec des groupes de 200 textes, 17 sont mieux classés avec la mesure TF.IDF, 2 par Jaccard et 3 groupes sont à égalité pour les deux mesures. Pour les jeux d'essais à 50 textes, la mesure Jaccard semble aussi bonne que la mesure TF.IDF pour classer les textes entre eux (2/7 pour Jaccard, 2/7 pour TF.IDF et égalité des deux mesures pour 3 jeux sur 7). Il semble donc que sur de petits ensembles de textes, la fréquence d'apparition des noms propres ne joue pas un si grand rôle ; la présence ou l'absence du nom propre dans les textes peut suffire.

Nous avons ensuite comparé les résultats des mesures utilisant des vecteurs de mots seuls ou des vecteurs de noms propres (1, 2, 3 et 4) pour voir quelles étaient les meilleures mesures. Sur les ensembles de 50 textes, la mesure TF.IDF sur les noms propres est meilleure que les mesures utilisant les mots seuls du texte (6 fois sur 7 tests). Pour les ensembles de 200 textes, les noms propres avec TF.IDF obtiennent 12 fois le meilleur score et sont à égalité 3 fois avec le vecteur de mots seuls et TF.IDF. Globalement, les vecteurs de noms propres et la mesure TF.IDF obtiennent de meilleurs résultats (17/29 et 4/29 à égalité avec les mots) que le vecteur des mots (8/29), les cas restants étant à égalité.

Comparons maintenant les résultats obtenus avec TF.IDF par les vecteurs contenant tous les mots et les vecteurs de noms propres (mesures 4 et 6). Les vecteurs contenant tous les mots obtiennent les meilleurs résultats (21/29) ; il y a cependant égalité dans 5 cas avec les noms propres. Pour des petits groupes de textes, les noms propres donnent des résultats aussi bons que les vecteurs de mots.

L'utilité de calculer la pureté semble douteuse car elle confirme presque exactement les résultats de l'entropie.

Il faut donc retenir que :

- Les vecteurs de noms propres obtiennent de meilleurs résultats que les vecteurs de mots seuls. Pourtant la taille des vecteurs de noms propres est très réduite par rapport à celle des vecteurs de mots. Donc la qualité informative des noms propres dans un texte par rapport aux autres mots est confirmée. Ils sont de bons candidats pour participer à une classification des textes avec des éléments plus linguistiques que de simples vecteurs de mots.
- Les vecteurs contenant tous les mots (avec noms propres) donnent les meilleurs résultats. Les mots, bien que moins significatifs que les noms propres, améliorent les résultats de la classification par rapport à une classification n'utilisant que les noms propres.

Par conséquent, nous avons testé des mesures de similarités fusionnant les différentes mesures basées sur des vecteurs de mots seuls, de noms propres et de l'ensemble de tous les mots pour voir si une amélioration peut être apportée par rapport à une mesure "simple" sur tous les mots en augmentant artificiellement l'importance des noms propres.

### 5.3 Comparaison des résultats obtenus par des mesures de similarités fusionnées

Rappelons la formule de fusion des similarités (Formule 14 déjà décrite au paragraphe 2.3).

$$sim(d, d') = \alpha \cdot sim_{mots}(d, d') + \beta \cdot sim_{np}(d, d')$$

*Formule 16 : Mesure de similarité fusionnée entre sous-vecteurs de mots et de noms propres*

#### 5.3.1 Fusion de similarité avec la mesure TF.IDF sur le vecteur des mots seuls et le vecteur des noms propres

Nous faisons varier les coefficients sur la mesure TF.IDF avec la similarité  $sim_{mots}$  et la similarité  $sim_{np}$  de noms propres de manière à ce que la somme des coefficients reste égale à 1. On commence avec un coefficient de 0,1 pour les mots seuls et 0,9 sur les noms propres ; on augmente et diminue les coefficients jusqu'à un coefficient de 0,9 pour les mots et 0,1 pour les noms propres.

Sur 29 jeux de tests, on obtient 17 meilleurs résultats lorsque les coefficients sur les noms propres et sur les mots valent 0,5. Etant donné la différence de taille moyenne des deux vecteurs, cela revient à donner une importance plus forte à la similarité sur les noms propres par rapport à celle sur les mots. Ce qui confirme à nouveau l'importance des noms propres.

### 5.3.2 *Fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots et le vecteur des noms propres*

Nous essayons maintenant de fusionner les mesures de similarité  $sim_{\text{tous\_mots}}$  du vecteur contenant tous les mots avec le vecteur  $sim_{np}$ . Les résultats de la fusion de similarité sont les meilleurs (19 fois sur 29) lorsque le coefficient est de 1 sur tous les mots et est de 0,7 sur les noms propres.

### 5.3.3 *Comparaison des mesures fusionnées avec la mesure TF.IDF sur le vecteur de tous les mots*

Nous avons vu en §5.2 que le vecteur contenant tous les mots obtenait les meilleurs résultats. Nous le comparons maintenant aux fusions de similarité des paragraphes §5.3.1 et §5.3.2.

La fusion de  $sim_{\text{mots}}$  et  $sim_{np}$  obtient les meilleurs résultats à 8 reprises (7 fois pour la fusion  $sim_{\text{ts\_mots}}$  et  $sim_{np}$ , et 6 fois pour la mesure TF.IDF sur tous les mots). De plus, La fusion de  $sim_{\text{mots}}$  et  $sim_{np}$  obtient des résultats égaux aux autres mesures à 7 occasions, ce qui fait d'elle la meilleure mesure mais il faut souligner que la différence avec les autres mesures semble faible.

## 6 Conclusion

Les résultats d'une classification peuvent être améliorés en affectant aux noms propres un poids plus important que les autres mots du texte par la technique de la fusion des similarités.

Comme nous le prouvons avec toutes ces mesures, les noms propres seuls ont une qualité "classificatrice" plus importante que les autres mots d'un texte journalistique. Un processus de classification les prenant en compte ainsi que d'autres indices linguistiques (groupes nominaux ou verbaux par exemple) pourraient améliorer encore la classification de ce type de textes.

## CONCLUSION ET PERSPECTIVES

---

Cette thèse est surtout consacrée à la reconnaissance automatique des noms propres.

Nous avons présenté tout d'abord les noms propres d'un point de vue linguistique : définition et productivité, typage morpho-syntaxique et sémantique. Une étude en corpus sur les trois principaux types de noms propres (noms de personnes, de lieux, d'organisations) montre qu'ils apparaissent en des quantités très différentes dans les journaux, et que, selon leur type, il est plus ou moins possible de les repérer. En effet, leurs structures internes complexes, mais aussi leurs contextes dans le discours (preuves internes et externes) diffèrent : il faut donc adapter les méthodes permettant de les repérer et de les catégoriser à leurs types. Aux deux extrêmes, les noms de lieux nécessitent des dictionnaires alors que les noms d'organisations autorisent une description de leurs contextes et de leurs structures internes.

Les systèmes les plus courants d'extraction automatique de noms propres utilisent des règles qui décrivent les noms propres et leurs grammaires locales. Ces systèmes obtiennent de très bons résultats, mais l'investissement humain en terme d'écriture de règles est important. De tels systèmes utilisent des informations hétérogènes : morphologie, syntaxe, dictionnaires.

Nous avons constaté que les travaux, sur le français, dans ce domaine peuvent encore être améliorés et étendus. C'est pourquoi nous avons créé un système d'extraction linguistique des noms propres en français, que nous avons nommé ExtracNP et qui est fondé sur les cascades de transducteurs. Celles-ci permettent de faire de nombreuses transformations sur les textes pour réaliser soit de l'analyse syntaxique, soit de l'extraction d'information.

Pour cela, nous utilisons le système Intex qui permet de décrire des grammaires de manière puissante et lisible à l'aide de transducteurs. Nous avons implémenté un outil, nommé CasSys, permettant la conception de cascades de transducteurs. CasSys offre des fonctionnalités qui s'ajoutent à celles d'Intex et sera mis à la disposition de la communauté des chercheurs utilisant ce logiciel.

CasSys est utilisé par le système extracNP d'extraction d'information pour générer deux cascades de transducteurs extrayant les noms propres : la première cascade permet de catégoriser et extraire une partie des noms propres d'un texte (ceux qui possèdent des indices permettant de les repérer), la seconde utilise les résultats de la première pour en trouver de nouveaux.

Avant d'appliquer ces cascades, des pré-traitements préparent le texte. Le découpage en phrases, que nous avons amélioré, obtient un rappel et une précision supérieurs à 99% pour les trois types de textes testés (*Le Monde*, *Ouest France*, *La femme de trente ans* de Balzac). Nous utilisons un certain nombre de dictionnaires existants ou créés par nos soins (prénoms, sigles) pour étiqueter les textes. Étant donné la très forte lexicalisation de nos grammaires, les étiquettes syntaxiques nous sont très peu utiles et nous envisageons de nous en passer complètement dans une version ultérieure.

Notre système ExtracNP et ses cascades de transducteurs obtiennent un rappel de 93% et une précision de 94% : ce système fournit actuellement les meilleurs résultats sur

le français et va pouvoir contribuer à l'extension des dictionnaires du projet Prolex de traitement automatique des noms propres, développé à l'université de Tours.

Les noms propres sont très importants pour comprendre le sens et le contenu des textes journalistiques. Nous voulions tenter de voir si cette intuition était vraie. Nous avons utilisé la méthode de classification hiérarchique ascendante (reconnue pour donner de très bons résultats, mais très lente, ce qui nous importe peu dans ce travail). Les résultats obtenus avec différentes mesures de similarités sont comparés à l'aide des mesures d'entropie et de pureté. Nous comparons tout d'abord les résultats obtenus par des mesures de similarités (Jaccard et TF.IDF) sur trois types de vecteurs différents : les vecteurs de noms propres, les vecteurs de mots seuls (sans les noms propres) et les vecteurs contenant tous les mots (y compris les noms propres). Il ressort que les noms propres seuls classent mieux les textes que les autres mots sur de petits ensembles de textes. Par contre, ce sont les mesures prenant en compte tous les types de mots qui sont les meilleures (ceci est en partie dû à la faible quantité des noms propres par rapport aux autres mots du texte). Finalement, nous montrons qu'une mesure de similarité portant sur les mots que l'on combine à une mesure de similarité sur les noms propres est légèrement meilleure. Les noms propres semblent donc pertinents pour le processus de classification automatique des textes.

À partir de nos travaux actuels, nous souhaitons poursuivre sur la recherche de relations entre noms propres et sur le traitement de l'anaphore. Nous envisageons aussi des collaborations multilingues pour écrire des grammaires adaptées à d'autres langues que le français. Nous avons commencé quelques essais avec Thierry Grass sur l'extraction des noms de personnes en allemand.

## ANNEXE A : FICHE TECHNIQUE DU SYSTEME CASYS

La Figure 46 montre l'interface du système CasSys. L'utilisateur spécifie le nom d'un fichier contenant la liste des transducteurs à appliquer sur le texte avec les options souhaitées (mode *merge* ou *replace*).

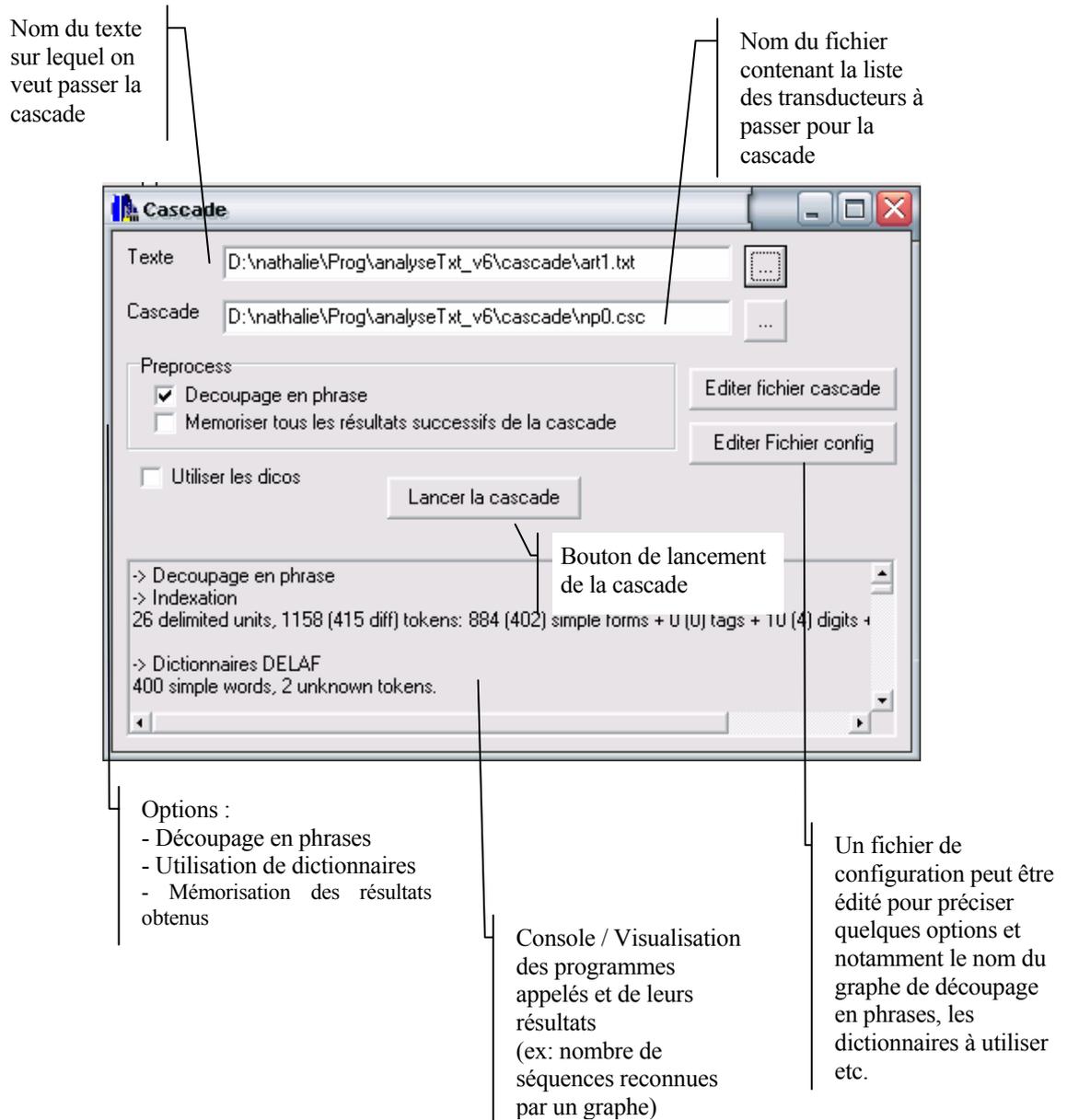


Figure 46 : Interface de CasSys

L'algorithme général de CasSys est le suivant :

---

### 1. Pré-traitements

- a. Découpage en phrases ou pas : programme **fst2txt**
- b. Indexation du texte : programme **indexer**
- c. Passage des dictionnaires ou pas : programme **dicos** et **dicoc**

### 2. Passage de la cascade de transducteurs : les graphes de la cascade sont listés dans un fichier et on précise pour chacun les modes de passage.

