Université de Marne-la-Vallée

**Laboratoire d'Automatique Documentaire et Linguistique,
Université Paris 7**

Recensement et description des mots composés – méthodes et applications

Agata Savary

Thèse de doctorat en Informatique Fondamentale

soutenue le 14 décembre 2000

Directeur de thèse **: Max Silberztein**

Jury :
Gaston Gross (rapporteur)
Maurice Gross
Franz Guenthner
John Humbley
Christian Jacquemin (rapporteur)
Eric Laporte
Max Silberztein

Dedykuję moim najdroższym
Cyprienne
Tytusowi
Xavier

Remerciements

Je voudrais remercier les nombreuses personnes qui ont contribué de différentes façons à mon travail et a mon évolution lors de mes études en thèse de doctorat.

Merci à Max Silberztein, mon directeur de thèse, qui a toujours été très disponible pour moi et enthousiaste par rapport à mon travail.

Merci aux membres, invités et amis du LADL pour l'ambiance chaleureuse, aide, patience, soutient et amitié. Plus particulièrement, merci à Maurice Gross, Blandine Courtois, Cédrick Fairon, Christian Leclère, Eric Laporte, Jean Senellart et Marianne.

Merci à Christian Jacquemin d'avoir été exigeant lors des lectures et des discussions sur ma thèse, et de m'avoir encouragée à découvrir de nombreux aspects de mon domaine de recherche.

Merci à Gaston Gross, John Humbley et Michel Mathieu-Colas pour leurs lectures, conseils et discussions concernant mon mémoire de thèse.

Merci à Franz Guenthner, et son équipe du CIS de m'avoir fait découvrir le domaine de la linguistique informatique.

Merci à Béatrice Daille, Chantal Enguehard, Didier Bourigault et Krzysztof Bogacki pour leur intérêt, aide et le temps qu'il m'ont consacré.

Merci à toute l'équipe informatique de la société LCI pour l'ambiance et l'aide. Merci à Tita Kyriacopoulou et à Adrien Assous pour la confiance lors de la réalisation de mon projet. Merci spécialement à Smith Charles pour son assistance si patiente.

Merci à Didier Arquès pour sa sagesse et sérénité lors de notre discussion.

Dziękuję Xavier za wsparcie i cierpliwość, oraz za lekturę mojego doktoratu.

Table des matières

Introduction

Chapitre 1 Objectifs et état de l'art

1.1 Objectifs

De nombreux travaux de référence dans le domaine du traitement automatique du langage naturel, comme par exemple les étiqueteurs grammaticaux, tiennent rarement compte du problème de composition dans les langues naturelles, ou bien le font à une petite échelle. En revanche, des applications du domaine de la terminologie computationnelle, comme l'extraction de termes, la reconnaissance de termes et de leurs variantes dans des textes, l'alignement de termes pour la création automatique de lexiques bilingues, etc., sont très concernés par le phénomène de composition dans les langages techniques. Souvent dans des telles applications on propose des algorithmes qui n'emploient pas ou très peu de connaissances linguistiques et terminologiques initiales. Leurs auteurs argumentent ce choix par le fait que la création de bases de telles connaissances est trop coûteuse. Pourtant, des bases de connaissances linguistiques et terminologiques existent – ce sont de nombreux dictionnaires traditionnels de la langue générale et des langages techniques, qu'il faut convertir en des formats utilisables par des programmes informatiques.

Dans l'étude décrite ci-dessous nous nous sommes penchée sur le recensement des mots composés à grande échelle, qui est selon nous indispensable pour les bons résultats de l'analyse automatique de textes. Nous avons essayé d'approfondir les questions suivantes :

1) Comment ce recensement peut être effectué ?

2) Est-il utile de le réaliser ?

Ces deux questions se reflètent dans la structure du mémoire. Dans la première partie, nous analysons différents problèmes posés par la description de mots composés dans des dictionnaires électroniques. Dans la deuxième partie, nous décrivons l'application des dictionnaires obtenus dans les tâches d'extraction terminologique et de correction orthographique de termes.

1.2 Cadre du travail

Les recherches décrites ci-dessous ont été menées par l'auteur dans le Laboratoire d'Automatique Documentaire et Linguistique (LADL) de l'Université Paris 7, et dans la société LCI Informatique, dans le domaine des mots composés et plus spécialement de la composition nominale en anglais.

Au sein du LADL, nous avons effectué des travaux concernant la construction de dictionnaires électroniques et l'analyse lexicale des textes. Nous nous sommes basée sur le système INTEX® qui a été créé par Max Silberztein en tant que cadre informatique pour la théorie linguistique du LADL. Ce système emploie des outils à états finis (automates et transducteurs) pour l'analyse lexicale des grands corpus. Il comprend entre autres un étiqueteur basé sur les lexiques DELA[1], un module de levée d'ambiguïtés à l'aide de grammaires locales, celui de la recherche de patrons syntaxiques dans un texte, et celui d'aide à la création de nouveaux lexiques électroniques au format DELA. Le programme de flexion automatique des mots composés que nous proposons dans le chapitre 4 de ce mémoire, a été

[1] Dictionnaires Electroniques du LAdl

destiné à compléter ce dernier module d'INTEX dans la tâche de construction automatique d'un dictionnaire électronique de mots composés fléchis, le DELACF, à partir d'un dictionnaire de mots composés sous formes lemmatisées, le DELAC. Pour ceci nous étudions les comportements flexionnels des mots composés en trois langues : le français, l'anglais et le polonais. Nous proposons une définition d'une flexion régulière et irrégulière des composés, basée sur la notion de constituants caractéristiques (tête), ainsi qu'une méthodologie de classement des composés selon la façon dont ils se fléchissent. Le programme de flexion obtenu est testé pour les noms composés du polonais et ensuite appliqué à la création du DELACF anglais de la langue générale et du DELACF anglais de termes informatiques.

Le dictionnaire DELAC de l'anglais d'environ 60 000 entrées a été mis en forme par nous à partir des listes des mots composés recensés par le professeur Maurice Gross (LADL), Mme McCarthy-Hamani, Katia Zellagui (Université de Besançon), Michael Walsh (Université de Dublin) et David Harte (Université de Dublin).

Le système INTEX permet, dans son module d'étiquetage, d'appliquer à un texte des dictionnaires sous trois formats différents : textuels, compactés, et des outils à états finis (automates et transducteurs). Ce dernier format facilite la description des composés productifs comme dates, déterminants, numéraux, etc. Nous avons construit une bibliothèque d'automates et transducteurs finis pour les numéraux cardinaux et ordinaux de l'anglais. Grâce aux symboles de sortie de ces transducteurs, nous pouvons attribuer à chaque numéral écrit en toutes lettres son correspondant en chiffres. Ceci permet de rendre compte de certaines ambiguïtés et équivalences entre ces deux types de représentation des numéraux.

Les trois derniers chapitres du mémoire décrivent les travaux effectués au sein de la société LCI Informatique dans le domaine de la terminologie. Il s'agit de la participation au projet LEXPERT (financé par l'ANVAR) de création d'une base de données terminologiques multilingues. Cette base, commercialisée sous le nom LexPro CD Databank (appelée aussi LexPro), versions 1.0 et 2.0, a été créée à partir de plusieurs dizaines de dictionnaires techniques de traduction mis sur un support informatique. Pour la version 3.0 de ce logiciel nous avons participé au développement des modules d'accès employant des techniques du traitement automatique du langage naturel, telles que la lemmatisation des termes et la correction orthographique (chapitre 9). Pour une version future, nous avons préparé un prototype d'un extracteur terminologique (chapitre 8). La société LCI étant l'un des partenaires scientifiques du LADL, nous avons introduit les dictionnaires électroniques DELA et certains programmes du système INTEX dans les fonctionnalités mentionnées de LexPro.

1.3 Travaux apparentés

1.3.1 Composition nominale

Différents aspects de la composition nominale ont été abordés par de nombreux travaux linguistiques dont nous présentons certains ci-dessous.

La notion de mot composé

La notion de mot composé a fait l'objet de nombreuses discussions et reste très controversée, comme le montrent les études comparatives de Corbin (1992), ainsi que d'Habert et Jacquemin (1993). Certains linguistes, comme par exemple Levi (1978), n'admettent pas qu'il soit possible de distinguer les noms composés des syntagmes nominaux libres. D'autres réalisent des analyses poussées des noms composés sans jamais donner la définition de cette

notion. Par exemple, pour Bauer (1988), la composition est « la formation d'un nouveau lexème par l'adjonction de deux lexèmes ou plus » (notre traduction), mais il ne fournit pas de définition claire du lexème.

Aussi, au niveau de la graphie existent des confusions car :

- pour certains elle est essentielle pour trancher entre les mots composés et les mots simples ; par exemple Silberztein (1993a, voir les définitions adoptées dans ce mémoire – section 2.2.2) considère que toute séquence composée doit contenir au moins un séparateur ; chez Bourigault (1994) un terme complexe doit contenir au moins une tête et une expansion, les deux étant séparés graphiquement dans un texte ; chez Jacquemin (1997) les termes et leurs variantes recherchées dans des textes sont reconnaissables par une grammaire dont les séquences soudées de lettres sont des unités indivisibles,
- pour d'autres elle n'a pas ce rôle ; une séquence soudée de plusieurs concepts simples peut être un composé comme *sunshine*, *electroscope* ou *anyone* chez Bauer (1983), *fireman, doorknob* ou *cutaway* dans OALDCE (1989), *survêtement*, *porteplume* ou *biologie* chez Grévisse (1993), d'ailleurs chez ce dernier une unité lexicale où les mots sont séparés par des blancs n'est plus appelée un composé mais une locution.

Il semble que la différence entre les deux approches est liée au fait que les trois premiers auteurs effectuent la reconnaissance automatique des composés, ce qui nécessite une définition stricte des unités atomiques de traitement, tandis que chez les trois derniers cette reconnaissance est faite par un lecteur humain.

Le critère référentiel de la composition est donné par Benveniste (1974): « il y a composition quand deux termes identifiables pour le locuteur se conjoignent en une unité nouvelle à signifié unique et constant ». Les constituants doivent être identifiables. Ainsi, *centimètre, portefeuille* ou *entresol* seraient clairement des composés, alors que pour *plafond* « le sentiment de la composition est déjà aboli ».

L'un des critères syntaxiques principaux de la composition est celui du figement. Selon G. Gross (1990) un groupe nominal est d'autant plus figé qu'il accepte moins de transformations prévues pour sa structure. Par exemple, la phrase :

[1] Le gouvernement a pris un train de mesures.

ne peut pas subir l'effacement du deuxième nom dans la suite *train de mesures* :

*[2] * Le gouvernement a pris un train.*

alors que cette transformation est attendue pour les noms de la structure *N de N*. Ainsi, *train de mesures* est considéré comme ayant un certain degré de figement. Le phénomène de figement est présenté d'une façon plus détaillée dans G. Gross (1996). En poursuivant cette approche, nous admettons dans ce mémoire que le plus petit degré de figement suffit pour considérer un syntagme comme composé.

J.-Cl. Anscombre (1990), analysant les noms composés du type *N à N*, conclut que « les tests habituellement préconisés pour détecter le figement ne permettent pas de distinguer les composés figés des autres ». Il propose d'analyser la « structuration événementielle », donc la sémantique, des noms composés (« à toute unité lexicale correspond au moins un schéma d'événement »), mais ne donne pas de définition opératoire de figement qui prenne en compte une telle analyse.

Cadiot (1992), en se concentrant également sur les composés français *N à N,* démontre la nécessité d'analyser les noms composés sous trois aspects parallèlement : la catégorie, le sens

lexical et la référence. Ces trois notions sont selon lui bien différentes. Cadiot propose ensuite une définition du nom composé du type N_1 *à* N_2 ou N_1 *à* N_2 *Adj* en termes de référents : *à* N_2 (*Adj*) doit être une qualification de N_1 permettant de donner un nom à une sous-classe définie des référents de ce N_1. Ceci impose deux conditions sur *à* N_2 (*Adj*) : « l'informativité minimale » (*animal à sang froid* est un nom composé mais *animal à sang* ne l'est pas), et « l'épure des images référentielles » (une *boîte à gants* est un composé, mais une *boîte à gants blancs* ne l'est pas car la qualification n'est pas ici suffisamment générale). Finalement, Cadiot rappelle les transformations syntaxiques admises par différents composés *N à N* qui permettent de distinguer au moins deux types sémantiques de l'usage de la préposition *à* : *à/POUR* et *à/AVEC*. Sur ce point (sémantique fondée sur la syntaxe) il rejoint le point de vue de G. Gross (1990 et 1996).

M. Noailly (1989) n'est pas d'accord avec « l'extension qu'avait prise le concept [du mot composé] » avec l'apparition des travaux de G. Gross (1988) entre autres. Elle suggère que la première condition nécessaire pour classer un syntagme nominal en tant que composé soit le test d'insertion du participe *dit*. Par exemple,

[3] les allocations familliales, le service national, le roman feuilleton

seraient des composés car l'insertion de *dit* y est « agréable » :

[4] les allocations dites familliales, le service dit national, le roman dit feuilleton

Mais ceci ne serait pas le cas pour

*[5] un bateau à voile (*un bateau dit à voile), un moulin à café (*un moulin dit à café)*

Pour certains auteurs la notion de composition n'est pas liée à celle du figement. Anscombre (1990), dans une phrase citée plus haut, admet l'existence de « composés figés » et « autres composés ». Pour Corbin (1992), les séquences complexes lexicalisées (selon les critères du figement syntaxique ou autres) ne sont pas des mots composés, s'ils peuvent être générés par des mécanismes syntaxiques. Downing (1977), sans jamais parler du phénomène de figement dans son analyse des procès de créations de composés anglais, admet qu'un nom composé anglais du type *Nom Nom* est « une simple concaténation de deux ou plus substantifs qui fonctionnent comme un troisième nominal » (notre traduction). Une définition du même type est donnée par Lyons (1978), pour qui « un lexème est composé si son thème est formé en combinant deux thèmes ou plus ». Les « composés-mots » se distinguent des « composés syntagmatiques » par le critère d'existence d'un seul accent primaire. Fabre et Sébillot (1996) proposent un calcul automatique du sens des composés nominaux anglais de la forme *NN*, « sachant que ce calcul s'applique à des séquences non lexicalisées ».

Remarquons également la confusion au sujet de la notion de la syntaxe et de la sémantique des mots composés. Plusieurs auteurs analysent la nature et les origines des noms composés en termes de la structure phrastique sous-jacente. C'est l'attitude de Benveniste (1974, p.145), pour qui « chaque type de composés est à étudier comme la transformation d'un type d'énoncé syntaxique libre », ainsi que de Levi (1978) et de Fabre et Sébillot (1996), que nous mentionnons plus loin. Pourtant, ce premier auteur considère son approche comme syntaxique (« la composition est une micro-syntaxe »), le deuxième comme syntaxico-sémantique, et les deux derniers comme sémantique (« déduction du sens du composé à partir des informations sur les constituants »).

Syntaxe des noms composés

Le LADL de l'Université Paris 7 et le LLI (Laboratoire de Linguistique Informatique) de l'Université de Villetaneuse ont une approche systématique au recensement et à la description du comportement syntaxique des mots composés. Leurs travaux dans ce domaine, menés dans le cadre transformationnel, ont commencé au début des années 1980 et ont mené à la définition du nom composé par son degré de figement par G. Gross (1988 et 1990), mentionnée ci-dessus. Le comportement syntaxique des noms composés (comme des mots simples et des autres mots composés) est décrit par des tables appelées *lexiques-grammaires*, où l'unité de l'analyse est une phrase, et où toutes les transformations admises pour un mot donné sont marquées par le trait « + », et les transformation interdites par le trait « - ». M. Mathieu-Colas (1988) a recensé plus de 500 types morphologiques de noms composés français. Des travaux linguistiques sur des classes particulières des composés (par exemple Monceaux 1994) ont révélé le nombre élevé des transformations qui concernent les noms composés. Des listes et des analyses des composés français autres que les noms ont été réalisées (par exemple M. Gross 1986 pour les adverbes, Piot 1978 pour les conjonctions composées). Le recensement systématique des composés à permis de produire un dictionnaire des mots composés du français qui compte 125 000 entrées (Jung 1990, Silberztein 1990). Les expressions figées ont fait l'objet de description par automates et transducteurs finis (Maurel 1989 pour les dates, M. Gross 1997 pour le langage de la bourse).

D'autre part l'équipe du professeur Gaston Gross de l'Université de Villetaneuse a défini la notion de classes d'objets (G. Gross 1994) et a mené le recensement d'éléments (aussi bien simples que complexes) de certaines classes comme les noms de professions, les objets de certains verbes comme *lire* etc.

Habert et Jacquemin (1995) analysent les traitements automatiques, à la fois syntaxiques et sémantiques, dans le cadre de grammaires d'unification. En évitant la notion controversée du mot composé, ils parlent des « constructions nominales à contraintes fortes », et comparent, d'une part, la représentation de leur construction syntaxique et de leur « sémantisme » dans trois formalismes : PATR-II, Lexical Functional Grammar, et les Grammaires d'Arbres Adjoints. D'autre part, le traitement de la variation des constructions nominales est décrits par deux autres formalismes : la Grammaire Contrôlée par l'Acceptabilité, et OLMES.

Sémantique des noms composés

Lyons (1978) et Levi (1978), abordent, dans le cadre générativiste, le sujet de la sémantique de la composition nominale. Selon Lyons les « lexèmes composés » sont complètement réguliers (i.e. leur sens et distribution sont prévisibles par des règles productives du système linguistique), tandis que les « lexèmes composés » proviennent des composés syntaxiques, mais ils sont institutionnalisés, et une partie de leur sens est spécialisée, donc non déductible par des règles générales (comme *country house* qui signifie une résidence possédée par une famille aristocratique en dehors de la capitale).

Levi (1978) effectue une étude des nominaux complexes (*complex nominals*) anglais qu'elle divise en trois groupes : les composés nominaux (*nominal compounds*) comme *apple cake, windmill*, les nominalisations, comme *Markovain solution, city planner*, et les phrases nominales à adjectif non prédicatif (*nonpredicating adjectif*) comme *electric shock, musical comedy*. Ces trois groupes contiennent des syntagmes obtenus, selon Levi, uniquement par deux processus productifs : la nominalisation du prédicat ou l'effacement du prédicat. Sont donc exclus de l'étude les formations lexicalisées, spécialisées, idiomatiques, et métaphoriques (*rock music, honeymoon* etc.). Levi traite au même titre des nominaux d'un

degré élevé de figement (*constitutional amendment*) que des syntagmes libres (*American attack*), son intérêt principal étant de prédire la totalité ou une partie des sens possibles d'un nominal à partir de ses structures logiques sous-jacentes.

Le même objectif est admis par Fabre et Sébillot (1996) qui proposent un calcul automatique du sens des composés nominaux anglais hors domaine, à partir de la forme et du sens de ses constituants. Dans une première phase, les mécanismes généraux engendrent toutes les interprétations admises. Elles sont sous forme d'un prédicat (représentant l'action sous-jacente au composé), muni des arguments (agent, thème, bénéficiaire, instrument, etc.). Certains arguments peuvent ne pas être spécifiés. Par exemple, le composé *rod feeding* est interprété comme *feed* (agent :-, thème :-, bénéficiaire : -, instrument : *rod).* Le prédicat est identifié soit morphologiquement, quand l'un des constituants est un nom déverbal, soit par l'attribution des noms rôles aux constituants (par exemple le nom *soap* est marqué comme typiquement lié au prédicat *wash*). Dans la deuxième phase, le choix souvent multiple de différents prédicats possibles pour un composé doit être limité par la classe sémantique des composants. La classe sémantique est déterminée selon les relations hyperonymiques explicitées dans la base lexicale WordNet (Miller 1993, voir section 1.3.2).

Finin (1986) emploie une méthodologie semblable pour l'interprétation des nominaux composés anglais dans le sens de Levi (1979). Dans un composé du type *Nom Nom ou Adj Nom*, le premier composant peut remplir l'un des « rôles » (agent, objet, instrument, location,…) du verbe sous-jacent. Le verbe sous-jacent est soit directement accessible dans le cas des noms déverbaux (*drinking water*, *engine repair*), soit reconstitué à partir de l'« activité caractéristique » des composants (e.g. dans *cat food* l'activité caractéristique de *food* est *to eat*). Après avoir généré toutes les interprétations possibles pour un nominal, la sélection du candidat le plus plausible doit se faire par l'analyse du discours, mais les méthodes de cette analyse ne sont pas précisées.

Formation des noms composés

Corbin (1992) se penche sur les mécanismes formels et sémantiques qui régissent la construction des mots composés, et c'est dans ces termes-là qu'elle définit les frontières internes (i.e. par rapport à la préfixation, la suffixation et la conversion) et externes (i.e. par rapport aux séquences complexes construites autrement que par des règles lexicales) de la composition nominale. Ainsi, « un mot composé est une unité lexicale complexe construite par des règles lexicales conjoignant des unités lexicales à pouvoir référentiel ». Par exemple *appui-tête* est, selon Corbin, un mot composé car il ne peut être engendré que par des mécanismes dérivationnels, tandis que *pomme de terre* ou *trompe-la-mort*, même s'ils sont lexicalisés, ne sont pas des mots composés car ils peuvent être obtenus par des mécanismes syntaxiques.

Downing (1977) se concentre sur les noms composés anglais du type *Nom Nom* qui ne sont pas lexicalisés. Elle s'oppose aux approches, telles que celle de Levi (1978), où l'analyse d'un composé nominal s'effectue par la reconstruction de la structure phrastique sous-jacente. L'acceptabilité de la création d'un nouveau nom composé est, selon Downing, soumise à des contraintes pragmatiques très diverses (e.g. *apple-juice seat* = un siège en face duquel se trouve un verre de jus), et non pas limitée à un nombre fini de structures sémantiques et syntaxiques à partir desquelles un composé peut être dérivé.

Selkirk (1982) présente également un point de vue opposé à celui de Levi (1978). Dans le cadre de la grammaire générative, elle propose une grammaire hors contexte pour décrire la formation des mots composés. Selon elle, les transformations, comme celles qui font dériver

un composé de sa structure phrastique sous-jacente, n'ont aucun rôle dans la création de composés anglais. Quand à l'interprétation des composés, Selkirk considère que seuls les composé « verbaux » (*cake baker, housecleaning, surface adherence,* etc.) sont intéressants linguistiquement. Pour les autres composés, le nombre de relations sémantiques possibles entre la tête et l'extension est tellement vaste qu'il est impossible de les décrire.

1.3.2 Construction de dictionnaires électroniques

Dans le domaine des dictionnaires existants sur support informatique il existe une confusion entre les *dictionnaires informatisés* et les *dictionnaires électroniques*. Les premiers sont des dictionnaires que nous appelons « usuels » dans ce mémoire, comme NSOED (1996), c'est-à-dire des ouvrages uni- ou multilingues destinés à un lecteur humain. Le format de leurs entrées est celui des dictionnaires semblables sur papier, mais leur « informatisation » permet de rassembler une quantité beaucoup plus grande de données, ainsi que de fournir un accès rapide et la possibilité de recherche d'informations selon différents critères. Comme nous expliquons dans la section 5.2, les dictionnaires de ce type contiennent beaucoup d'informations implicites qu'un lecteur humain est censé pouvoir déduire. Par *dictionnaires électroniques* nous comprenons des bases de données lexicales où toutes les informations sont explicites car elles sont destinées à l'usage des programmes informatiques. Ces bases visent la modélisation de la langue, ce qui les distingue des lexiques électroniques particuliers créés pour les besoins d'applications particulières. Les lexiques électroniques particuliers de la langue générale, comme celui de Rank Xerox de Grenoble (Chanod et Tapanainen 1995), sont le plus souvent élaborés pour des étiqueteurs grammaticaux ou pour des correcteurs orthographiques. Ils se limitent, pour la plupart, à indiquer la catégorie grammaticale, accompagnée éventuellement des traits flexionnels (nombre, genre, etc.) des mots simples et leurs formes fléchies. L'exhaustivité de tels lexiques ne semble pas être un but en soi. Faute de trouver un mot dans le lexique, on emploie des algorithmes qui essayent de deviner la catégorie de ce mot selon différents critères (par exemple la terminaison, la catégorie des mots voisins, etc.). Les mots composés n'y sont pris en compte que d'une façon marginale. Ceci n'est pas le cas pour les lexiques électroniques spécialisés, comme celui du système FASTER (Jacquemin 1997). Ils contiennent de nombreux termes composés, mais ils sont dédiés à leur domaine technique et ne couvrent donc que certaines unités lexicales de la langue générale.

Le LADL est le centre principal de recherche sur les dictionnaires électroniques du français. Il s'occupe de la description et de sa mise à jour des unités lexicales de la langue : les mots simples et composés sont recensés dans le système DELA (Courtois 1990, Silberztein 1990) ; leur flexion est décrite explicitement et exhaustivement (codes flexionnels – voir section 2.3) ; leur comportement syntaxique est décrit par des lexiques-grammaires (Leclère 1990) ; leur phonétisation est réalisée par un dictionnaire phonétique (Laporte 1988) ; les expressions figées sont décrites par des automates et des transducteurs (voir section 2.5) ; les noms propres géographiques sont recensés dans des dictionnaires du type DELA (73 000 entrées de la géographie française, Maurel et al. 1995, Maurel et Piton 1999).

Les principes du LADL de description lexicale sont partagés par d'autres laboratoires, européens et asiatiques, qui constituent le réseau RELEX. Ainsi les dictionnaires électroniques de l'anglais, de l'italien (Monteleone 1997), de l'allemand (Guenthner 1996), du portugais, de l'espagnol, du russe, du polonais, du bulgare, du slovaque, du grec moderne (Sklavounou 1999), du coréen, du thaï et de l'ancien français (Bat-Zeev Shyldkrot 1996) existent déjà ou sont en cours de construction. Un projet de dictionnaire électronique bilingue allemand-français est en cours entre le CIS (Centrum für Informations- und Sprachverarbeitung) de l'Université de Munich et le LLI de l'Université de Villetaneuse. Dans

un autre projet concernant un dictionnaire bilingue français-espagnol, Blanco (1997) a effectué un alignement et un classement de 25.000 noms composés français et de leurs équivalents en espagnol.

Quant aux dictionnaires électroniques anglais à large couverture, le projet WordNet® sous direction du professeur George A. Miller est en cours à l'Université de Princeton. Il s'agit d'un dictionnaire sémantique de mots anglais, simples et composés, qui sont représentés dans trois structures hiérarchiques : une pour les noms, une pour les verbes et une pour les adjectifs. A l'intérieur de chaque structure les mots sont organisés en de petits ensembles de synonymes (*synsets*) correspondant à des concepts lexicaux, entre lesquels des relations de différents types sont introduites. Par exemple, le *synset* suivant : {*beak, bill, neb*} (bec) est un hyponyme (sorte) de {*mouth, muzzle*} (bouche, museau), qui à son tour est un méronyme (partie) de {*face, countenance*} (visage, figure) et un hyponyme de {*orifice, opening*} (ouverture). WordNet contient (état de 1993) 51 500 mots simples et 44 100 mots composés (« collocations ») organisés en 70 100 *synsets.* Les noms constituent la catégorie dominante de ce lexique : 57 000 formes simples et composés dans 48 800 *synsets*.

Notons également que dans le contexte des dictionnaires traditionnels (sur papier) l'importance de la composition semble plus acquise dans la tradition britannique et américaine que dans la tradition française. Témoignent de ce fait les ouvrages dédiés à ce sujet comme *Oxford Dictionary of Current Idiomatic English,* et J. Seidl, W. McMordie *English Idioms and How to Use Them*. D'autre part, Walker et Amsler (1986), décrivant une méthode d'identification automatique du domaine d'un texte donné à l'aide des informations provenant d'un dictionnaire traditionnel, démontrent la nécessité d'identification et de recensement d'unités complexes (« aggregates of terms ») pour les bons résultats de leur approche.