- a. Pour chaque graphe, on utilise **recor** ou **recon**.
  - b. Les résultats de recor ou recon sont :
    - soit réinsérés dans le texte initial avec le mode choisi par l'utilisateur : **enrich**
    - soit ajoutés à un fichier index et une étiquette remplace le motif dans le texte. Ce fichier index porte le nom du fichier Texte correspondant et à l'extension **idx**.
- 

L'utilisation du système de cascade de transducteurs CasSys nécessite Intex et l'installation des variables d'environnement d'Intex (manuel Intex, chapitre 15. *Utiliser Intex en ligne de commande*).

Le **fichier de configuration** de CasSys doit contenir les noms et chemins du graphe de découpage en phrases et des dictionnaires que l'on souhaite passer sur le texte.

```
pathSentence="d:\intex\french\sentence.fst"
DELAF="~\delafm.bin" "~\Chiffres romains.fst" "~\CNP.fst"
"~\CNPS.fst" "~\monDelaf.dic"
DELACF="~\CNPCA.fst" "~\Dnum.fst" "~\prenoms-prolex.dic"
"~\prolintex.bin" "~\profession.dic" "~\sigles-prolex.dic"
"~\prolintexPlus.dic"
```

Un **deuxième fichier est nécessaire** : celui qui contient la liste des graphes de la cascade de transducteur. Le format d'une ligne de ce fichier est le suivant :

*option1 option2 option3 option4 nom\_et\_chemin\_du\_graphe*

- Option1 : permet de préciser si le texte est un texte délimité en phrases (présence de {S} ou pas). Selon cette option, le programme CasSys devra utiliser le programme recor ou recon d'Intex.

- Option2 : précise si le graphe doit être passé en mode merge ou replace.
- Option3 : choix du mode fusion ou remplacement d'Intex, ou demande de placer les motifs repérés dans le fichier index.
- Option4 : précise si les programmes recor ou recon doit utiliser les étiquettes placées par les dictionnaires sur chaque mot.

Exemple :

```
# trouver les prefixes annonçant des noms de personnes
d f 1 1 ..\grf\person\ctxt_tit_tcivil.grf
d f 1 1 ..\grf\person\ctxt_tit_tministre.grf
d f 1 1 ..\grf\person\ctxt_tit_tnoble.grf
# trouver les noms de personnes
d f 1 1 ..\grf\person\person1.grf
d f 1 1 ..\grf\person\person2.grf
d f 1 1 ..\grf\person\person3.grf
# repérer des adresses dans les textes
d f 1 1 ..\grf\adresse\tadresse.grf
```

Deux interfaces permettent de générer le fichier de configuration et la liste des graphes de la cascade.



## ANNEXE B : METHODOLOGIE DE VERIFICATION DU DECOUPAGE EN PHRASES

---

Au cours de notre étude sur la segmentation des textes en phrases, nous nous sommes rendus compte que les erreurs communes au graphe sentence actuellement fourni avec Intex et notre propre graphe étaient très rares.

À certains endroits, le délimiteur de phrases {S} est placé alors qu'il ne devrait pas l'être ; ces bruits sont, par exemple :

- Un sigle suivi d'un mot commençant une majuscule

*La C.F.D.T.{S} Basse-Normandie lance un mouvement de grève.*

- Des points de suspension suivis d'un mot commençant par une majuscule

*La littérature vendéenne est souvent signée des plus grands  
noms : Hugo, Balzac, Chateaubriand, Nerval et même ...{S} Jules  
Verne.*

*Remarque* : nous n'avons trouvé aucune erreur de type silence (un {S} manquant) commune aux deux graphes.

Étant donné le peu d'erreurs communes aux deux graphes, nous proposons de ne regarder que les erreurs non communes pour vérifier les améliorations apportées par le nouvel automate. Ceci permet d'automatiser en partie la vérification des résultats et d'évaluer les résultats sur une très grande quantité de textes. Il suffit de comparer les deux textes obtenus après découpages des phrases avec les deux graphes. Lorsqu'on trouve, dans un texte, le symbole {S} de fin de phrase et qu'il n'est pas à cette position dans le second texte, on affiche une concordance des deux textes à cet endroit et on propose un menu dans lequel l'utilisateur choisit quelle solution est la bonne :

- la solution du premier graphe sentence,
- la solution de notre graphe sentence,
- aucune des deux solutions,
- ou les deux.

Lorsque le texte est entièrement vérifié, le programme calcule lui-même les statistiques correspondantes (rappel, précision).

Exemple 1 : la première solution est la bonne.

*...canalisation.{S}Paul BODIN. {S} (1) ONIVINS et INAO : Offices des  
vins...*

*...canalisation.{S}Paul BODIN. (1) ONIVINS et INAO : Offices des vins...*

Exemple 2 : la deuxième solution est la bonne.

*...et mise à prix 400 000 F. Un Marquet de 1921, Le port d'Alger et la ville, ...*

*...et mise à prix 400 000 F. {S} Un Marquet de 1921, Le port d'Alger et la ville, ...*

Exemple 3 : les deux solutions sont bonnes mais l'automate de découpage des phrases ne place pas le symbole {S} au même endroit. L'utilisateur ne doit donc pas compter ceci comme une erreur.

*...Luxembourgeois, Hollandais. {S}.. On attendait encore des Américains, ...*

*...Luxembourgeois, Hollandais... {S} On attendait encore des Américains, ...*

Ou les deux solutions sont bonnes mais l'automate de découpage des phrases ne place pas le symbole {S} dans un des deux cas. Le placement ou non du symbole {S} dans le cas ci dessous ne relève pas d'une erreur mais du choix fait pour réaliser le découpage des phrases. L'utilisateur ne doit donc pas compter ceci comme une erreur.

*...Sa conclusion est double : « Oui, la Révolution a réussi sa déchristianisation...*

*...Sa conclusion est double : {S} « Oui, la Révolution a réussi sa déchristianisation...*

Exemple 4 : les deux solutions sont fausses mais le {S} n'est pas placé exactement au même endroit (un espace en plus dans la deuxième solution), l'automate de découpage des phrases place un symbole {S} inutile. Ceci doit être compté comme une erreur.

*... en lien avec F.R. {S} 3 Normandie s'associera au Prix du polar et ...*

*... en lien avec F.R. {S}3 Normandie s'associera au Prix du polar et...*

Ce système semi-automatisé permet de faire une vérification très rapide d'un corpus beaucoup plus grand. Les erreurs communes étant très peu fréquentes il n'est pas très préjudiciable de ne pas les prendre en compte (0 % de silences communs dans les 3 textes étudiés, 0.30 % de bruits communs pour le journal Ouest France mais seulement 0.04 % pour Le Monde et La Femme de trente ans).

## ANNEXE C : LEMMATISATION DES TEXTES

---

De manière classique, les vecteurs de mots sont créés en éliminant les mots vides<sup>75</sup> (peu significatifs) : cette élimination diminue la longueur des vecteurs et doit augmenter la dissimilarité entre les documents.

Il faut ensuite *lemmatiser* ou *raciniser* les mots. Lemmatiser signifie enlever la flexion et pour obtenir une forme canonique du mot, par exemple la forme infinitive pour un verbe (ex : *chantera* → *chanter*). *Raciniser*<sup>76</sup> (*stemming*) signifie trouver le morphème racine du mot (ex : *chant*, *chanter*, *chanson*).

La lemmatisation cherche donc à créer une liste des formes simples du texte. Différentes méthodes de lemmatisation sont expliquées dans [Frakes, Baeza-Yates, 1992]. Le système SIAC (Segmentation et Indexation Automatique de Corpus), par exemple, lemmatise les textes avec le logiciel ECSTA<sup>77</sup> en utilisant un dictionnaire de formes fléchies [Bellot, 2000].

Nous avons choisi de lemmatiser nos textes grâce au système Intex. Voici comment nous procédons.

Nous créons la liste des mots lemmatisés du texte accompagnés de leurs fréquences d'apparition dans le texte. Cette liste est obtenue sans désambiguïsation des étiquettes portées par les mots. Ce système doit donner un résultat moins bon qu'une lemmatisation sur un texte désambiguë, mais est meilleur que la racinisation (*stemming*).

Pour chaque texte, nous produisons une liste de mots avec leur fréquence de la forme :

*adapter 1*  
*agir 2*  
*analyste 1*  
*autre 1*  
*etc.*

Les mots sont lemmatisés à l'aide des dictionnaires très complets de la langue française [Courtois, Silberztein, 1990]. Pour chaque forme fléchie rencontrée dans le texte, nous ne choisissons qu'un de ces lemmes en fonction d'une heuristique expliquée dans l'algorithme ci-dessous.

---

<sup>75</sup> Par contre, pour le domaine de l'extraction d'information, [Riloff, 1995] remarque que certains mots vides sont importants. Par exemple, la suppression de la négation aboutit à représenter une phrase par son contraire.

<sup>76</sup> L'algorithme de Porter, par exemple, est un algorithme de *stemming* très utilisé [Porter, 1980].

<sup>77</sup> LIA, Avignon.

Voici l'algorithme de notre programme de lemmatisation :

---

Indexation du texte (programme **indexer**).

Passage des dictionnaires de formes fléchies sur le texte (programme **dicos**).

On obtient alors un fichier ne contenant que les mots présents dans le texte (i.e. le dictionnaire des mots du texte). On supprime certaines entrées de ce dictionnaire :

Si les formes fléchies de deux mots sont égales, on utilise l'heuristique suivante :

- i. soit ces mots ont la même catégorie (nom, verbe ou adjectif) auquel cas on garde n'importe laquelle des entrées.
- ii. soit ils n'ont pas la même catégorie : on privilégie d'abord les noms, les verbes puis les autres catégories.

On remplace les mots du texte par les lemmes choisis par notre heuristique (programme **etiqq**) et on indexe à nouveau le texte (**indexer**).

On crée la liste des mots du texte avec leur fréquence (programme **tokenlist**).

On applique le dictionnaire dont on a supprimé des entrées sur cette liste des mots du texte (programme **dicos**).

On applique un transducteur reconnaissant les mots vides sur cette liste en mode remplacement (programme **recor**) pour les éliminer (programme **enrich**)

On retire les lignes "doublons" (ex : *Français 3, français 2* sont deux mêmes lignes car seule la majuscule change et on remplace ces deux occurrences par *français 5*).

---

Donnons, pour l'exemple, le texte suivant :

*Le débat sur la politique monétaire : M. Giscard d'Estaing plaide pour le maintien du franc dans le SME*

*M. Valéry Giscard d'Estaing estime, dans Paris-Match (daté 7 janvier), qu'il existe " un complot ou plutôt une sorte de culture-complot, qui incite toutes sortes de gens \_ des opérateurs, des analystes, des commentateurs \_ à agir dans le même sens, au même moment, pour tenter de faire sauter les derniers verrous qui protègent encore le système monétaire européen ". Le président de l'UDF souligne que " les uns agissent par conviction idéologique ", parce qu'ils jugent le système des taux de change flottants " mieux adapté aux réalités brusquement changeantes du monde moderne ", et que d'autres prônent la fin du SME parce qu'" ils ont compris que, si le système monétaire européen saute, c'en est fini pour longtemps du projet d'union monétaire ". Pour M. Giscard d'Estaing, " ...*

Le résultat obtenu sur le texte est le suivant :

adapter 1	maintien 1
agir 2	match 1
analyste 1	moderne 1
autre 1	moment 1
change 1	monde 1
changeant 1	monétaire 4
commentateur 1	opérateur 1
complot 2	paris 1
comprendre 1	plaider 1
conviction 1	politique 1
culture 1	président 1
dater 1	projet 1
débat 1	prôner 1
dernier 1	protéger 1
Estaing 3	réalité 1
estime 1	saute 1
européen 2	sauter 1
exister 1	sens 1
fin 1	SME 2
fini 1	sorte 2
flottant 1	souligner 1
franc 1	système 3
gens 1	taux 1
Giscard 3	tenter 1
idéologique 1	UDF 1
inciter 1	union 1
janvier 1	Valéry 1
juger 1	verrou 1
longtemps 1	



## **ANNEXE D : CORPUS DE TEST DES SIMILARITES ET RESULTATS**

---

Notre corpus est construit les résultats de requêtes sur des ensembles de textes des campagnes Amaryllis.

Deux séries de thèmes sont utilisées. En voici la liste :

### **Série de thèmes 1**

---

- Th 1 : séparation de la Tchécoslovaquie
- Th 2 : la guerre en Yougoslavie
- Th 3 : le sang contaminé
- Th 4 : l'Allemagne face à la Xénophobie
- Th 5 : Le chômage en France : solutions ?
- Th 6 : le SIDA
- Th 7 : la guerre en Somalie
- Th 8 : la guerre civile en Algérie
- Th 9 : la mafia en Italie
- Th 10 : préparation du PS aux législatives
- Th 11 : le chômage en France
- Th 12 : Les attentats et conflits politiques en Afrique du S
- Th 13 : Les actes terroristes des intégristes musulmans e
- Th 14 : Les conflits armés entre certaines Républiques de
- Th 15 : La rébellion kurde en Turquie, en Irak et en Iran e
- Th 16 : Les troubles politiques et civils au Sénégal en 19
- Th 17 : L'opposition de la communauté internationale au j
- Th 18 : La situation économique et politique à Cuba en 1
- Th 19 : La situation politique au Cambodge en 1993
- Th 20: Le terrorisme et la violence aux Etats-Unis d'Amér
- Th 21 : Le combat pour les droits de l'homme dans le mor
- Th 22 : Les divergences d'opinion au sein de la coalition l
- Th 23 : La parité du franc à l'intérieur du S.M.E.
- Th 24 : La réforme du code de la nationalité française
- Th 25 : La crise économique en France
- Th 26 : La drogue en France

**série de thèmes 2**

- 
- Th 1 : La construction monétaire
  - Th 2 : Politique internationale en matière de Défense
  - Th 3 : L'extrême droite en Europe
  - Th 4 : Les contradictions du service public
  - Th 5 : L'économie libérale et les entreprises françaises
  - Th 6 : L'économie libérale et les centrales syndicales françaises
  - Th 7 : La femme dans la société d'aujourd'hui
  - Th 8 : Le puritanisme
  - Th 9 : L'affaire Botton
  - Th 10 : La réglementation de la pêche
  - Th 11 : La pollution
  - Th 12 : La xénophobie
  - Th 13 : La crise de l'immobilier en France
  - Th 14 : Médias et déontologie
  - Th 15 : La politique protectionniste américaine
  - Th 16 : La réforme pénale en France
  - Th 17 : Le règlement des conflits en Europe de l'Est
  - Th 18 : Les écoutes téléphoniques
  - Th 19 : la réforme de la Constitution
  - Th 20 : La situation en Irlande du Nord
  - Th 21 : La protection sociale en France
  - Th 22 : La politique et l'Ecologie
  - Th 23 : La crise en Russie
  - Th 24 : L'industrie sidérurgique
  - Th 25 : Le règlement des conflits en Afrique
  - Th 26 : La sauvegarde des régimes de retraite

Nous avons listé les textes donnés par l'évaluation Amaryllis comme résultat des requêtes correspondants à ces thèmes sur les corpus OFIL1 et OFIL2 (ce qui nous donne des textes classés par thèmes). Par exemple, la requête du thème 2 de la première série de thèmes sur le corpus OFIL2 doit permettre de trouver 37 textes de ce corpus (d'après les résultats d'Amaryllis).

Nous avons créé, grâce à ces différents groupes de textes connus, des ensembles de textes à classer. Les groupes à classer (ex : of1th1\_3) sont nommés comme suit :

- of1 ou of2 pour des textes provenant respectivement des corpus OFIL1 ou OFIL2,
- th1 ou th2 selon que les textes répondent à des thèmes de la première série de thèmes ou de la deuxième série,
- un numéro (1, 2, 3 etc.)

Les tableaux suivants listent les ensembles de textes mélangés, que nous allons regrouper grâce à nos mesures, ainsi que le nombre de textes qu'ils contiennent et le nombre de groupes idéal qu'ils contiennent. Nous avons créé une vingtaine d'ensembles de textes contenant 200 textes et 6 ensembles de 50 textes. Nous n'avons pu concevoir d'ensembles plus gros étant donné le peu de corpus dont nous disposons (environ 3000 textes différents).

nom des groupes de textes à classer	of1th1_2	of1th1_3	of1th1_4	of1th1_5	of1th1_6	of1th1_7	of1th1_8	of1th1_9	of1th1_10	of1th1_11	of1th1_12	of1th1_13
nombre de textes	195	194	202	203	204	203	192	197	54	50	55	52
nombre de Classes	13	8	8	12	8	7	9	10	6	4	4	5

nom des groupes de textes à classer	of2th1_2	of2th1_3	of2th1_4	of2th1_5	of2th1_6	of2th1_7	of2th1_8	of2th1_9	of2th1_10	of2th1_11	of2th1_12
nombre de textes	205	194	194	201	198	206	204	202	53	52	55
nombre de Classes	7	10	10	9	9	9	9	10	4	4	4

nom des groupes de textes à classer	of1th2_5	of1th2_6	of1th2_7	of1th2_8	of1th2_9	of1th2_10	of1th2_11	of1th2_12
nombre de textes	197	201	201	194	197	207	199	193
nombre de Classes	6	6	6	5	6	6	6	5

Avant de présenter les résultats obtenus, voici la description de toutes les similarités testées :

Nom donné à la mesure	Description
cl_mnp_tfidf_m0_1np0_9	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,1 et le vecteur des noms propres coefficienté 0,9
cl_mnp_tfidf_m0_2np0_8	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,2 et le vecteur des noms propres coefficienté 0,8
cl_mnp_tfidf_m0_3np0_7	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,3 et le vecteur des noms propres coefficienté 0,7
cl_mnp_tfidf_m0_5np0_5	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,5 et le vecteur des noms propres coefficienté 0,5
cl_mnp_tfidf_m0_7np0_3	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,7 et le vecteur des noms propres coefficienté 0,3
cl_mnp_tfidf_m0_9np0_1	fusion de similarité avec la mesure TF.IDF sur le vecteur des mots coefficienté 0,9 et le vecteur des noms propres coefficienté 0,1
cl_mots_jacc	mesure Jaccard sur le vecteur des mots
cl_mots_tfidf	mesure TF.IDF sur le vecteur des mots
cl_np_jacc	mesure Jaccard sur le vecteur des noms propres
cl_np_tfidf	mesure TF.IDF sur le vecteur des noms propres
cl_tmnp_tfidf_m1np0_1	fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots coefficienté 1 et le vecteur des noms propres coefficienté 0,1
cl_tmnp_tfidf_m1np0_2	fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots coefficienté 1 et le vecteur des noms propres

Nom donné à la mesure	Description
	coefficienté 0,2
cl_tmnp_tfidf_m1np0_3	fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots coefficienté 1 et le vecteur des noms propres coefficienté 0,3
cl_tmnp_tfidf_m1np0_5	fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots coefficienté 1 et le vecteur des noms propres coefficienté 0,5
cl_tmnp_tfidf_m1np0_7	fusion de similarité avec la mesure TF.IDF sur le vecteur de tous les mots coefficienté 1 et le vecteur des noms propres coefficienté 0,7
cl_tsmots_jacc	mesure Jaccard sur le vecteur de tous les mots
cl_tsmots_tfidf	mesure TF.IDF sur le vecteur de tous les mots

Comme nous l'avons expliqué au Chapitre 6, les nombres de classes auquel nous avons coupé l'arbre varient en fonction des jeux d'essais et sont indiqués dans les tableaux. Pour pouvoir étudier les valeurs d'entropie et de pureté obtenus à l'aide des noms propres, nous sommes obligés de couper l'arbre pour un grand nombre de classes (environ quarante). C'est pourquoi nous étudions les résultats obtenus pour deux hauteurs de coupure d'arbre à chaque fois. Lorsque nous coupons l'arbre à une hauteur ayant un faible nombre de classes, les résultats des entropies et puretés pour les noms propres n'apparaissent donc pas dans nos tableaux.

**Les résultats de l'entropie sont listés dans les tableaux suivants.**

entropie (200 textes) nombre de classes	of1th1_200_2		of1th1_200_3		of1th1_200_4		of1th1_200_5	
	15	35	15	37	16	44	16	41
cl_mnp_tfidf_m0_1np0_9	0,104	0,036	0,066	0,026	0,245	0,115	0,107	0,046
cl_mnp_tfidf_m0_2np0_8	0,103	0,036	0,050	0,027	0,222	0,104	0,113	0,044
cl_mnp_tfidf_m0_3np0_7	0,117	0,041	0,039	0,022	0,191	0,086	0,096	0,050
cl_mnp_tfidf_m0_5np0_5	0,099	0,025	0,022	0,010	0,176	0,074	0,081	0,038
cl_mnp_tfidf_m0_7np0_3	0,132	0,034	0,063	0,028	0,180	0,055	0,138	0,039
cl_mnp_tfidf_m0_9np0_1	0,157	0,049	0,130	0,033	0,181	0,052	0,169	0,053
cl_mots_jacc	0,421	0,225	0,502	0,258	0,287	0,105	0,420	0,226
cl_mots_tfidf	0,236	0,064	0,303	0,126	0,161	0,072	0,271	0,095
cl_np_jacc	-	0,050	-	0,009	-	0,138	-	0,061
cl_np_tfidf	-	0,036	-	0,021	-	0,129	-	0,035
cl_tmnp_tfidf_m1np0_1	0,170	0,045	0,154	0,048	0,177	0,063	0,169	0,053
cl_tmnp_tfidf_m1np0_2	0,105	0,038	0,065	0,023	0,178	0,046	0,143	0,045
cl_tmnp_tfidf_m1np0_3	0,137	0,032	0,078	0,031	0,159	0,048	0,094	0,036
cl_tmnp_tfidf_m1np0_5	0,130	0,036	0,111	0,030	0,174	0,056	0,105	0,053
cl_tmnp_tfidf_m1np0_7	0,090	0,031	0,058	0,021	0,182	0,061	0,087	0,029
cl_tsmots_jacc	0,312	0,121	0,226	0,133	0,335	0,134	0,237	0,110
cl_tsmots_tfidf	0,123	0,042	0,039	0,014	0,162	0,070	0,082	0,034

entropie (200 textes) nombre de classes	of1th1_200_6		of1th1_200_7		of1th1_200_8		of1th1_200_9	
	16	44	16	34	16	38	16	47
cl_mnp_tfidf_m0_1np0_9	0,154	0,070	0,109	0,087	0,052	0,032	0,214	0,096
cl_mnp_tfidf_m0_2np0_8	0,134	0,070	0,096	0,070	0,066	0,035	0,206	0,091
cl_mnp_tfidf_m0_3np0_7	0,116	0,050	0,127	0,084	0,052	0,017	0,207	0,051
cl_mnp_tfidf_m0_5np0_5	0,130	0,054	0,081	0,051	0,061	0,024	0,179	0,051
cl_mnp_tfidf_m0_7np0_3	0,172	0,063	0,078	0,053	0,051	0,032	0,169	0,062
cl_mnp_tfidf_m0_9np0_1	0,292	0,126	0,145	0,102	0,179	0,093	0,311	0,115
cl_mots_jacc	0,486	0,247	0,308	0,207	0,397	0,229	0,485	0,215
cl_mots_tfidf	0,328	0,138	0,171	0,085	0,408	0,169	0,284	0,126
cl_np_jacc	-	0,084	-	0,123	-	0,020	-	0,179
cl_np_tfidf	-	0,068	-	0,081	-	0,027	-	0,108
cl_tmnp_tfidf_m1np0_1	0,207	0,126	0,152	0,095	0,178	0,105	0,306	0,104
cl_tmnp_tfidf_m1np0_2	0,213	0,084	0,087	0,067	0,089	0,048	0,231	0,088
cl_tmnp_tfidf_m1np0_3	0,179	0,084	0,109	0,059	0,072	0,043	0,231	0,085
cl_tmnp_tfidf_m1np0_5	0,100	0,027	0,108	0,081	0,060	0,037	0,205	0,065
cl_tmnp_tfidf_m1np0_7	0,141	0,043	0,079	0,056	0,057	0,032	0,204	0,076
cl_tsmots_jacc	0,334	0,100	0,200	0,130	0,152	0,062	0,356	0,173
cl_tsmots_tfidf	0,116	0,056	0,112	0,056	0,047	0,024	0,182	0,070

entropie (200 textes) nombre de classes	of2th1_200_3		of2th1_200_4		of2th1_200_5		of2th1_200_6	
	15	33	15	41	16	40	14	33
cl_mnp_tfidf_m0_1np0_9	0,163	0,048	0,316	0,145	0,124	0,067	0,012	0,011
cl_mnp_tfidf_m0_2np0_8	0,159	0,042	0,301	0,129	0,112	0,066	0,008	0,008
cl_mnp_tfidf_m0_3np0_7	0,136	0,039	0,217	0,102	0,121	0,056	0,016	0,008
cl_mnp_tfidf_m0_5np0_5	0,135	0,035	0,164	0,078	0,087	0,052	0,016	0,006
cl_mnp_tfidf_m0_7np0_3	0,181	0,040	0,278	0,069	0,137	0,061	0,040	0,019
cl_mnp_tfidf_m0_9np0_1	0,219	0,070	0,268	0,096	0,116	0,072	0,181	0,051
cl_mots_jacc	0,565	0,277	0,429	0,140	0,370	0,159	0,481	0,230
cl_mots_tfidf	0,384	0,119	0,315	0,122	0,245	0,087	0,371	0,106
cl_np_jacc	-	0,048	-	0,179	-	0,081	-	0,020
cl_np_tfidf	-	0,047	-	0,146	-	0,064	-	0,014
cl_tmnp_tfidf_m1np0_1	0,261	0,091	0,268	0,096	0,152	0,076	0,175	0,044
cl_tmnp_tfidf_m1np0_2	0,157	0,050	0,243	0,091	0,137	0,081	0,083	0,018
cl_tmnp_tfidf_m1np0_3	0,154	0,044	0,247	0,081	0,110	0,067	0,044	0,016
cl_tmnp_tfidf_m1np0_5	0,161	0,038	0,226	0,085	0,111	0,053	0,047	0,026
cl_tmnp_tfidf_m1np0_7	0,199	0,038	0,205	0,070	0,092	0,050	0,028	0,011
cl_tsmots_jacc	0,265	0,144	0,365	0,109	0,157	0,092	0,220	0,116
cl_tsmots_tfidf	0,108	0,028	0,166	0,093	0,108	0,065	0,022	0,009

entropie (200 textes) nombre de classes	of2th1_200_7		of2th1_200_8		of2th1_200_9	
	15	46	13	50	16	35
cl_mnp_tfidf_m0_1np0_9	0,153	0,079	0,241	0,121	0,093	0,060
cl_mnp_tfidf_m0_2np0_8	0,139	0,051	0,230	0,110	0,092	0,059
cl_mnp_tfidf_m0_3np0_7	0,129	0,038	0,247	0,103	0,092	0,057
cl_mnp_tfidf_m0_5np0_5	0,101	0,040	0,231	0,089	0,040	0,021
cl_mnp_tfidf_m0_7np0_3	0,133	0,047	0,234	0,096	0,050	0,036
cl_mnp_tfidf_m0_9np0_1	0,140	0,040	0,282	0,103	0,165	0,067
cl_mots_jacc	0,251	0,079	0,424	0,139	0,378	0,171
cl_mots_tfidf	0,127	0,047	0,275	0,125	0,266	0,105
cl_np_jacc	-	0,097	-	0,124	-	0,054
cl_np_tfidf	-	0,049	-	0,111	-	0,070
cl_tmnp_tfidf_m1np0_1	0,140	0,040	0,286	0,111	0,114	0,054
cl_tmnp_tfidf_m1np0_2	0,117	0,050	0,270	0,106	0,080	0,040
cl_tmnp_tfidf_m1np0_3	0,097	0,045	0,273	0,093	0,083	0,041
cl_tmnp_tfidf_m1np0_5	0,113	0,052	0,247	0,094	0,040	0,030
cl_tmnp_tfidf_m1np0_7	0,116	0,052	0,212	0,080	0,033	0,021
cl_tsmots_jacc	0,168	0,047	0,319	0,104	0,169	0,108
cl_tsmots_tfidf	0,066	0,019	0,162	0,075	0,069	0,032

entropie (200 textes) nombre de classes	of1th2_200_5		of1th2_200_6		of1th2_200_7		of1th2_200_8	
	8	43	13	53	11	39	9	41
cl_mnp_tfidf_m0_1np0_9	0,464	0,219	0,221	0,107	0,156	0,084	0,231	0,102
cl_mnp_tfidf_m0_2np0_8	0,383	0,160	0,142	0,069	0,209	0,087	0,179	0,086
cl_mnp_tfidf_m0_3np0_7	0,315	0,136	0,098	0,050	0,161	0,083	0,175	0,086
cl_mnp_tfidf_m0_5np0_5	0,201	0,090	0,083	0,032	0,106	0,047	0,189	0,082
cl_mnp_tfidf_m0_7np0_3	0,152	0,053	0,046	0,015	0,098	0,054	0,127	0,032
cl_mnp_tfidf_m0_9np0_1	0,212	0,055	0,025	0,007	0,227	0,055	0,225	0,076
cl_mots_jacc	0,412	0,105	0,071	0,027	0,543	0,192	0,393	0,096
cl_mots_tfidf	0,336	0,069	0,019	0,006	0,415	0,123	0,177	0,082
cl_np_jacc	-	0,295	-	0,220	-	0,135	-	0,153
cl_np_tfidf	-	0,232	-	0,111	-	0,066	-	0,102
cl_tmnp_tfidf_m1np0_1	0,233	0,054	0,027	0,007	0,151	0,047	0,168	0,067
cl_tmnp_tfidf_m1np0_2	0,226	0,057	0,039	0,011	0,117	0,053	0,120	0,047
cl_tmnp_tfidf_m1np0_3	0,292	0,047	0,035	0,015	0,078	0,043	0,165	0,053
cl_tmnp_tfidf_m1np0_5	0,151	0,062	0,051	0,015	0,077	0,041	0,127	0,036
cl_tmnp_tfidf_m1np0_7	0,208	0,077	0,050	0,015	0,101	0,047	0,113	0,040
cl_tsmots_jacc	0,328	0,103	0,045	0,014	0,236	0,062	0,423	0,134
cl_tsmots_tfidf	0,205	0,029	0,041	0,007	0,081	0,023	0,062	0,027