Au LADL un système DELA de l'anglais est en cours d'élaboration. Blandine Courtois s'occupe de l'élargissement du dictionnaire des mots simples créé par Klarsfeld et McCarthy-Hammani (1992) et complété par Monceaux (1995). Le présent mémoire décrit la création du dictionnaire de mots composés.

Dans la tradition polonaise des dictionnaires électroniques, les mots simples sont étudiés d'une façon systématique par plusieurs auteurs. Le professeur K. Bogacki de l'Université de Varsovie a créé un DELAS polonais selon les principes du LADL (150 000 lemmes). L'équipe du prof. Z. Vetulani (Z. Vetulani, Walczak, Obrębski, G. Vetulani 1998) a décrit le comportement morphologique de 41 500 substantifs simples, également selon la méthode du LADL, et construit un programme de leur lemmatisation par calcul (i.e. sans générer la liste de formes fléchies). Cette description, convertie en Prolog et complétée par des données sur d'autres catégories grammaticales, a été appliquée dans le système POLINT de dialogue homme-machine (Vetulani 1996) et dans un essai de traduction automatique polonais-français (Jassem 1996). Une autre approche a été adoptée par J. Bień et K. Szafran (1996) de l'Université de Varsovie qui ont implémenté un analyseur morphologique des mots simples basé sur un lexique de 100 000 mots. Quant aux mots composés du polonais, leur recensement a été entrepris par les travaux sur des dictionnaires traditionnels unilingues dont le plus important est celui de Skorupka (1967), mais ils n'ont pas fait l'objet de traitement automatique, mis à part Chrobot (1996).

1.3.3 Outils à états finis

Les outils à états finis, dans leurs versions classiques et étendues (Kornai 1999), sont employés dans le TALN d'une façon naturelle, des nombreux aspects du langage étant adaptés à cette représentation (M. Gross 1989b, Mohri 1997), et efficace en temps d'accès et en

espace mémoire. Voici les domaines d'applications de ces outils et des exemples de leurs travaux de référence :

- la reconnaissance de la parole : Pereira et Riley (1997) ;
- la phonologie computationnelle : Kaplan et Kay (1994), Laporte (1997) ;
- la morphologie : Koskenniemi (1997), Silberztein (1993a), Clemenceau (1997) ;
- l'étiquetage grammatical et la levée d'ambiguïtés morphologiques : Roche et Schabes (1997), Silberztein (1993a), Laporte et Monceaux (1997), Chanod et Tapanainen (1994), Roche (1992) ;
- l'analyse syntaxique : Roche (1993), Schulz et Mikołajewski (1999) ;
- la correction orthographique : Oflazer (1996), Daciuk (1998) ;

Dans le cadre de la morphologie computationnelle par outils à états finis, dans lequel se place une partie de ce mémoire, il existe deux types principaux d'approches :

- les méthodes lexicales basées sur les dictionnaires électroniques - cette première approche est représentée par exemple par la méthodologie du LADL appliquée à des langues mentionnée dans la section 1.3.2 ; elle admet qu'une analyse morphologique d'un mot n'est possible que si ce mot apparaît dans le dictionnaire morphologique ;
- les méthodes heuristiques basées sur des systèmes de règles morphologiques – la méthode la plus importante est ici le *two-level system* désigné initialement par K. Koskenniemi pour le finnois, et appliqué pour de nombreuses langues comme l'anglais (Karttunen et Wittenburg 1983), le russe, le suédois, l'allemand, le danois, le basque, l'estonien ; il admet un système de règles comme seul point de départ pour l'analyse morphologique.

Les deux approches ont des avantages (+) et des inconvénients (-), dont nous faisons une discussion ci-dessous.

Two-level system :

+ l'analyse de chaque mot, connu et inconnu, est possible,
+ la méthode s'adapte bien aux langues agglutinantes comme le finnois et le hongrois, ainsi qu'aux phénomènes dérivationnels dans d'autres langues (préfixation, suffixation etc.),
– l'analyse d'un mot risque d'être incorrecte,
– la construction de l'ensemble de règles est compliquée – pour être sûr de chaque règle il faut consulter l'ensemble de tous les mots auxquels elle s'applique ; ainsi, il faudrait en fait disposer d'une liste complète de mots,
– la maintenance d'un tel grand système de règles est difficile – il y a un risque d'introduction de règles contradictoires ou concurrentes,
– pour les langues où la flexion n'est pas excessivement riche (comme les langues germaniques et les langues romanes), les transducteurs créés à partir des règles et utilisés pour l'analyse morphologique sont de tailles plus importantes que les dictionnaires-automates dans la deuxième approche ;

Système de dictionnaires tel DELA :

+ le système de codes flexionnels est peu complexe donc facile à apprendre et à maintenir – la vérification des règles est facile car on procède mot par mot,

+ les mots analysables, i.e. présents dans le dictionnaire, sont analysés sans erreur,
− aucune analyse n'est possible pour les mots inconnus (sauf si des méthodes heuristiques ou probabilistes sont rajoutées quand la consultation de dictionnaire donne un échec),
− le coût de création d'un dictionnaire est important.

La fusion des deux approches dans le cadre de l'analyse morphologique a été proposée par Clemenceau (1997). Un dictionnaire des formes fléchies, sert ici à reconnaître d'une façon sûre toutes les unités recensées, tandis qu'un système de règles du type *two-level system* permet d'analyser les dérivations inconnues des mots présents dans le dictionnaire.

1.3.4 Morphologie flexionnelle des mots composés

Quant à la morphologie flexionnelle des mots composés, concernée par le chapitre 4 de ce mémoire, peu d'études détaillées existent. Grévisse (1993) trouve que le pluriel des noms composés est « le domaine le plus difficile de toute la grammaire française ». Les seuls composants qui puissent prendre la marque du pluriel dans un composé sont les noms et les adjectifs. Dans les noms composés du type *Nom Nom* les deux composants se mettent au pluriel s'il s'agit d'une apposition (*oiseaux-mouches*) sauf exception où deux variantes sont possible (*chênes-lièges*, *chênes-liège*). Seul le premier composant varie si l'un des composants est complément de l'autre (*timbres-poste, coups d'oeil*), mais il y a beaucoup d'exceptions à double pluriel : *noms de lieu(x), toiles d'araignée(s)*, etc. Dans les composés du type *Nom Adj* et *Adj Nom* les deux éléments varient (*grands-pères*) sauf *grand-mères, petits-beurre, terre-pleins*, etc. Dans le cas des *Verbe Nom*, le verbe ne varie pas, et le complément est invariable s'il n'est pas l'objet direct (*réveille-matin*). Grévisse propose de fléchir le deuxième composant toujours quand il est objet direct (*couvre-lits, essuie-mains*, etc.).

Dans les outils TALN, le problème de génération de formes fléchies d'un mot composé, ou bien, inversement, du rattachement d'un composé fléchi à son lemme, est d'habitude traité par des règles générales qui ne tiennent pas compte d'exceptions. Par exemple, dans le module morphologique du système WordNet (www.cogsci.princeton.edu module *morphy)* les formes fléchies comme *attorneys general* sont lemmatisées par la lemmatisation de chacun des composants. Ceci ne permet pas de retrouver le lemme correct d'entrée comme *customs duty* qui est réduit à **custom duty*. Les problèmes de ce type ne peuvent être traités correctement que par recensement et description explicite.

1.3.5 Reconnaissance et acquisition de termes

Comme le démontre Lehrberger (1986), un langage spécialisé (*sublanguage*) n'est pas un sous-ensemble du langage naturel « standard ». L'intersection des deux est d'habitude non vide, mais il existe des phrases qui appartiennent au langage standard et non pas au langage spécialisé et inversement. Autrement dit, un langage spécialisé présente certaines restrictions et déviations par rapport au langage standard. Par exemple, de nombreux mots du vocabulaire standard n'apparaissent jamais dans des textes spécialisés, ou bien apparaissent avec un nombre réduit de catégories et de sens. D'autre part, leur syntaxe peut changer, e.g. dans les manuels d'utilisation de logiciels informatiques le verbe *save* apparaît avec les prépositions *in* et *into*, alors qu'en anglais standard seule la première possibilité est admise.

En conséquence, le traitement automatique du langage spécialisé devrait être effectué par des outils adaptés au domaine (*subject matter*) du texte traité. Cette condition, considérée comme trop coûteuse, est rarement remplie par les applications TALN existantes. La tendance est plutôt contraire : proposer des systèmes applicables à chaque domaine technique sans

nécessité d'adaptation. Si cette solution est « robuste », elle ne devrait servir, selon nous, que comme outil intermédiaire pour l'élaboration d'autres outils adaptés au domaine. Par exemple, un extracteur terminologique « général » peut servir à l'élaboration ou l'enrichissement de lexiques et grammaires spécialisés.

Dans le domaine de la terminologie computationnelle, surtout l'extraction terminologique a une bibliographie très riche grâce aux enjeux qu'elle représente : l'indexation automatique des textes, la recherche documentaire, la création de lexiques et dictionnaires uni- et multilingues, la traduction automatique et l'aide à la traduction. Les travaux existants dans ce domaine peuvent être classés selon différents critères :

1) La reconnaissance des termes contre l'acquisition (ou extraction) des termes.

   Le premier type d'outils sert à retrouver dans des textes les termes déjà connus. Pour cela, il suffit de disposer d'un outil d'analyse lexicale tenant compte des unités complexes, comme INTEX (Silberztein 1993a). La difficulté de réalisation plus fine de cette tâche provient du fait qu'un terme peut apparaître sous différentes variantes (e.g. *birth date* = *date of birth*). La variation terminologique a été traitée par Jacquemin (1997) et Jacquemin, Klavans et Tzoukermann (1997). Du côté de l'extraction des termes nous pouvons classer un nombre important de travaux, comme le système Termino de David et Plante (1990), Xtract de Smadja (1993), Acabit de Daille (1994), ANA de Enguehard et Pantera (1994), Justeson et Katz (1995), Lexter de Bourigault (1994), FASTER de Jacquemin (1997 ; la deuxième partie de son étude), et aussi une méthode particulière d'extraction de termes complexes par repérage de leurs acronymes par Bowden et al. (1998).

2) L'extraction statistique des termes contre l'extraction par analyse structurale.

   La première approche est fondée sur l'idée que les mots qui forment une unité lexicale ont tendance à apparaître ensemble plus souvent que d'autres combinaisons de mots. Les travaux de référence sont par exemple ceux de Enguehard et Pantera (1994), Justeson et Katz (1995), Nakagawa et Mori (1998). La deuxième approche, visant le repérage de toutes les occurrences de termes, non seulement les plus fréquentes, est représentée par David et Plante (1990), Bourigault (1994), Jacquemin (1997), Ladoucer et Cochrane (1996), et nous même. Des modèles hybrides qui emploient conjointement des outils linguistiques (étiqueteur lexicaux, grammaires locales, analyseurs syntaxiques) et statistiques, sont par exemple ceux de Daille (1994), et Smadja (1993).

3) L'extraction « initiale » de termes contre l'enrichissement terminologique.

   La plupart des outils d'extraction adoptent cette première approche, i.e. ils admettent le corpus et éventuellement un outil linguistique général (étiqueteur grammatical, grammaire locale, ou analyseur syntaxique) comme les seuls points de départ pour la recherche de termes. La deuxième approche tient compte de l'existence d'une base terminologique initiale comme point de départ pour la recherche de nouveaux termes. Elle est représentée par Jacquemin (1997), ainsi que partiellement par Enguehard et Pantera (1994), et est aussi admise dans ce mémoire (chapitre 8).

1.3.6 Correction orthographique

Le traitement d'erreurs d'orthographe est l'un des problèmes du TALN les plus anciens. Il a une bibliographie très riche dont une bonne analyse a été effectuée par Kukich (1992). Elle divise les méthodes de correction orthographique en trois types :

1) la reconnaissance de mots incorrects,

2) la correction de fautes d'orthographe hors contexte,

3) la correction de fautes d'orthographe en fonction du contexte.

Les outils du premier type se limitent à repérer les mots orthographiquement incorrects, sans proposer leur correction. Ils sont basés soit sur l'analyse des n-grams (i.e. séquences de lettres qui sont interdites ou très peu probables dans un mot, comme *shj*), soit sur la consultation d'un dictionnaire (e.g. McIlroy 1982). Dans ce deuxième cas, le problème majeur posé pour les approches présentées par Kukich est celui de la taille du dictionnaire et de son temps d'accès. A l'heure actuelle ce problème ne se présente guère grâce à l'emploi d'outils à états finis, qui permettent de représenter un dictionnaire avec un taux de compression très important, et avec un temps d'accès proportionnel à la longueur du mot recherché (et donc indépendant de la taille du dictionnaire).

Les outils du deuxième type ont pour but de rechercher, pour un mot considéré incorrect, des mots connus semblables. Cette ressemblance peut être définie de différentes façons. Dans le contexte de reconnaissance optique de caractères (OCR), elle est liée aux ressemblances de lettres et de suites de lettres (e.g. *m* et *ni*). Pour les textes tapés sur un clavier il peut s'agir d'erreurs d'origines phonétiques, alors le mot erroné et la correction proposée doivent avoir la prononciation identique ou proche (par exemple la méthode de Laporte et Silberztein 1989 est basée sur cette idée). Dans le cas de fautes de frappe, le rapport entre un mot correct et incorrect est souvent exprimé en termes de quatre opérations élémentaires sur des lettres, proposées par Damerau (1964) : l'omission (*rapport* → **rpport*), l'insertion (*rapport* → **ramppport*), le remplacement (*rapport* → **rqpport*) et l'interversion (*rapport* → **rpaport*). Parmi les outils du deuxième type, un progrès important a été réalisé par Oflazer (1996) qui propose une méthode basée sur la consultation « tolérante » d'un dictionnaire sous format d'automate fini. Sa comparaison avec une approche élaborée par nous, se trouve dans le chapitre 9.

Les correcteurs orthographiques qui tiennent compte du contexte sont les plus récents parmi les trois types présentés par Kukich. Leurs avantages les plus importants par rapport aux méthodes hors contexte sont :

- la possibilité de repérage de mots mal orthographiés qui résultent en d'autres mots corrects (e.g. *from* → *form*),
- la limitation du nombres de correction proposées pour un mot erroné.

Un correcteur contextuel parfait devrait effectuer une analyse syntaxique, sémantique et pragmatique au niveau de la phrase entière, ce qui est d'autant plus difficile que la phrase analysée peut ne pas être grammaticale. C'est pourquoi certains systèmes admettent des solutions intermédiaires, c'est à dire basées sur l'analyse du contexte local (un syntagme nominal, un verbe avec ses arguments, un sujet avec son prédicat, etc.). Par exemple, Schwindt (1990) dans son système d'apprentissage de l'allemand assisté par ordinateur, a construit une grammaire d'unification contenant des « règles de fautes » qui décrivent les erreurs grammaticales commises souvent en allemand par les français (règles d'accord entre les noms et les adjectifs, la rection des verbes, etc.). Chaque règle de faute est accompagnée d'une règle de correction, qui est activée quand cette première reconnaît un contexte local erroné.

Chapitre 2 Analyse lexicale des mots composés par le système INTEX®

2.1 Introduction

Le point de départ pour toutes les recherches que nous présentons a été le système INTEX®, crée par Max Silberztein en tant que cadre informatique de la théorie linguistique du LADL. Dans ce chapitre, nous allons présenter les unités de traitement et les algorithmes mis au point dans INTEX, en soulignant particulièrement ceux qui concernent les mots composés.

INTEX est un système multilingue de traitement automatique de grands corpus (jusqu'à plusieurs millions de mots). Il permet entre autres :

- d'effectuer l'analyse lexicale et syntaxique de textes à l'aide de dictionnaires électroniques à large couverture et des grammaires locales, contenus dans le système ou créés par l'utilisateur,
- de rechercher dans le texte des motifs syntaxiques créés à l'aide des informations contenues dans les étiquettes grammaticales.

Les principes et les détails du fonctionnement des interfaces et algorithmes d'INTEX sont présentés dans Silberztein (1993a), Silberztein (1997) et Silberztein (1999-2000).

L'une des particularités d'INTEX, en comparaison avec d'autres systèmes d'analyse lexicale, est la présence et l'importance des algorithmes spécialisés pour la reconnaissance des mots composés et expressions figées. Dans le présent chapitre nous donnons une description de ces algorithmes, précédée de la présentation des définitions et des formats des données linguistiques utilisés par INTEX. Nous éclairons notre discussion par des exemples en français, en anglais et en polonais[2].

Dans la section 2.2 nous donnons les définition des unités de traitement. Dans les sections 2.3, 2.4 et 2.6 nous présentons le principe de construction et de compactage des dictionnaires électroniques des mots simples (DELAS, DELAF) et des mots composés (DELAC, DELACF). La section 2.5 décrit une représentation des mots composés d'un certain degré de productivité par automates finis. Nous traitons le phénomène de l'ambiguïté des mots composés dans la section 2.8. Finalement dans les sections 2.9 et 2.10 nous décrivons l'analyse lexicale des textes qui tient compte des mots composés.

2.2 Définitions

Pour les traitements automatiques des textes en langues naturelles par le système INTEX, nous décrivons d'abord les objets que ce traitement concerne. Il s'agit des notions de l'alphabet, des séparateurs, d'une forme simple et composée, du mot simple et composé, et de la tête (ou constituants caractéristiques) d'un mot composé. Pour illustrer ces notions nous prenons en compte trois langues : le français, l'anglais et le polonais. Nous présentons également des exemples d'unités lexicales qui ne se laissent pas décrire par les définitions admises.

[2] La version commerciale actuelle d'Intex ne contient pas encore de modules du polonais.

2.2.1 Lettres de l'alphabet et séparateurs

Pour chaque langue, l'**alphabet**, i.e. l'ensemble de lettres utilisées, doit être défini. Dans le cas de beaucoup de langues européennes, l'alphabet contient les 52 lettres de l'alphabet latin : a, A, b, B, c, C, d, D, e, E, f, F, g, G, h, H, i, I, j, J, k, K, l, L, m, M, n, N, o, O, p, P, q, Q, r, R, s, S, t, T, u, U, v, V, w, W, x, X, y, Y, z, Z, ainsi que leurs versions diacritiques et les ligatures, comme à, À, â, Â, ä, Ä, ç, Ç, è, È, é, É, ê, Ê, ë, Ë, î, Î, ï, Ï, ô, Ô, ö, Ö, œ, Œ, ù, Ù, û, Û, ü, Ü pour le français, ou ą, Ą, ć, Ć, ę, Ę, ł, Ł, ń, Ń, ó, Ó, ś, Ś, ź, Ź, ż, Ż pour le polonais. Tous les autres caractères comme les chiffres (0, 1, 2, 3, 4, 5, 6, 7, 8, 9), les caractères de ponctuation (les points, les virgules, les deux-points, les apostrophes, les traits d'unions, les points d'interrogations, les parenthèses etc.), les symboles mathématiques (le « plus », le « moins », le « pour-cent », etc.) et autres caractères non-alphabétiques sont considérés comme **séparateurs**.

Déjà, ce stade d'énumération d'éléments de l'alphabet pose un premier problème, car rien n'empêche un texte en une langue donnée de contenir des mots, donc des lettres, d'une autre langue. Il faut donc définir un alphabet universel comprenant tous les alphabets du monde. Cette idée a conduit à définir les systèmes Unicode, qui consacrent deux ou quatre octets au codage de chaque caractère, ce qui permet de représenter au moins $2^{16} = 65\,536$ caractères différents. N'ayant pas encore adopté ce système, INTEX se sert du jeu de caractères sur un octet défini par le standard ANSI, qui peut représenter $2^8 = 256$ caractères différents. Ainsi, pour les buts pratiques et selon les corpus de textes disponibles, nous nous limitons à l'alphabet [6] pour le français et l'anglais, et à l'alphabet [7] pour le polonais :

[6] {a, A, à, À, â, Â, ä, Ä, b, B, c, C, ç, Ç, d, D, e, E, è, È, é, É, ę, Ę, ë, Ë, f, F, g, G, h, H, i, I, î, Î, ï, Ï, j, J, k, K, l, L, m, M, n, N, ñ, Ñ, o, O, ô, Ô, ö, Ö, œ, Œ , p, P, q, Q, r, R, s, S, t, T, u, U, ù, Ù, û, Û, ü, v, V, w, W, x, X, y, Y, z, Z}

[7] {a, A, ą, Ą, b, B, c, C, ć, Ć, d, D, e, E, ę, Ę, f, F, g, G, h, H, i, I, j, J, k, K, l, L, ł, Ł, m, M, n, N, ń, Ń, o, O, ó, Ó, p, P, q, Q, r, R, s, S, ś, Ś, t, T, u, U, v, V, w, W, x, X, y, Y, z, Z, ź, Ź, ż, Ż}

2.2.2 Mot simple et mot composé

Le choix de l'alphabet pour la langue traitée mène à la définition suivante d'une forme simple et par la suite du mot simple :

Définition 1a :
Une forme simple est une séquence consécutive non vide de caractères de l'alphabet apparaissant entre deux séparateurs.

Ceci est une définition purement orthographique. Par exemple, " letre " et " possiblle " sont des formes simples même si elles ne sont pas des mots du français. Nous introduisons donc la définition du mot simple :

Définition 1b :
Un **mot simple** de la langue traitée est une forme simple qui constitue une entrée d'un des dictionnaires de mots simples fléchis de cette langue.

Sans recours à la définition de ce dictionnaire (voir section 2.3), que nous pouvons voir comme une simple liste de mots, nous remarquons le deuxième problème dans les notions introduites ici : les mots non reconnus à cause de l'incomplétude du dictionnaire seront traités

comme incorrects. Une solution possible est celle de développer des algorithmes qui " devinent " les propriétés (e.g. la catégorie grammaticale) du mot inconnu par des calculs approximatifs fondés sur les terminaisons, le contexte, la statistique etc. Une autre approche (admise par le LADL pour le système DELA, et donc également pour le système INTEX), plus coûteuse, est celle de continuer constamment la correction et l'actualisation des dictionnaires. Nous pouvons opter pour une combinaison plus ou moins équilibrée de ces deux méthodes, en fonction de la précision et de la robustesse nécessaires dans l'application visée.

Voici des exemples de mots simples de l'anglais ([8]), du français ([9]) et du polonais ([10]) :

[8] *frame, consequences, done, airmail, better, of*

[9] *pomme, hier, vole, entête, SNCF, France*

[10] *znak, dobrego, nigdy*

D'une façon analogue, nous définissons les notions d'une forme composée et d'un mot composé :

Définition 2a :
Une **forme composée** est une séquence consécutive d'au moins deux formes simples et blocs de séparateurs.

Une forme composée a donc la forme suivante :

[11] $<b_1> <b_2> ... <b_n>$, où
- n >= 2,
- pour chaque 1 <= i <= n, $<b_i> \in S^+$ ou $<b_i> \in A^+$ (A est l'alphabet et S est l'ensemble de séparateurs)
- pour chaque 1 <= i < n , $<b_i> \in S^+ \Rightarrow <b_{i+1}> \in A^+$.

Pour des raisons opérationnelles de la méthodologie DELA et du système INTEX, deux conditions supplémentaires ont été introduites pour les formes composées :

1) Une forme composée ne doit pas contenir de virgules ni de points, ces deux caractères ayant le statut de métacaractères dans les entrées des dictionnaires DELA (voir section 2.3)[3].
2) Une forme composée doit commencer par une lettre. Cette mesure facilite les algorithmes du système INTEX de reconnaissance de mots dans des textes.

La définition 2a, comme la définition 1a, est aussi purement orthographique. Par exemple, *machine à* ou *a vu 5* sont des formes composées, mais ne sont pas des mots du français. Nous introduisons donc la définition du mot composé :

Définition 2b :
Un **mot composé** de la langue traitée est une forme composée qui constitue une entrée du dictionnaire des mots composés fléchis de cette langue.

La présentation détaillée du dictionnaire mentionné dans la définition ci-dessus se trouve dans la section 2.4. Pour les mots composés nous nous heurtons au même problème que pour les

[3] Ce problème peut être résolu par un mécanisme de protection de métacaractères semblable à celui utilisé en informatique à l'aide du métacaractère « \ ».

mots simples – celui de l'incomplétude du dictionnaire. Nous pouvons y remédier partiellement par l'utilisation des outils d'enrichissement terminologique comme celui qui fait l'objet du chapitre 8, mais un tel outil n'est pas fourni dans la version actuelle (4.21) d'INTEX.

Voici des exemples des mots composés en anglais, français et polonais :

[12] *black sheep, rock'n'roll, after all, ambassadors-at-large, Empire State Building*

[13] *pommes de terre, aujourd'hui, c'est-à-dire, peaux rouge, Canal +*

[14] *bialy kruk* (une chose rare ou introuvable, litt. "un corbeau blanc"), *plan 5-letni, znaków zapytania* (point d'interrogation – génitif, pluriel), *promieniowanie γ* (rayonnement gamma)

La grande majorité des séquences contenant des séparateurs ne sont pas concernées par les deux conditions rajoutées à la définition 2a. Mais dans des langages techniques nous trouvons un certain nombre d'unités qui commencent par un séparateur ou qui contiennent une virgule ou un point. Voici des exemples dont certains sont tirés du domaine de l'informatique traité en détails dans la section 7.5.3.

[15] (ang.) *λ-calculus*

[16] (ang.) *1DIR+*

[17] (ang.) *P.O.D.*

[18] (ang.) *multiple instructions, multiple data*

D'autre part, la définition 2a, même sans rajout des deux conditions, n'admet pas de cas où un mot est constitué uniquement de séparateurs, comme les exemples suivants (toujours du domaine de l'informatique) :

[19] *_1576* (nom d'un virus de PC)

[20] *1963* (nom d'un virus de PC, ou un déterminant composé numérique)

Les séquences de [15] à [20] ne peuvent pour l'instant figurer ni dans un dictionnaire du type DELAS, ni du type DELACF, ce qui prouve que le traitement automatique des termes simples et composés ne peut pas toujours être effectué avec les mêmes méthodes que celles élaborées pour les mots simples et composés du langage général. L'élimination des problèmes posés par les exemples [17] et [18] peut être obtenue soit par le mécanisme de protection de métacaractères, soit par le changement du format des dictionnaires : d'une liste textuelle d'entrées (voir sections 2.3 et 2.4) vers une base de donnée relationnelle.

Les quatre définitions introduites plus haut sont précises et formelles. Néanmoins, nous n'échappons pas aux questions linguistiques bien connues au sujet de la composition. Car si c'est grâce aux dictionnaires que nous statuons sur les mots simples et composés, il faut d'abord préciser selon quels critères ces dictionnaires sont créés, c'est-à-dire décider quelles formes simples et composées doivent y être introduites.

Quant aux mots simples, les principes du système DELA admettent toutes les formes simples apparaissant dans des textes de référence. Par exemple, pour les dictionnaires du français général nous utilisons surtout le journal « Le Monde ». Nous ne prenons évidemment pas en compte les fautes de frappe et d'orthographe qui peuvent se trouver dans ces textes, bien que le statut de certaines graphies soit parfois arbitraire.

Les formes composées demandent souvent beaucoup d'analyse avant de pouvoir être classées comme mots composés. Différents critères sont utilisés par différents linguistes qui se prononcent à ce sujet (voir section 1.3.1). L'argumentation de G. Gross (1988) fondée sur l'analyse transformationnelle a influencé de façon importante les auteurs du système DELA. Elle introduit la notion du **degré de figement** d'une séquence qui est d'autant plus élevé que cette séquence accepte moins de transformations syntaxiques prévues pour sa structure. Puisque le nombre de transformations à analyser peut facilement atteindre une centaine (voir A. Monceaux 1994), le nombre de tous les types possibles de figement s'approche à 2^{100}. Pour le traitement automatique du langage naturel nous avons besoin d'une définition opératoire qui ne peut pas distinguer tous ces types de figement. On a donc décidé que même le plus petit degré de figement suffit pour considérer une séquence comme figée. Autrement dit, si une séquence interdit au moins une des règles transformationnelles caractéristiques à sa structure, cette séquence doit apparaître dans le dictionnaire des mots composés.