entropie (200 textes) nombre de classes	of1th2_200_10		of1th2_200_11		of1th2_200_12	
	12	54	16	49	14	45
cl_mnp_tfidf_m0_1np0_9	0,203	0,100	0,175	0,089	0,100	0,066
cl_mnp_tfidf_m0_2np0_8	0,215	0,086	0,137	0,070	0,097	0,057
cl_mnp_tfidf_m0_3np0_7	0,232	0,054	0,140	0,065	0,100	0,043
cl_mnp_tfidf_m0_5np0_5	0,190	0,035	0,099	0,050	0,070	0,021
cl_mnp_tfidf_m0_7np0_3	0,115	0,023	0,080	0,037	0,068	0,004
cl_mnp_tfidf_m0_9np0_1	0,175	0,021	0,137	0,039	0,044	0,004
cl_mots_jacc	0,417	0,062	0,317	0,117	0,237	0,039
cl_mots_tfidf	0,130	0,025	0,164	0,065	0,136	0,008
cl_np_jacc	-	0,145	-	0,210	-	0,083
cl_np_tfidf	-	0,145	-	0,090	-	0,056
cl_tmnp_tfidf_m1np0_1	0,246	0,029	0,105	0,042	0,044	0,004
cl_tmnp_tfidf_m1np0_2	0,098	0,014	0,099	0,046	0,023	0,004
cl_tmnp_tfidf_m1np0_3	0,133	0,027	0,097	0,031	0,047	0,004
cl_tmnp_tfidf_m1np0_5	0,116	0,023	0,100	0,048	0,068	0,011
cl_tmnp_tfidf_m1np0_7	0,111	0,020	0,090	0,040	0,060	0,011
cl_tsmots_jacc	0,161	0,048	0,266	0,043	0,122	0,035
cl_tsmots_tfidf	0,173	0,014	0,047	0,021	0,043	0,000

entropie (50 textes) nombre de classes	of1th1_50_10		of1th1_50_11		of1th1_50_12		of1th1_50_13	
	6	20	4	6	5	14	6	12
cl_mnp_tfidf_m0_1np0_9	0,383	0,116	0,000	0,000	0,000	0,000	0,083	0,027
cl_mnp_tfidf_m0_2np0_8	0,355	0,129	0,000	0,000	0,000	0,000	0,083	0,027
cl_mnp_tfidf_m0_3np0_7	0,355	0,129	0,000	0,000	0,000	0,000	0,083	0,027
cl_mnp_tfidf_m0_5np0_5	0,299	0,083	0,000	0,000	0,000	0,000	0,083	0,024
cl_mnp_tfidf_m0_7np0_3	0,263	0,044	0,000	0,000	0,000	0,000	0,134	0,024
cl_mnp_tfidf_m0_9np0_1	0,298	0,098	0,000	0,000	0,025	0,000	0,271	0,050
cl_mots_jacc	0,605	0,246	0,346	0,036	0,174	0,028	0,395	0,188
cl_mots_tfidf	0,422	0,148	0,000	0,000	0,063	0,016	0,369	0,131
cl_np_jacc	-	0,158	0,000	0,000	-	0,000	-	0,015
cl_np_tfidf	-	0,116	0,000	0,000	-	0,000	-	0,024
cl_tmnp_tfidf_m1np0_1	0,298	0,098	0,000	0,000	0,025	0,000	0,271	0,050
cl_tmnp_tfidf_m1np0_2	0,256	0,045	0,000	0,000	0,025	0,000	0,237	0,030
cl_tmnp_tfidf_m1np0_3	0,254	0,044	0,000	0,000	0,000	0,000	0,131	0,015
cl_tmnp_tfidf_m1np0_5	0,246	0,036	0,000	0,000	0,000	0,000	0,104	0,000
cl_tmnp_tfidf_m1np0_7	0,299	0,083	0,000	0,000	0,000	0,000	0,090	0,000
cl_tsmots_jacc	0,503	0,232	0,040	0,000	0,096	0,016	0,176	0,072
cl_tsmots_tfidf	0,335	0,074	0,000	0,000	0,000	0,000	0,218	0,035

entropie (200 textes) nombre de classes	of2th1_50_10		th1_50_11		of2th1_50_12	
	7	16	5	14	4	14
cl_mnp_tfidf_m0_1np0_9						
cl_mnp_tfidf_m0_2np0_8	0,059	0,039	0,158	0,030	0,133	0,103
cl_mnp_tfidf_m0_3np0_7	0,037	0,026	0,023	0,000	0,133	0,103
cl_mnp_tfidf_m0_5np0_5	0,048	0,016	0,000	0,000	0,111	0,072
cl_mnp_tfidf_m0_7np0_3	0,095	0,049	0,030	0,000	0,111	0,063
cl_mnp_tfidf_m0_9np0_1	0,387	0,144	0,060	0,043	0,139	0,074
cl_mots_jacc	0,279	0,085	0,180	0,065	0,262	0,153
cl_mots_tfidf	0,370	0,169	0,072	0,043	0,164	0,124
cl_np_jacc	-	0,016	-	0,174	-	0,092
cl_np_tfidf	-	0,000	-	0,089	-	0,113
cl_tmnp_tfidf_m1np0_1	0,387	0,144	0,072	0,043	0,139	0,074
cl_tmnp_tfidf_m1np0_2	0,333	0,077	0,060	0,043	0,100	0,086
cl_tmnp_tfidf_m1np0_3	0,228	0,071	0,044	0,000	0,100	0,072
cl_tmnp_tfidf_m1np0_5	0,095	0,049	0,044	0,000	0,117	0,063
cl_tmnp_tfidf_m1np0_7	0,054	0,016	0,000	0,000	0,111	0,072
cl_tsmots_jacc	0,092	0,056	0,169	0,030	0,156	0,136
cl_tsmots_tfidf	0,103	0,055	0,000	0,000	0,126	0,067

Les résultats de la pureté sont listés dans les tableaux suivants :

pureté (200 textes) nombre de classes	of1th1_200_2		of1th1_200_3		of1th1_200_4		of1th1_200_5	
	15	35	15	37	16	44	16	41
cl_mnp_tfidf_m0_1np0_9	0,836	0,892	0,933	0,928	0,782	0,807	0,892	0,867
cl_mnp_tfidf_m0_2np0_8	0,821	0,913	0,948	0,928	0,797	0,822	0,872	0,882
cl_mnp_tfidf_m0_3np0_7	0,805	0,923	0,954	0,933	0,817	0,847	0,892	0,892
cl_mnp_tfidf_m0_5np0_5	0,851	0,949	0,969	0,964	0,847	0,866	0,911	0,926
cl_mnp_tfidf_m0_7np0_3	0,821	0,933	0,928	0,943	0,817	0,881	0,862	0,916
cl_mnp_tfidf_m0_9np0_1	0,795	0,913	0,887	0,938	0,851	0,906	0,813	0,911
cl_mots_jacc	0,533	0,687	0,577	0,742	0,743	0,827	0,611	0,714
cl_mots_tfidf	0,646	0,892	0,732	0,845	0,866	0,871	0,695	0,867
cl_np_jacc	-	0,856	-	0,964	-	0,757	-	0,852
cl_np_tfidf	-	0,892	-	0,933	-	0,787	-	0,897
cl_tmnp_tfidf_m1np0_1	0,785	0,923	0,876	0,918	0,817	0,886	0,813	0,911
cl_tmnp_tfidf_m1np0_2	0,856	0,933	0,928	0,948	0,832	0,901	0,847	0,911
cl_tmnp_tfidf_m1np0_3	0,815	0,938	0,928	0,938	0,847	0,906	0,897	0,926
cl_tmnp_tfidf_m1np0_5	0,826	0,933	0,892	0,938	0,817	0,881	0,892	0,906
cl_tmnp_tfidf_m1np0_7	0,836	0,944	0,938	0,948	0,842	0,876	0,906	0,931
cl_tsmots_jacc	0,590	0,810	0,825	0,856	0,698	0,787	0,739	0,833
cl_tsmots_tfidf	0,815	0,892	0,943	0,948	0,856	0,861	0,906	0,901

pureté (200 textes) nombre de classes	of1th1_200_6		of1th1_200_7		of1th1_200_8		of1th1_200_9	
	16	44	16	34	16	38	16	47
cl_mnp_tfidf_m0_1np0_9	0,882	0,892	0,906	0,872	0,948	0,943	0,782	0,838
cl_mnp_tfidf_m0_2np0_8	0,892	0,882	0,926	0,897	0,927	0,953	0,787	0,838
cl_mnp_tfidf_m0_3np0_7	0,897	0,922	0,867	0,867	0,943	0,969	0,777	0,904
cl_mnp_tfidf_m0_5np0_5	0,897	0,917	0,926	0,921	0,927	0,953	0,812	0,904
cl_mnp_tfidf_m0_7np0_3	0,863	0,897	0,921	0,921	0,953	0,948	0,822	0,904
cl_mnp_tfidf_m0_9np0_1	0,740	0,843	0,862	0,897	0,828	0,875	0,680	0,838
cl_mots_jacc	0,574	0,740	0,700	0,778	0,630	0,708	0,558	0,701
cl_mots_tfidf	0,696	0,819	0,867	0,906	0,599	0,807	0,716	0,817
cl_np_jacc	-	0,853	-	0,823	-	0,927	-	0,721
cl_np_tfidf	-	0,897	-	0,872	-	0,932	-	0,792
cl_tmnp_tfidf_m1np0_1	0,721	0,843	0,857	0,897	0,849	0,875	0,685	0,848
cl_tmnp_tfidf_m1np0_2	0,848	0,882	0,936	0,921	0,901	0,927	0,772	0,873
cl_tmnp_tfidf_m1np0_3	0,828	0,863	0,897	0,921	0,932	0,943	0,761	0,878
cl_tmnp_tfidf_m1np0_5	0,902	0,936	0,892	0,882	0,932	0,927	0,782	0,898
cl_tmnp_tfidf_m1np0_7	0,868	0,936	0,931	0,916	0,943	0,943	0,787	0,893
cl_tsmots_jacc	0,681	0,877	0,813	0,852	0,859	0,906	0,670	0,777
cl_tsmots_tfidf	0,892	0,917	0,892	0,916	0,958	0,943	0,822	0,878

pureté (200 textes) nombre de classes	of2th1_200_3		of2th1_200_4		of2th1_200_5		of2th1_200_6	
	15	33	15	41	16	40	14	33
cl_mnp_tfidf_m0_1np0_9	0,825	0,904	0,701	0,779	0,864	0,887	0,974	0,945
cl_mnp_tfidf_m0_2np0_8	0,825	0,928	0,711	0,804	0,891	0,900	0,995	0,975
cl_mnp_tfidf_m0_3np0_7	0,845	0,933	0,799	0,851	0,881	0,910	0,990	0,970
cl_mnp_tfidf_m0_5np0_5	0,840	0,938	0,845	0,881	0,910	0,915	0,990	0,980
cl_mnp_tfidf_m0_7np0_3	0,809	0,943	0,737	0,881	0,886	0,905	0,975	0,975
cl_mnp_tfidf_m0_9np0_1	0,747	0,928	0,686	0,856	0,900	0,891	0,823	0,944
cl_mots_jacc	0,464	0,680	0,582	0,789	0,652	0,806	0,545	0,753
cl_mots_tfidf	0,588	0,881	0,716	0,835	0,781	0,876	0,611	0,879
cl_np_jacc		0,928	-	0,753	-	0,851	-	0,944
cl_np_tfidf		0,918	-	0,789	-	0,896	-	0,960
cl_tmnp_tfidf_m1np0_1	0,711	0,902	0,686	0,856	0,881	0,886	0,843	0,955
cl_tmnp_tfidf_m1np0_2	0,830	0,943	0,716	0,861	0,866	0,881	0,919	0,975
cl_tmnp_tfidf_m1np0_3	0,840	0,943	0,742	0,861	0,910	0,900	0,960	0,980
cl_tmnp_tfidf_m1np0_5	0,820	0,948	0,814	0,861	0,896	0,905	0,970	0,965
cl_tmnp_tfidf_m1np0_7	0,794	0,938	0,799	0,871	0,905	0,920	0,980	0,970
cl_tsmots_jacc	0,727	0,814	0,655	0,840	0,836	0,886	0,803	0,848
cl_tsmots_tfidf	0,866	0,948	0,845	0,871	0,910	0,896	0,985	0,980

pureté (200 textes) nombre de classes	of2th1_200_7		of2th1_200_8		of2th1_200_9	
	15	46	13	50	16	35
cl_mnp_tfidf_m0_1np0_9	0,904	0,853	0,731	0,820	0,917	0,917
cl_mnp_tfidf_m0_2np0_8	0,854	0,869	0,775	0,824	0,926	0,926
cl_mnp_tfidf_m0_3np0_7	0,854	0,893	0,770	0,838	0,926	0,931
cl_mnp_tfidf_m0_5np0_5	0,893	0,913	0,765	0,858	0,965	0,965
cl_mnp_tfidf_m0_7np0_3	0,874	0,888	0,784	0,848	0,960	0,950
cl_mnp_tfidf_m0_9np0_1	0,845	0,913	0,716	0,828	0,837	0,921
cl_mots_jacc	0,782	0,854	0,578	0,799	0,619	0,782
cl_mots_tfidf	0,840	0,913	0,745	0,809	0,762	0,881
cl_np_jacc	-	0,825	-	0,770	-	0,901
cl_np_tfidf	-	0,879	-	0,809	-	0,911
cl_tmnp_tfidf_m1np0_1	0,845	0,913	0,730	0,809	0,891	0,941
cl_tmnp_tfidf_m1np0_2	0,845	0,898	0,721	0,828	0,921	0,946
cl_tmnp_tfidf_m1np0_3	0,908	0,903	0,716	0,848	0,916	0,946
cl_tmnp_tfidf_m1np0_5	0,903	0,888	0,775	0,848	0,965	0,955
cl_tmnp_tfidf_m1np0_7	0,883	0,898	0,784	0,868	0,970	0,970
cl_tsmots_jacc	0,835	0,908	0,711	0,824	0,842	0,876
cl_tsmots_tfidf	0,917	0,922	0,843	0,858	0,931	0,950

pureté (200 textes) nombre de classes	of1th2_200_5		of1th2_200_6		of1th2_200_7		of1th2_200_8	
	8	43	13	53	11	39	9	41
cl_mnp_tfidf_m0_1np0_9	0,640	0,751	0,846	0,831	0,915	0,896	0,845	0,866
cl_mnp_tfidf_m0_2np0_8	0,685	0,807	0,915	0,861	0,841	0,891	0,861	0,881
cl_mnp_tfidf_m0_3np0_7	0,726	0,832	0,940	0,881	0,876	0,900	0,887	0,892
cl_mnp_tfidf_m0_5np0_5	0,853	0,878	0,945	0,881	0,945	0,925	0,861	0,892
cl_mnp_tfidf_m0_7np0_3	0,893	0,898	0,955	0,910	0,935	0,925	0,918	0,923
cl_mnp_tfidf_m0_9np0_1	0,843	0,919	0,980	0,925	0,816	0,930	0,835	0,902
cl_mots_jacc	0,624	0,868	0,945	0,886	0,557	0,801	0,696	0,887
cl_mots_tfidf	0,741	0,893	0,985	0,925	0,627	0,866	0,871	0,907
cl_np_jacc	-	0,635	-	0,706	-	0,866	-	0,835
cl_np_tfidf	-	0,721	-	0,836	-	0,915	-	0,851
cl_tmnp_tfidf_m1np0_1	0,848	0,919	0,980	0,920	0,891	0,930	0,892	0,918
cl_tmnp_tfidf_m1np0_2	0,827	0,904	0,960	0,920	0,930	0,920	0,928	0,928
cl_tmnp_tfidf_m1np0_3	0,736	0,909	0,965	0,925	0,955	0,935	0,907	0,923
cl_tmnp_tfidf_m1np0_5	0,888	0,904	0,955	0,905	0,960	0,930	0,918	0,923
cl_tmnp_tfidf_m1np0_7	0,853	0,898	0,970	0,896	0,935	0,925	0,933	0,923
cl_tsmots_jacc	0,726	0,843	0,970	0,891	0,766	0,905	0,686	0,825
cl_tsmots_tfidf	0,797	0,929	0,970	0,925	0,945	0,950	0,943	0,907

pureté (200 textes) nombre de classes	of1th2_200_10		of1th2_200_11		of1th2_200_12	
	12	54	16	49	14	45
cl_mnp_tfidf_m0_1np0_9	0,845	0,821	0,879	0,864	0,927	0,902
cl_mnp_tfidf_m0_2np0_8	0,850	0,831	0,899	0,879	0,922	0,912
cl_mnp_tfidf_m0_3np0_7	0,841	0,865	0,874	0,884	0,933	0,927
cl_mnp_tfidf_m0_5np0_5	0,874	0,903	0,925	0,925	0,948	0,933
cl_mnp_tfidf_m0_7np0_3	0,932	0,918	0,945	0,925	0,938	0,943
cl_mnp_tfidf_m0_9np0_1	0,879	0,923	0,899	0,910	0,974	0,938
cl_mots_jacc	0,628	0,865	0,719	0,844	0,803	0,922
cl_mots_tfidf	0,908	0,908	0,884	0,894	0,891	0,927
cl_np_jacc	-	0,667	-	0,749	-	0,860
cl_np_tfidf	-	0,783	-	0,864	-	0,896
cl_tmnp_tfidf_m1np0_1	0,836	0,913	0,925	0,905	0,974	0,938
cl_tmnp_tfidf_m1np0_2	0,942	0,928	0,920	0,910	0,974	0,938
cl_tmnp_tfidf_m1np0_3	0,908	0,923	0,930	0,915	0,964	0,943
cl_tmnp_tfidf_m1np0_5	0,928	0,913	0,930	0,925	0,933	0,943
cl_tmnp_tfidf_m1np0_7	0,923	0,928	0,935	0,910	0,948	0,953
cl_tsmots_jacc	0,874	0,870	0,794	0,894	0,896	0,922
cl_tsmots_tfidf	0,812	0,928	0,970	0,935	0,964	0,927

pureté (50 textes) nombre de classes	of1th1_50_10		of1th1_50_11		of1th1_50_12		of1th1_50_13	
	6	20	4	6	5	14	6	12
cl_mnp_tfidf_m0_1np0_9	0,667	0,722	0,722	0,960	0,982	0,927	0,923	0,865
cl_mnp_tfidf_m0_2np0_8	0,685	0,722	0,722	0,960	0,982	0,927	0,923	0,904
cl_mnp_tfidf_m0_3np0_7	0,685	0,722	0,722	0,960	0,982	0,945	0,923	0,904
cl_mnp_tfidf_m0_5np0_5	0,741	0,759	0,759	0,960	0,982	0,982	0,923	0,904
cl_mnp_tfidf_m0_7np0_3	0,778	0,870	0,870	0,960	0,982	0,945	0,885	0,923
cl_mnp_tfidf_m0_9np0_1	0,704	0,796	0,796	0,980	0,964	0,927	0,731	0,904
cl_mots_jacc	0,463	0,611	0,611	0,700	0,873	0,927	0,692	0,731
cl_mots_tfidf	0,667	0,778	0,778	0,980	0,945	0,927	0,731	0,827
cl_np_jacc	-	0,630	0,630	0,980	-	0,873	-	0,885
cl_np_tfidf	-	0,704	0,704	0,960	-	0,909	-	0,865
cl_tmnp_tfidf_m1np0_1	0,704	0,796	0,796	0,960	0,964	0,927	0,731	0,904
cl_tmnp_tfidf_m1np0_2	0,759	0,852	0,852	0,960	0,964	0,927	0,750	0,923
cl_tmnp_tfidf_m1np0_3	0,759	0,870	0,870	0,960	0,982	0,945	0,865	0,962
cl_tmnp_tfidf_m1np0_5	0,796	0,870	0,870	0,960	0,982	0,945	0,904	0,942
cl_tmnp_tfidf_m1np0_7	0,741	0,796	0,796	0,960	0,982	0,964	0,923	0,923
cl_tsmots_jacc	0,537	0,685	0,685	0,960	0,927	0,873	0,865	0,827
cl_tsmots_tfidf	0,630	0,796	0,796	0,960	0,982	0,945	0,827	0,885

pureté (200 textes) nombre de classes	of2th1_50_10		of2th1_50_11		of2th1_50_12	
	7	16	5	14	4	14
cl_mnp_tfidf_m0_1np0_9						
cl_mnp_tfidf_m0_2np0_8	0,962	0,868	0,885	0,885	0,885	0,885
cl_mnp_tfidf_m0_3np0_7	0,981	0,906	0,981	0,885	0,981	0,885
cl_mnp_tfidf_m0_5np0_5	0,943	0,925	0,981	0,904	0,981	0,904
cl_mnp_tfidf_m0_7np0_3	0,943	0,943	0,981	0,942	0,981	0,942
cl_mnp_tfidf_m0_9np0_1	0,679	0,868	0,962	0,904	0,962	0,904
cl_mots_jacc	0,792	0,868	0,885	0,904	0,885	0,904
cl_mots_tfidf	0,660	0,811	0,962	0,923	0,962	0,923
cl_np_jacc	-	0,868	-	0,808	-	0,808
cl_np_tfidf	-	0,925	-	0,865	-	0,865
cl_tmnp_tfidf_m1np0_1	0,679	0,868	0,962	0,904	0,962	0,904
cl_tmnp_tfidf_m1np0_2	0,755	0,906	0,962	0,904	0,962	0,904
cl_tmnp_tfidf_m1np0_3	0,811	0,925	0,962	0,942	0,962	0,942
cl_tmnp_tfidf_m1np0_5	0,943	0,943	0,962	0,942	0,962	0,942
cl_tmnp_tfidf_m1np0_7	0,962	0,962	0,981	0,942	0,981	0,942
cl_tsmots_jacc	0,943	0,943	0,846	0,923	0,846	0,923
cl_tsmots_tfidf	0,943	0,925	0,981	0,942	0,981	0,942

## PUBLICATIONS

---

- Friburger N., Maurel D. (2002), *Finite-State Transducer Cascade to Extract Named Entities in Texts*, In Theoretical Computer Science, en soumission.
- Friburger N., Maurel D. (2002), *Similarités entre textes basées sur les noms propres*, Conférence Internationale sur la fouille de texte (CIFT 2002), Hamamet, Tunisie, 20-23 octobre 2002, p.1-13.
- Friburger N., Maurel D. (2002), *Textual similarity based on proper names*, Mathematical Formal Information Retrieval (MFIR'2002), Tampere, Finlande, 15 août 2002, p.155-167.
- Friburger, N. (2002), *Cascade de transducteurs pour INTEX : Un nouvel outil*, In 5<sup>ème</sup> Journées Intex, Marseille, France, 11-12 juin 2002.
- Friburger N., Maurel D. (2001), *Finite-State Transducer Cascade To Extract Proper Nouns in Texts*, In Proceedings of 2<sup>nd</sup> Conference on Implementing and Application of Automata (CIAA'2001), Pretoria, South Africa, 23-25 juillet 2001. (à paraître dans LNCS)
- Friburger N., Maurel D. (2001), *Pré-traitement pour l'extraction de connaissances sur les anthroponymes*, In Journées francophones d'Extraction et de Gestion des Connaissances (EGC 2001), Nantes, France, 17-19 janvier 2001, p.239-249.
- Friburger N. (2000), *Pré-traitement pour une fouille de textes basée sur les noms propres*, In Récital 2000, Lausanne, Suisse, 13-14 juin 2000, p.471-476.
- Friburger N., Dister A., Maurel D. (2000), *Améliorer le découpage des phrases sous Intex*, dans Revue Informatique et Statistique dans les Sciences Humaines (RISSH), Belgique, Vol. 36, n°1-4, p. 181-199.
- Friburger N., Silberztein M. (1999), *Le débogueur de grammaires sous Intex*, In Linguisticae Investigationes, Cédric Fairon Ed., Vol. 22, p. 399-410.



## BIBLIOGRAPHIE

---

- Aberdeen, J., Burger, J., Day, D., Hirschman, L., Robinson, P. et Vilain, M. (1995). *MITRE: Description of the Alembic system used for MUC-6*, Dans Proceedings of the Sixth Message Understanding Conference (MUC-6), Columbia, Maryland, p.141-155.
- Abney, S. (1991). *Parsing By Chunks*, Dans Principle-Based Parsing, Berwick, R., Abney, S. et Tenny, C. ed., Computation and Psycholinguistics, Kluwer Academic Publishers, Dordrecht, p. 257-278.
- Abney, S. (1996). *Partial Parsing via Finite-State Cascades*, Dans Proc. of Workshop on Robust Parsing, 8th European Summer School in Logic, Language and Information, Prague, Czech Republic, p. 8-15.
- Abney, S., Collins, M. et Singhal, A. (2000). *Answer extraction*, Dans Proc. of ANLP'2000, p. 296-301.
- Aït-Mokhtar, S. et Chanod, J. (1997). *Incremental Finite State Parsing*, Dans Proc. of ANLP'97, p. 72-79.
- Allerton, D. (1987). *The linguistic and sociolinguistic status of proper names*, Journal of Pragmatics, vol. 11, p. 61-92.
- Appelt, D., Hobbs, J., Israel, D. et Tyson, M. (1993). *Fastus: A Finite-State processor for information extraction from real-world text*, Dans Proc. of IJCAI'93, p. 1172-1178.
- Bauer, G. (1998). *Deutsche Namenkunde*, Hans-Gert Roloff ed., Dans Germanistische Lehrbuchsammlung, Weidler Buchverlag, Berlin, 356 p.
- Béchet, F., Nasr, A., Genet, F. (2000). *Tagging Unknown Proper Names Using Decision Trees*, *Proceedings of the 38th Meeting of the Association for Computational Linguistics*, p. 77-84.
- Belleil, C. (1997). *Reconnaissance, typage et traitement des coréférences des toponymes français et de leurs gentilés par dictionnaire électronique relationnel*, Thèse soutenue à l'Université de Nantes, Nantes.
- Belleil, C. et Maurel, D. (1997). *Un dictionnaire relationnel des noms propres liés à la géographie, consultés par transducteurs*, Dans *Meta : Journal des traducteurs*, Presses de l'Université de Montréal, Vol. 42(2), Montréal, Canada, p. 273-282.
- Bellot, P. et El-Bèze, M. (2001). *Classification locale non supervisée pour la recherche documentaire*, Dans *Traitement Automatique des Langues (TAL)*, Hermès, Vol. 41(2), p. 355-366.
- Bellot, P. (2000). *Méthodes de classification et de segmentation locales non supervisées pour la recherche documentaire*, Thèse de Doctorat de l'Université d'Avignon, janvier 2000.

- Besançon, R. (2001). Intégration de connaissances syntaxiques et sémantiques dans les représentations vectorielles de textes, Thèse soutenue à l'Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Suisse, 190 p.
- Bikel, D. M., Miller, S., Schwartz, R. et Weischedel, R. (1997). *Nymble: a high-performance learning name-finder.*, Dans Proc. of Fifth Conference on Applied Natural Language Processing, p. 194-200.
- Black, W. J., Rinaldi, F. et Mowatt, D. (1998). *FACILE: Description of the NE System Used for MUC-7*, Dans Proc. of MUC-7, [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Boisen, S., Crystal, M. R., Schwartz, R., Stone, R. et Weischedel, R. (2000). *Annotating resources for Information Extraction*, Dans Actes 2 nd International Conference on Linguistic Resources and Evaluation (LREC'2000), Athens, Greece, 31 May- 2 June 2000, p. 1211-1214.
- Boley, D. (1998). *Principal Direction Divisive Partitioning*, Data Mining and Knowledge Discovery, Vol.2(4), p. 325-344.
- Borkowski, C. (1967). An Experimental System for Automatic Identification of Personal Names and Personal Titles in Newspaper Texts, American Documentation, Vol. 18(3), p. 131-138.
- Borthwick, A., Sterling, J., Agichtein, E. et Grishman, R. (1998). *NYU: Description of the MENE Named Entity System as used in MUC-7*, Dans Proc. of the Seventh Message Understanding Conference (MUC-7), [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Brill, E. (1995). *Unsupervised learning of disambiguation rules for Part of Speech tagging*, Dans Proceedings of the Third Workshop on Very Large Corpora, Cambridge, MA, p. 1-13.
- Burnard, L. et Sperberg-McQueen, C. (1995). *Encoding for Interchange: An Introduction to the TEI*, Dans Technical Report TEI J31, [www.tei-c.org/Lite/](http://www.tei-c.org/Lite/), London.
- Chanod, J. et tapanainen, P. (1996). *A Robust Finite-State Parser for French*, Dans Proc. of Workshop on Robust Parsing, Prague, Czech, p. 16-25.
- Chinchor, N. A. (1997). *Overview of MUC-7 / MET-2*, [http://www.itl.nist.gov/iaui/894.02/related\\_projects/muc/proceedings/muc\\_7\\_toc.html#appendices](http://www.itl.nist.gov/iaui/894.02/related_projects/muc/proceedings/muc_7_toc.html#appendices),
- Church, K. (1988). *A stochastic parts program and noun phrase parser for unrestricted text*, Dans Proceedings of the Second Conference on Applied Natural Language Processing, Computer Speech and Language ed., Vol.5, p. 136-143.
- Ciravegna, F. et Lavelli, A. (1999). *Full text parsing using cascades of rules: An information extraction perspective*, Dans Proceedings of EACL'99, Bergen, Norway, p. 102-109.
- Clifton, C. et Cooley, R. (1999). *TopCat: data mining for topic identification in a text corpus*, Dans Proc. of the 3rd European Conference of Principles and Practice of Knowledge Discovery in Databases, Prague, Czech Republic, September 15-18, 1999, Lectures Notes in Artificial Intelligence, Vol. 1704, Springer-Verlag.