Cette définition, suffisamment rigoureuse, n'est pas facile à utiliser lors du recensement des mots composés puisqu'elle prend en compte un grand nombre de transformations. C'est pourquoi nous allons renoncer à l'étude transformationnelle exhaustive des candidats pour des mots composés, et nous allons nous contenter de quelques règles formelles et intuitives générales.

Premièrement, nous admettons le principe de **non compositionnalité**. Nous disons qu'une forme composée est un mot composé si elle peut être considérée **comme unité atomique syntaxique et/ou sémantique**, c'est-à-dire si ses propriétés syntaxiques, sémantiques et/ou distributionnelles ne peuvent pas être calculées à partir de ceux de ses constituants. L'atomicité syntaxique est visible dans l'exemple d'un *peau rouge* qui est un nom composé au masculin, alors que son composant nominal *peau* est au féminin. Or, selon les règles standard de construction, le genre et le nombre d'un groupe nominal composé de cette façon proviennent du nom et de l'adjectif qui le constituent. De la même façon, un *merle blanc* est atomique sémantiquement car ceci n'est pas un merle qui est blanc, mais une chose ou une personne rare, introuvable. Comme exemple de l'atomicité distributionnelle, prenons le *cordon bleu* qui est humain malgré sont composant nominal inanimé *cordon*.

Ce raisonnement est simple pour les expressions métaphoriques, mais nous comptons comme mots composés beaucoup plus que ces expressions-là. La signification de nombreuses séquences est souvent liée à celles de leurs composants, mais cette liaison peut être imprévisible. Par exemple en anglais, on rencontre plusieurs noms composés dont le sens « a quelque chose à voir » avec l'air, mais cette relation est à chaque fois différente :

[21] **air-bed** : mattress that can be filled with air[4]

[22] **air brake** : brake worked by air pressure

[23] **air-conditioning** : system controlling the temperature of the air

[24] **air force** : branch of the armed forces that uses aircraft

[25] **air hostess** : stewardess in a passenger aircraft

[26] **air pocket** : partial vacuum in the air

Air-bed se rapporte à la présence d'air à l'intérieur d'un objet, tandis qu'*air pocket* signifie son absence etc. De plus, il n'est pas possible de « calculer » les mots soulignés dans les définitions à partir de la seule présence des constituants de la forme composée, alors qu'ils y

[4] Les définitions selon OALDCE (1989).

sont indispensables pour la compréhension des termes. C'est pourquoi ces séquences doivent être recensées et décrites en tant que noms composées.

Une discussion détaillée et riche en exemples de cette notion du nom composé se trouve chez Silberztein (1993b). Il y propose quatre critères de distinction entre ce qu'on appelle « noms composés lexicalisés » et les groupes nominaux libres. A part les trois critères que nous venons d'expliquer brièvement - l'atomicité syntaxique, sémantique, et distributionnelle - il prend en compte aussi l'institutionnalisation de l'usage qui fait que certains termes comme *chefs d'état* sont utilisés couramment sans jamais être remplacés par leurs équivalents syntaxiques et sémantiques comme **chefs de pays*.

Les critères présentés ci-dessus ont guidé les lexicographes et nous-même dans la tache de recensement de mots composés anglais qui a résulté en le dictionnaires électronique DELAC (chapitre 5). Néanmoins, suite à l'analyse des travaux sur la notion de mot composé (section 1.3.1), et à nos propres expériences en création et application de dictionnaires électroniques de mots composés, nous constatons que la question de recensement explicite d'unités complexes en vu d'analyse automatique de textes dépend fortement de l'application. Par exemple, l'analyse lexicale de textes techniques doit être effectuée avec des dictionnaires de termes pertinents du domaine, et nous pouvons dire que les termes composés d'un domaine ne le sont pas forcément pour un autre domaine. Du point de vue de l'aide à la traduction, les constructions figées pertinentes (contenues dans le glossaire du texte à traduire) dépendent non seulement du domaine, mais du corpus, du client, et du contrat de traduction (voir section 8.3).

La règle de non compositionnalité, telle que nous l'admettons, a pour conséquence une ambiguïté systématique des mots composés, discutée plus en détail dans la section 2.8. Nous avons dit qu'une séquence composée « peut être considérée » comme unité atomique syntaxique et/ou sémantique. Mais la plupart des composés peuvent aussi apparaître dans certains contextes en tant que groupes nominaux libres, comme dans la phrase :

[27] Elle a rangé le cordon bleu dans le tiroir.

Quant à l'aspect orthographique de notre définition du mot composé, remarquons que, pour les besoins computationnels, nous exigeons l'existence d'au moins un séparateur à l'intérieur d'un mot composé. Ainsi les mots français *gentilhomme* et *pourboire*, les mots anglais *railway, fireman,* et *rattlesnake*, ainsi que les mots polonais *ognioodporny, gołosłowny* et *ogniomistrz* figurent dans les dictionnaires des mots simples.

Dans certaines langues, les séparateurs sont soit limités, comme en allemand - une grande partie de groupes nominaux s'écrivent sans séparateur - soit inexistants, comme en thaï - il n'existe ici aucun séparateur ni de mots, ni de phrases.

Dans le cas de l'allemand, les mots composés sémantiquement ou syntaxiquement peuvent avoir le même statut, et être traités par les mêmes algorithmes que les mots simples, i.e. être recensés dans des dictionnaires que l'on applique aux textes. Un gros problème est posé par contre par les groupes nominaux libres qui eux aussi s'écrivent souvent sans séparateur. Pour pouvoir les reconnaître, il est nécessaire de disposer non seulement des listes de mots simples, mais aussi des modifications qu'ils subissent quand ils se soudent, par exemple le mot *Schulerinnerung* (souvenir de l'école) est composé de deux morphèmes simples *Schule* (école) et *Erinnerung* (souvenir), ce qui nécessite l'omission de la lettre *e*. Des algorithmes de division de séquences soudées en composants sont requis pour une analyse lexicale efficace.

La langue thaï est un cas extrême d'ambiguïté lexicale. La difficulté de son traitement est du même ordre que celle rencontrée dans le traitement de la parole car aucun critère fixe ne peut

permettre d'identifier les unités de traitement, même au niveau des phrases. Ainsi, les définitions 1a et 2a d'une forme simple et composée ne sont pas utilisables. Les mots (sans distinction entre les mots simples et composés) sont toutes les séquences de caractères qui sont recensées dans le dictionnaire. Remarquons que l'utilisation de dictionnaires relativement complets permettra de repérer certains endroits où un séparateur hypothétique de mots ou phrases est obligatoire ou interdit.

2.2.3 Constituants caractéristiques des mots composés.

Pour continuer la description des particularités qui aident à faire la distinction entre les noms composés et les groupes nominaux libres, nous introduisons la notion de la tête d'un composé de la même façon qu'elle est définie pour tous les syntagmes.

Définition 3 :
La **tête** (dite aussi les **constituants caractéristiques**) d'un mot composé est constituée des composants qui ont les mêmes traits morphologiques que le mot composé lui-même.

Par exemple, en français la plupart des noms composés du type *Nom Adjectif* ont le même genre et nombre que ceux du nom et de l'adjectif qui les constituent, comme

[28] *étoile filante*

qui est au féminin singulier, tout comme *étoile* et *filante*[5].

En polonais, cet accord inclut aussi le cas[6], par exemple

[29] *dusza towarzystwa* (un boute-en-train)

du type *Nom Nom*$_{(au\ génitif)}$ est au nominatif singulier féminin, comme son nom caractéristique *dusza*, et non pas comme *towarzystwa* qui est au génitif singulier neutre.

En anglais, seul le nombre est concerné. Par exemple :

[30] *by-product* (dérivé)

est au singulier, tout comme son nom caractéristique *product*.

Voici un résumé pour les trois langues des constituants caractéristiques (soulignés dans les exemples) des classes les plus productives de noms composés :

---

[5] La notion de tête d'un syntagme n'est pas la même pour différents auteurs. Par exemple, pour Bourigault (1994) dans un terme complexe français du type *Nom Adj* la tête ne contient que le nom, l'adjectifs étant son expansion.

6 Il existe 7 cas en polonais: le nominatif, de génitif, le datif, l'accusatif, l'instrumental, le locatif, et le vocatif.

Structure	Tête	Exemples
Nom Adjectif	Nom et Adjectif	*carte postale, vaches maigres*
Adjectif Nom	Adjectif et Nom	*petit ami, belle-mère, grand-angle*
Nom_1 de Nom_2	Nom_1	*chemin de fer, soif des honneurs*
Nom_1 à Nom_2	Nom_1	*barbe à papa, tir à l'arc, café au lait*
Nom_1 Préposition Nom_2	Nom_1	*saut en hauteur, preuve par l'absurde*
Préposition Nom	Nom	*avant-garde, contre-exemple*
Nom_1 Nom_2	Nom_1	*moissonneuse-batteuse, bateau-mouche*
Verbe Nom	pas de tête	*porte-avions, garde-voie*

Tab.1 Les classes des noms composés et leurs têtes en français

Structure	Tête	Exemples
Nom_1 Nom_2	Nom_2	*gas station, pony- tail, police state*
Adjectif Nom	Nom	*black hole, absolute zero*
Nom_1 of Nom_2	Nom_1	*table of contents, man-of-war*
Nom_1 Préposition Nom_2	Nom_1	*brother-in-law, skeleton in the cupboard*
Préposition Nom	Nom	*by-product, at-home*

Tab.2 Les classes des noms composés et leurs têtes en anglais

Structure	Tête	Exemples
Nom Adjectif	Nom et Adjectif	*choroba morska, nożyce krawieckie*
Adjectif Nom	Adjectif et Nom	*babie lato, pierwsze skrzypce*
Nom_1 $Nom_{2(génitif)}$	Nom_1	*gorączka złota, rachunek prawdopodobieństwa*
Nom_1 Nom_2	Nom_1	*dama karo, majster klepka, hocki-klocki*
Nom_1 Préposition Nom_2	Nom_1	*cisza przed burzą, cukier w kostkach*

Tab.3 Les classes des noms composés et leurs têtes en polonais

La notion des constituants caractéristiques d'un composé est étroitement liée à celle de la non-compositionnalité syntaxique que nous avons discutée dans la section précédente. Le fait que l'on ne puisse pas calculer les propriétés syntaxiques d'une séquence à partir de celles de ses constituants signifie qu'au moins l'un des cas suivants se produit :

1) la tête est inexistante (dans ce cas-là on parle de composés *exocentriques*[7]) ou n'est pas celle qui est prévue pour la structure donnée (par exemple *peau-rouge*, voir section 3.2)
2) au moins une des formes fléchies prévues pour la structure donnée est interdite (*bits and pieces* etc.),
3) au moins une des formes fléchies se construit de façon irrégulière, c'est-à-dire ce n'est pas précisément la tête qui se fléchit (*bateau-mouche* etc.).

Nous discutons plus en détail ces problèmes dans le chapitre 3.

2.3 Dictionnaires des mots simples et transducteurs de flexion

Le traitement des mots composés doit être précédé d'une description exhaustive des mots simples. La méthodologie de description des mots simples qui a été implémentée dans le système DELA a été appliquée pour de nombreuses langues (le français, l'anglais, l'allemand, l'espagnol, le portugais, l'italien, le grec, le russe, le coréen, l'ancien français) et fait l'objet de plusieurs ouvrages (voir par exemple Courtois 1990 et Silberztein 1993a pour le français, Sklavounou 1999 pour le grec moderne). Nous n'allons faire ici qu'un bref résumé de cette méthodologie. Elle contient deux phases.

1) La construction du dictionnaire DELAS[8]. Ceci est une liste de tous les mots simples sous leurs formes lemmatisées, i.e. l'infinitif pour les verbes, le masculin singulier (au nominatif le cas échéant) pour les adjectifs et les déterminants, le singulier (au nominatif le cas échéant) pour les noms (ou le pluriel si le singulier n'existe pas), le positif pour les adjectifs et les adverbes admettant la graduation etc., ainsi que de tous les mots simples invariables comme les prépositions, les adverbes invariables etc. Chacun des mots simples est accompagné d'un **code flexionnel** qui décrit entièrement la façon d'obtenir toutes les formes fléchies de ce mot à partir de sa forme lemmatisée. Par exemple, l'entrée du DELAS français :

[7] Terme utilisé par exemple par Grévisse (1993).
[8] Dictionnaire Electronique du LAdl pour le mots Simples

[31] nouveau,A72

représente le lemme *nouveau* qui est un adjectif, et qui se fléchit selon le schéma suivant : le masculin singulier (*nouveau*) est équivalent au lemme (i.e. aucune terminaison n'est à rajouter), pour obtenir le féminin singulier (*nouvelle*) il faut effacer les deux dernières lettres du lemme et rajouter la terminaison *lle*, pour le pluriel au masculin (*nouveaux*) il faut rajouter la séquence *x*, et pour le pluriel au féminin (*nouvelles*) il faut effacer les 2 dernières lettres et rajouter *lles*. Ce principe est décrit par la combinaison suivantes de symboles : ***(<E>**:ms, **2lle**:fs, **x**:mp, **2lles**:fp)*. Les séquences qui précèdent les deux-points sont des opérations à effectuer sur le lemme. Les séquences qui suivent les deux-points sont les traits morphologiques des formes fléchies obtenues. Le symbole <*E*> représente la séquence vide. Un chiffre (ici ***2***) indique le nombre de lettres qu'il faut effacer à la fin du lemme. Les suites de lettres (ici ***lle***, ***x***, ***lles***) sont les terminaisons qu'il faut rajouter à la fin du lemme (éventuellement réduit par des effacements). Les traits morphologiques correspondent au masculin (*m*), au féminin (*f*), au singulier (*s*) et au pluriel (*p*). Ce schéma est équivalent à un transducteur fini dit **transducteur de flexion** représenté par le graphe[9] au format INTEX de l'illustration Fig.1.

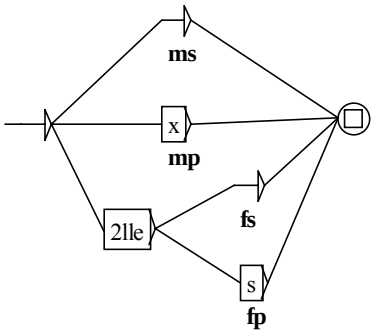


Fig.1. Transducteur de flexion A72

L'alphabet d'entrée de ce transducteur est constitué du symbole <*E*> pour la séquence vide, des chiffres (qui peuvent être remplacés par l'occurrence de plusieurs caractères *L*) pour le passage à gauche, et de tout l'alphabet de la langue (voir section 2.2.1). L'alphabet de sortie contient les codes pour tous les traits flexionnels de la langue, comme dans le tableau ci-dessous. Pour le polonais ces codes ne sont pas précisément ceux du dictionnaire sur lequel nous nous basons, celui du prof. K. Bogacki de l'Université de Varsovie. Nous introduisons certaines modifications (e.g. pour les genres masculins) afin de pouvoir appliquer l'algorithme de flexion des mots composés décrit dans le chapitre 4.

Types de flexion	Formes	Français	Anglais	Polonais
Nombre	singulier	s	s	s
	pluriel	p	p	p
Genre	masculin	m	-	-
	féminin	f	-	f

[9] Un graphe d'Intex est équivalent à un transducteur fini, ce que l'on peut voir si pour chaque nœud du graphe: a) on ajoute un état avant ce nœud; b) on transforme le nœud en une transition qui sera étiquetée par le symbole à l'intérieur de ce nœud; c) le nœud le plus à gauche devient l'état initial; d) le nœud entouré d'un cercle devient l'état final. La direction des transitions est toujours celle du côté droit d'un nœud vers le côté gauche d'un autre nœud.

	masculin humain	-	-	o
	masculin animé	-	-	z
	masculin inanimé	-	-	r
	neutre	-	-	n
Cas	nominatif	-	-	M
	génitif	-	-	D
	datif	-	-	C
	accusatif	-	-	B
	instrumental	-	-	I
	locatif	-	-	L
	vocatif	-	-	W
Personne	première	1	1	1
	deuxième	2	2	2
	troisième	3	3	3
Temps et mode	infinitif	W	W	F (*czytać*)
	indicatif présent	P	P	H (*czytam*)
	indicatif imparfait	I	I	P (*czytałem*)
	passé simple	J	-	-
	passé impersonnel	-	-	S (*czytano*)
	indicatif futur	F	-	U (*przeczytam*[10])
	participe présent	G	G	-
	participe présent adjectival	-	-	J (*czytający*)
	participe présent adverbial	-	-	K (*czytając*)
	participe passé	K	K	-
	participe passé adjectival	-	-	Q (*czytany*)
	participe passé adverbial	-	-	T (*przeczytawszy*[11])
	subjonctif présent	S	-	-
	subjonctif	T	-	-

[10] *przeczytam* n'est pas l'indicatif futur du verbe imperfectif *czytać*, mais du verbe perfectif *przeczytać*. En polonais les verbes imperfectifs n'ont pas la forme de l'indicatif futur, et les verbes perfectifs n'ont pas d'indicatif présent.

[11] *przeczytawszy* n'est pas le participe passé adverbial du verbe imperfectif *czytać*, mais du verbe perfectif *przeczytać*. En polonais les verbes imperfectifs n'ont pas la forme du participe passé adverbial, et les verbes perfectifs n'ont pas de participe présent adverbial.

	imparfait			
	conditionnel présent	C	-	Z (*czytałbym*)
	impératif	Y	-	G (*czytaj*)
Gradation	positif	-	∅[12]	∅[13]
	comparatif	-	C	c
	superlatif	-	S	u

Tab.4 Traits flexionnels du français, anglais et polonais

Tous les mots simples qui se fléchissent de la même façon partagent le même code flexionnel, par exemple :

[32] beau,N72

A ce formalisme simple d'effacement et de concaténation de lettres à la suite d'un lemme (i.e. d'une entrée du DELAS) peuvent se joindre d'autres opérations élémentaires selon les besoins de la langue traitée. Par exemple, INTEX introduit l'opération de recopie *C* et du passage à droite *R* pour décrire plus facilement le changement systématique d'une voyelle dans un ensemble de mots de base qui ne partagent pas la même désinence. Ainsi, en français les verbes *céder* et *espérer*, dans lesquels l'accent aigu est remplacé par l'accent grave au singulier présent (*cède*, *espère*), ont le même code flexionnel V7 décrit par le schéma *(**4èRCRC**:P1s:P3s, **4èRCRCs**:P2s, **2ons**:P1p, **1z**:P2p, **4èRCRCnt**:P3p, ...)*. Par exemple, pour obtenir la première ou la troisième personne du singulier présent (*:P1s:P3s*) du mot *céder*, il faut, en se plaçant à la fin du mot, reculer de 4 lettres à gauche, inscrire la lettre *è*, aller d'une lettre à droite (*R*), recopier (*C*) la lettre courante (*d*), et répéter les deux dernières actions. Ceci est illustré par le tableau ci-dessous :

Opération	Lemme	Résultat
	*céder^*	*céder*
4	*c^éder*	*c*
è	*c^éder*	*cè*
R	*cé^der*	*cè*
C	*cé^der*	*cèd*
R	*céd^er*	*cèd*
C	*céd^er*	*cède*

Si l'on n'utilisait pas les opérations *R* et *C* il aurait fallu deux codes flexionnels différents : *(**4ède**:P13:P3s, **4èdes**:P2s,...)* pour *céder* et *(**4ère**:P1s:P3s, **4ères**:P2s,...)* pour *espérer* car ces deux mots n'ont pas la même terminaison. Après l'insertion du *è* à la place du *é* il faut recopier la lettre *d* dans le premier cas et la lettre *r* dans le deuxième. Dans certains cas l'augmentation du nombre des codes flexionnels pourrait être importante, ce qui rendrait plus

[12] Code nul: si un adjectif ou un adverbe apparaît sans marque de gradation alors le positif est sous-entendu.

[13] Code nul: si un adjectif ou un adverbe apparaît sans marque de gradation alors le positif est sous-entendu.

difficile leur gestion. Par exemple, en allemand il faudrait ajouter près de 300 codes de flexion nouveaux si les opérations ***R*** et ***C*** n'étaient pas disponibles.

En polonais, les opération *R* et *C* peuvent intervenir dans des exemples comme *stopa* ou *głowa.* Dans le DELAS de prof. Bogacki ces deux mots ont deux codes différents, N225 et N20 respectivement, car leur terminaisons sont différentes. Si l'on emploi les opérations ***R*** et ***C*** ces deux codes peuvent être fusionnés de la façon suivante :

*(**<E>**:Mfs, **1y**:Dfs:Mfp:Bfp:Wfp, **1ie**:Cfs:Lfs, **1ę**:Bfs, **1ą**:Ifs, **1o**:Wfs, **3óRC**:Dfp, **1om**:Cfp, **mi**:Ifp, **ch**:Lfp)*.

Le premier trait flexionnel représente le cas (voir Tab.4). La forme du féminin pluriel génitif (*stóp, głów*) se construit par le retrait de 3 lettres a gauche, le rajout de la lettre *ó*, passage d'une lettre à droite et la recopie de la lettre courante.

D'autres opérations, à part le passage à gauche, l'insertion d'une lettre, la recopie et le passage à droite sont envisageables si les besoins d'une langue particulière le justifient. Par exemple, Sklavounou (1999) a proposé l'opération de désaccentuation pour les noms et les adjectifs du grec moderne. La plupart des noms de cette langue forment l'accusatif en désaccentuant une voyelle. Si cette opération pouvait être ajoutée à celles déjà disponibles pour la description de flexion cela permettrait de limiter considérablement le nombre de codes flexionnels. L'opération de désaccentuation s'appliquerait sans doute aussi avec succès aux noms ou adjectifs espagnols qui portent un accent écrit a la dernière syllabe et le perdent lors de la mise au pluriel ou au féminin, comme *lección -> lecciones, siamés -> siameses, siamesa, siamesas*[14].

Précisons que les opérations supplémentaires comme la recopie ou la désaccentuation ne changent en rien la puissance descriptive obtenue avec seulement le passage à gauche et l'insertion d'une lettre. Le seul intérêt de ces ajouts est de rendre l'ensemble des codes flexionnels d'une langue le plus petit et le plus intuitif possible. D'autre part, de tels ajouts dans un système informatique comme INTEX peuvent représenter un surcoût important, il faut donc que leur introduction soit bien motivée.

2) Génération automatique du dictionnaire DELAF[15]. Ce dictionnaire est une liste de toutes les formes fléchies de tous les mots simples répertoriés dans le DELAS. Grâce aux codes flexionnels accompagnant les formes lemmatisées dans le DELAS, la génération du DELAF se fait entièrement automatiquement. Par exemple, à l'entrée [31] du DELAS français correspondent dans le DELAF les 4 entrées suivantes :

[33] nouveau,.A:ms

[34] nouvelle,nouveau.A:fs

[35] nouveaux,nouveau.A:mp

[36] nouvelles,nouveau.A:fp

[14] L'opération de désaccentuation peut être simulée par les actions suivantes : a) dans tous les mots du DELAS on substitue chaque lettre accentuée par une lettre sans accent suivie d'un apostrophe, e.g. *ó* → *o'* ; b) dans les codes flexionnels on représente chaque opération de désaccentuation par l'opération d'effacement de l'apostrophe ; c) on restitue les accents non effacés dans les formes fléchies obtenues. Par exemple, le mot espagnol *lección* est transformé en *leccio'n*, et ensuite par l'opération ***2RCes*** nous obtenons son pluriel *lecciones.*

[15] Dictionnaire Electronique du LAdl pour le mots simples Fléchis

Chaque entrée contient une forme fléchie suivie d'une virgule, suivie de son étiquette grammaticale constituée de sa forme lemmatisée (éventuellement vide si les deux sont identiques), suivie du point, suivi de la catégorie, suivie des deux-points, suivi des traits flexionnels.

Voici d'autres exemples des DELAF français, anglais et polonais :

[37] (fr.) *mesdames,madame.N:fp*

[38] (fr.) *cédaient,céder.V:I3p*

[39] (ang.) *lay,lie.V:I* (mentir)

[40] (ang.) *lay,.V:P1s:P2s:P1p:P2p:P3p* (poser)

[41] (ang.) *that,.PRON*

[42] (pol.) *piekła,piec.V:P3fs* (cuire au four)

[43] (pol.) *piekła,piekło.N:Dns:Mnp:Bnp:Wnp* (enfer)

[44] (pol.) *niskich,niski.A:Dop:Dzp:Drp:Dfp:Dnp:Lop:Lzp:Lrp:Lfp:Lnp* (bas)

Quand plusieurs formes fléchies sont identiques pour un mot, comme dans l'exemple *niskich*, leurs étiquettes sont factorisées et toutes les séquences de traits flexionnels correspondantes apparaissent séparées par les deux-points.

Pour l'analyse morphologique automatique seul le dictionnaire DELAF est utilisé car ce sont les formes fléchies qui apparaissent dans des textes. Néanmoins, le dictionnaire DELAS est nécessaire pour la maintenance (le rajout de nouveaux mots et la correction des anciens) ainsi que pour la flexion automatique du dictionnaire des mots composés (voir section suivante).

2.4 Dictionnaires des mots composés

La description des mots composés se fait, comme pour les mots simples, par leur recensement explicite.

3) Le dictionnaire DELAC[16] est une liste de formes lemmatisées de mots composés. Par exemple, l'entrée suivante du DELAC français

[45] *abbaye(N21) cistercienne(A41),N+NA:fs/-+*

représente le fait que le mot composé *abbaye cistercienne* est un nom (*N*) de la typologie *Nom Adjectif* (*+NA*) féminin singulier (*fs*) et admettant la flexion en nombre et non pas en genre (-+). Les composants simples qui varient lors de la flexion du composé entier (dans la plupart de cas ceci sont les constituants caractéristiques - voir section 2.2.3) sont accompagnés de leurs codes flexionnels identiques à ceux qui apparaissent dans le dictionnaire DELAS. Ce format d'entrée a été admis pour le dictionnaire DELAC français (voir Silberztein 1990). La génération du DELACF a été faite par un algorithme spécialisé pour le français (i.e. inutilisable pour d'autres langues), qui exigeait entre autres le dédoublement des entrées à pluriel irrégulier comme *blanc d'œuf - blancs d'œuf, blanc d'œufs* (voir section 3.3, exemples [132]-[136]), et même la flexion manuelle de certains cas difficiles (*peau rouge*).

Lors du travail sur le programme universel de flexion automatique d'un DELAC vers un DELACF (chapitre 4), nous avons démontré que les informations fournies pour les constituants caractéristiques doivent contenir, en plus de leurs codes flexionnels, aussi leurs

[16] Dictionnaire Electronique du LAdl pour les mots Composés.

lemmes et leurs traits morphologiques (voir section 4.2). Ainsi, la forme complète de l'entrée [45] est la suivante :

[46] abbaye(abbaye,N21:fs) cistercienne(cistercien,A41:fs),N+NA:fs/+N

Nous avons également proposé, pour les besoins de la flexion automatique, de diviser le DELAC en sous-listes regroupant les composés qui se fléchissent de la même façon (voir section 4.4).

Voici d'autres exemples d'entrées des DELAC français, anglais et polonais.

[47] (fr.) *cousin(cousin.N32:ms) germain(germain.N32:ms),N+NA:ms/+N+G*

[48] (fr.) *allocation(allocation.N21:fs)-chômage(chômage.N1:ms),N+NN:fs/+N*

[49] (fr.) *auto-immun(immun.A21:ms).A:ms/+G*

[50] (ang.) *blue collar worker(worker.N1:s),N+ANN:s/+N* (ouvrier)

[51] (ang.) *ups and downs,N+XAndX:p* (les hauts et les bas)

[52] (ang.) *as soon as possible,ADV* (le plus vite possible)

[53] (pol.) *słomiany(słomiany.A1:Mos) wdowiec(wdowiec.N137:Mos), N+AN:Mos /+f+N+C*
(litt. *un veuf de paille* = un homme dont la femme est partie en voyage)

[54] (pol.) *zamki(zamek.N1244:Mrp) na lodzie,N+NPrepN:Mrp/+C*
(litt. des *châteaux sur glace* = projets irréalisables)

[55] (pol.) *czarno na białym.ADV* (noir sur blanc)

Remarquons que :

- les traits morphologiques de la forme canonique d'un mot composé ne sont pas forcément les mêmes que ceux des formes canoniques de ses constituants caractéristiques, par exemple *zamki na lodzie* ([54]) est au pluriel tandis que la forme canonique *zamek* du nom caractéristique *zamki* est au singulier;
- les étiquettes grammaticales ne sont pas nécessaires là où le composé ne possède que sa forme canonique, comme *ups and downs* ([51]);
- *cousin germain* ([47]) peut être marqué comme subissant toute flexion en genre (*+G*) car il n'y a que deux genres en français, tandis que *słomiany wdowiec* ([53]) reçoit que la marque *+f* car sa forme féminine existe[17] mais non pas les autres genres du polonais : le masculin animé, le masculin inanimé et le neutre.

4) Le dictionnaire DELACF[18] est une liste de toutes les formes fléchies des mots composés contenus dans le DELAC. Par exemple, l'entrée [47] du DELAC français se fléchie en nombre et en genre (*+N+G*) ce qui donne les quatre formes suivantes :

[56] cousin germain,.N+NA:ms

[57] cousine germaine, cousin germain.N+NA:fs

[58] cousins germains, cousin germain.N+NA:mp

[59] cousines germaines, cousin germain.N+NA:fp

[17] Bogacki n'admet pas dans son DELAS la possibilité de flexion en genre de noms polonais.
[18] Dictionnaire Electronique du LAdl pour les mots Composés Fléchis.

Le format est identique à celui du DELAF et la génération du DELACF à partir du DELAC est automatique grâce aux étiquettes grammaticales indiquées dans le DELAC et à la division du DELAC en sous-listes selon le type de flexion des mots composés. L'opération de flexion automatique des composés est décrite dans le chapitre 4.

Comme pour les mots simples, seul le dictionnaire des formes fléchies des composés est utilisé pour l'analyse morphologique automatique des textes. Silberztein (1993a, pp. 98-100) montre qu'il est préférable de reconnaître les composés dans des textes par la consultation d'un DELACF de taille souvent très importante (voir tableau ci-dessous) plutôt que par calcul, i.e. par un algorithme qui permet de retrouver la forme lemmatisée d'un composé à partir de sa forme fléchie en faisant des hypothèses sur les catégories, les codes flexionnels et les traits flexionnels des composants simples : un tel algorithme devrait être très lourd et inefficace à cause des mots ambigus et des exceptions.

Le tableau ci-dessus donne les nombres actuels d'entrées dans les 4 types de dictionnaires pour les trois langues considérées :

	Français	Anglais	Polonais
DELAS	120 000	166 000	150 000
DELAF	675 000	289 000	2 347 000
DELAC	126 000	60 000	1 100
DELACF	271 000	110 000	-[19]

Tab.5 Nombres d'entrées de dictionnaires électroniques

Le travail sur le DELAC polonais a été seulement commencé. Néanmoins, l'analyse des 1100 mots composés polonais a été importante pour l'élaboration de l'algorithme de flexion automatique de mots composés (chapitre 4).

2.5 Description des mots et expressions composés par expressions rationnelles et automates finis

Le recensement des mots composés n'est pas pratique pour les constructions qui :

- ont un nombre important de variantes orthographiques,
- ont un degré relativement élevé de productivité,
- utilisent un petit nombre de mots simples pour en créer un grand nombre de combinaisons.

Le premier cas concerne les mots comme

[60] (fr.) *tsar, tzar, csar, czar*

dont le nombre de variantes se multiplie encore dans des composés comme

[61] *tsar de Russie, tsar de toutes les Russies*

[19] Le DELACF polonais n'a pas été créé.

pour donner 4 * 2 = 8 variantes possibles qui sont plus facilement présentables dans le graphe de la figure Fig.3. Le nœud grisé de ce graphe cache le graphe de la figure Fig.2. Ce mécanisme d'imbrication permet à un graphe d'obtenir la puissance équivalente à celle d'un réseau récursif de transition (recursive transition network) si l'on admet la récursivité des références aux sous-graphes. Pour la description du lexique nous nous limitons aux imbrications non récursives, ainsi le formalisme reste toujours équivalent aux automates finis car tout graphe avec un nœud grisé peut être remplacé par un graphe plus complexe mais sans noeud grisé. Ceci sera obtenu par les remplacements successifs des nœuds grisés par les graphes qu'ils représentent et le nombre de ces remplacements sera toujours fini. Si on utilise un transducteur au lieu d'un simple automate, les sorties du transducteur peuvent servir à rattacher les différentes variantes orthographiques à une des variantes choisie comme canonique, comme *tsar* dans l'exemple Fig.2.

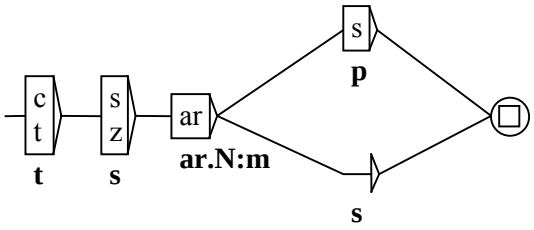


Fig.2. La représentation des variantes orthographiques du mot simple *tsar* par un graphe

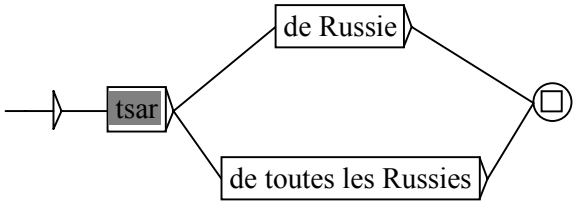


Fig.3. Les variantes orthographiques du mot composé *tsar de Russie*

Les automates finis, ainsi que les expressions rationnelles auxquelles ils sont équivalents, permettent une représentation commode de la productivité pour les séries suivantes[20] :

[62] (fr.) *vitamine (A+B+C+D+E)*

[63] (pol.) *komputer (pierwszej+drugiej+trzeciej+czwartej+piątej) generacji*
(ordinateur de la (première + deuxième + troisième + quatrième + cinquième) génération)

[64] (pol.) *rzut (kulą+dyskiem +oszczepem+młotem)*
(le lancer du (poids+disque+javelot+marteau))

[65] (pol.) *bieg na ((60+100+600+800+1000+1500+2000) metrów + 1 kilometr + (2 + 3 + 4) kilometry + (5 + 10) kilometrów) (<E>+przez plotki+z przeszkodami)*
(course de (haies+(60+...) mètres+(2+3+4+5+10) kilomètres))

[66] (ang.) *<NB>-year-old*[21]

[20] Dans les expressions rationnelles <E> symbolise la séquence vide, les parenthèses signifient le choix d'un parmi les éléments séparés par un plus.

[67] (ang.) (*sonata + fugue + symphony + study + toccata + passacaglia + cantata + suite + (<E> + violin + piano) concerto + oratorio + nocturne + waltz + polonaise + mazurka + prelude + requiem + mass) in ((C + D + E + F + G + A + B) + ((D + E + G + A + B) flat) + ((C + D + F + G + A) sharp)) (<E> + major + minor)*

Le problème majeur posé par la représentation de mots et expressions composés par automates finis (ou expressions rationnelles) est celui de la description des formes fléchies. Les expressions [62]-[67] ci-dessus ne contiennent que les formes de base des composés, elle ne sont donc pas directement utilisables dans l'analyse lexicale. L'inclusion de toutes les formes fléchies dans les automates mêmes est toujours possible (e.g. *(sonata+sonatas+fugue+fugues+symphony+symphonies+etc.) in (C+...)*)[22], mais là où la flexion est plus riche et où il faut tenir compte des accords entres plusieurs constituants, cette solution exige la multiplication importante du nombre de noeuds et de transitions.

Des familles importantes de composés ont déjà été décrites par des automates finis, par exemples les dates du français par Maurel (1989), ou les termes anglais de la bourse par Gross (1997). Dans le chapitre 6 nous présentons une bibliothèque d'automates et transducteurs finis pour la reconnaissance des déterminants numéraux cardinaux et ordinaux de l'anglais.

En résumé, la représentation des séquences figées par automates et transducteurs finis a deux avantages :

- les variantes du même mot (comme *tsar*) ou les expressions du même type (numéraux, dates, déterminants...) sont regroupées dans un seul graphe,
- les nombreuses combinaisons de mots sont plus faciles à représenter et à gérer par les différents chemins d'un graphe que par une longue liste.

2.6 Compactage des dictionnaires

Les dictionnaires DELAF et DELACF décrits dans les sections 2.3 et 2.4 doivent atteindre des tailles très importantes pour une bonne couverture de la langue traitée (voir tableau Tab.5 section 2.4). L'espace mémoire aussi bien que le temps de consultation d'une liste de plusieurs centaines de milliers d'entrées sont trop coûteux pour que l'analyse lexicale puisse se faire d'une façon efficace. C'est pourquoi sous INTEX, comme dans de nombreuses autres applications TALN, les gros dictionnaires sont représentés sous format d'automates finis déterministes et minimaux. Un tel automate reconnaît tous les mots (et seulement ceux) appartenant au dictionnaire textuel d'origine. Les dictionnaires-automates minimaux et déterministes ont deux grands avantages : leurs tailles sont très petites et leurs temps d'accès, i.e. de recherche d'un mot, est linéaire en fonction de la longueur du mot recherché. Le taux de compactage d'un dictionnaire textuel vers un automate est le plus élevé quand le dictionnaire contient beaucoup d'entrées avec les préfixes et les suffixes communs. C'est pourquoi les effets sont les plus spectaculaires pour les dictionnaires DELAF (et non pas les

[21] *<NB>* signifie n'importe quel nombre décrit par l'un des graphes présentés dans le chapitre 7.

[22] Cette notation peut être simplifiée de la façon suivante : (*<sonata>+<fugue>+<symphony>+…*) *in* (*C+D+E+...*) où un lemme entre crochets représente n'importe quelle forme fléchie de ce lemme.

DELACF) et spécialement pour les langues à une flexion riche comme le polonais - voir tableau Tab.6.

	Français	Anglais	Polonais
DELAF textuel	20,8 Mo	-	80 Mo
DELAF compacté	1,2 Mo	1 Mo	1,8 Mo
DELACF textuel	12 Mo	4,9 Mo	-
DELACF compacté	5,6 Mo	1,25 Mo	-

Tab.6 Tailles des dictionnaires textuels et compactés

INTEX utilise la méthode « traditionnelle » de compactage d'un DELAF textuel en un automate fini déterministe. Cette méthode, consiste en deux phases : la construction d'un automate déterministe (d'habitude non minimal), puis sa minimisation. Dans la première phase, la liste textuelle d'entrées du DELAF est convertie en un *trie*, i.e. un automate déterministe acyclique sous forme d'arbre (arbre lexicographique généralisé), où l'état initial est la racine de l'arbre, et les états terminaux sont ses feuilles. L'image Fig.4 présente un *trie* construit pour un dictionnaire contenant six mots anglais : *ant, ants, apse, apses, pat, pats*. Dans la deuxième phase, le *trie* est minimisé selon la méthode reprise de Hopcroft et Ullman (1979). Comme d'autres algorithmes de minimisation d'automates analysés par Watson (1995, pp. 191-214), cette méthode est basée sur la propriété d'équivalence (E) d'états dans un automate. Deux états p et q sont équivalents ($(p,q) \in E$) si et seulement si leurs langages droits sont égaux, le langage droit d'un état étant l'ensemble de tous les suffixes reconnus à partir de cet état jusqu'à l'un des états finals, par exemple sur la figure Fig.4 les ensembles d'états qui ont les mêmes langages droits sont {5, 9, 13}, {4, 8, 12}, et {3, 11}. Si tous les états équivalents sont fusionnés nous obtenons l'automate minimal, qui est présentés sur l'image Fig.5.

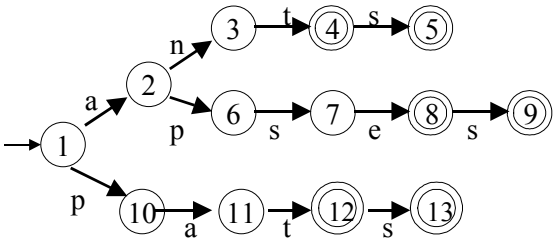


Fig.4. Un trie

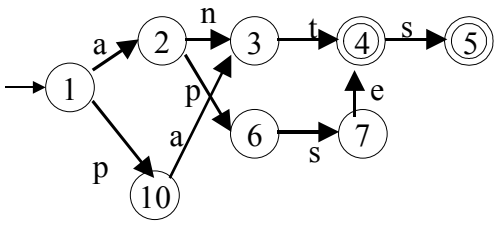


Fig.5. Un automate minimal

Cette méthode de compactage d'un dictionnaire textuel vers un automate en deux phases exige l'espace mémoire suffisante au moins pour contenir le *trie* qui résulte de la première phase. Ceci pose un problème de performance pour de très gros dictionnaires, et donc surtout pour des langues à flexion riche, car le *trie* peut s'avérer plus grand que la taille de mémoire disponible.

Pour remédier à ce problème, des méthodes de construction incrémentale d'automates ont été élaborées. D.Revuz (1991) a proposé un algorithme qui permet, à partir d'un dictionnaire textuel, d'obtenir un automate *pseudo-minimal* de taille très réduite par rapport à un *trie* correspondant. Les mots sont ajoutés progressivement dans l'ordre lexicographique inverse (i.e. la comparaison de lettres se fait de droite à gauche). Pour chaque nouveau mot, il faut trouver le plus long suffixe commun de ce mot avec tous les mots qui sont déjà dans l'automate. Ce suffixe commun est factorisé. L'insertion du préfixe du nouveau mot, qui reste après l'isolation du suffixe commun, se fait de gauche à droite en utilisant les états déjà existants ou en les dédoublant, en fonction du nombre de leurs prédécesseurs (pour un exemple, voir Revuz 1991, pp. 42-45). Après l'obtention de l'automate pseudo-minimal, a lieu la phase de minimisation qui est basée sur la propriété d'équivalence de langages droits décrite plus haut. La complexité de l'algorithme est linéaire en fonction de la taille du lexique pour la première phase, et linéaire en fonction du nombre d'états dans l'automate pseudo-minimal pour la deuxième phase.

Une méthode plus efficace a été élaborée par Daciuk et al. (2000). Elle ne contient qu'une seule phase. Les mots sont rajoutés dans l'ordre lexicographique de telle façon qu'après chaque nouvelle insertion l'automate obtenu reste minimal. Cet algorithme est basé sur l'observation que, lorsque les mots arrivent dans l'ordre lexicographique, les seuls états qui doivent éventuellement être modifiés pour insérer un nouveau mot sont ceux qui se trouvent sur le chemin du mot précédent. Pour un mot à insérer, il faut d'abord trouver le plus long préfixe commun avec les mots qui sont déjà dans l'automate. A partir du dernier état de ce préfixe, il faut parcourir le chemin du mot précédent (de droite à gauche), et vérifier si certains de ses états ne peuvent pas être remplacés par des états équivalents (du point de vu de leurs langages droits) qui existent déjà dans l'automate. Finalement, le suffixe du mot courrant est ajouté à l'automate, et la procédure recommence pour un mot suivant. La figure Fig.6 présente un exemple de construction incrémentale d'un automate minimal pour le même ensemble de mots que ceux de la figure Fig.4.

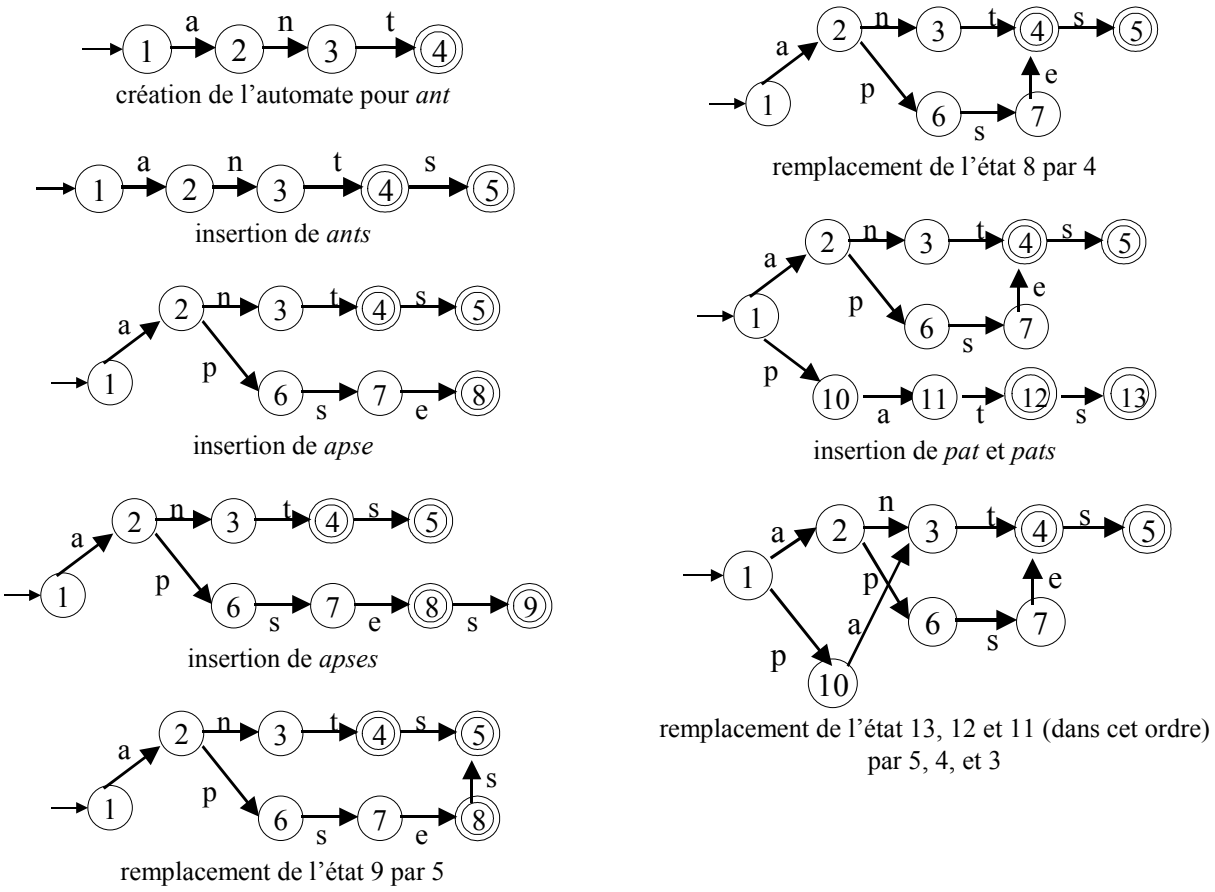


Fig.6. Construction incrémentale d'un automate minimal.

Nous avons brièvement présenté les algorithmes de construction d'automates déterministes minimaux à partir d'une liste textuelle de mots. Un dictionnaire DELAF, tel qu'il est utilisé par INTEX, n'est pas un automate pur, car la reconnaissance d'un mot donne lieu à une production de l'étiquette grammaticale attribuée à ce mot. En ce sens un DELAF compacté ressemble à un transducteur. Le problème de minimisation de transducteurs finis est plus complexe que pour les automates finis. Un transducteur peut être vu comme un automate si l'on considère que son alphabet est constitué de paires (*e,s*), où *e* est un symbole d'entrée et *s* est une séquence de symboles de sortie. Dans ce cas les algorithmes de minimisation d'automates s'appliquent aussi aux transducteurs. Mais dans un tel automate-transducteur minimal la recherche peut ne plus être déterministe. Un transducteur déterministe du point de vue de l'alphabet d'entrée s'appelle un transducteur séquentiel. Si, dans un transducteur séquentiel, nous admettons la possibilité de produire une ou plus séquences de symboles de sortie après chaque état terminal, nous obtenons ainsi un transducteur sous-séquentiel. Les problèmes liés à la minimisation des transducteurs séquentiels et sous-séquentiels ont été décrits par Mohri (1997).

Dans un DELAF compacté sous INTEX aucun symbole de sortie n'est attaché aux transitions. La production de l'étiquette grammaticale de ne se fait qu'après avoir atteint l'état final. C'est pourquoi la minimisation peut être effectuée par un algorithme prévu pour des automates finis proprement dits, avec la seule différence que les états terminaux sont distingués par les différentes productions qui leur sont attribuées.

2.7 Couverture

Les systèmes du traitement automatique du langage naturel tiennent rarement compte du fait que les mots composés devraient être reconnus en tant qu'unités atomiques. Seules les études et les applications informatiques du LADL s'occupent de la reconnaissance des composés à grande échelle. Elles ont démontré (Senellart 1996) que près de 30% de formes simples contenues dans des textes du langage général (tel que celui du journal « Le Monde ») font partie de mots et expressions composées, et ne doivent donc pas être traitées séparément.

La méthode d'analyse choisie dans INTEX est celle de ne reconnaître que les mots composés connus a priori, i.e. contenus dans les dictionnaires. Les résultats dépendent donc entièrement de la qualité et de la complétude des dictionnaires dont nous disposons. Remarquons que les résultats du recensement des mots composés sont les plus gratifiants au début : plus le DELAC est complet plus il est difficile d'améliorer la couverture des textes en continuant ce recensement. Ceci est dû au fait que les mots les plus fréquents sont très peu nombreux et les mots apparaissant le plus rarement sont très nombreux.

Nous avons illustré cette hypothèse dans l'annexe D. Nous y présentons les résultats de l'analyse lexicale de 99 mégaoctets de texte (plus de 15 millions de formes simples) du journal *Herald Tribune* (l'année 1994). A l'aide du dictionnaire DELAC de l'anglais général décrit dans le chapitre 5, le système INTEX a trouvé plus de 524 000 occurrences de mots composés qui correspondent à environ 26 000 formes composées différentes. Nous avons analysé les mots composés les plus fréquents et les moins fréquents. Il n'y a que 55 formes composées différentes (0,2%) avec la fréquence supérieure à 1000, mais elles représentent 25% de toutes les occurrences des composés dans le texte. D'autre part les formes composées peu fréquentes sont très nombreuses : il y a plus de 22 000 formes avec la fréquence inférieure à 20, et elles correspondent à 18% de toutes les occurrences.

Cette propriété a fait objet d'une discussion sur la loi de Zipf qui dit que la fréquence f_i d'un événement P_i est égale à K/i^a, où *i* est le rang de cet événement fixé par la fréquence décroissante (i.e. l'événement le plus fréquent a le rang 1 et l'événement le moins fréquent a le rang *n* égal aux nombre total d 'événements). Senellart (1999) démontre que la loi de Zipf reste valable pour des textes pas trop gros et pour les mots des rangs intermédiaires. Les mots de fréquences très faibles sont sous-représentés par rapport à la loi attendue de Zipf ce qui exclut leur acquisition par des méthodes statistiques. Ils peuvent donc être acquis uniquement suite au recensement manuel.

2.8 Mots composés ambigus et non ambigus

Disposer de dictionnaires exhaustifs de mots composés ne garantit pas la bonne identification des composés dans un texte. Toute suite contiguë de formes simples apparaissant en tant que mot composé dans un dictionnaire n'est pas forcément une occurrence d'un mot composé dans un texte. En effet, la plupart des composés sont ambigus avant que l'analyse syntaxique et sémantique du texte concerné soit effectuée. Cette ambiguïté peut être due à plusieurs phénomènes.

1) Nous ne connaissons pas les frontières entre les syntagmes de la phrase. Ainsi, les mots qui se trouvent à la frontière de deux syntagmes peuvent entrer dans un faux composé, comme dans les exemples suivants :

[68] (ang.) *It was not easy to break dance performances in front of the whistling audience.*

(*break dance* = un type de danse moderne ; *Il n'était pas facile d' interrompre des spectacles de danse devant le publique qui sifflait.*)

[69] (ang.) *After all the preparations have been done we finally left.*

(*after all* = de toute façon ; *Une fois toutes les préparations accomplies, nous sommes finalement partis.*)

[70] (pol.) *Z uwagi na gospodarstwo domowe prace zostały odłożone na później.*

(*gospodarstwo domowe* = foyer ; *Pour les besoins de travaux de ferme les travaux ménagers ont été suspendus.*)

2) Les séquences potentiellement composées peuvent se chevaucher. Ceci peut avoir lieu à l'intérieur du même syntagme ou à la frontière entre deux syntagmes, comme dans

[71] (fr.) *Il m'a donné une*$[_1$*pomme de* $[_2$*terre*$]_1$ *cuite*$]_2$.

[72] (ang.) *That was when I met the* $[_1$*black* $[_2$*arts*$]_1$ *student*$]_2$ *for the first time.*

(*black arts* = sciences occultes ; *arts student* = étudiant en lettres)

[73] (pol.) *Sprzątając* $[_1$*dom* $[_2$*książki*$]_1$ *kucharskie*$]_2$ *ustawiła na półce w kuchni.*

(*dom książki* = librairie; *książka kucharska* = livre de cuisine;
En rangeant la maison, elle a mis les livres de cuisine sur l'étagère.)

3) Une séquence potentiellement composée apparaît en tant que syntagme libre ou partiellement libre. Ceci arrive surtout avec les composés métaphoriques, comme

[74] (fr.) *La table ronde allait très bien dans notre petit salon.*

[75] (ang.) *You shouldn't have touched this. Look at your green fingers!*

(*green fingers* = la main verte ; *Tu n'aurais pas dû toucher ça, regarde tes doits verts !*)

[76] (pol.) *Zimne nogi i blada twarz twarz jej córki wzbudziły podejrzenie o problemy z krążeniem.*

(*zimne nogi* = de la gelée à la viande; *blada twarz* = visage pâle (lang. des Indiens); *Les jambes froides et le visage pâle de sa fille ont fait supposer des problèmes cardiaques*)

[77] (pol.) *Zbyt ciężki karabin maszynowy uniemożliwiał mu szybkie przemieszczanie się.*

(*karabin maszynowy* = mitrailleuse; *ciężki karabin maszynowy* = mitrailleuse lourde; *La mitrailleuse trop lourde rendait les déplacements rapides impossibles*)

4) Certains mots composés ont plusieurs sens différents. Ces mots apparaîssent dans le dictionnaire sous plusieurs entrées, à chaque fois avec une étiquette différente. Par exemple,

[78] (pol.) *kura domowa*

peut signifier soit une *mère poule* et donc recevoir l'étiquette avec la marque distributionnelle *+Hum* (humain), soit une espèce de poule qui correspond à la marque *+Anl* (animal).

Un mot composé est non ambigu s'il est composé quel que soit son contexte d'occurrence. Le nombre de composés non ambigus est petit par rapport au nombre de tous les composés. On y compte surtout les séquences qui contiennent des constituants n'ayant pas de statut indépendant. Nous en parlons, pour l'anglais, dans la section 3.6 (exemples [190]) et la section 5.4.1 (exemples [295]). En voici des exemples pour l'anglais, le français et le polonais :

[79] (ang.) *good-lookingness, walkie-talkie*

[80] (fr.) *aujourd'hui, co-incidence, prud'homme*

[81] (pol.) *czapka-niewidka* (une casquette qui rend invisible)*, rendez-vous, joint venture*

Les deux derniers composés sont non ambigus en polonais alors qu'ils ne le sont pas en français et en anglais respectivement, car ils contiennent des mots étrangers qui n'existent pas en polonais.