- Coates-Stephens, S. (1993). *The Analysis and Acquisition of Proper Names for the Understanding of Free Text*, Dans *Computers and the Humanities*, Kluwer Academic Publishers, Vol. 26(5-6), Hingham, MA, p. 441-456.
- Collins, M. et Singer, Y. (1999). *Unsupervised models for named entity classification*, Dans *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*.
- Courtois, B. et Silberztein, M. (1990). *Dictionnaire électronique des mots simples du français*, Langue Française, Vol. 87, Larousse, Paris, p. 11-22.
- Cowie, J. et Lehnert, W. (1996). *Information Extraction*, Dans *ACM Special Issue on Natural Language Processing*, Vol. 39(1), p. 80-101.
- Cucchiarelli, I., Luzi, D. et Velardi, P. (1998). *Automatic semantic tagging of unknown proper names*, Dans *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and Proceedings of the 17th International Conference on Computational Linguistics*, Montreal, Canada, p. 286-292.
- Cucerzan, S. et Yarowsky, D. (1999). *Language independent named entity recognition combining morphological and contextual evidence*, Dans *Proc. of 1999 Joint SIGDAT Conference on EMNLP and VLC*, p. 90-99.
- Cutting, D. R., Karger, D. R., Pedersen, J. O., et Tukey, J. W. (1992). *Scatter/Gather : A Cluster-based Approach to Browsing Large Document Collections*, *Proceedings of SIGIR'92*, ACM Press, New York, p. 318-329.
- Daille, B. et Morin, E. (2000). *Reconnaissance automatique des noms propres de la langue écrite : les récentes réalisations*, Dans *Traitement Automatique des Langues*, Maurel, D. et Gueunthner, F. ed., Vol. 41(3), p. 601-621.
- Daille, B. et Williams, G. (2001). *Extraction de collocations à partir de textes*, Dans *Actes de TALN'2001*, Tours, France, p. 3-16.
- Dalianas, H. et Aström, E. (1998). *SweNam - A swedish Named Entity Recognizer. Its construction, training and evaluation*, Dans *Proc. of MUC-7*, [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Dejong, G. (1982). *An Overview of the Frump System*, W.B. Lehnert and M.H. Ringle ed., Dans *Strategies for Natural Language Processing*, Erlbaum, p. 149-176.
- Dister, A. (1997). *Problématique des fins de phrase en traitement automatique du français*, Dans *Champs Linguistiques*, Dehays, J., Rosier, L. et Tilkin, F. ed., Duculot, Paris, p. 470.
- Eggert, E., Maurel, D. et Belleil, C. (1998). *Allomorphies et suppléments dans la formation des gentilés : application au traitement informatique*, *Cahiers de lexicologie*, Vol. 73(2), p. 167-179.
- Fairon, C. et Senellart, J. (1999). *Réflexions sur la localisation, l'étiquetage, la reconnaissance et la traduction d'expressions linguistiques complexes*, Dans *Actes de TALN'99*, Cargèse, Corse, p. 135-143.
- Fairon, C. (2000). *Structures non-connexes. Grammaire des incises en français : description linguistique et outils informatiques*, Dans *Thèse de doctorat en informatique*, Université Paris 7, 183 p.

- Flaux, N. (1991). *L'antonomase du nom propre ou la mémoire du référent*, Langue Française, Vol. 92, p. 26-45.
- Forsgren, M. (1994). *Nom propre, référence, prédication et fonction grammaticale*, Dans Nom propre et nomination (Actes du colloque de Brest), Noailly, M. ed., p. 95-106.
- Fourour, N. (2002). *Nemesis, un système de reconnaissance incrémentielle des entités nommées pour le français*, Dans Actes de TALN'2002, Nancy, France, p. 265-274.
- Fox, E. A. (1983). *Extending the Boolean and Vector Space Models of Information Retrieval with Pnorm Queries and Multiple Concept Types*, Dans PhD thesis, University Microfilms, Ann Arbor, MI., Cornell.
- Frakes, W. B. et Baeza-Yates, R. (1992). *Information Retrieval: Data Structures and Algorithms*, Prentice-HALL, North Virginia.
- Fuhr, N. et Buckley, C. (1991). *A Probabilistic Learning Approach for Document Indexing*, Dans ACM Transactions on Information Systems, Vol. 9(3), p. 223-248.
- Gaizauskas, R., Wakao., Humphreys, K., Cunningham, H. et Wilks, Y. (1995). *University of Sheffield: Description of the LaSIE System as used for MUC-6*, Dans Proceedings of the Sixth Message Understanding Conference (MUC-6), Morgan Kaufmann, p. 207-220.
- Gala Pavia, N. (2000). Using Incremental Finite-State Architecture to create a Spanish Shallow Parser, Dans Proc. of TALN'2000, Lausanne, Suisse, p. 477-482.
- Gale, W., K. Church, K. et Yarowsky, D. (1992). *One sense per discourse*, Dans Proceedings of the DARPA Speech and Natural Language Workshop, Harriman, New York, p. 233-237.
- Gallippi, A. (1996). *Learning to Recognize Names Across Languages*, Dans Proc. of the 16th International Conference on Computational Linguistics (COLING'96), Copenhagen, Danemark, p. 424-429.
- Garric, N. et Maurel, D. (2000). *Désambiguïsation des noms propres déterminés par l'utilisation des grammaires locales*, Dans Revue française de Linguistique appliquée, Vol.5(2), p. 85-100.
- Gary-Prieur, M. N. (1994). *Grammaire du nom propre*, Presse universitaire de France, Paris, 252 p.
- Gershman, A. (1977). *Conceptual Analysis of Noun Groups in English*, Dans Proc. of IJCAI, p. 132-138.
- Grass, T. (2000). *Typologie et traductibilité des noms propres de l'allemand vers le français à partir d'un corpus journalistique*, Maurel, D. et Gueunthner, F. ed., Dans t.a.l., Vol. 41(3), p. 643-669.
- Grass, T., Maurel, D. et Piton, O. (2002). *Description of a Multilingual Database of Proper Names*, Dans Proc. of Portal 2002, LNCS, 23-26 juillet 2002, Faro, Portugal, p. 137-150.
- Grevisse, M. et Goosse, A. (1986). *Le Bon Usage*, Duculot, Gembloux, Belgique, 1768 p.

- Grishman, R. Sundheim, B. (1996). *Message Understanding Conference - 6: a brief history*, Dans Proc. of 16th International Conference on Computational Linguistics (COLING-96), Morgan Kaufmann, California, p. 466-471.
- Grishman, R. (1997). *Information Extraction : Techniques and Challenges*, Information Extraction: A multidisciplinary Approach to an Emerging Information Technology, Vol.1299, p. 10-27.
- Gross, G. (1994). *Classes d'objets et description des verbes*, Dans Langages, Larousse, Vol.115, p. 15-30.
- Grover, C., Matheson, C., Mikheev, A. et Moens, M. (2000). *LT TTT - A flexible Tokenisation Tool*, in Proceedings of the Second Language Resources and Evaluation Conference (LREC 2000), 31 May-2 June 2000, Athens, Greece.
- Guerrin, C. (1998). *Etude socio-toponymique de la variation dans les noms de communes françaises entre 1943 et 1996*, Thèse soutenue à l'Université de Rouen, France.
- Hayes, P. (1994). *NameFinder: Software that Finds Names in Text*, Dans Proc. of the 4th RIAO Conference of Computer Assisted Information Searching on the Internet, Vol. 1, m Oct. 1994, New York, p. 762-774.
- Hiemstra, D. (2001). *Using language models for information retrieval*, Dans these, university of Twente, Enschede, 164 p.
- Hobbs, J. R., Appelt, D., Bear, J., Israel, D., Kameyama, M., Stickel, M. et Tyson, M. (1996). *FASTUS: A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text*, Roche, E. et Schabes, Y. ed., Dans Finite State Devices for Natural Language Processing, MIT Press, Cambridge, MA, p. 383-406.
- Jacquemin, C. et Bush, C. (2000). *Combining Lexical and Formatting Cues for Named Entity Acquisition from the Web*, Dans Proc. Joint SIGDAT Conference on Empirical Methods in NLP and Very Large Corpora, Hong Kong, p. 181-189.
- Jacquemin, C. (2001). *Spotting and Discovering Terms through Natural Language Processing*, MIT Press, Cambridge, Mass, 357 p.
- Joachims, T. (1997). *A probabilistic analysis of the Rocchio algorithm with TFIDF for text categorization*, Dans International Conference on Machine Learning (ICML).
- Jonasson, K. (1994). *Le nom propre. Constructions et interprétations*, Duculot, Paris, 255 p.
- Jones, B. E. (1994). *Exploring the Role of Punctuation in Parsing Natural Text*, Dans Proceedings of COLING'94, p. 421-425.
- Karkaletsis, V., Spyropoulos, C. et Petasis, G. (1999). *Named Entity Recognition from Greek Texts: the GIE Project*, Tzafestas, S. ed., Dans Advances in Intelligent Systems: Concepts, Tools and Applications, Kluwer Academic Publishers, p. 131-142.
- Kim, J. et Evens, M. W. (1996). *Efficient Coreference Resolution for Proper Names in the Wall Street Journal Text*, Dans online Proceedings of MAICS'96, Bloomington, <http://www.cs.indiana.edu/event/maics96/Proceedings/Kim/kim.html>.

- Kleiber, G. (1991). *Du nom propre non modifié au nom propre modifié : le cas de la détermination des noms propres par l'adjectif démonstratif*, Langue française, Vol. 92, p. 82-103.
- Kleiber, G. (1994). *Sur la définition des noms propres : une dizaine d'années après*, Dans Nom propre et nomination (Actes du colloque de Brest), Noailly, M. ed., p. 11-36.
- Kokkinakis, D. et Kokkinakis, S. J. (1999). *A Cascaded Finite-State Parser for Syntactic Analysis of Swedish*, Dans Proc. of the 9th EACL, Bergen, Norway, p. 245-248.
- Kosseim, L. et Poibeau, T. (2001). *Proper Name Extraction from non-journalistic Texts*, Dans Proc. of the 11th Conference Computational Linguistics in the Netherlands, W. Daelemans, K. Sima'an, J. Veenstra and J. Zavrel Ed., Tilburg, Pays-Bas, p. 144-157.
- Krovetz, R. et Croft, W. (1992). *Lexical Ambiguity and Information Retrieval*, ACM Transactions on Information Systems, Vol. 10(2), p. 115-141.
- Krupka, G. R. et Hausman, K. (1998). *IsoQuest Inc.: Description of the NetOwlTM Extractor System as Used for MUC-7*, Dans Proc. of the Seventh Message Understanding Conference (MUC-7), [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Kuhns, R. (1988). *A news Analysis System*, Dans Proc. of COLING'1988, Budapest, Hungary, p. 351-355.
- Laporte, E. (1997). Les mots : un demi-siècle de traitement sur les mots. Dans TAL *État de l'art*, Hermès, Vol. 38(2), p. 47-68.
- Lin, D. (1998). *Using collocation statistics in information extraction*, Dans Proceedings of the Seventh Message Understanding Conference (MUC-7), [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Liu, F. et Hass, L. (1988). *Synthetic speech technology for enhancement of voice-store-abd-forward-systems*, Dans Proc. of the American Voice Input/Output Society.
- Maiorano, S. et Wilson, T. (1996). *Multilingual Entity task (MET): Japanese results*, Dans Proc. of Tipster Test Program (Phase II).
- Mani, I.; MacMillan, R. T. (1996). *Identifying Unknown Proper Names in Newswire Text*, Boguraev, B. et Pustejovsky, J. ed., Dans Corpus Processing for Lexical Acquisition, MIT Press, Cambridge, MA, p. 41-59.
- Maurel, D., Belleil, C., Eggert, E. et Piton, O. (1996). *Le projet PROLEX : réalisation d'un dictionnaire électronique relationnel des noms propres du français*, Dans Proc. of GDR-PRC Communication Homme-Machine Séminaire Lexique, Grenoble, p. 174-175.
- Maurel, D. et Piton, O. (1999). *Un dictionnaire de noms propres pour Intex : Les noms propres géographiques*, Linguisticae Investigationes, Vol. 22, p. 277-287.
- Maurel, D. et Gueunthner, F. éd. (2000). *Le traitement automatique des noms propres*, Dans TAL, Hermès, Vol. 41(3), p. 600-836.