2.9 Algorithmes de l'analyse lexicale des mots composés

Les algorithmes de l'analyse lexicale que nous utilisons, sont décrits en détail par M. Silberztein (1997 et 1999-2000). Nous en présentons ici seul le schéma général[23] sur la figure Fig.7 et nous décrivons plus précisément celles des fonctions qui concernent les mots composés.

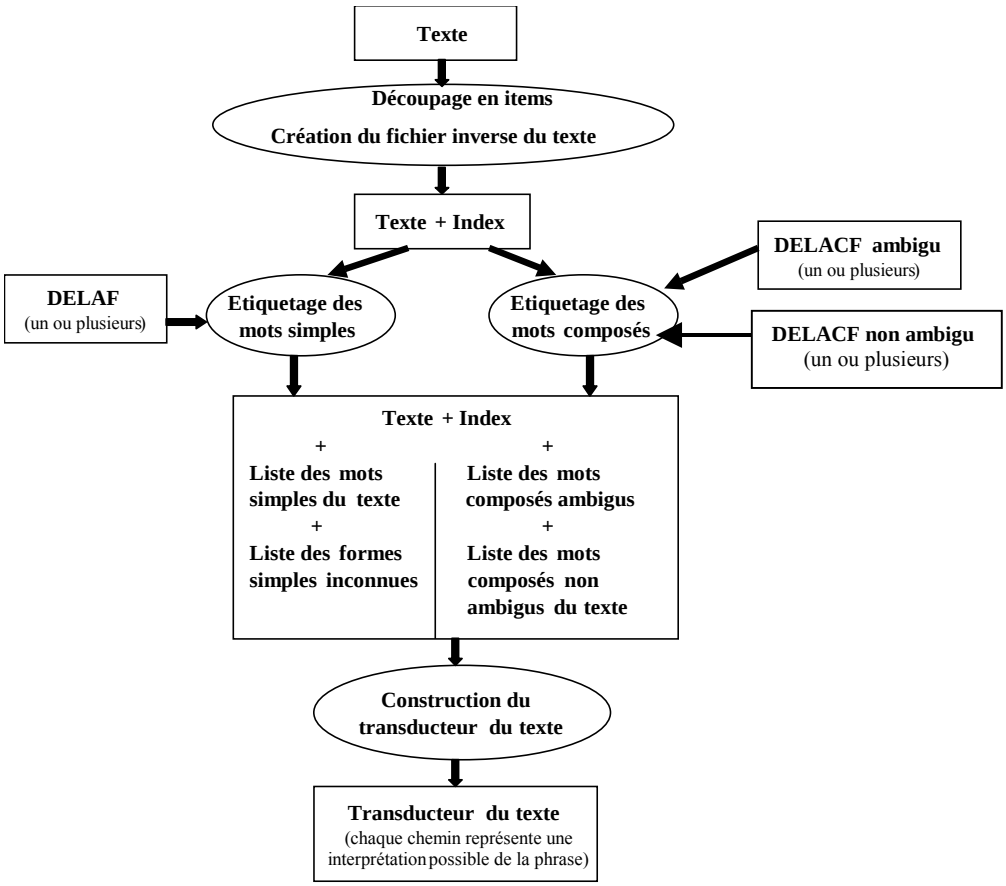


Fig.7. Schéma de fonctionnement de l'analyse lexicale

[23] Nous présentons le schéma de fonctionnement de l'ancienne version (3.4) d'INTEX car c'est celle-là qui a été prise en compte lors de l'élaboration de l'extracteur terminologique décrit dans le chapitre 8. Dans la version récente (4.21) le schéma de fonctionnement diffère de celui de la version 3.4 de deux points de vue :
- les composés non ambigus sont rangés dans un dictionnaire spécial utilisé pour le prétraitement du texte,
- les dictionnaires DELAF/DELACF sont appliqués au texte selon 3 (et non pas 2) niveaux de priorité.

Sur le schéma de fonctionnement, les éléments ovales représentent les différentes fonctions de l'analyse lexicale, tandis que les éléments rectangulaires correspondent aux entrées et sorties de ces fonctions. Tout le processus est exemplifié dans l'annexe A.

Deux phases préliminaires du traitement sont celles de l'identification des items du texte et de la constitution de son fichier inverse (index). Les items du texte sont les suites contiguës soit de lettres, soit de séparateurs (voir la section 2.2.1). Le fichier inverse du texte donne pour chaque item la liste de toutes ses occurrences. La création de ce fichier nécessite un temps supplémentaire de traitement mais elle accélère, surtout pour des corpus volumineux, les autres étapes de l'analyse.

L'étiquetage s'occupe ensuite de la reconnaissance des mots simples et composés du texte grâce à la consultation d'un ou plusieurs dictionnaires DELAF/DELACF. Chacun des dictionnaires DELAF peut avoir l'un des deux niveaux de priorité : 1 ou 0. Le mot courant du texte est d'abord recherché dans tous les dictionnaires à priorité 1. S'il y figure au moins une fois il reçoit les étiquettes grammaticales provenant de ces dictionnaires-là, et sa recherche est terminée. Sinon, les dictionnaires à priorité 0 sont consultés. Le mécanisme des priorités permet, en fonction de l'application, d'éliminer des étiquettes peu probables pour certains mots.

Les dictionnaires DELACF sont divisés en ceux contenant les mots composés ambigus et ceux avec des composés non ambigus. En résultat de l'étiquetage, nous obtenons quatre listes d'étiquettes – la liste des mots simples du texte, la liste des mots composés non ambigus, la liste des composés ambigus, et la liste des mots simples inconnus.

Tous ces éléments-là servent à construire les transducteurs finis du texte, un transducteur par phrase. Chaque noeud d'un de ces transducteurs représente alors un mot, simple ou composé, avec son étiquette grammaticale (le lemme et les traits morphologiques), et tous les chemins qui mènent du noeud initial au noeud final correspondent à toutes les interprétations possibles de la phrase.

2.10 Représentation des composés par transducteurs

Si l'on ne prenait pas en compte les mots composés, le transducteur d'une phrase, à la fin de l'étiquetage, se réduirait à un alignement de toutes les interprétations de tous les mots simples de cette phrase[24]. L'existence des composés complique cette image, car certaines séquences sont regroupées et donc tous les chemins n'ont pas forcément la même longueur - voir le transducteur dans l'annexe A.

A cause du fait que la plupart des composés sont ambigus, comme ceci a été démontré dans la section 2.8, la complexité d'un transducteur du texte peut être considérable. Examinons le composé de l'exemple suivant :

[82] (pol.) *Objaśniono nam jak posługiwać się LKM-em (lekkim karabinem maszynowym).*
(*On nous a expliqué comment se servir d'une mitrailleuse légère.*)

Si l'on veut être sûr que l'analyse lexicale ne passe sous silence aucune interprétation correcte, toutes les combinaisons des différentes étiquettes doivent être représentées. Ainsi,

[24] Ceci est vrai à la fin de la phase de l'étiquetage, mais pas après l'étape suivante qui est la levée d'ambiguïtés par des grammaires locales, dont la description dépasse le cadre de ce mémoire.

nous obtenons un transducteur dont la partie représentant le composé souligné se trouve sur la figure Fig.8.

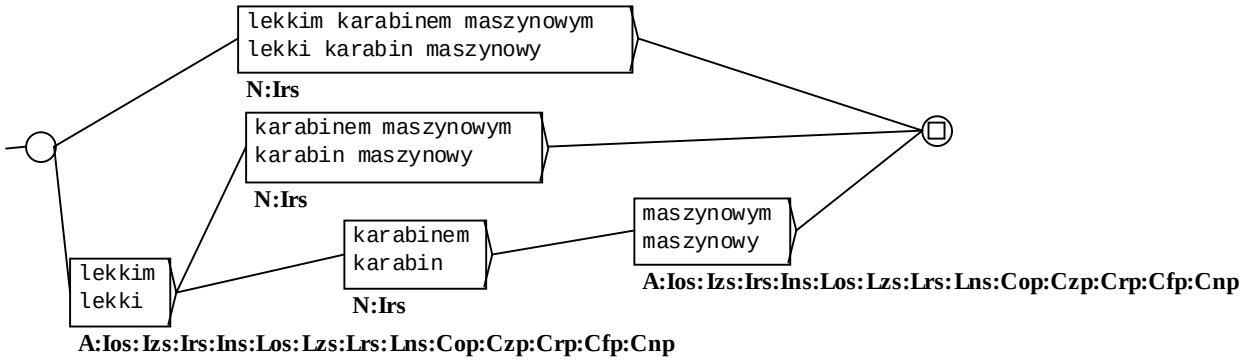


Fig.8. Transducteur du mot composé *lekkim karabinem maszynowym*

Les étiquettes avec les lemmes et les catégories communes ont été mises en facteur. Grâce à cela ce sous-graphe ne contient que 3 chemins possibles mais ils correspondent à 1 + 13 + 13*13 = 183 interprétations possibles (qui se multiplient encore par le nombre d'ambiguïtés dans le reste de la phrase). Cependant une seule interprétation est valable, celle qui considère la séquence *lekkim karabinem maszynowym* comme un mot composé non ambigu et admet donc seul le chemin supérieur du graphe. Nous pouvons imposer cette interprétation en incluant le mot composé en question dans un dictionnaire des mots composés non ambigus (i.e. celui qui a la priorité sur les autres dictionnaires), mais ceci nous ferait courir le risque de mauvaise analyse dans des phrases où *lekki karabin maszynowy* apparaît en tant que syntagme libre (voir exemple [77]). Nous choisissons donc dans un premier temps de garder tous les chemins du graphe Fig.8 car la levée d'ambiguïté fiable ne peut pas être faite à ce stade de traitement.

Pour certaines applications, nous pouvons néanmoins utiliser avec beaucoup de succès la stratégie d'analyser la plupart des composés recensés comme non ambigus (à certaines exceptions près, comme par exemple les adverbes courts du français : *en fait, du tout*, etc.). Les cas comme dans les exemples [68]-[77] seront alors mal représentés mais ils sont peu fréquents (en général les rédacteurs évitent de tels jeux de mots).

2.11 Conclusion

En conclusion de la discussion sur les notions de base admises dans INTEX, nous pouvons dire que :

1) Comme le constate G. Gross (1988, p.59), la distinction entre les mots simples et les mots composés est arbitraire, et ne reflète en aucun cas la complexité des concepts que ces mots représentent.
2) Seuls les dictionnaires permettent une (des) division(s) du texte en unités de traitement. C'est par la présence d'une séquence dans un dictionnaire (et non pas par calcul) que l'on peut analyser cette séquence comme unité atomique dans un texte.
3) Les définitions du mot simple et du mot composé admise dans INTEX sont des notions adaptées aux exigences que pose un traitement automatique.

Nous avons présenté les outils de l'analyse lexicale des mots composés. A plusieurs stades de cette analyse INTEX utilise des automates et des transducteurs finis (les codes flexionnels, le compactage des dictionnaires, la description d'expressions figées, la représentation des interprétations possibles d'une phrase) qui s'avèrent être des outils naturels et efficaces de description des différents phénomènes linguistiques.

Entre autres, la représentation par transducteurs finis du texte étiqueté permet de :

- tenir compte des mots composés du texte et de leurs chevauchement éventuels,
- exprimer les ambiguïtés irrésolubles au stade de l'analyse lexicale,
- lever partiellement les ambiguïtés en gardant la cohérence des possibilités restantes (contrairement à l'étiquetage linéaire qui attribue à chaque forme du texte une seule étiquette grammaticale).

Première partie

Mots composés – problèmes linguistiques et méthodes de recensement

Chapitre 3 Propriétés linguistiques des noms composés

3.1 Introduction

L'étude à grande échelle des noms composés soulève un certain nombre de problèmes qu'il faut résoudre lors du développement des méthodes automatiques de traitement. Dans la section 2.2.3 nous en avons mentionné trois qui sont liés aux constituants caractéristiques et à la flexion, et nous les analysons dans les sections 3.2-3.5. D'autres exemples de phénomènes communs pour les noms composés dans des langues telles que le français, l'anglais et le polonais sont traités dans les sections 3.6-3.9.

3.2 Inexistence et irrégularités des constituants caractéristiques

Le cas le plus simple de l'inexistence de constituant caractéristique concerne les noms composés qui n'ont pas la structure d'un groupe nominal et ne peuvent donc pas avoir de tête qui indique les traits morphologiques de la séquence. Les exemples sont nombreux. Nous avons déjà mentionné qu'en français toute la classe *Verbe Nom* est de ce type, comme dans :

[83] *un porte-serviettes*

[84] *un perce-neige*

La forme du masculin singulier de ces composés ne provient ni des verbes (qui n'ont pas de genre) ni des noms qui sont au pluriel et/ou au féminin.

Des exemples similaires sont ceux des composés construits à partir de pronoms ou de prépositions, comme :

[85] *un rendez-vous*

[86] *un entre-deux*

En anglais, des classes de structures atypiques pour les groupes nominaux sont aussi celles avec des verbes, comme :

[87] *a take-in* (cheat, deception)

[88] *a forget-me-not* (myosotis)

[89] *a drive-yourself* (véhicule loué et conduit par le locataire)

ainsi que d'autres constructions, comme

[90] *a four-in-hand* (attelage à quatre chevaux)

[91] *a good-for-nothing* (bon à rien)

En polonais, les composés déverbaux, comme

[92] *Bog zaplać* (un Dieu-vous-le-rende)

sont peu nombreux, mais d'autres constructions diverses sans composant caractéristique existent. Dans

[93] *sam na sam* *(un tête-à-tête)*

qui est du type *AdjPrep*Adj, non seulement le deuxième adjectif n'a pas le cas exigé par la préposition *na*, mais en plus tout le composé est au neutre, alors que les deux adjectifs se trouvent au masculin.

Il existe d'autres exemples, moins nombreux, qui ont la structure d'un groupe nominal mais dont les traits morphologiques ne se déduisent pas de ceux de leurs composants en position de tête. En français ceci est le cas dans des mots comme :

[94] Nom Adj : *un peau-rouge*

[95] Adj Nom : *un rouge-gorge*

[96] Nom à Nom : *un tête-à-tête*

[97] Prèp Nom : *un après-guerre*

[98] Nom Adj : *un terre-plein*

qui sont au masculin malgré leur constituant nominal en position de tête (souligné) qui est au féminin.

D'autres cas, comme

[99] *une deux-chevaux*

[100] *un deux-pièces*

sont différents de leur nom en position de tête, aussi bien en genre qu'en nombre.

An anglais, les mots du type *AdjNom :*

[101] *a lesser yellowlegs*

[102] *a cross-roads* (un carrefour)

sont au singulier. Pourtant, les constituants en position de tête sont au pluriel.

Dans certains noms composés qui ont une structure correcte de groupe nominal, nous remarquons que la tête existe mais n'est pas celle qui est prévue pour cette structure. Ceci arrive en français dans :

[103] *une grand-mère*

[104] *une franc-maçonne*

[105] *des saint-émillions*

où l'adjectif qui devrait normalement être caractéristique ne s'accorde pas avec le nom. Pour les exemples [103] et [104] nous pouvons résoudre ce problème en considérant que *franc* et *grand* sont ici des adjectifs invariables en genre (comme *standard*) différents de *grand.N32* et *franc.N47*. Une autre possibilité est de dire que dans les trois exemples seul le nom est le constituant caractéristique (ce qui ne réduit pas pour autant l'irrégularité de flexion, dont nous parlons dans la section suivante).

En polonais, dans le mot

[106] *herod baba* (litt. une femme-Herod = un virago)

du type *NomNom* le deuxième nom se fléchi à la place du premier et reprend donc son rôle de tête.

3.3 Inexistence et irrégularités des formes fléchies

L'existence et la construction de toutes les formes fléchies attendues pour les noms composés sont pour la plupart systématiques, c'est-à-dire conformes au schéma syntaxique de la séquence. Nous parlons dans ce cas de composés à flexion régulière.

Définition 4 :
Un nom composé (et plus généralement un mot composé) est à **flexion régulière** si les deux conditions suivantes sont remplies :
1) toutes les formes fléchies de ce composé, prévues par ses constituants caractéristiques, existent,
2) seuls les constituants caractéristiques varient au cours de la flexion.

Rappelons que si la deuxième condition est remplie, les traits syntaxiques de chaque constituant caractéristique sont, par définition, les mêmes que ceux du mot composé, et ceci dans chaque forme fléchie de ce composé. Tel est le cas pour les noms comme :

[107] (ang.) *court painter, point of view* (point de vue)*, black hole* (trou noir)*, binary coded decimal*

[108] (fr.) *cousin germain, belle-mère, coup de fil, machine à laver*

[109] (pol.) *brat cioteczny* (cousin germain)*, słomiany wdowiec* (homme dont la femme est partie en voyage)*, dom książki* (librairie)*, fotel na biegunach* (fauteuil à bascule)

où toutes les formes fléchies prévues pour les groupes nominaux existent et se construisent par la mise à la forme correspondante de la tête du groupe :

- le pluriel en anglais, français et polonais :

  [110] (ang.) *court painters, points of view, black holes, binary coded decimals*

  [111] (fr.) *cousins germains, belles-mères, coups de fil, machines à laver*

  [112] (pol.) *bracia cioteczni, słomiani wdowcy, domy książki, fotele na biegunach*

- le féminin en français et polonais, mais seulement là où le nom de tête a une flexion en genre :

  [113] (fr.) *cousine germaine*

  [114] (pol.) *słomiana wdowa*

- les 7 cas en polonais :

  [115] (pol.) *brat cioteczny, brata ciotecznego, bratu ciotecznemu, brata ciotecznego, bratem ciotecznym, bracie ciotecznym, bracie cioteczny; słomiany wdowiec, słomianego wdowca,* etc.*; dom książki, domu książki,* etc.*; fotel na biegunach, fotelu na biegunach,* etc.

Toutes les combinaisons des différents traits morphologiques sont également possibles ici, par exemple :

[116] (fr.) *cousines germaines* - pluriel féminin

[117] (pol.) *słomianej wdowy, słomianym wdowom*, etc. - singulier féminin génitif, singulier féminin datif, etc.

soit au total 2 formes en anglais, 2 ou 4 formes en français, et 14 ou 28 formes en polonais[25].

Définition 5 :
Un nom (mot) composé est à **flexion irrégulière**, si au moins un des deux cas suivants se présente :
1) l'une au moins des formes fléchies prévues par la tête de ce composé est interdite,
2) dans au moins l'une des formes fléchies il existe un constituant non caractéristique qui subit une flexion.

Comme illustration de la première condition en anglais, prenons

[118] *bits and pieces* (des morceaux)

qui, en tant que composé, n'admet pas le singulier, alors que *bit and piece* est une construction syntaxiquement correcte. La même contrainte existe en polonais pour

[119] *zimne nogi* (litt. "des jambes froides" = un plat de viande en gelée)

qui, mis au singulier, perd son sens idiomatique.

Nous pouvons nous interroger sur l'existence du pluriel pour des nombreux noms composés. Il n'est pas clair que cela a du sens de parler de plusieurs :

[120] (fr.) *? Armées Rouges*

[121] (fr.) *? gaz naturels*

[122] *(ang.) ? grey matters* ("les matières grises")

[123] *(ang.) ? senses of humor* (sens d'humour)

[124] *(pol.) ? telegraficzne skróty* ("les styles télégraphiques")

alors que l'existence du pluriel des noms *sens, raison*, *matter*, *skrót* n'est pas problématique. Nous allons généralement admettre un pluriel partout où il n'est pas formellement interdit, et où il ne changerait pas la signification du composé.

Selon la deuxième condition de la définition 5, certaines formes fléchies de composés peuvent se construire de façon irrégulière. Beaucoup de mots sans constituant caractéristique sont irréguliers dans ce sens, comme les *VerbeNom* en français, qui possèdent, pour la plupart, le pluriel construit par la mise au pluriel du nom (qui n'est cependant pas un composant caractéristique) :

[125] *un perce-neige, des perce-neiges*

En français et en polonais, une partie importante de la classe de la structure *Nom Nom* est aussi irrégulière, car les deux noms se fléchissent, alors que seul le premier est caractéristique :

[126] *bateaux-mouches,* etc.

[127] *człowieka kota, ludzi kotów,* etc. (homme-chat)

[25] La déclinaison complète en polonais, par exemple celle des adjectifs, comprend 70 formes (7 cas, 2 nombres, 5 genres: le masculin humain, le masculin animé, le masculin inanimé, le féminin, et le neutre). Néanmoins, la flexion en genre des noms est limitée à la mise au féminin de certains noms masculins humains.

D'autres exemples moins nombreux existent dans des classes à prépositions - les *Nom de Nom* en français et les *Nom of Nom* en anglais - ainsi que dans les *Nom Nom*$_{(génitif)}$ en polonais. Les hésitations sur la mise au pluriel du deuxième nom (qui n'est pas un constituant caractéristique) dans ces mots viennent souvent de considérations sémantiques. On se demande si la notion désignée par un composé donné au pluriel concerne un seul ou plusieurs compléments. Par exemple, pour

[128] (ang.) *head of government*

[129] (pol.) *akt urodzenia*

s'agit-il de plusieurs chefs (*heads*) du même gouvernement (*government*) ou bien des gouvernements différents ? Demande-t-on des actes (*akty*) de naissance (*urodzenia*) d'une seule ou plusieurs personnes ? Les deux situations sont possibles, d'où existence des deux formes du pluriel, dont la deuxième est irrégulière :

[130] heads of government, heads of governments

[131] akty urodzenia, akty urodzeń

En français ces variantes sont souvent de nature orthographique car la prononciation ne change pas d'une forme à l'autre, par exemple (selon Grevisse 1988, p. 852) :

[132] une toile d'araignée - des toiles d'araignée, des toiles d'araignées

[133] un nom de lieu - des noms de lieu, des noms de lieux

Des variantes orthographiques peuvent exister d'ailleurs pour les deux nombres, comme

[134] simulateur de vol, simulateur de vols, simulateurs de vol, simulateurs de vols

La deuxième et la quatrième forme sont alors irrégulières.

Dans les deux exemples déjà mentionnés en français - *grand-mère* et *franc-maçon* - l'irrégularité de la flexion est due aux adjectifs non caractéristiques qui s'accordent éventuellement en nombre alors qu'ils restent invariables en genre :

[135] grand-mère, grand-mères | grands-mères

[136] franc-maçon, franc-maçonne, francs-maçons, francs-maçonnes

Dans la section suivante nous présentons un aperçu des autres problèmes liés à la formation du pluriel irrégulier en anglais.

Remarquons finalement que :

1) ni la non-existence d'une forme particulière pour un composé,

2) ni l'existence de plusieurs variantes de la même forme

ne signifient automatiquement l'irrégularité de la flexion car il peut s'agir des contraintes propres à l'un des composants simples. Par exemple, l'impossibilité de mettre les mots comme

[137] coup de fil

[138] funérailles nationales

respectivement au féminin et au singulier provient du fait que les noms de tête *coup* et *funérailles* ne possèdent pas de féminin ou de singulier. D'une façon similaire, les deux variantes du pluriel instrumental du mot polonais :

[139] związane ręce - związanymi rękami, związanymi rękoma
(impossibilité d'agir, litt. "les mains liées")

correspondent aux deux façons équivalentes de fléchir le nom de tête *ręka*.

3.4 Irrégularités de la mise au pluriel en anglais

Pour de nombreux exemples de noms composés en anglais il y a des variations du pluriel. Parfois, deux ou trois formes équivalentes sont utilisées en même temps. La seule analyse détaillée que nous avons trouvée de la formation du pluriel des composés en anglais date d'il y a presque 35 ans. Jespersen (1965, II, pp. 20-35) y présente différentes exceptions aux règles générales de la construction du pluriel, avec des exemples d'occurrences dans des textes littéraires, dont certains du XIX siècle. Les ouvrages plus récents, comme ceux de Fowler (1983, pp. 456-7), Webster (1976, p. 25a) et Quirk et al. (1972, pp. 174-5), traitent ce problème de façon marginale.

3.4.1 Nom Adjectif

Le premier cas d'irrégularité est celui de la structure *Nom Adjectif* qui est atypique en anglais. Les composés de ce type admettent généralement deux formes du pluriel : la première avec le nom au pluriel, et la deuxième avec le nom invariable et l'adjectif fléchi comme s'il était un nom. Cette dernière forme est d'habitude considérée comme plus moderne.

[140] attorneys general = attorney generals (ministres de la justice des USA)

[141] battles royal = battle royals (mêlées générales)

[142] courts martial = court martials (cours martiales)

Dans certains cas, seule la forme plus traditionnelle est retenue, comme

[143] notaries public (notaire)

[144] heirs presumptive (héritiers présomptifs)

[145] accounts current (comptes courants)

et parfois l'on retrouve trois variantes :

[146] letters patent = letter patents = letters patents (lettres patentes)

3.4.2 Nom Nom

Il s'agit ici d'appositions[26] qui se mettent au pluriel de façon irrégulière, par la mise au pluriel du nom terminal, et aussi du premier nom en *-man* ou *-woman* :

[147] man-servant - men-servants (valets)

[148] woman writer - women writers

[149] gentleman farmer - gentlemen farmers (fermier amateur)

Ces deux marques semblent exceptionnelles, car par exemple *lady writer* et *boy friend* prennent seulement le *-s* au deuxième nom :

[26] Nous allons plutôt éviter ce terme qui semble mal défini, surtout en anglais où il est difficilement distinguable de la composition germanique standard (une série de noms dont les initiaux sont attributs du terminal).

[150] lady writers

[151] boy friends (petits amis)

Pour des titres du type *lord justice* et *lieutenant-general,* la formation du pluriel semble libre. Nous donnons une liste des variantes admises dans les ouvrages mentionnés plus haut. Les cas irréguliers sont ceux des mots suivants :

[152] lord justices, lords justices (juges au Court d'Appel)

[153] Lord Chancellors, Lords Chancellor (ministres de la justice)

[154] Lord(-) Lieutenants, Lords Lieutenant, Lords Lieutenants, lords-lieutenants
(représentants de la Couronne dans un comté de Grande-Bretagne)

[155] Knights Hospitallers
(Chevaliers de l'Hôpital de St Jean, l'ordre militaire et religieux fondé au 11^{e} siecle)

Les exemples des cas réguliers du même type sont les suivants :

[156] Lord Mayors (maires des grandes villes de Grande-Bretagne)

[157] major(-)generals (généraux de division)

[158] Lieutenant-Colonels, lieutenant colonels (sous-lieutenants)

3.4.3 Composés déverbaux

Les noms composés provenant de verbes avec une particule prennent presque toujours le *-s* à la fin :

[159] hand-outs (documents), *take-offs* (décollages), *break-ins* (cambriolages), *take-aways* (repas à emporter), *hold-ups, stand-bys* (remplaçants)

mais Jespersen (1965) cite quelques exceptions à cette règle comme :

[160] answers-back (réponse)

Ce cas pourrait entrer plutôt dans la classe *Nom Particule_Adverbiale*, où le premier nom prend régulièrement la marque du pluriel :

[161] *passers-by* (passants), *hangers-on* (parasites)

Les composés provenant de verbes accompagnés de leurs objets et d'autres constructions verbales plus complexes prennent toujours le *-s* à la fin :

[162] cure-alls (panacées), *find-faults* (personnes qui cherchent des fautes), *tell-tales* (personnes qui révèlent des secrets), *forget-me-nots* (myosotis), *pick-me-ups* (remontants), *ne'er-do-wells* (des bons à rien), *has-beens* (hommes finis), *stay-at-homes* (casaniers), *johny-come-latelies* (nouveaux venus)

3.4.4 Nom Préposition Nom

Dans cette construction le premier nom étant caractéristique, c'est lui qui se met généralement au pluriel. Il y a quelques exceptions :

[163] jack-in-the-boxes (diables à ressort), *will-o'-the-wisps* (feus follets), *dog-in-the-mangers* (empêcheurs de tourner en rond)

3.4.5 Phrases nominales avec une conjonctions

Celles-ci sont irrégulières dans le cas de noms de boisson :

[164] whisky-and-sodas, whiskies and sodas

[165] gin-and-tonics

[166] gins-and-French

3.4.6 Emprunts

Les composés étrangers, comme les emprunts simples, peuvent garder leur pluriel d'origine, le former à l'anglaise, ou bien rester invariables :

[167] nouveau riche > nouveaux riches (selon OALDCE 1989)

[168] objet d'art > objets d'art (selon OALDCE 1989)

[169] opera buffa > operas buffa, operе buffe (selon NSOED 1996)

[170] beau ideal > beaus ideal, beau ideals (selon Webster 1976)

3.5 Irrégularités des numéraux cardinaux polonais

Un cas particulier des groupes nominaux, donc aussi des noms composés, en polonais est celui où la tête contient un **numéral cardinal**. Les numéraux imposent des contraintes très spécifiques à d'autres éléments de la phrase.