- McDonald, D. D. (1996). *Internal and External Evidence in the Identification and Semantic Categorisation of Proper Names*, Boguraev; Pustejavsky ed., Dans Corpus processing for lexical acquisition, p. 32-43.
- Mikheev, A., Grover, C. et Moens, M. (1998). *Description of the LTG system used for MUC -7*, Dans Proc. of 7th Message Understanding Conference (MUC-7), [http://www.itl.nist.gov/iad/894.02/related\\_projects/muc/](http://www.itl.nist.gov/iad/894.02/related_projects/muc/).
- Mikheev, A., Moens, M. et Grover, C. (1999). *Named Entity Recognition without Gazetteers*, Dans Proc. of the Ninth Conference of the European Chapter of the Association for Computational Linguistics, Bergen, Norway, p. 1-8.
- Miller, D., Schwartz, R., Weischedel, R. et Stone, R. (1999). *Named Entity Extraction from Broadcast News*, Dans Proc. of DARPA Broadcast News Workshop, Herndon, VA, <http://www.nist.gov/speech/publications/darpa99/html/abstract.htm#ie-20>.
- Molino, J. (1982). *Le nom propre dans la langue*, Dans Langage, Larousse, Vol. 66, p. 5-21.
- Moore, J., Eui-Hong, H., Boley, D., Gini, M., Gross, R., Hashings, K., Karypis, G., Kumar, V. et Mobasher, B. (1997). *Web Page Categorization and Feature Selection Using Association Rule and Principal Component Clustering*, Dans Proc. of Workshop on Information Technologies and System, University de Minnesota, Mineapolis.
- Noailly, M. (1991). *"L'énigmatique Tombouctou": nom propre et position de l'épithète*, Langue Française.
- Noailly, M. (1994). *Un nom propre, deux morphologies : pour quoi dire ?*, Dans Nom propre et nomination (Actes du colloque de Brest), Noailly, M. ed., p. 75-84.
- Noailly (1995). *Nom propre et nomination*, Actes du colloque de Brest, Noailly, M. ed., Toulouse, 377 p.
- Paik, W., Liddy, E. D., Yu E.; McKenna M., et McKenna M., (1996). *Categorizing and Standardizing Proper Nouns for efficient Information Retrieval*, Boguraev, B. et Pustejovsky, J. ed., Dans Corpus processing for lexical acquisition, Massachussets Institute of technology, p. 61-73.
- Piton, O. et Maurel, D. (1997). *Le traitement informatique de la géographie politique internationale*, Dans Bulag, numéro spécial, Besançon, p. 321-328.
- Poibeau, T. (1999). *Evaluation des systèmes d'extraction d'information : une expérience sur le français*, Langues (spécial sur l'évaluation), Vol. 2(2), p. 110-118.
- Poibeau, T. (1999). *Repérage des entités nommées : un enjeu pour les systèmes de veille*, Dans Actes du Colloque TIA'99 : Terminologie et Intelligence Artificielle, Vol.19, p. 43-51.
- Popescu-Belis, A. (1999). *Evaluation de la résolution de la référence : critiques et propositions*, Dans TAL, Vol. 40(2), p. 117-146.
- Porter, M. (1980). *An algorithm for suffix stripping*, Program, Vol. 14(3), p. 130-137.
- Rau, L. F. (1991). *Extracting company names from text*, Dans Proc. of the Sixth IEEE Conference on Artificial Intelligence Applications, p. 29-32.

- Rau, L., Jacobs, P. et Zernik, U. (1991). *Information extraction and text summarization using linguistic knowledge acquisition*, Dans *Information processing and management*, Vol. 25(4), p. 419-428.
- Ravin, Y. et Wacholder, N. (1997). *Extracting Names From Natural-Language Text*, Dans *Research Report RC 20338*, IBM.
- Rey, A. (1977). *Le lexique : images et modèles. Du dictionnaire à la lexicologie*, Armand Colin, Paris, 307 p.
- Rey-Debove, J. (1994). *Nom propre, lexique et dictionnaires de langue*, Dans *Nom propre et nomination (Actes du colloque de Brest)*, Noailly, M. ed., p. 107-122.
- Riloff, E. et Lehnert, W. (1994). *Information Extraction as a Basis for High-Precision Text Classification*, *ACM Transactions on Informations Systems*, Vol. 12(3), p. 296-333.
- Riloff, E. (1995). *Little Words can make a Big Difference for Text Classification*, Dans *Proc. of the 18th Annual International ACM SIGIR Conference on Research and Developpement in Information*, p. 130-136.
- Roche, E. et Schabes, Y. (1995). *Deterministic Part-Of-Speech Tagging with Finite State Transducers*, Dans *Computational Linguistics*, Mitsubishi Electric Research Laboratories / cambridge Research Center, Vol. 21(2), p. 227-253.
- Salton, G. *et al.* (1975), *A Vector Space Model for Automatic Indexing*, *CACM*, Vol. 18(11), p. 613-620.
- Salton, G. et Buckley, C. (1988). *Term Weighting Approaches in Automatic Text Retrieval*, *Information Processing & Management*, Vol. 24(5), p. 513-523.
- Sampson, G. (1989). *How fully does a machine-usable dictionary cover English text*, Dans *Literary and Linguistic Computing*, Vol.4(1), p. 29-35.
- Savoy, J. (2000). *Rapport d'expérience: Amaryllis 1998-2000*. Workshop Amaryllis, Paris.
- Schütze, H. et Silverstein, C. (1997). *Projections for Efficient Document Clustering*, Dans *Proc. of the 20th International ACM SIGIR Conference on Research and Development in Information Retrieval*, p. 74-81.
- Sekine, S. et Eriguchi, Y. (2000). *Japanese Named Entity extraction Evaluation - Analysis of Results*, Dans *Proceedings of Computational Linguistics (Coling'2000)*, Saarbrücken, Germany, p. 25-30.
- Senellart, J. (1998). *Locating Noun Phrases with Finite State Transducers*, Dans *Proc. of COLING-ACL'98*, p. 1212-1219.
- Shaw, J. et Fox, E. (1995). *Combination of Multiple Searches*, Dans *Proc. of The Third Text Retrieval Conference (TREC 3)*, National Institute of Standards and Technology Special Publication, p. 105-109.
- Siegfried, S. et Bernstein, J. (1991). *Synonyme: The Getty's New Approach to Pattern Matching for Personal Names*, *Computers and the Humanities*, Vol.25(4), p. 211-226.
- Silberztein, M. (1993). *Dictionnaires électroniques et analyse automatique de textes - Le système INTEX*, Masson, Paris, 240 p.

- Smeaton, A., Burnett, M., Crimmins, F. et Quinn, G. (1998). *An Architecture for Efficient Document Clustering and Retrieval on a Dynamic Collection of Newspaper Texts*, Dans Proc. of the BCS-IRSG Colloquium, Springer Workshops in Computing.
- Steinbach, M., Karypis, G. et Kumar, V. (2000). *A comparison of document clustering techniques*, Dans Proc. of KDD Workshop on Text Mining, University of Minnesota.
- Stevenson, M. et Gaizauskas, R. (2000). *Using Corpus-derived Name Lists for Named Entity recognition*, Dans Proc. of Applied NLP and the N. American Chapter of the ACL, Seattle, USA, p. 290-295.
- Strehl, A., Ghosh, J. et Mooney, R. (2000). *Impact of Similarity Measures on Web-Page Clustering*, Proceedings of the 17th National Conference on Artificial Intelligence, Workshop of Artificial Intelligence for Web Search (AAAI 2000), 30-31 July 2000, Austin, Texas, USA, p. 58-64.
- Sundheim, B. M. (1995). *Overview of Results of the MUC-6 Evaluation*, Dans Proceedings of the Sixth Message Understanding Conference (MUC-6), p. 13-31.
- Taghva, K. et Gilbreth, J. (1995). *Recognizing Acronyms and their Definitions*, Dans Technical Report 95-03, Information Science Research Institute, University of Nevada, Las Vegas.
- Trouilleux, F. (1997). *Identification et classement automatique des noms propres en français*, Rapport de DEA, sous la direction de J.P. Chanod, Clermont-Ferrand.
- Trouilleux, F. (1999). *Un problème pour l'identification automatique : la limite droite des noms propres en français*, Dans Actes des Journées d'études de l'Atala sur le traitement automatique des noms propres, Paris.
- Tür, G., Hakkani-Tür, D. et Oflazer, K. (2000). *Name Tagging Using Lexical, Contextual and Morphological Information*, In Proceedings of the Workshop on Information Extraction Meets Corpus Linguistics at Second International Conference on Language Resources and Evaluation (LREC 2000), May 2000, Athens, Greece.
- van Rijsbergen (1979). *Information Retrieval (2nd edition)*, Butterworths, London.
- Vogel, S. et Ney, H. (2000). *Translation with Cascaded Finite-State Transducers*, Dans Proc. of ACL Conf. (Assoc. for Comput. Linguistics), Hongkong, p. 23-30.
- Voorhees, E. (1986). Implementing agglomerative hierarchical clustering algorithms for use in document retrieval, *Information Processing and Management*, Vol. 22, p. 465-76.
- Voorhees, E. (1999). *Natural Language Processing and Information Retrieval*, Dans *Information Extraction: Towards Scalable, Adaptable Systems*, Paziienza, M. ed., Springer, New York, p. 32-48.
- Wacholder, N., Ravin, Y. et Choi, M. (1997). *Disambiguation of Proper Names in Text*, Dans Proc. of the Fifth Conference on Applied Natural Language Processing, p. 202-208.
- Wakao, T., Gaizauskas, R. et Wills, Y. (1996). *Evaluation of an Algorithm for the Recognition and Classification of Proper Names*, Dans Proc. of the 16th

- International Conference on Computational Linguistics (COLING96), Copenhagen, p. 418-423.
- Willett, P. (1988). *Recent trends in hierarchic document clustering: a critical review*, Information Processing and Management, Vol. 24(5), p. 577-597.
- Wolinski, F., Vichot, F. et Dillet, B. (1995). *Automatic Processing of Proper Names in Texts*, ed., Dans Proc. of the Seventh Conference of the European Chapter of the Association for Computational Linguistics (EACL'95) , University of College Dublin, Dublin, Ireland, p. 23-30.
- Yang, Y. et Pedersen, J. (1997). *A Comparative Study on Feature Selection in Text Categorization*, Dans Proc. of the 14th International Conference on Machine Learning ICML97, p. 412-420.
- Zabeeh, F. (1968). *What's in a Name*, An Inquiry into the Semantics and Pragmatics of Proper Names, La Haye, Martinus Nijhoff.
- Zamir, O. et Etzioni, O. (1999). *Grouper: a dynamic clustering interface to Web search results*, Dans Proc. of the Eighth International World Wide Web Conference, Toronto, Canada.
- Zhao, Y. et Karypis, G. (2001). *Criterion Functions for Document Clustering*, Dans Technical Report TR-01-40, University of Minnesota.

## TABLE DES FIGURES

---

Figure 1: La Deixis .....	6
Figure 2 : Continuum nom propre - nom commun.....	8
Figure 3 : Description générale d'un système d'extraction d'information .....	24
Figure 4 : Représentation de la base de connaissances d'Exoseme.....	36
Figure 5 : Représentation Intex de la phrase "Parallèlement le billet vert donne des signes de faiblesses." .....	45
Figure 6 : Représentation d'une grammaire de la négation pré-verbale.....	45
Figure 7 : Dictionnaire des nombres de 2 à 99 .....	46
Figure 8 : Graphe Intex représentant la grammaire des nombres de 100 à 999.....	47
Figure 9: Transducteur exemple1 .....	49
Figure 10 : Transducteur exemple2.....	49
Figure 11: Transducteur repSigPoint .....	51
Figure 12 : Transducteur replaceSig .....	51
Figure 13 : Transducteur translettres.....	52
Figure 14: Graphe Sentence .....	58
Figure 15 : Notre graphe Sentence .....	60
Figure 16 : Graphe cas2.....	61
Figure 17 : Graphe MotifAntro .....	62
Figure 18 : Graphe sigles.....	63
Figure 19 : Graphe cas3 .....	64
Figure 20 : Graphe MotifSymboles.....	64
Figure 21 : Graphe Cas4.....	65
Figure 22 : Graphe Parentheses.....	67
Figure 23 : Un prénom composé .....	73
Figure 24 : Architecture de ExtracNP .....	77
Figure 25 : Contexte "ministre" .....	82
Figure 26 : Graphe décrivant les prénoms composés .....	82
Figure 27 : Graphe décrivant les prénoms en partie inconnus .....	83
Figure 28 : Graphe représentant les patronymes simples .....	83
Figure 29 : Graphe reconnaissant M. suivi d'un patronyme .....	84
Figure 30 : Graphe reconnaissant M. suivi d'un prénom puis d'un patronyme.....	84
Figure 31 : Coordination de noms de personnes.....	87
Figure 32 : Graphe MotifPerson.....	87
Figure 33 : Coordination sur les prénoms et noms d'une même famille .....	87

---

Figure 34 : Grammaire des établissements scolaires.....	88
Figure 35 : Graphe simplifié reconnaissant des formes de noms d'organisations à base descriptive.....	91
Figure 36 : Morphologie générale d'un nom d'organisation.....	92
Figure 37 : Graphe reconnaissant une partie des noms d'organisation étrangère.....	92
Figure 38 : Graphe reconnaissant les sigles et leur développement s'il est présent.....	94
Figure 39 : Reconnaissance des noms géographiques de lieux.....	95
Figure 40 : noms de lieux accompagnés d'un point cardinal.....	95
Figure 41 : Graphe reconnaissant des noms de pays.....	96
Figure 42 : Matrice des similarités des 4 textes.....	109
Figure 43 : Matrice des similarités après la première fusion.....	110
Figure 44: Arbre obtenu avec le critère Complete Link.....	111
Figure 45 : Arbre obtenu avec le critère Single Link.....	111
Figure 46 : Interface de CasSys.....	119

## TABLE DES FORMULES

---

Formule 1 : Rappel .....	25
Formule 2 : Précision.....	25
Formule 3 : F-mesure, formule générale.....	26
Formule 4 : F-mesure.....	26
Formule 5 : Mesure Jaccard .....	105
Formule 6 : Inverse Document Frequency.....	106
Formule 7 : Poids $w_{ik}$ d'un terme $k$ dans le document $i$ par TF.IDF .....	106
Formule 8: Similarité entre deux textes $d_i$ et $d_j$ .....	106
Formule 9: Similarités fusionnées.....	106
Formule 10 : Mesure de similarité fusionnée entre sous-vecteurs de mots et de noms propres.....	107
Formule 11 : Critère Complete Link.....	110
Formule 12 : Entropie d'un groupe de textes .....	112
Formule 13 : Entropie totale d'un ensemble de groupes.....	112
Formule 14 : Pureté d'un groupe de textes .....	112
Formule 15 : Pureté totale .....	112
Formule 16 : Mesure de similarité fusionnée entre sous-vecteurs de mots et de noms propres.....	115



## TABLE DES TABLEAUX

---

Tableau 1 : Preuves externes (contextes) et noms propres .....	17
Tableau 2: Résultats de notre étude selon le type du nom propre .....	19
Tableau 3 : Proportions des différentes preuves selon le type de nom propre (en %).....	19
Tableau 4 : Les proportions des différentes sortes de points en corpus .....	70
Tableau 5 : Précision, rappel et F-mesure des résultats sur les différents corpus .....	71
Tableau 6 : Description d'une partie des transducteurs reconnaissant les noms de personnes .....	85
Tableau 7 : Résultat de l'extraction en nombre d'occurrences de chaque type de noms propres dans le journal Le Monde.....	97
Tableau 8 : Résultat de l'extraction en nombre d'occurrences de chaque type de noms propres dans le journal Ouest France.....	98
Tableau 9 : Rappel et précision pour le journal Le Monde (en %) .....	98
Tableau 10 : Rappel et précision pour le journal Ouest France (en %) .....	99
Tableau 11 : Rappel, précision et F-mesure globaux pour les deux journaux (en %).....	99



# INDEX

---

---

## A

**Aberdeen** · 29, 32  
**Abney** · 29, 31, 41, 42, 43, 57  
abréviation · 11, 21, 30, 36, 61, 63, 65, 66, 70, 71, 73  
**Aït-Mokthar** · 43  
*Alembic* · 29, 32  
**Allerton** · 12  
Amaryllis · 108, 129  
ambiguïté · 11, 36, 37, 39, 63, 104  
analyse syntaxique · 42, 43, 47  
*Answer extraction* · 29, 31  
anthroponyme · 10, 13, 70  
antonomase · 8  
**Appelt** · 43  
arbre · 109