1) Pour **0** - le syntagme nominal qui suit se met au génitif pluriel, pour tout cas et tout genre, tandis que *zero* reste toujours au neutre singulier, mais se décline :

[171] Mamy zero szans. (*zero* : accusatif neutre singulier; *szans* : génitif féminin pluriel)
(*Nous n'avons aucune chance (*litt. *nous avons zéro de chances*).)

[172] *Wyruszył na poszukiwania z zerem informacji.* (*zerem* : instrumental neutre singulier; *informacji* : génitif féminin pluriel)
(*Il est parti avec zéro information*)

Quand le groupe nominal commençant par *zero* se trouve en position du sujet, le verbe s'accorde avec *zero* donc il se met au singulier neutre.

[173] W koszyku zostaje zero gruszek. (*zero* : nominatif neutre singulier; *gruszek* : génitif féminin pluriel, *zostaje* : neutre 3e personne du singulier)
(*Il reste zéro pomme dans le panier.* (litt. *Zéro de pomme reste dans le panier.*))

2) Pour **1** - le groupe nominal entier se met au singulier et s'accorde avec le nom principal pour tous les cas :

[174] Jeden dodatkowy uczestnik nie sprawia różnicy. (*jeden dodatkowy uczestnik* : nominatif masculin humain singulier)
(*Un participant supplémentaire ne fait pas de différence*)

[175] Zabrakło mi jednego punktu. (*jednego punktu* : génitif masculin inanimé singulier)
(*J'ai manqué d'un point.*)

3) Pour **2**, **3** et **4** - le groupe nominal entier se met au pluriel et s'accorde avec le nom principal pour tous les cas :

[176] Kupiliśmy dwa składane krzesła. (*dwa składane krzesła* : accusatif neutre pluriel)
(*Nous avons acheté deux chaîses pliantes.*)

[177] Opowieść o trzech psach przestraszyła moją córkę. (*trzech psach* : locatif masculin animé pluriel)
(*L'histoire sur les trois chiens a fait peur à ma fille.*)

[178] Wszystkim czterem chłopcom należala się nagroda. (w*szystkim czterem chłopcom* : datif féminin pluriel)
(*Tous les quatre garçons méritaient une récompense*)

De plus, au masculin humain un groupe nominal en position de sujet peut être soit au nominatif

[179] Dwaj bracia rozstali się. (*dwaj bracia :* nominatif masculin humain pluriel*; rozstali się :* 3 personne du pluriel masculin)
(*Les deux frères se sont quittés.*)

soit au génitif[27], le verbe apparaît alors au singulier neutre :

[180] Dwóch braci rozstało się. (*dwóch braci :* génitif masculin humain pluriel*; rozstało się :* singulier neutre)

4) Pour les **autres** - le groupe nominal entier s'accorde avec le nom principal pour tous les cas sauf au nominatif et au vocatif (qui sont invariables au pluriel), et sauf à l'accusatif pour tous les genres sauf le masculin humain :

[181] Zakład współpracuje z dwudziestoma fachowcami. (*dwudziestoma fachowcami* : instrumental masculin humain pluriel)
(*la société travaille avec 20 spécialistes*)

[182] Baśń z tysiąca i jednej nocy (*tysiąca i jednej nocy* : génitif féminin pluriel)
(*compte de mille et une nuit*)

Eventuellement, pour les numéraux supérieurs à 100, les composants initiaux peuvent être invariables pendant que les deux derniers composants s'accordent avec le nom principal.

[183] Nie sposób zapoznać się z tysiąc pięćset osiemdziesięcioma dziewięcioma uczestnikami. (*tysiąc pięćset* : nominatif; *osiemdziesięcioma dziewięcioma* : instrumental; *uczestnikami* : instrumental) = *Nie sposób zapoznać się z tysiącem pięciuset osiemdziesięcioma dziewięcioma uczestnikami.* (*tysiącem pięciuset osiemdziesięcioma dziewięcioma* : instrumental; *uczestnikami* : instrumental)
(*Il n'est pas possible de faire connaissance des 1589 participants.*)

Les cas du **nominatif**, de l'**accusatif** et du **vocatif** sont compliqués. Pour le masculin humain le nominatif et le vocatif n'existent pas. Les groupes nominaux en position du sujet se mettent alors toujours au génitif, comme dans l'exemple [180] ci-dessus. L'accusatif du masculin humain est régulier.

Pour les autres genres le nominatif, l'accusatif et le vocatif sont égaux et on y distingue 2 cas :

a) pour **22-24**, **32-34**, ..., **92-94**, **102-104**, **122-124**, **132-134**, ..., **192-194**, ..., **902-904**, **922-924**, ..., **992-994**, **1002-1004**, **1022-1024**, etc. - le groupe nominal entier s'accorde avec le nom principal

[27] Ou à l'accusatif, les deux formes étant égales pour le masculin humain pluriel.

[184] *Na liście figurowały tysiąc sto sześćdziesiąt dwie osoby.* (*tysiąc sto sześćdziesiąt dwie osoby* : nominatif féminin pluriel)
(*Mille cent soixante deux personnes figuraient sur la liste.*)

b) Pour les autres : **5-21**, **25-31**, **35-41**, ..., **95-101**, **105-121**, **125-131**, ..., **995-1001**, etc., le numéral est au nominatif, tandis que le groupe nominal qui suit est au génitif. Le verbe se met alors au singulier neutre :

[185] *Od początku miesiąca pojawiło się sto czternaście spadających gwiazd.* (*sto czternaście* : nominatif; *spadających gwiazd* : génitif féminin pluriel; *pojawiło się* : 3e personne du singulier neutre)
(*Cent quatorze étoiles filantes sont apparues depuis le début du mois.*)

Même si les règles présentées ci-dessus sont générales pour tous les emplois des numéraux cardinaux polonais, elles sont tellement complexes et surprénnantes que, pour les buts de description de mots composés, nous pouvons les considérer comme exceptions plutot que régularités. Ainsi, nous pouvons dire que les noms composés contenant 1, 2, 3 ou 4 sont réguliers (dans les sens de la définition 4, section 3.3), aussi bien du point de vue des constituants caractéristiques que des formes fléchies. Par exemple pour :

[186] *trzej muszkieterowie* (trois mousquetaires)

[187] *cztery ściany* (une pièce enfremée, litt. “quatres murs”)

toutes les formes fléchies (7) existent, les deux constituants sont caractéristiques, et ils s’accordent dans tous les cas. Evidemment, il n’y a pas de forme du singulier de ces composés, mais ceci n’est pas une irrégularité, car les numéraux *trzej* et *cztery* (qui sont caractéristiques) n’ont pas de singulier eux-mêmes.

Pour les numéraux supérieurs à 4, nous distinguons le masculin humain des autres genres (voir point 4) ci-dessus). Les noms masculins humains sont irréguliers (dans le sens de la définition 5, section 3.3), car le nominatif et le vocatif n’existent pas, mais les deux composants caractéristiques s’accordent dans tous les autres cas :

[188] *dwunastu apostołów* (douze apôtres - génitif), *dwunastu apostołom* (datif), *dwunastu apostołów* (accusatif), *dwunastoma apostołami* (instrumental), *dwunastu apostołach* (locatif)

Pour les autres genres tous les cas existent, et les deux constituants caractéristiques s’accordent si le numéral se termine par 2, 3 ou 4 (flexion régulière). Sinon les constituants caractéristiques s’accordent sauf au nominatif, à l’accusatif et au vocatif (flexion irrégulière) :

[189] *dziesięć przykazań* (*dix commandement* – nominatifs=accusatif=vocatif) – *dziesięć* est au nominatif (=accusatif=vocatif), *przykazań* est au génitif

3.6 Morphologie dérivationnelle et conversion

Les mots composés, comme les mots simples, peuvent être soumis à la **dérivation** par préfixation ou suffixation pour créer des nouvelles entités syntaxico-sémantiques. Analysons quelques exemples an anglais :

[190] *up-to-date* > *up-to-dateness* (l’état d’être à jour), *ivory-tower* > *ivory-towerist*, *captain-general* > *captain-generalcy* (le poste d’un capitain-général), *forty-nine* > *forty-niner* (chercheur d'or en Californie pendant la ruée de 1849), *good-looking* > *good-lookingness*, etc.

Aucun des constituants soulignés n'existe en tant que mot simple indépendant. Ils ne peuvent apparaître que dans les composés ci-dessus. Il s'agit donc bien de mots composés entiers, i.e. traités « en bloc », qui subissent la dérivation, ce qui témoigne de leur degré élevé de figement.

Un phénomène semblable est lié à la **conversion**, définie par Bauer (1983) comme l'utilisation d'un mot classé normalement dans une certaine catégorie comme s'il appartenait à une autre catégorie, sans aucun changement de la forme de ce mot. Ceci arrive très fréquemment en anglais où presque chaque catégorie peut être soumise à la conversion et où la conversion peut produire des formes dans presque chaque catégorie. Par exemple, un nom devient verbe : *a commission > to commission*, et l'inverse : *to interrupt > an interrupt*, une conjonction ou une préposition peut devenir un verbe ou un nom : *But me no buts! She liked to remember her life's ups and downs. They couldn't afford to up the prices too much if they wanted to stand up to the competition.*

Les composés nominaux fondés sur des structures non nominales, comme ceux mentionnés dans les sections 3.2 (exemples [83]-[93]) et 3.4.3, sont un terrain privilégié pour la conversion. Dans beaucoup de ces composés le degré de figement est plus élevé à cause du fait que leurs constituants ne subissent pas la conversion en dehors des composés. Par exemple dans :

[191] *fours in hand* (attelages à quatre chevaux), *good-for-nothings* (bons à rien), *set-tos* (disputes), *cure-alls* (panacée), *has-beens* (hommes finis), *ne'er-do-wells* (bons à rien), *johny-come-latelies* (nouveaux venus), *drive-yourselves* (des véhicules loués et conduits par ceux qui les louent), *coffee-ands* (=*coffees and doughnuts*, cafés et beignets), *carry-ons* (cirques), *court martials* (cour martiale)

les marques soulignées du pluriel ne peuvent pas accompagner les mêmes mots dans des groupes nominaux libres car ces formes n'existent pas en tant que noms simples.

Les mots composés peuvent aussi comporter des formes dérivées à partir de mots obtenus par conversion. Par exemple, dans les adjectifs composés du type :

[192] *bottle-arsed* (plus large à l'une extrémité qu'à l'autre), *bowler-hatted* (qui porte un chapeau melon), *ill-omened* (peu propice)

les éléments soulignés sont des participes obtenus par conversion de noms en verbes et rajout du suffixe *-ed*. Ici encore, ils ne peuvent pas prendre les mêmes formes en dehors des composés.

Les mots composés entiers peuvent aussi être à la base de conversion. Par exemple, les adjectifs et les participes composés suivants :

[193] (ang.) *missing in action > the missing in action* (portés disparus)

[194] (ang.) *kicking-off > the kicking-off*

[195] (pol.) *ciężko ranny* (un blessé grave)

[196] (pol.) *umyslowo chory* (un malade mental)

se nominalisent sans aucune modification de forme, et en anglais sans ajout de marque de pluriel, de la même façon que ceci arrive avec les adjectifs simples comme

[197] (ang.) *poor-> the poor* (les pauvres - employé seulement au pluriel)

[198] (pol.) *chory, chorzy* (le/les malade/s)

3.7 Variantes orthographiques

Celles-ci sont relativement nombreuses et pas toujours prévisibles (voir Mathieu-Colas 1990 pour le cas du français). Les phénomènes qui interviennent dans leur création peuvent être de différentes natures. Analysons des exemples de l'anglais où l'orthographe des mots simples qui constituent un composé, ou du composé lui-même, peut varier en fonction de

– la région

[199] *dialling code* (AB[28]) = *dialing code* (AA[29]) (indicatif)

[200] *armour plate* (AB) = *armor plate* (AA) (blindage)

[201] *apple of Peru* (AB) = *apple-peru* (AA) (une plante péruvienne)

– l'emploi des majuscules et minuscules

[202] *Boolean algebra = boolean algebra* (algèbre booléenne)

[203] *know-nothingism = Know-Nothingism* (ignorance volontaire)

– l'insertion ou omission de séparateurs

[204] *curling-tongs = curling tongs* (fer à friser)

[205] *corn root-worm = corn rootworm*

[206] *air control man = air controlman* (controlleur du traffic aérien)

Les phénomènes ci-dessus peuvent aussi se produire simultanément, comme dans :

[207] *China-root = chinaroot* (un arbuste de l'Extrême-Orient)

Il n'est pas toujours clair si l'existence de versions orthographiques pour des mots simples implique la pertinence des variantes respectives des composés. Par exemple, dans le cas des deux formes équivalentes

[208] *goofer = goopher* (une formule magique)

seule une variante de composé est explicitement indiquée dans le dictionnaire[30]

[209] *goofer dust* (une poudre utilisée pour jeter un sort *; goopher dust* n'est pas indiqué)

Un autre problème concerne le placement des composés nominaux en positions de modifieurs d'autres noms. Par exemple dans,

[210] *working class* (classe ouvrière)

les deux composants sont séparés par un blanc, mais dans

[211] *working-class origins* (d'origine de la classe ouvrière)

ce composé prend un trait d'union. Cette règle n'est pas généralisable, par exemple :

[212] *air traffic, air traffic control* (trafic aérien, contrôle du trafic aérien)

[28] AB = anglais britannique
[29] AA = anglais américain
[30] En l'occurrence, *New Shorter Oxford English Dictionary* (NSOED 1996)

3.8 Variations de l'ordre des constituants

La langue polonaise, où l'ordre des mots dans une phrase est relativement libre, autorise aussi un degré de liberté de l'ordre des constituants dans les noms composés *Nom Adjectif* et *Adjectif Nom*.

– Certains d'entre eux admettent aussi bien l'ordre Nom Adjectif que Adjectif Nom :

  [213] bezwzględna większość = większość bezwzględna (la majorité absolue)

  [214] woda bieżąca = bieżąca woda (l'eau courante)

– Certains « préfèrent » l'un de ces ordres à l'autre :

  [215] węgiel kamienny plutôt que *kamienny węgiel* (du charbon)

  [216] podeszły wiek plutôt que *wiek podeszły* (l'âge avancé)

– D'autres acceptent seulement l'un de ces ordres (sauf parfois avec une accentuation spéciale dans la phrase). L'inversion de cet ordre donne alors soit une séquence incorrecte, comme pour :

  [217] dobre imię (une bonne réputation)*, *imię dobre*

  soit un autre groupe nominal dont le sens n'est pas celui du composé d'origine :

  [218] pan młody (le nouveau marié)*, młody pan* (un jeune homme)

  [219] liczba mnoga (le pluriel)*, mnoga liczba* (un grand nombre)

  Ce dernier phénomène a lieu aussi en français, ou certains adjectifs se placent à gauche du nom dans des emplois figés et à droite avec leur sens libre[31] :

  [220] un grand acteur (qualité), *un acteur grand* (taille)

  [221] un grand homme (célèbre)*, un homme grand* (taille)

3.9 Autres variantes terminologiques

Dans les sections 3.7 et 3.8 nous avons vu que certains noms composés ont plus d'une forme graphique correcte. Ainsi, leur reconnaissance automatique dans des textes peut s'avérer difficile même si l'on dispose de dictionnaires électroniques suffisamment complets, car il n'est pas toujours possible de recenser toutes les variantes de chaque entrée.

Aux variations de l'ordre des mots et de l'orthographe se rajoutent d'autres transformations présentes surtout dans des termes de langages spécialisés. Ces phénomènes ont été classés d'une façon détaillée par Jacquemin (1997) pour l'anglais et par Daille (1994) pour le français. Les classements des types de transformations introduits par les deux auteurs ne sont pas identiques. En voici un résumé, complété par nos propres exemples :

1) Surcompositions.

   Un composé se voit ajouter un modifieur ou une tête pour créer un terme plus complexe :

   [222] (ang.) *link control > [link control] cards*

   [223] (ang.) *fibre channel > [fibre channel] [storage system]*

[31] Une étude détaillée du placement des adjectifs avant ou après nom en français a été faite par Garrigues (1997).

[224] (fr.) *réseaux de microstations > connection de [réseaux de microstations]*

[225] (pol.) *wartość dodana* (valeur ajoutée) > *podatek od [wartości dodanej]* (taxe sur valeur ajoutée)

[226] (pol.) *stan cywilny* (état civil) > *urząd [stanu cywilnego]* (office d'état civil)

2) Insertions.

Un modifieur est inséré à l'intérieur d'un composé. Ceci peut être vu aussi comme phénomène inverse - l'ellipse d'un élément inférieur :

[227] (ang.) *power unit > power distribution unit*

[228] (ang.) *automatic formatting > automatic disk formatting*

[229] (fr.) *précision de pointage > précision globale de pointage*

[230] (pol.) *audycja na żywo* (emission en direct) > *audycja radiowa na żywo* (emission de radio en direct)

Une surcomposition et une insertion peuvent avoir lieu simultanément :

[231] (ang.) *cache control > cache page control information*

3) Coordinations.

Un terme est coordoné avec un autre terme par la mise en facteur de la partie commune[32] :

[232] (ang.) *minimum configuration + maximum configuration > minimum and maximum configuration*

[233] (fr.) *élévateurs de fréquence + abaisseurs de fréquence > élévateurs et abaisseurs de fréquence*

[234] (pol.) *data urodzenia + miejsce urodzenia > data i miejsce urodzenia* (date et lieu de naissance)

4) Variantes morphosyntaxiques.

La morphologie des constituants et la structure syntaxique du composé changent mais la relation de synonymie entre l'ancien et le nouveau terme et préservée :

[235] (ang.) *date of birth > birth date* (date de naissance)

[236] (fr.) *tension des artères > tension artérielle*

[237] (pol.) *podatek dochodowy* (impôt sur le revenu) > *podatek od dochodu*

5) L'emploi de différentes formes dérivationnelles

[238] (ang.) *faith curer = faith curist* (qui fait guéir par la foi et la prière, sans médicaments conventionnels)

[239] (ang.) *medical examiner* (médecin légiste) ?= *medical examinator*
(les dictionnaires usuels de référence indiquent les deux variantes équivalentes du mots simples, *examiner = examinator*, mais seulement une variante du mot composé, *medical examiner* ; la variante *medical examinator* n'y figure pas explicitement)

6) Abréviations

[240] (ang.) *phys-ed = physical education* (éducation physique)

[32] Domingues (1998/9) analyse les problèmes de reconnaissance de termes coordonnés.

[241] (ang.) *choc-bar* = *chocolate-bar* (barre chocolatée)

[242] (ang.) *coin-op* = *coin-operated* (laverie automatique)

3.10 Conclusion

Certains des problèmes linguistiques présentés ci-dessus ont trouvé des solutions dans les structures de données et algorithmes informatiques que nous décrivons dans les chapitres suivants.

Dans la suite de ce mémoire (chapitre 3) nous nous penchons sur la description formelle de la flexion régulière et irrégulière des mots composés.

Nous verrons comment le phénomène de la morphologie dérivationnelle et de la conversion en anglais est représenté dans le dictionnaire des mots composés anglais (chapitre 5).

Le rattachement des variantes graphiques a été réalisé par Mathieu-Colas (1990) pour les mots simples et composés du français, et par Monceaux (1995) pour certains cas de mots simples de l'anglais. Le même problème n'a pas encore été traité dans notre système de dictionnaires pour les composés anglais. Ceci concerne également les autres variantes terminologiques exemplifiées dans la section 3.9. Actuellement, les différentes variantes – lorsqu'elle sont recensées – sont des entrées indépendantes. Dans le chapitre 8 nous proposons une méthode d'enrichissement terminologique qui permet de reconnaître dans des textes certaines surcompositions des termes déjà présents dans les dictionnaires.

Chapitre 4 Flexion automatique des mots composés

4.1 Introduction

Nous avons mentionné (section 2.4) que pour l'analyse automatique des mots composés il faut disposer d'un dictionnaire du type DELACF car ce sont les formes fléchies et non pas les formes canoniques des composés qui apparaissent dans des textes. Nous souhaitons que, une fois le DELAC constitué, le DELACF puisse être généré automatiquement.

Nous avons effectué dans les sections 3.2 à 3.5 l'étude des régularités et des irrégularités de flexion des noms composés. Maintenant nous présentons une méthode originale de description de cette flexion qui permet la génération automatique des formes fléchies des mots composés.

4.2 Contenu d'une entrée du DELAC

Après ce que nous avons dit dans la section 2.2.3 (définition 3) et 3.3 (définition 4), nous voyons que pour fléchir un mot composé il faut fléchir sa tête, comme dans *cousin au deuxième degré, cousine au deuxième degré, cousins au deuxième degré, cousines au deuxième degré*. Il est donc nécessaire, pour chaque composé, d'indiquer sa tête ainsi que de fournir pour chacun des constituants caractéristiques 3 sortes d'information :

1) Le **code flexionnel** provenant du DELAS.

   Il n'est pas toujours évident de le marquer correctement. Premièrement, les mots simples sont ambigus, par exemple le nom anglais *brother* a deux formes du pluriel *brothers* et *brethren* correspondant à deux codes flexionnels différents : *N1* et *N1;1*. Il faut donc savoir que le pluriel du nom composé *brother-in-law* se construit avec ce premier code (voir Silberztein 1993a, pp. 91-94).

   Deuxièmement, certains composés contiennent des composants qui n'ont pas de statut de mots simples variables indépendants et qui donc soit n'existent pas du tout dans le DELAS (voir exemples [190]), soit ne sont codés que comme des mots invariables (exemples [191]). Ceci est par exemple le cas de *comedia dell'arte* et *jazz-band* en polonais, ou de *stand-by,* et *up-to-dateness* en anglais. Afin de garder la cohérence entre le DELAS et le DELAC, nous devons introduire dans le DELAS les entrées « artificielles » : pour le polonais *comedia.N2* et *band.N116,* pour l'anglais *by.N1* et *dateness.N3* qui nous permettront de fléchir les composés en question (voir aussi la discussion des cas particuliers chez Silberztein 1993a, pp. 77-79)

2) Le **lemme**.

   Le code flexionnel décrit comment obtenir les formes fléchies à partir de la forme lemmatisée, tandis que beaucoup de mots composés contiennent dans leur forme lemmatisée des formes simples qui elles ne sont pas des lemmes, comme ceci à déjà été remarqué dans la section 2.4 (point 3, premier commentaire). Ainsi, dans *carte blanche*, l'adjectif caractéristique *blanche* est une forme non lemmatisée à partir de laquelle nous devons produire une autre forme fléchie *blanches* pour obtenir le pluriel *cartes blanches.* Pour cet exemple on pourrait envisager une procédure de déduction de la forme lemmatisée *blanc* à partir de *blanche* et de son code *N8*, mais ceci n'est pas toujours possible sans

information complémentaire. Par exemple, pour mettre au pluriel féminin le mot *mémoire vive,* il faut d'abord retrouver le masculin singulier *vif* auquel on appliquera ensuite l'opération ***1ves**:fp* de son transducteur de flexion pour obtenir *vives.* Or, la terminaison *-f* de *vif* est perdue lors de la production de *vive* et ne peut pas être reconstituée à partir du code flexionnel. C'est pourquoi chaque composant variable d'un mot composé doit être accompagné de sa forme lemmatisée.

3) Les **traits flexionnels**.

Il sont nécessaires pour indiquer ceux du mot composé entier (voir définition 3, section 2.2.3). En particulier, une forme peut être ambiguë, i.e. avoir des jeux de traits flexionnels différents pour le même lemme. Par exemple, dans le nom polonais *dom dziecka* (orphelinat), le premier nom (caractéristique) peut être aussi bien au nominatif qu'à l'accusatif.

Ces trois informations sont exactement celles que l'on associe à chaque mot simple fléchi dans le DELAF. En conséquence, les entrées du DELAC sont comme dans l'exemple [46] repris de la section 2.4 :

[243] abbaye(abbaye,N21:fs) cistercienne(cistercien,A41:fs).N+NA:fs/+N

Mais ceci ne suffit pas pour les cas irréguliers que nous présentons dans les sections 3.2 -3.5. Par exemple, *blanc d'oeuf*, qui admet deux variantes au pluriel (*blancs d'oeuf, blancs d'oeufs*), ne se fléchit pas comme *pomme de terre*, alors que ces deux noms composés appartiennent à la même classe typologique *NdeN*. Si nous décrivons ce premier en tant que

[244] blanc(blanc.N1:ms) d'oeuf.N+NdeN:ms/+N

l'algorithme de flexion ne peut pas fléchir *oeuf* pour obtenir *blancs d'oeufs*. D'autre part, si nous écrivons

[245] blanc(blanc.N1:ms) d'oeuf(oeuf.N1:ms).N+NdeN:ms/+N

nous perdons l'information que seul le premier nom *blanc* est caractéristique (donc que *blancs d'oeuf* est bien au pluriel), et en plus nous ne savons pas que la forme **blanc d'oeufs* n'est pas correcte.

Silberztein (1993a, pp. 100-103) décrit des algorithmes employés pour obtenir le DELACF français à partir du DELAC par des utilitaires du type AWK ou SED. Ces algorithmes-là, fonctionnels pour la plupart des mots composés français, ne permettent pas la flexion automatique des exceptions, et ils ne sont pas réutilisables pour d'autres langues.

Ci-dessous, nous présentons une méthode universelle de flexion automatique des mots composés.

4.3 Fichiers de flexion

Pour chaque langue traitée, nous utilisons un fichier contenant ses types de flexion (genre, nombre, etc.) avec, pour chaque type, l'énumération de ses formes. Ce fichier pour le français comprendra les informations suivantes :

N : s,p
R : m,f
P : 1,2,3
T : W,P,I,J,F,G,K,S,T,C,Y

Les caractères initiaux en majuscules suivis du deux-points représentent ici les types de flexion : le nombre (*N*), le genre (*R*), la personne (*P*), le temps et le mode (*T*). Les caractères après les deux-points ont les mêmes significations que dans le tableau Tab.4 : singulier (*s*), pluriel (*p*), masculin (*m*), féminin (*f*), première (*1*), deuxième (*2*), troisième (*3*) personne, infinitif (*W*), etc. De même, les fichiers de flexion pour le polonais et pour l'anglais contiennent respectivement 6 et 4 types des flexion codés : nombre (*N*), genre (*R*), cas (*A*), personne (*O*), temps et mode (*E* ou *T*), gradation (*Y*).

N : s,p
R : o,z,r,f,n
A : M,D,C,B,I,L,W
O : 1,2,3
E : F, H, P, S, U, J, K, Q, T, Z, G
Y : ∅,c,u

N : s,p
O : 1,2,3
T : W,P,I,G,K
Y : ∅,C,S

Le choix des codes pour les types des flexions et pour les traits flexionnels peut être différent de celui présenté ci-dessus à condition que

- chaque code contienne un seul caractère,
- les codes pour les types de flexion ainsi que pour les traits flexionnels soient non ambigus,
- les codes employés dans les dictionnaires soient cohérents avec ceux du fichier de flexion.

4.4 Fichiers-dictionnaires

Le dictionnaire DELAC est divisé en sous-fichiers selon la façon dont les mots composés se fléchissent. Chacun de ces sous-fichiers contient :

- une entête avec
  - la description des constituants caractéristiques,
  - la description de la flexion irrégulière, le cas échéant,
- une liste des mots composés dont la flexion correspond à la description de l'entête du fichier.

Les cas les plus simples sont ceux de la flexion régulière telle qu'elle est définie dans la section 3.3 (définition 4). Ils concernent la majorité des composés dans les trois langues présentées.

4.4.1 Français

En français la majorité des noms composés du type *Nom Adj* et *Adj Nom* sont regroupés dans un seul fichier dont voici un petit extrait :

[246] *#+/+*
abbaye(abbaye.N21:fs) cistercienne(cistercien.A41:fs),N+NA:fs/+N
cousin(cousin.N32:ms) germain(germain.A32:ms),N+NA:ms/+N+G
jeune(jeune.A31:fs) fille(fille.N21:fs),N+AN:fs/+N
merle(merle.N1:ms) blanc(blanc.A47:ms),N+NA:ms/+N
petit(petit.A32:ms) ami(ami.N32:ms),N+AN:ms/+N+G
Pays-bas,N+NA:mp
...

La première ligne signifie que chacun des mots composés qui suivent contient deux formes simples caractéristiques. Aucune information flexionnelle complémentaire n'est nécessaire, car l'on a affaire à une flexion régulière : pour obtenir le pluriel ou le féminin du composé (dans le cas où la flexion en nombre ou en genre est admise par les marques *+N +G*), il faut mettre ses deux constituants respectivement au pluriel ou au féminin. Les codes flexionnels ne sont pas utiles pour les mots sans flexion comme *Pays-bas*. Ce dernier substantif est à flexion irrégulière selon la définition 5 (section 3.3), car il n'admet pas le singulier qui est pourtant possible pour cette séquence en tant qu'un groupe nominal libre (*un pays bas*). Néanmoins, placer ce composé dans le même fichier-dictionnaire que les *NomAdj* réguliers n'introduira pas d'erreur de flexion car celle-ci n'est pas admise dans l'entrée elle-même (manque de marque *+N*).

Un autre fichier à flexion régulière contiendra la plupart des noms du type *Nom Prep Nom* :

[247] *#+/-/-*
avocat(avocat.N32:ms) de le diable,N+NdeN:ms/+N+G[33]
boîte(boîte.N21:fs) à musique,N+NaN:fs/+N
champs d'honneur,N+NA:ms
frère(frère.N1:ms) de lait,N+NdeN:ms/+N
preuve(preuve.N21:fs) par absurde,N+NPrepN:fs/+N
...

Ici l'entête (également limitée à une seule ligne) décrit le fait que les mots composés sont de longueur 3, et que seul le premier composant est caractéristique.

Une autre classe importante du français, celle des *PrepNom* et certains *NomNom* sera décrite par l'entête indiquant que seul le deuxième composant est caractéristique :

[248] *#-/+*
auto-stoppeur(stoppeur.N35:ms),N+NN:ms/+N+G
avant-garde(garde.N21:fs),N+PrepN:fs/+N
avant-gardiste(gardiste.X31:ms),N+PrepN:ms/+N+G
avant-goût(goût.N1:ms),N+PrepN:ms/+N
baby-sitter(sitter.X31:ms),N+NN:ms/+N+G
contre-révolution(révolution.N21:fs),N+NPrepN:fs/+N
sous-maître(maître.N39:ms),N+PrepN:ms/+N+G
...

Regardons maintenant quelques fichiers à flexion irrégulière en français. Le cas de la classe *Nom Nom*, qui est irrégulière toute entière (voir section 3.3, exemple [126]), est décrit par l'entête suivante :

[33] Le DELACF français contient les déterminants sans élision.

[249] *#+/-*
#p:p/p
allocation(allocation.N21:fs)-chômage(chômage.N1:ms),N+NN:fs/+N
bateau(bateau.N3:ms)-mouche(mouche.N21:fs),N+NN:ms/+N
coton(coton.N1:ms)-tige(tige.N21:fs),N+NN:fs/+N
moissonneuse(moissonneuse ,N21:fs)-lieuse(lieuse ,N21:fs),N+NN:fs/+N
...

La première ligne indique, comme toujours, le nombre de composants et marque les composants caractéristiques (ici seulement le premier). La deuxième ligne indique que le pluriel est irrégulier : pour l'obtenir il faut mettre au pluriel le premier composant et aussi le deuxième (qui n'est pas caractéristique). Pour le reste de la flexion du sous-fichier ci-dessus on pourrait introduire des lignes analogues :

#s:s/-
#m:m/-
#f:f/-

pour indiquer que le singulier, le masculin et le féminin sont réguliers. Mais ces informations sont données par défaut, c'est-à-dire que si l'entête du fichier ne contient aucune ligne concernant la forme flexionnelle qui nous intéresse, cette forme est régulière et donnée par la première ligne, ici #+/-. En revanche, si une telle ligne existe, comme ici pour le pluriel, elle masque la flexion régulière. Alors, le pluriel régulier **bateaux-mouche* correspondant à *p:p/-* n'existe pas. Si l'on veut avoir les deux formes, il faut les inclure explicitement dans l'entête comme cela a été fait pour *blanc d'oeuf, chef d'état* etc. :

[250] *#+/-/-*
#p:p/-/-
#p:p/-/p
blanc(blanc.N1:ms) d'oeuf(oeuf.N1:ms), N+NdeN:ms/+N
chef(chef.N31:ms) d'état(état.N1:ms),N+NdeN:ms/+N+G
...

Les lignes 2 et 3 indiquent les façons d'obtenir les formes du pluriel :

- régulier : *blancs d'oeuf*, *chefs d'état*, où seulement le premier composant se met au pluriel, et les deux autres restent invariables,
- irrégulier : *blancs d'oeufs*, *chefs d'états*, où non seulement la tête se fléchit, mais aussi le troisième composant qui est non caractéristique

Comme nous venons de mentionner, la deuxième ligne est obligatoire ici, car chaque description de flexion irrégulière bloque la flexion régulière correspondante. Cela veut dire que l'entête de la forme

#+/-/-
#p:p/-/p
blanc(blanc.N1:ms) d'oeuf(oeuf.N1:ms), N+NdeN:ms/+N
chef(chef.N31:ms) d'état(état.N1:ms),N+NdeN:ms/+N+G
...

admettrait uniquement le pluriel irrégulier *blancs d'oeufs, chef d'états*.

L'entête peut être plus longue, comme c'est le cas pour *simulateur de vol* :

#+/-/-
#s:s/-/-
#s:s/-/p
#p:p/-/-
#p:p/-/p
simulateur(simulateur.N1:ms) de vol(vol.N1:ms), N+NdeN:ms/+N
...

où les lignes de 2 à 5 décrivent respectivement le singulier *simulateur de vol*, *simulateur de vols*, et le pluriel *simulateurs de vol, simulateurs de vols.*

Jusqu'à présent nous avons discuté seulement de la construction du pluriel. Ceci est dû au fait que la flexion irrégulière en français concerne surtout le nombre. Voici des exemples où l'irrégularité touche aussi les formes féminines :

[251] *#+/+*
#f:m/f
franc(franc.N47:ms)-maçon(maçon.N41:ms),N+NdeN:ms/+N+G
franc(franc.A47:ms)-tireur(tireur.N35:ms), N+NdeN:ms/+N+G
...

La deuxième ligne indique que le féminin (pour les deux nombres) s'obtient de façon irrégulière en gardant le premier constituant au masculin : *franc-maçonne, francs-maçonnes, franc-tireuse, francs-tireuses.*

Examinons aussi la classe qui n'a pas de constituant caractéristique, celle des *VerbeNom* en français. Comme les *NomNom*, cette classe est irrégulière toute entière. Le pluriel d'un *VerbeNom* est soit identique au singulier, comme dans *porte-avions*, soit créé par la mise au pluriel du deuxième constituant, comme dans *tire-bouchon* > *tire-bouchons*. Les deux cas peuvent être décrits par la même entête :

[252] *#-/-*
#p:-/p
porte-avions(avion.N1:mp),N+VN:ms/+N
tire-bouchon(bouchon.N1:ms),N+VN:ms/+N
...

4.4.2 Anglais

Voici des extraits des classes de noms composés à flexion régulière. La première et la deuxième contiennent entre autres les séquences où le nom caractéristique est à la fin et ses modifieurs le précèdent. Seul ce dernier nom se met alors au pluriel : *anchor men, bas-reliefs, athlete's feet,* etc. On y trouve aussi des structures particulières qui ont été nominalisées et prennent la marque du pluriel au dernier composant : *take-offs, has-beens, forget-me-nots, stay-at-homes*.

[253] *#-/+*
anchor man(man.N8:s),N+NN:ms/+N (présentateur)
bas-relief(relief.N7:s),N+XN:s/+N[34] (bas-relief)
by-election(election.N1:s),N+PrepN:s/+N (élection partielle)
has-been(been.N1:s),N+XX:s/+N (homme fini)
take-off(off.N1:s),N+VPrep:s/+N (décollage)
...

[254] *#-/-/+*
althlete's foot(foot.N43;1:s),N+NsN:s/+N (champignons aux pieds)
blue-collar worker(worker.N1:s),N+ANN:s/+N (ouvrier)
forget-me-not(not.N1:s),N+XXN:s/+N (myosotis)
gin-and-tonic(tonic.N1:s),N+NandN:s/+N (gin tonic)
stay-at-home(home.N1:s),N+XXN:s/+N (casanier)
students' union(union.N1:s),N+NsN:s/+N (syndicat étudiant)
...

Dans la troisième classe se trouvent typiquement les *NomPrépositionNom,* où seul le premier nom se met au pluriel (*points of view, gins-and-French*, etc.), et dans la quatrième les *NomNom* (exemples [147]-[149], [155]), où les deux composants se fléchissent (*men-servants*, *women writers*, *knights-templars,* etc.).

[255] *#+/-/-*
point(point.N1:s) of view,N+NofN:s/+N (point de vue)
brother(brother.N1:s)-in-law,N+NPrepN:s/+N (beau-frère)
gin(gin.N1:s)-and-French,N+NandN:s/+N
man(man.N8:s)-of-war,N+NofN:s/+N (navire de guerre)
...

[256] *#+/+*
man(man.N8:s)-servant(servant.N1:s),N+NN:s/+N (valet)
woman(woman.N8:s) writer(writer.N1:s),N+NN:s/+N
gentleman(gentleman.N8:s) farmer(farmer.N1:s),N+NN:s/+N (fermier amateur)
knight(knight.N1:s)-templar(templar.N1:s),N+NN:s/+N (Templier)
...

Dans certains *NomAdj* et *NPrep*, comme ceux des exemples [143]-[145] et [161], seul le premier composant varie au cours de la flexion : *notaries public, passers-by,* etc.

[257] *#+/-*
notary(notary.N3:s) public,N+NA:s/+N (notaire)
heir(heir.N1:s) presumptive,N+NA:s/+N (héritier présomptif)
passer(passer.N1:s)-by,N+NPrep:s/+N (passant)
...

Les noms composés anglais avec le pluriel irrégulier ont été décrits d'une façon détaillée dans les sections 3.2, 3.3 et 3.4. Voici les extraits des fichiers-dictionnaires de ces noms.

Le premier admet deux formes du pluriel, où le nom caractéristique est toujours au pluriel et son complément peut l'être ou non (voir exemple [130]) : *heads of government, heads of governments.*

[34] *XN* signifie qu'il y a deux composants et l'on ne précise pas la catégorie du premier.

[258] *#+/-/-*
#p:p/-/-
#p:p/-/p
head(head.N1:s) of government(government.N1:s),N+NofN:s/+N
...

La marque du pluriel peut apparaître alternativement sur le premier ou le deuxième élément (exemples [140]-[142], et [153]) : *battles royal, battle royals,* etc.

[259] *#-/-*
#p:p/-
#p:-/p
battle(battle.N1:s) royal(royal.N1:s),N+NN:s/+N (mêlée générale)
beau(beau.N1:s) ideal(ideal.N1:s),N+NN:s/+N
...

ou encore sur les deux éléments à la fois (exemples [146] et [154]) : *letters patent, letter patents, letters patents,* etc.

[260] *#-/-*
#p:p/-
#p:-/p
#p:p/p
letter(letter.N1:s) patent(patent.N1:s),N+NN:s/+N (lettres patentes)
Lord(lord.N1:s) Lieutenant(lieutenant.N1:s),N+NN:s/+N
(représentant de la couronne dans un comté de Grande-Bretagne)
...

Certains noms ont les formes du singulier et du pluriel identiques alors que leurs composants en position de tête possèdent deux formes différentes (exemples [101]-[102]) : *a cross-roads, two cross-roads,* etc.

[261] *#-/-*
#p:-/-
cross-roads,N+XN:s/+N (carrefour)
lesser yellowlegs,N+XN:s/+N
pince-nez,N+XX:s/+N
...

Remarquons que ce cas n'est pas le même que celui des composés sans singulier ou sans pluriel. Ici à une entrée du DELAC correspondent bien deux entrées factorisées dans le DELACF :

cross-roads,.N+XN:s:p

tandis que les mots comme *street furniture* sont représentés dans le DELAC comme réguliers mais sans flexion en nombre :

#-/+
street furniture,N+VN:s (mobilier urbain)

et n'ont qu'une forme correspondante dans le DELACF :

street furniture,.N+VN:s

4.4.3 Polonais

En polonais, les classes les plus répandues des noms composés, celles des *AdjNom* et *NomAdj*, sont pour la plupart régulières : les deux constituants étant caractéristiques tous les deux se fléchissent : *anielska cierpliwość* (patience d'ange)*, anielskiej cierpliwości, anielską cierpliwością, anielską cierpliwość*,..., *film rysunkowy* (dessin animé)*, filmu rysunkowego, filmy rysunkowe, ...*Voici le fichier-dictionnaire correspondant :

[262] *#+/+*
anielska(anielski.A3:Mfs) cierpliwość(cierpliwość.N5203:Mfs),N+AN:Mfs/+N+C
artystyczny(artystyczny.A1:Mrs) nieład(nieład.N5105:Mrs),N+AN:Mrs/+N+C
(désordre artistique)
brylantowe(brylantowy.A1:Mrp) gody(gody.N4:Mrp),N+AN:Mrp/+C
(noces de diament)
film(film.N1:Mrs) rysunkowy(rysunkowy.A1:Mrs),N+NA:Mrs/+N+C
pan(pan.N119:Mos) młody(młody.A6:Mos),N+NA:Mos/+N+C+f
(jeune marié)
wybory(wybór.N1265:Mrp) powszechne(powszechny.A1:Mrp),N+NA:Mrp/+C
(élections générales)
...

Une autre classe régulière importante est celle des $NomNom_{gén}$, où le premier nom est caractéristique et le deuxième reste toujours au génitif : *egzamin dojrzałości* (baccalauréat, litt. « l'examen de maturité »)*, egzaminu dojrzałości, egzaminy dojrzałości, ...* De la même façon fonctionne la classe $NomNom_{instr}$ avec le deuxième substantif toujours à l'instrumental : *rzut oszczepem* (lancer de javelot)*, rzutu oszczepem, rzutowi oszczepem, ...*

[263] *#+/-*
egzamin(egzamin.N1146:Mrs) dojrzałości,N+NNgén:Mrs/+N+C
miara(miara.N229:Mfs) rzeczy,N+NNgén:Mfs/+N+C
(sens de l'importance des choses, litt. « la mesure des choses »)
rzut(rzut.N10:Mrs) oszczepem,N+NNinst:Mrs/+N+C
powożenie(powożenie.N31:Mns) końmi,N+NNinst:Mns/+N+C
(conduite d'un char tiré par des chevaux)
...

Le premier nom est également caractéristique dans les composés du type *NomPrépositionNom* dont la plupart sont réguliers : *cukier w kostkach* (sucre en cubes)*, cukru w kostkach, cukrowi w kostkach, ...*

[264] *#+/-/-*
cukier(cukier.N1113:Mrs) w kostkach,N+NNgén:Mrs/+N+C
zamki(zamek.N1112:Mrp) na lodzie,N+NPrepN:Mrs/+C
(rêves irréalisables, litt. « des châteaux sur glace »)
...

Examinons maintenant des exemples des composés à flexion irrégulière. Parmi les $NomNom_{gén}$ on peut distinguer trois groupes irréguliers du point de vue du nombre : celui qui exige la mise au pluriel des deux noms, comme *ojcowie rodzin* (pères de famille)*, głowy rodzin* (chefs de famille) :

[265] *#+/-*
#p:p/p
głowa(głowa.N20:Mfs) rodziny(rodzina.N22:Dfs),N+NNgén:Mfs/+N+C
ojciec(ojciec.N136:Mos) rodziny(rodzina.N22:Dfs),N+NNgén:Mos/+N+C
(père de famille)
...

celui qui admet le pluriel régulier et irrégulier comme l'exemple [131] :

[266] *#+/-*
#p:p/-
#p:p/p
akt(akt.N10:Mrs) urodzenia(urodzenia.N31:Dns),N+NNgén:Mrs/+N+C
(acte de naissance)
głos(głos.N130:Mrs) sumienia(sumienie.N31:Dns),N+NNgén:Mrs/+N+C
(voix de la concience)
...

et celui qui admet deux formes du singulier et deux du pluriel : *pranie mózgu* (lavage de cerveaux)*, pranie mózgów, prania mózgu, prania mózgów* :

[267] *#+/-*
#s:s/-
#s:s/p
#p:p/-
#p:p/p
miłość(miłość.N5203:Mfs) bliźniego(bliźni.N?[35]*:Dos),N+NNgén:Mfs/+N+C*
(l'amour de l'autrui)
pranie(pranie.N31:Mns) mózgu(mózg.N?:Drs),N+NNgén:Mns/+N+C
bohater(bohater.N1110:Mos) dnia(dzień.N141:Drs),N+NNgén:Mos/+N+C+f
(le héros du jour)
...

La classe des *NomNom* (avec les deux composant au nominatif) est irrégulière toute entière comme c'était le cas pour le français (exemple [249]). Mais cette fois-ci l'entête du fichier doit être plus longue car le deuxième composant s'accorde avec le premier non seulement en nombre mais aussi en cas : *człowiek kot, człowieka kota, ludzie koty, ludzi kotów,* etc.

[268] *#+/-*
#p:p/p
#g:g/g
#d:d/d
#a:a/a
#i:i/i
#l:l/l
#v:v/v
czapka(czapka.N?:Mfs) niewidka(niewidka.N?:Mfs),N+NN:Mfs/+N+C
(casquette qui rend invisible)
człowiek(człowiek.N5106:Mos) kot (kot.N1173:Mzs),N+NN:Mos/+N+C
(homme-chat)

[35] Un point d'interrogation figure pour les constituants dont nous ne connaissons pas les codes flexionnels, car nous n'avons pas le DELAS polonais à notre disposition. Ces codes doivent être indiqués pour la version finale du DELAC.

figiel(figiel.N?:Mrs)-migiel(migiel.N?:Mrs),N+NN:Mrs/+N+C (frasques)
kombajn(kombajn.N1146:Mrs) bizon (bizon.N1146:Mzs),N+NN:Mrs/+N+C
(moissonneuse-batteuse)
majster(majster.N?:Mos)-klepka(klepka.N?:Mfs),N+NN:Mos/+N+C
(un incompétent)
...

Cette longue entête peut être simplifiée :

[269] *#+/-*
#p:p/p
#C:C/C
czapka(czapka.N?:Mfs) niewidka(niewidka.N?:Mfs),N+NN:Mfs/+N+C
człowiek(człowiek.N5106:Mos) kot (kot.N1173:Mzs),N+NN:Mos/+N+C
figiel(figiel.N?:Mrs)-migiel(migiel.N?:Mrs),N+NN:Mrs/+N+C
kombajn(kombajn.N1146:Mrs) bizon (bizon.N1146:Mzs),N+NN:Mrs/+N+C
majster(majster.N?:Mos)-klepka(klepka.N?:Mfs),N+NN:Mos/+N+C
...

La troisième ligne de l'entête remplace les lignes 3-8 de l'entête précédente et signifie que chacun des cas énumérés dans le fichier de flexion du polonais par *A:M,D,C,B,I,L,W* (voir section 4.3) s'obtient par la mise à ce cas de chacun des composants.

Dans les exemples [92]-[93] nous avons vu des noms composés construits à partir des parties de discours autres que noms et n'ayant pas la structure d'un groupe nominal. Ces composés sont souvent invariables alors qu'ils sont employés comme des syntagmes nominaux fléchis. Par exemple,

[1] *Owo sam na sam z synem utkwiło mu w pamięci.* (nominatif singulier)
(*Ce tête-à-tête avec son fils lui est resté dans la mémoire*)

[2] *Nigdy nie zapomniał tego sam na sam* (génitif singulier)
(*Il n'a jamais oublié ce tête-à-tête*)

[3] *Potrzeba było wielu sam na sam by dobrze się zrozumieć.* (accusatif pluriel)
(*Il a fallu plusieurs tête-à-tête pour bien se comprendre*)

Il faut donc coder *sam na sam* comme nom sans constituant caractéristique admettant une flexion en nombre et cas, où chaque forme fléchie est identique au lemme. Voici l'extrait du fichier correspondant :

[270] *#-/-/-*
#N:-/-/-
#C:-/-/-
sam na sam,N+PrPrepPro:Mns/+N+C (tête-à-tête)
trzy po trzy,N+NumPrepNum:Mns/+N+C (des bobards)
bla bla bla,N+OnomOnomOnom:Mns/+N+C
...

Les lignes 2 et 3 de l'entête représentent le fait que pour obtenir n'importe quel nombre (*N*) et n'importe quel cas (*C*) il faut garder les trois composants sans modification par rapport à la forme de base.

Regardons maintenant des exemples de noms composés construits à partir de numéraux cardinaux, qui, comme nous l'avons vu dans la section 3.5, sont liés à des principes de flexion

très particuliers. Ceux contenant le 1, 2, 3, ou 4 sont réguliers aussi bien du point de vue des constituants caractéristiques que des formes fléchies :

[271] *#+/+*
cztery(cztery.NU20:Mfp) litery(litera.N228:Mfp),N+NumN:Mfp/+C
(cul, litt. « quatre lettres »)
cztery(cztery.NU20:Mfp) ściany(ściana.N22:Mfp),N+NumN:Mfp/+C
(pièce enfermée litt. « quatres murs »)
trzej(trzej.NU14:Mop) muszkieterowie(muszkieter.N1110:Mop),N+NumN:Mop/+C
(trois mousquetaires)
...

[272] *#+/+/+*
dwie(dwaj.Num?:Mfp) lewe(lewy.A1:Mfp) ręce(ręka.N226:Mfp),N+NumAN:Mfp/+C
(gaucher à deux mains litt « deux mains gauches »)
...

Pour les numéraux supérieurs à 4, nous distinguons les masculins humains des autres genres. Les masculins humains sont irréguliers, car le nominatif et le vocatif n'existent pas, ce qui est exprimé pour les deux premières entrées ci-dessous par les marques *+C-n-v*. Pour les autres genres, quand le numéral ne se termine pas par 2, 3 ou 4, les constituants caractéristiques ne s'accordent pas au nominatif, à l'accusatif et au vocatif . C'est pourquoi le nom ne peut pas être considéré comme constituant caractéristique, ce qui est exprimé par la première ligne de l'entête.

[273] *#+/-*
#d:d/d
#i:i/i
#l:l/l
czterdziestu(czterdzieści.NU18:Dop) rozbójników(rozbójnik.N1111:Dop),
N+NumN:Dop/+C-n-v (quarante voleurs)
dwunastu(dwanaście.NU9:Dop) apostołów(apostoł.N?:Dop),N+NumN:Dop/+C-n-v
(douze apôtres)
dziesięć(dziesięć.NU15:Mnp) przykazań(przykazanie.N?:Dnp),N+NumN:Mnp/+C
(dix commandements)
dziesięć(dziesięć.NU15:Mrp) srebrników(srebrnik.N?:Drp), N+NumN:Mfp/+C
...

Les lignes 2, 3 et 4 de l'entête expriment que le datif, l'instrumental et le locatif s'obtiennent en accordant les deux composants. Les 4 autres cas sont « réguliers », car le nom reste au génitif et seul le numéral se fléchit.

4.5 Algorithme de flexion

Une fois que les mots composés ont été décrits, la génération de toutes les formes fléchies se fait entièrement automatiquement. Cette partie du mémoire décrit l'algorithme utilisé pour la flexion automatique des composés.

4.5.1 Exploration d'un transducteur de flexion

L'un des modules du programme effectue l'exploration en profondeur d'un transducteur de flexion. Il enregistre dans une structure toutes les séquences reconnaissables de traits morphologiques avec leurs terminaisons associées. Par exemple, pour le transducteur *A72* du français (exemple [31] et Fig.1) nous produisons l'ensemble de terminaisons : *{<E>:ms,*

x:mp, 2lle:fs, 2lles:fp}. Cette exploration complète d'un transducteur flexionnel se fait une seule fois pour tous les composants simples du DELAC qui ont le code flexionnel donné. Les structures de données pour les transducteurs lus sont rangées dans une table de hachage, la fonction de hachage étant la somme de codes ascii de caractères constituant le code du transducteur, modulo une constante fixée à 1001.

Le format textuel d'un transducteur de flexion est présenté par Silberztein (1997) pp. 75-76. Voici les structures obtenues pour le code *A72*, ainsi que l'algorithme d'exploration :

Etat	**Etat est terminal**	**Nombre de prédécesseurs**	**Nombre de successeurs**	**Liste des transitions**		
1	non	0	2	(<E> : ms , 2)	(x : mp , 2)	(2lle : <E> , 3)
2	oui	2	0			
3	non	1	1	(<E> : fs , 2)	(s : fp , 2)	

Fig.9. Structure du transducteur de flexion *A72.*

Nombre de séquences reconnaissables	**Séquences reconnaissables**			
4	<E>:ms	x : mp	2lle : fs	2lles : fp

Fig.10. Structures des séquences reconnaissables par le transducteur *A72.*

```
lire_transducteur(code)
début
    si( le transducteur pour code déjà lu)
        <retourner sa structure> ;
    sinon
        nbe← <nombre d'états du transducteur> ;
        pour e de 1 à nbe
            pour chaque <transition sortante de e>
                <enregistrer la transition> ;
            fin pour ;
        fin pour ;
        explorer_transd() ;
fin ;
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
explorer_transd()                    //Explorer un transducteur
début
    pour e de 1 à nbe
        VISITÉS[e]← 0 ;          //Compte le nombre de fois qu'un état a été visité
        EXPLORÉS[e] ← 0 ;        //Tableau d'états explorés
        CHEMIN[e] ← ∅ ;          //L'ensemble de chemins partants de e
    fin pour
    explorer(état initial 1) ;
fin
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
explorer(e)          //Explorer le transducteur à partir de l'état e
début
    VISITÉS[e]← VISITÉS[e]+1 ;
    si (état est terminal)
        CHEMIN[e] ← CHEMIN[e] ∪ (<E>/ :<E>)   //Ajouter chemin vide
    fin si ;
    pour chaque (transition t=(ét,s) partante de e)  //ét - l'étiquette de transition, s - état d'arrivée
        si (EXPLORÉS[s]=0 et VISITÉS[s]<100)    //La constante 100 est pour éviter des cycles.
            explorer(s) ;
        fin si ;
        pour chaque (ch ∈ CHEMIN[s])
            CHEMIN[e] ← CHEMIN[e] ∪  (ét ° ch) ;   //Concatener l'étiquette ét avec chaque
                                                    //séquence reconnaissable à partir de s
        fin pour ;
    fin pour ;
    EXPLORÉS[e] ← 1 ;
fin
```

Fig.11. Algorithme d'exploration d'un graphe en profondeur

Complexité : L'algorithme *explorer*, dans le pire des cas, doit parcourir entièrement chaque chemin du transducteur entre l'état initial et l'état final, afin de concaténer tous les symboles rencontrés. De plus les formes égales qui sont mises en facteur doivent être séparées (e.g. l'étiquette *s:mp:fp* dans le transducteur *N31,* Fig.12, décrit deux chemins différents *s:mp* et *s:fp*). Ainsi, la complexité de l'algorithme est proportionnelle à la somme de longueurs de tous les chemins acceptant, soit

$\sum_{(ch : \text{chemin entre l'état initial et final})} longueur(ch) < nb\_chemins * max\_long\_chemins$, où

- *nb_chemins* est le nombre de tous les chemins acceptants, ce qui est équivalent au nombre de toutes les formes fléchies, *nb_formes*, décrites par le transducteur,

– *max_long_chemins* = $max_{(ch\ :\ chemin\ entre\ l'état\ initial\ et\ final)}$ *(longueur(ch))* est la longueur maximale de tous les chemins acceptants du transducteur.