---

## B

balises · 48–50, 51, 55  
**Bauer** · 13  
*BBN IdentiFinder* · 29, 31  
**Béchet** · 32  
**Belleil** · 3, 11, 37  
**Bellot** · 107, 125  
**Besançon** · 104, 106  
**Bikel** · 29, 31, 35  
**Black** · 28, 30  
**Boisen** · 31  
**Boley** · 111  
**Borkowski** · 23  
**Borthwick** · 29, 31  
**Brill** · 32  
bruit · 25–26, 59, 66, 68, 70  
**Burnard** · 55

---

## C

CAH · 109  
candidat nom propre · 31, 74  
cascade de transducteurs · 4, 30, 32, 41–44, 47, 48, 50, 52, 54, 56, 71, 78, 84, 86, 87, 88, 94, 97, 100, 96–102, 120  
*Cass* · 42  
*CasSys* · 4, 41, 44, 47, 48, 50, 54, 56, 77, 119, 120  
*catégorisation* · 108  
**Chanod** · 43  
**Chinchor** · 26  
chunks · 42, 57  
**Church** · 32  
**Ciravegna** · 44

civilité · 60, 80, 81  
classification ascendante hiérarchique · 109  
classification non supervisée · 103  
cluster hypothesis · 108  
clustering · 103  
**Coates-Stephens** · 3, 14, 23, 28, 30, 34, 35, 37  
**Collins** · 29, 31  
collocation · 32, 104  
contexte · 78  
droit · 16, 20, 38, 78, 80  
gauche · 16, 19, 59, 81, 84  
local · 39 Voir aussi *preuve externe*  
corpus · 5, 18, 21, 24, 29, 30, 31, 32, 36, 43, 83, 96, 97, 108, 129  
Cosine · Voir *mesure Cosine*  
**Courtois** · 72, 125  
**Cowie** · 35  
critère  
Complete Link · 110  
Group Average Link · 110  
Single Link · 110  
Ward · 110  
crochets · 66, 67  
**Cucchiarelli** · 29, 32  
**Cucerzan** · 29, 31, 37  
**Cutting** · 109

---

## D

**Daille** · 5, 13, 21, 27, 104  
**Dalianas** · 29, 32  
découpage en phrases · 47, 57–71, 77, 96, 120, 124  
Deixis · 6  
**Dejong** · 23  
dendrogramme · 109  
dérivation · 9  
désambiguïsation · 24, 39, 40, 57, 125  
détermination · 9  
dictionnaire · 3, 10, 20, 28, 30, 34–36, 44, 46, 47, 57, 71–74, 77, 96, 100, 120, 125  
relationnel · 11  
**Dister** · 57

---

## E

ECRAN · 33  
**Eggert** · 3, 9  
ENAMEX · 26, 33  
entité  
nommée · 8  
entité nommée · 3, 23, 24, 26–27, 31, 27–40, 52, 55, 79  
entropie · 111–13, 114, 132  
ergonyme · 13

étiquetage · 30, 32, 33, 34, 57, 71, 74, 101  
étiquette · 46, 48, 49, 57, 63, 72, 74, 81, 120  
*Exoseme* · 28, 31, 38  
extension à droite · 38  
*ExtracNP* · 4, 77, 100, 107  
extraction  
  des entités nommées · 25, 26  
  d'information · 8, 23, 24, 30, 42, 43, 44, 57, 103,  
  104

---

## F

*FACILE* · 28, 30, 44  
Fairon · 18, 72  
*FASTUS* · 28, 30, 43  
fichier Index · 53  
**Flaux** · 8  
F-mesure · 25–26, 27, 28, 29  
forme canonique · 72, 125  
formes fléchies · 72, 74, 125  
**Forsgren** · 17  
**Fourour** · 29, 33  
**Fox** · 106  
**Frakes** · 125  
fréquence · 104, 105, 107, 114, 125  
*Frump* · 23  
*FUNES* · 28, 30, 35  
fusion  
  de similarité · 47, 106, 107, 110, 115, 131

---

## G

**Gaisauskas** · 30  
**Gala Pavia** · 43  
**Gale** · 40  
**Gallipi** · 32  
**Gallippi** · 29  
**Garric** · 3, 9  
**Gary-Prieur** · 5, 9, 17  
gentils · 3, 7, 11, 18, 20, 96  
**Gershman** · 23  
grammaire locale · 30, 42, 44, 72, 74, 78, 79, 100, 101  
graphe · 48, 44–51, 53, 54, 55, 64, 71, 74, 84–96, 120,  
  123  
graphe Sentence · 68, 69, 71, 123  
graphie · 11  
**Grass** · 3, 13, 15, 94, 100  
**Grevisse** · 5, 6  
**Grishman** · 24, 27  
**Gross** · 3, 11  
**Guerrin** · 10  
**Gueunthner** · 5  
guillemets · 66, 69

---

## H

**Hayes** · 34, 35  
**Hobbs** · 28, 30  
*homonymie* · 15  
*hyperonymie* · 7  
*hypertype* · 13  
*hyponymie* · 13

---

## I

*IFSP* · 43  
*Intex* · 3, 18, 41, 44–50, 56, 57, 59, 69, 70, 71, 72, 74,  
  78, 120, 123, 125

---

## J

Jaccard · Voir *mesure Jaccard*  
**Jacquemin** · 21, 37  
**Jonasson** · 6, 9, 12  
**Jones** · 57  
journal  
  Le Monde · 18, 29, 33, 40, 69, 70, 96, 98, 100, 108  
  Ouest France · 69, 96, 98, 100

---

## K

**Karkaletsis** · 28, 30  
**Kim** · 79  
**Kleiber** · 5, 9, 67  
**Kokkinakis** · 43  
**Kosseim** · 31  
**Krovetz** · 104  
**Krupka** · 28, 30  
**Kuhns** · 23

---

## L

LADL · 44  
**Laporte** · i  
*LaSIE* · 28, 30  
lemmatisation · Voir *lemme*  
lemme · 72, 104, 125  
lexicalisation · 8  
lieu · Voir *nom de lieu*  
limite droite · 38, 89, 90, 91, 93, 99  
**Lin** · 29, 32  
**Liu** · 10  
longest pattern matching · 47  
*LTG System* · 29, 33

---

## M

**Maiorano** · 26  
majuscule · 15, 63  
**Mani** · 10  
**Maurel** · 3, 5, 9, 10, 15, 73  
**McDonald** · 15, 16, 28, 30, 37, 38, 59  
*MENE* · 29, 31  
mesure  
  Cosine · 105, 107  
  de similarité · 4, 103–11, 112, 113, 115, 116, 131,  
  132  
  ensembliste · 105  
  Jaccard · 105, 113  
  Okapi · 107  
  TF.IDF · 105, 113, 115, 131, 132  
MET · 26  
**Mikheev** · 29, 33, 34, 35

**Miller** · 29  
mode  
  merge d'Intex · 47, 48, 50, 57, 120  
  replace d'Intex · 47, 48, 119, 120  
modèle vectoriel · 103, 105  
**Molino** · 6  
**Moore** · 111  
morphologie · 9, 20, 32, 37, 74, 92, 93, 100  
mots déclencheurs · 34, 35  
mots vides · 36, 42, 125  
MUC · 14, 23–27, 28, 29, 32, 33, 35, 36, 44

---

## N

*NAS* · 23  
NE · 26  
*Nemesis* · 29, 33  
*NERC* · 28  
*NetOwl Extractor* · 28, 35  
**Noailly** · 5, 9, 17  
nœuds · 46  
nom  
  commun · 5, 6, 7, 8, 15  
  de lieu · 14, 26  
  de personne · 14, 80, 83, 84, 107  
  de profession · 80  
  d'objet · 14  
  d'organisation · 14, 36, 107  
  d'origine · 14  
nom propre · 3, 5–21, 23, 26–39, 41, 43, 47, 57, 74,  
  77, 78, 79–101, 103, 104, 107, 114, 115, 131  
  coordination · 17  
  en apposition · 17  
  épiphète · 17  
  morpho-syntaxe · 12  
  sujet ou attribut · 17  
  typologie · 12  
*Nominator* · 28, 30, 35, 38  
NUMEX · 26

---

## O

organisation · Voir *nom d'organisation*

---

## P

**Paik** · 13, 34  
particule · 83  
patronyme · 81, 83  
personne · Voir *nom de personne*  
phénonyme · 13  
**Piton** · 3, 10  
PNF · 28, 38  
**Poibeau** · 27, 29, 33, 34  
point  
  d'exclamation · 59  
  d'interrogation · 59, 66  
  -virgule · 58, 59  
polysémie · 40, 104  
ponctuation · 30, 57, 66, 67, 68  
pondération · 107  
  globale · 105

  locale · 105  
**Popescu-Belis** · 25  
**Porter** · 125  
praxonyme · 13  
précision · 25–26, 27, 28, 29, 33, 34, 35, 42, 44, 65,  
  70, 97, 98, 99, 100, 101, 123  
prénom · 14, 16, 18, 19, 34, 39, 53, 55, 60, 61, 62, 63,  
  72, 74, 79–87, 99, 107  
preuve  
  externe · 16–18, 28, 33, 34, 55, 72, 78, 79, 87, 88,  
  93, 94, 99, 101  
  interne · 15–17, 18, 19, 20, 23, 30, 33, 35, 36, 37,  
  40, 72, 78, 79, 80, 83, 88, 91, 94, 100  
Prolex · 3, 10, 72, 73, 93  
pureté · 111–13

---

## R

racine · 125  
rappel · 25–26, 27, 28, 29, 33, 34, 35, 39, 44, 70, 97,  
  98, 99, 101, 104, 123  
**Rau** · 10, 23, 36  
**Ravin** · 35  
recherche d'information · 103  
règles · 28, 30  
**Rey** · 10  
**Rey-Debove** · 7, 8  
**Riloff** · 104, 125  
**Roche** · 41

---

## S

**Salton** · 103, 105  
*SAM* · 23  
**Sampson** · 10  
**Savoy** · 108  
**Sekine** · 26, 27  
*SemTex* · 29, 33  
**Senellart** · 29, 32  
sens · 6  
**Shaw** · 106  
*SIAC* · 125  
**Siegfried** · 23  
sigles · 9, 21, 26, 36, 37, 50, 51, 59, 60, 61, 62, 63, 70,  
  73, 93, 99  
**Silberztein** · 3, 44, 57  
silence · 25–26, 68, 70, 123  
similarité · Voir *mesure de similarité*  
**Smeaton** · 103, 110  
sous-reconnaissance · 38, 101  
sous-vecteur · 106, 107  
**Steinbach** · 111  
**Stevenson** · 28, 35, 36  
*stopwords* · Voir *mots vides*  
**Strehl** · 111, 112  
suffixation · 9, 36  
sur-reconnaissance · 37  
*SweNam* · 29, 32  
symboles · 41, 60, 61, 63, 64  
*Synoname* · 23  
synonymie · 104

---

## ***T***

**Taghva** · 37  
*ThingFinder* · 28, 39  
TIMEX · 26  
Tipster · 24  
titre · 80  
toponyme · 13  
traduction · 42, 43  
trait  
    morphologique · 72  
    sémantique · 72  
transducteurs · 30, 33, 41, 42, 43, 44, 46, 47, 53, 74,  
    78, 84, 119  
TREC · 103, 108  
**Trouilleux** · 21, 28, 30, 35, 38, 39, 89

---

## ***V***

**van Rijsbergen** · 25, 108

vecteur · 104, 106, 107, 114, 115, 125  
virgule · 38, 59, 66, 68  
**Vogel** · 43  
**Voorhees** · 104, 105, 109, 110

---

## ***W***

**Wacholder** · 28, 30, 38, 39  
**Wakao** · 28, 30, 34, 36  
**Willett** · 110  
**Wolinski** · 28, 31, 36, 38, 39  
*Wordnet* · 106

---

## ***Z***

**Zabeeh** · 13  
**Zamir** · 110  
**Zhao** · 111, 112



---

## Reconnaissance automatique des noms propres

### Application à la classification automatique de textes journalistiques

---

#### Résumé

La quantité d'information disponible sur Internet ou sur support informatique est de plus en plus abondante.

Dans les textes journalistiques, les noms propres représentent 10% des mots ; ils sont très importants pour une compréhension précise des textes, mais ils sont actuellement très peu représentés dans les ressources lexicales disponibles. Le travail réalisé ici cherche à automatiser leur extraction et leur catégorisation et s'intègre dans le projet *Prolex* de traitement automatique des noms propres. Nous avons implanté un système, nommé *CasSys*, qui permet l'utilisation de cascade de transducteurs à l'aide des fonctionnalités et des ressources du logiciel *Intex*. *CasSys* permet, par exemple, l'analyse syntaxique d'un texte ou l'extraction d'information. Le formalisme des transducteurs est particulièrement intéressant pour la description des noms propres. Le principe de la cascade permet de gérer les phénomènes d'ambiguïtés, de segmentation et de catégorisation des noms propres. Par cette méthode, nous avons obtenu une précision de 94% avec un rappel de plus de 93%.

À titre d'application, nous avons testé l'utilisation des noms propres dans la classification automatique de textes journalistiques : l'information dont ils sont porteurs les rend particulièrement intéressants pour obtenir une classification de qualité. Nous avons testé différentes mesures de similarité, basées sur les noms propres, en les évaluant à travers une classification hiérarchique.

**Mots clefs** : Extraction d'information, entités nommées, TAL, classification non supervisée

---

## Automatic Recognition of Proper Names

### An Application in Automatic Clustering of Journalistic Texts

---

#### Abstract

The quantity of available information on the Internet or on electronic support is more and more important.

In the journalistic texts, proper names represent 10% of the words; they are very important for a precise understanding of the texts, but they are rarely presented in the available lexical resources. The work realized here tries to automate their extraction and their categorization and will be integrated to the project *Prolex* of automatic process of the proper nouns. We have created a system, named *CasSys* that allows the use of transducer cascades by means of the features and the resources of the software *Intex*. *CasSys* allows, for example, the syntactic analysis or information extraction on a text. The formalism of transducers is particularly interesting for the description of the proper nouns. The principle of the cascade allows to manage the phenomena of ambiguities, segmentation and categorization of the proper nouns. By this method, we obtained a precision of 94 % with a recall of more than 93 %.

As an application, we tested the use of the proper names in the clustering of journalistic texts: the information they carry makes them particularly interesting to obtain a clustering of quality. We tested various measures of similarity, based on the proper names, by estimating them through a hierarchical clustering.

**Keywords**: Information Extraction, named entities, NLP, clustering

---

## LI / Laboratoire d'Informatique de Tours

Equipe Bases de Données et Traitement des Langues Naturelles

64 avenue Jean Portalis

37000 Tours

<http://www.li.univ-tours.fr>