Dans le cas des trois langues considérées cette longueur, pour les transducteurs de flexion, ne dépasse pas 5. Ainsi, la complexité est de l'ordre *O(nb_formes)* (avec la constante multiplicative égale à 5).

Quant aux formes fléchies décrites par un transducteur, leur nombre est égal à :

– 1 ou 2 pour la plupart de constituants de mots composés anglais (2 pour les noms, 1 pour les autres catégories) ;

– 2 ou 4 pour la plupart de constituants de mots composés français (4 pour les noms et adjectifs variables en genre, 2 pour les autres) ;

– 14, 28 ou 70 pour la plupart de constituants de mots composés polonais (14 pour les noms invariables en genre, 28 pour les noms variables en genre, 70 pour les adjectifs).

La complexité de l'algorithme *explorer_transd* est de

*O(nb_états + nb_chemins * max_long_chemins)*

où *nb_états* est le nombre d'états du transducteur.

Dans l'algorithme *lire_transducteur* la complexité de la partie de lecture d'un transducteur est proportionnelle au nombre de transitions *nb_trans*. Ainsi, la complexité totale est de

*O(nb_trans + nb_états + nb_chemins * max_long_chemins)*

Pour les trois langues considérées, le nombre de transitions et le nombre d'états dans un transducteur de flexion ne dépassent pas le double du nombre de chemins acceptants, donc la complexité est de l'ordre *O(nb_formes)* (constante multiplicative égale à 9).

Remarquons que chaque transducteur de flexion est acyclique. Ceci permet de dresser une liste de toutes les séquences reconnaissables ce que nous faisons dans l'algorithme ci-dessus. Cet algorithme est aussi applicable à des transducteurs généraux cycliques. Dans ce cas-là le nombre de toutes les séquences reconnaissable peut être infini. Notre algorithme va en produire un certain nombre fini et ne va pas entrer dans une boucle infinie, grâce à la constante introduite pour limiter le nombre de fois qu'un état peut être visité (condition *VISITÉS[**s**]<100* dans *explorer(**e**)*).

4.5.2 Flexion des mots simples

Un autre module du programme produit les formes fléchies désirées d'un mot simple selon son transducteur de flexion. Si le transducteur donné n'a pas encore été employé nous effectuons sa lecture et exploration. Dans la structure des séquences reconnaissables par le transducteur (Fig.10) nous cherchons celles qui concernent les formes fléchies désirées. Nous produisons chaque forme fléchie en appliquant sa (ses) terminaison(s) appropriée(s) au lemme.

Par exemple, pour fléchir le nom composé français

[274] nouveau(nouveau.A72:ms) riche(riche.N31:ms),N+AN:ms/+N

nous avons besoin du masculin singulier et du masculin pluriel de chacun des composants. Les séquences reconnaissables des deux transducteurs concernés, A72 (Fig.1) et N31 (Fig.12), sont respectivement : *{<E>:ms, x:mp, 2lle:fs, 2lles:fp}* et *{<E>:ms, s:mp, <E>:fs, s:fp}*. Les deux premières séquences de chaque ensemble, appliquées aux lemmes *nouveau* et *riche*, permettent de produire les formes fléchies désirées : *nouveau:ms, nouveaux:mp* et *riche:ms, riches:mp*.

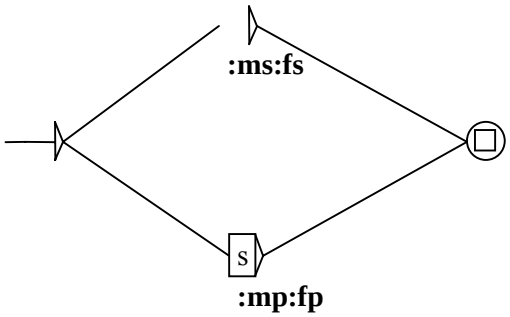


Fig.12. Transducteur de flexion N31

Remarquons qu'une forme fléchie d'un mot simple peut posséder plusieurs variantes réalisées par des chemins différents du transducteur de flexion. Par exemple, le substantif polonais *ręka* a deux formes de l'instrumental pluriel : *rękami* et *rękoma*, produites par deux chemins différents, *mi:Ifp, 1oma:Ifp*. L'algorithme de flexion ne doit pas s'arrêter après avoir trouvé une seule de ces formes.

Voici l'algorithme de flexion d'un mot simple :

```
flechir_mot_simple(mot, code, formes_désirées)
début
    FF[mot] ← ∅ ;              //Ensemble de formes fléchies du mot avec leurs traits morphologiques
    lire_transducteur(code) ;  //Si le transducteur déjà lu pour un autre mot, retrouver sa structure.
                               //Sinon, créer sa structure et explorer à partir d'un fichier fsa.
    pour chaque ((terminaison,traits) ∈ CHEMIN[1])
        pour chaque (f ∈ formes_désirées)
            si(traits équivalents à f)
                forme_fléchie ← produire_forme(mot, terminaison) ;
                FF[mot] ← FF[mot] ∪ (forme_fléchie : f) ;
            fin si ;
        fin pour ;
    fin pour ;
fin
```

Fig.13. Algorithme de flexion de mots simples

Pour la clarté de la description, nous présentons l'algorithme *produire_forme* de production d'une forme fléchie en langage C :

```
char *produire_forme(char *mot, char *term) {

int stack_cnt, mot_cnt, n, i;
char stack[50];
static char mot_flechi[512];

strcpy(mot_flechi,mot);
mot_cnt = strlen(mot_flechi) - 1;
stack_cnt = 0;
while (*term != '\0') {
        n = 0;
        while ( ('1' <= *term) && (*term <= '9') && (*term != '\0') ) {
                n = 10*n + (int)(*term - '0');
                term++;
        }
        for (i=1; i<=n; i++)
                stack[stack_cnt++] = mot_flechi[mot_cnt--];

        switch (*term) {
                case 'L': stack[stack_cnt++] = mot_flechi[mot_cnt--]; break;
                case 'C': mot_flechi[++mot_cnt] = stack[--stack_cnt]; break;
                case 'R': stack_cnt--; break;
                default: mot_flechi[++mot_cnt] = *term;
        }
        term++;
}
mot_flechi[++mot_cnt] = '\0';
return(mot_flechi);
}
```

Fig.14. Algorithme de production d'une forme fléchie d'un mot simple

Complexité : Soit *long_mot* la longueur du mot à fléchir, et *long_term* la longueur de la terminaison à appliquer au mot. Une terminaison décrit la production d'une forme fléchie par un passage de zéro, une ou plusieurs positions à gauche, et ensuite par rajout d'un suffixe. Ce suffixe est soit indépendant du lemme, soit obtenu à partir du suffixe du lemme par une combinaison de trois opérations – le passage à droite, la recopie d'une lettre, l'insertion d'une lettre. En principe, la production du suffixe se fait sans retour en arrière. Donc la complexité de l'algorithme *produire_forme* est proportionnelle à la longueur du suffixe de la forme fléchie, dans le pire des cas à la longueur de la forme fléchie entière : *O(long_forme_fléchie)*.

L'algorithme *flechir_mot_simple* nécessite la lecture et l'exploration du transducteur de flexion (section précédente), mais ceci se fait une seule fois pour chaque transducteur apparaissant dans le DELAC à fléchir. C'est pourquoi la complexité de *flechir_mot_simple* ne dépend pas directement de celle de *lire_transducteur*. Elle est proportionnelle à :

$$nb\_formes * nb\_formes\_à\_produire * (\sum_{(f : forme\_à\_produire)} longueur(f)) \leq$$

$$\leq nb\_formes * nb\_formes\_à\_produire^2 * max\_long\_formes, \text{ où}$$

- *nb_formes* est, comme dans la section précédente, le nombre de toutes les formes fléchies du mot,

- *max_long_formes* est la longueur maximale de toutes les formes fléchies à produire ;
- *nb_formes_à_produire* est le nombre de toutes les formes fléchies du composant simple à produire pour la flexion du mot composé courant ; dans le pire des cas toutes les formes existantes sont à produire, donc la complexité est de :

  $O(nb\_formes^3 * max\_long\_formes)$

Dans le cas des trois langues analysées, cette complexité peut atteindre *O(max_long_formes)* avec la constante multiplicative égale à 8, 64, et 343 000 pour l'anglais, pour le français, et pour le polonais respectivement.

4.5.3 Flexion des mots composés

La flexion d'un dictionnaire DELAC comporte trois étapes.

1) La lecture du fichier de flexion (voir section 4.3) et la création d'une structure de données avec les types de flexion et les traits morphologiques. Voici l'algorithme et la structure créée pour le polonais :

```
lire_fichier_flexion(langue)
début
        pour chaque <ligne du fichier de flexion>
                <enregistrer le code du type de flexion> ;
                <enregistrer les codes des traits morphologiques> ;
         fin pour ;
fin
```

Fig.15. Algorithme de lecture de fichier de flexion

Complexité : *O(nb_type_flexion+nb_traits_morpho)*, où

- nb_type_flexion est le nombre de type de flexion : 4 pour le français, 4 pour l'anglais, et 6 pour le polonais,
- nb_traits_morpho, nombre de différents traits morphologiques : 18 pour le français, 13 pour l'anglais, et 31 pour le polonais.

Types de flexion	Traits morphologiques										
N	**s**	**p**									
R	**o**	**z**	**r**	**f**	**n**						
A	**M**	**D**	**C**	**B**	**I**	**L**	**W**				
O	**1**	**2**	**3**								
E	**F**	**H**	**P**	**S**	**U**	**J**	**K**	**Q**	**T**	**Z**	**G**
Y	**∅**	**c**	**u**								

Fig.16. Structure de codes pour les types de flexion et les traits morphologiques polonais

2) Pour chaque fichier-dictionnaire, la lecture de son entête. Voici l'algorithme et trois structures obtenues pour le fichier de l'exemple [267] du polonais :

```
lire_entete()
début
    <lire la première ligne, compter le nombre de constituants, marquer les numéros
                des constituants caractéristiques – remplir structure TETE> ;
    pour chaque <ligne de l'entête>
        <lire la forme à produire> ;
        <marquer les constituants variables et les formes qu'ils prennent – remplir
        structure CREATION> ;
    fin pour ;
    pour chaque <ligne lue>
        <attacher dans la structure FORMES le numéro de la ligne à la forme
        qu'elle décrit>
fin
```

Fig.17. Algorithme de lecture de l'entête d'un fichier-dictionnaire

Complexité : elle est proportionnelle à la longueur de l'entête en termes du nombre de formes décrites (donc des lignes de l'entête) et du nombre de constituants : *O(nb_lignes_entête * nb_const).*

Nombre de constituants	Constituants caractéristiques
2	1

Structure TETE

Forme	Lignes (qui décrivent comment obtenir cette forme)	
s	1	2
p	3	4

Structure FORMES

Ligne	Forme	Nombre de constituants	Constituants fléchis	
1	s	2	1 s	
2	s	2	1 s	2 p
3	p	2	1 p	
4	p	2	1 p	2 p

Structure CREATION

Fig.18. Structures TETE, FORMES, et CREATION de description de flexion irrégulière

3) Pour chaque mot composé, la production de toutes ses formes fléchies en deux phases :

   - La détermination de la liste de toutes les formes fléchies qui doivent être produites, selon les informations flexionnelles apparaissant dans l'entrée. Reprenons l'un des exemples [267] du polonais :

   [275] bohater(bohater.N1110:Mos) dnia(dzień.N141:Drs),N+NNgén:Mos/+N+C+f
   (héros du jour)

   Nous devrions trouver toutes les combinaisons des 2 nombres, 7 cas, et 2 genres (masculin humain *o* de la forme de base et le féminin marqué *+f*), soit les 28 formes suivantes : *Mos, Mop, Dos, Dop, Cos, Cop, Bos, Bop, Ios, Iop, Los, Lop, Wos, Wop, Mfs, Mfp, Dfs, Dfp, Cfs, Cfp, Bfs, Bfp, Ifs, Ifp, Lfs, Lfp, Wfs, Wfp.*

   Voici l'algorithme :

```
formes_a_produire()
début
    F ← {traits_morpho_du_lemme} ;
    F'← ∅ ;
    pour chaque <P = possibilité de flexion marquée dans l'entrée>
        si (P est un type de flexion)
            pour chaque (f∈ F)
                pour chaque(p∈ P)
                    g ← <f où le trait du type P a été remplacé par p> ;
                    F'← F'∪ g ;
                fin pour ;
            fin pour ;
        sinon            //P est un seul trait morphologique, e.g. +f
            pour chaque (f∈ F)
                g ← <f où le trait du même type que P a été remplacé par P> ;
                F' ← F'∪ g ;
            fin pour ;
        F ← F' ;                //F = ensemble de formes à produire
    fin pour ;
fin
```

Fig.19. Algorithme de génération de traits morphologiques des formes à produire

Complexité : Dans le pire des cas, où toutes les combinaisons de traits de différents types de flexion sont possibles, la complexité est proportionnelle au produit des tailles de tous les types de flexion :

$\Pi_{(t \in T)}$ *Nb_Traits_Morpho(t)* , où

- T est l'ensemble de types de flexion : {*N,R,P,T*} pour le français, {*N,O,T,Y*} pour l'anglais, {*N,R,A,O,E,Y*} pour le polonais,
- *Nb_Traits_Morpho(t)* est le nombre de traits morphologiques dans type t ; pour le français il est égal à 2, 2, 3, 11 pour les 4 types respectivement, pour l'anglais il est égal à 2, 3, 5, et 3 respectivement, et pour le polonais à 2, 5, 7, 3, 11, et 3.

La plupart de mots composés sont concernés uniquement par la flexion en nombre et genre en français, en nombre en anglais, et en nombre, genre et cas en polonais. Ainsi, la complexité est de l'ordre *O(1)* avec la constante multiplicative égale à 4, 2 et 70 pour le français, pour l'anglais, et pour le polonais respectivement.

- La production des formes fléchies dont les traits flexionnels ont été déterminés dans la phase précédente. Ici, nous devons tenir compte de l'entête du fichier pour savoir réaliser la flexion irrégulière. L'exemple ci-dessus, *bohater dnia*, se trouve dans un fichier commençant par l'entête suivante :

```
[276]  #+/-
       #s:s/-
       #s:s/p
       #p:p/-
       #p:p/p
```

Cette information nous sert à déterminer la façon de produire toutes les formes contenant le trait *s* ou *p*. Pour cela, chaque ligne permanente est complétée, sur les

positions où se trouvent les constituant caractéristique, par d'autres traits morphologiques. Par exemple, pour obtenir les deux variantes de la forme *Lop* (locatif masculin humain pluriel) nous cherchons toutes le lignes qui décrivent le pluriel, ici lignes 4 et 5, et nous les complétons par les traits *Lo* :

#Lop:Lop/-
#Lop:Lop/p

Les traits libres, i.e. non spécifiés, du deuxième composant - le cas, le genre et le nombre dans la première variante, le cas et le genre dans la deuxième - seront ceux que ce composant prend dans la forme canonique du mot composé. Donc pour *bohater dnia* on obtiendra

#Lop:Lop/Drs
#Lop:Lop/Drp

De la même façon, toute autre forme fléchie s'obtient de deux façons. Nous avons donc l'ensemble de schémas de flexion comme suit :

#Mos:Mos/Drs	*#Mop:Mop/Drs*	*#Mfs:Mfs/Drs*	*#Mfp:Lfp/Drs*
#Mos:Mos/Drp	*#Mop:Mop/Drp*	*#Mfs:Mfs/Drp*	*#Mfp:Lfp/Drp*
#Dos:Dos/Drs	*#Dop:Dop/Drs*	*#Dfs:Dfs/Drs*	*#Dfp:Dfp/Drs*
#Dos:Dos/Drp	*#Dop:Dop/Drp*	*#Dfs:Dfs/Drp*	*#Dfp:Dfp/Drp*
#Cos:Cos/Drs	*#Cop:Cop/Drs*	*#Cfs:Cfs/Drs*	*#Cfp:Cfp/Drs*
#Cos:Cos/Drp	*#Cop:Cop/Drp*	*#Cfs:Cfs/Drp*	*#Cfp:Cfp/Drp*
#Bos:Bos/Drs	*#Bop:Bop/Drs*	*#Bfs:Bfs/Drs*	*#Bfp:Bfp/Drs*
#Bos:Bos/Drp	*#Bop:Bop/Drp*	*#Bfs:Bfs/Drp*	*#Bfp:Bfp/Drp*
#Ios:Ios/Drs	*#Iop:Iop/Drs*	*#Ifs:Ifs/Drs*	*#Ifp:Ifp/Drs*
#Ios:Ios/Drp	*#Iop:Iop/Drp*	*#Ifs:Ifs/Drp*	*#Ifp:Ifp/Drp*
#Los:Los/Drs	*#Lop:Lop/Drs*	*#Lfs:Lfs/Drs*	*#Lfp:Lfp/Drs*
#Los:Los/Drp	*#Lop:Lop/Drp*	*#Lfs:Lfs/Drp*	*#Lfp:Lfp/Drp*
#Wos:Wos/Drs	*#Wop:Wop/Drs*	*#Wfs:Wfs/Drs*	*#Wfp:Wfp/Drs*
#Wos:Wos/Drp	*#Wop:Wop/Drp*	*#Wfs:Wfs/Drp*	*#Wfp:Wfp/Drp*

Voici l'algorithme :

```
créer_schemas()
début
    pour chaque (f∈ F)                    //F est le résultat de l'algorithme précédent (Fig.19)
        PROD[f] ← ∅ ;                     //Ensemble de schémas de création de f
        prod ← <schéma de flexion régulière>;          //e.g. f:f/Drs pour « bohater dnia »
        pour chaque (ligne dans la structure CREATION qui concerne f)
            <actualiser et éventuellement dédoubler prod>     //e.g. f:f/Drs ; f: f/Drp
        fin pour ;
        PROD[f] ← PROD[f] ∪ prod ;
    fin pour ;
fin
```

Fig.20. Algorithme de création de schémas de flexion

Complexité : elle dépend du nombre de constituants et de formes fléchies qui doivent être produites de façon irrégulière ; ce nombre est limité par le nombre de lignes dans l'entête du fichier-dictionnaire *nb_lignes_entête*. Dans le pire des cas, c'est-à-dire si chaque forme est créée d'une façon irrégulière, la complexité ne dépasse pas :

*O(nb_formes_mc * nb_const * nb_lignes_entête).*

Ayant créé tous les schémas de production de formes fléchies, nous pouvons déterminer, pour chaque constituant simple, toutes ses formes fléchies nécessaires pour la flexion du mots composés. Pour l'exemple [275] il est nécessaire de créer les 28 formes fléchies du mot *bohater,* mais seulement deux formes fléchies, *Drs* et *Drp*, du mot *dzień*. Leur production se fait par le module *flechir_mot_simple* décrit plus haut. Nous combinons ensuite les formes fléchies selon les schémas de flexion et nous obtenons les formes fléchies du composé. Pour l'exemple [275] nous obtenons deux variantes de chaque forme, e.g. pour le locatif masculin humain pluriel ce sont : *bohaterach dnia* et *bohaterach dni*. L'algorithme est comme suit :

```
produire_formes_fléchies()
début
    pour chaque (constituant c du mot composé mc à fléchir)
        TFF[c] ← <ensemble de traits de toutes les formes fléchies de c mentionnée dans PROD> ;
        FF[c] ← flechir_mot_simple(c, code_flex[c], TFF[c]) ;
              //FF[c] obtient les formes fléchies de c nécessaires pour l'ensemble de flexion de mc
    fin pour ;
    pour chaque (p∈ PROD)
        <générer une forme fléchie de mc selon le schéma p> ;     //e.g. bohaterach dnia,N:Lop
    fin pour ;
fin ;
```

Fig.21. Algorithme de production de formes fléchies d'un mot composé

Complexité : La première partie de l'algorithme, celle qui détermine les formes fléchies d'un constituant et les produit, a la complexité de l'ordre :

$O(\sum_{(c : constituant)} (nb\_schémas * nb\_formes(c)^3 * max\_long\_formes(c))$, où

- *nb_formes(c)* est le nombre de toutes les formes fléchies du constituant *c,*
- *max_long_formes(c)* est la longueur maximale de toutes les formes fléchies,
- *nb_schémas* est le nombre de schémas dans l'ensemble *PROD* (Fig.20) ; au pire des cas, i.e. quand toutes les formes sont créées d'une façon irrégulière, le nombre de schémas est égal au nombre de lignes de l'entête multiplié par le nombre de formes du mot composé ; alors la complexité ci-dessus est égale à :

$O(nb\_lignes\_entête * nb\_formes\_mc * \sum_{(c : constituant)} (nb\_formes(c)^3 * max\_long\_formes(c)) <$

$O(nb\_lignes\_entête * nb\_formes\_mc * nb\_const * max\_nb\_formes\ simples^3 * max\_long\_formes\_simples)$, où

- *max_long_formes_simples* est la longueur maximale de formes fléchies d'un composant simple,
- *max_nb_formes_simples* est le nombre maximal de formes fléchies d'un composant, qui pour la plupart des mots composés dans les trois langues considérées est le même que le nombre de toutes les formes

fléchies possibles d'un mot composé, *nb_formes_mc*, d'où la complexité :

$O(nb\_lignes\_entête * nb\_const * nb\_formes\_mc^4 * max\_long\_formes\_simples)$

La deuxième partie de l'algorithme, celle de génération de formes fléchies selon les schémas nécessite la recherche des formes adéquates des composants simples dans leurs ensembles *FF*. Pour chaque schéma dans *PROD* et pour chaque composant *c* il faut comparer les traits désirés de *c* à ceux de chaque forme dans *FF(c)*, au pire ceci nécessite *nb_formes(c)* de comparaison. La complexité de la dernière boucle est donc de l'ordre :

$O(nb\_schémas * \sum_{(c : constituant)} nb\_formes(c)) <$

$O(nb\_schémas * nb\_const * max\_nb\_formes\_simples)$, où

Dans le pire des cas, comme plus haut, *nb_schémas* est égal à *nb_lignes_entête * nb_formes_mc* et la complexité de la dernière boucle de l'algorithme est de :

$O(nb\_lignes\_entête * nb\_formes\_mc^2 * nb\_const)$

En conséquence, la complexité de l'algorithme de génération de formes fléchies d'un mot composé est de l'ordre :

$O(nb\_lignes\_entête * nb\_const * nb\_formes\_mc^4 * max\_long\_formes\_simples + nb\_lignes\_entête * nb\_formes\_mc^2 * nb\_const) =$

$O(nb\_const * nb\_lignes\_entête * nb\_formes\_mc^2 * (nb\_formes\_mc^2 * max\_long\_formes\_simples + 1) =$

$O(nb\_const * nb\_lignes\_entête * nb\_formes\_mc^4 * max\_long\_formes\_simples)$

Et voici l'algorithme entier de flexion d'un DELAC :

```
flechir_DELAC()
début
    lire_fichier_flexion(langue) ;
    début
        pour chaque <fichier_dictionnaire>
            lire_entete() ;
            pour chaque <mot composé>
                formes_a_produire() ;
                créer_schemas() ;
                produire_formes_fléchies() ;
            fin pour ;
        fin pour ;
    fin ;
fin
```

Fig.22. Algorithme de flexion d'un DELAC

4.5.4 Complexité

Nous avons mentionné que la lecture et l'exploration des transducteurs de flexion se fait une seule fois pour chaque transducteur. La complexité de cette opération peut donc être comptée

à part. En prenant en compte les complexités des algorithmes contenus dans *flechir_DELAC,* nous obtenons la complexité de *flechir_DELAC* de l'ordre :

$$O(\Sigma_{(t : \text{transducteur de flexion})} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) + \quad //lire\_transducteur$$
$$nb\_type\_flexion + nb\_traits\_morpho + \quad // lire\_fichier\_flexion$$
$$\Sigma_{(fd : \text{fichier-dictionnaire})} (\quad //pour\ chaque\ <fichier\_dictionnaire>$$
$$nb\_lignes\_entête_{fd} * nb\_const_{fd} + \quad // lire\_entete$$
$$\Sigma_{(mc : \text{mot composé dans fd})} (\quad // pour\ chaque\ <mot\ composé>$$
$$\Pi_{(t \in T)} Nb\_Traits\_Morpho(t) + \quad // formes\_a\_produire$$
$$nb\_formes\_mc * nb\_const_{fd} * nb\_lignes\_entête_{fd} + \quad // créer\_schemas$$
$$nb\_const_{fd} * nb\_lignes\_entête_{fd} * nb\_formes\_mc^4 * max\_long\_formes\_simples)$$
$$// produire\_formes\_fléchies$$
$$))$$

La lecture de l'entête de flexion peut être négligée car elle est constante et peu coûteuse pour chaque langue considérée. Par simplifications nous obtenons la complexité suivante :

$$O(\Sigma_{(t : \text{transducteur de flexion})} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) +$$
$$\Sigma_{(fd : \text{fichier-dictionnaire})} ($$
$$nb\_lignes\_entête_{fd} * nb\_const_{fd} +$$
$$nb\_mots\_composés_{fd} * ($$
$$\Pi_{(t \in T)} Nb\_Traits\_Morpho(t) +$$
$$nb\_formes\_mc * nb\_const_{fd} * nb\_lignes\_entête_{fd} *$$
$$(1 + nb\_formes\_mc^3 * max\_long\_formes\_simples)))) =$$

$$O(\Sigma_{(t : \text{transducteur de flexion})} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) +$$
$$\Sigma_{(fd : \text{fichier-dictionnaire})} ($$
$$nb\_lignes\_entête_{fd} * nb\_const_{fd} +$$
$$nb\_mots\_composés_{fd} * ($$
$$\Pi_{(t \in T)} Nb\_Traits\_Morpho(t) +$$
$$nb\_formes\_mc^4 * nb\_const_{fd} * nb\_lignes\_entête_{fd} * max\_long\_formes\_simples))) =$$

$$O(\Sigma_{(t : \text{transducteur de flexion})} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) +$$
$$\Sigma_{(fd : \text{fichier-dictionnaire})} (nb\_lignes\_entête_{fd} * nb\_const_{fd}) +$$
$$nb\_mots\_composés * \Pi_{(t \in T)} Nb\_Traits\_Morpho(t) +$$
$$nb\_formes\_mc^4 * max\_long\_formes\_simples *$$
$$(\Sigma_{(fd : \text{fichier-dictionnaire})} (nb\_mots\_composés_{fd} * nb\_const_{fd} * nb\_lignes\_entête_{fd})))$$

Rappelons que :

- nb_trans_t , $nb\_états_t$, $nb\_chemins_t$, $max\_long\_chemins_t$ sont respectivement les nombres de transitions, d'états, de chemins acceptants et la longueur maximale des chemins acceptants du transducteur *t,*
- $nb\_lignes\_entête_{fd}$, $nb\_mots\_composés_{fd}$, nb_const_{fd} sont respectivement le nombre de lignes de l'entête d'un fichier-dictionnaire, le nombre de composés y contenus, et le nombre de constituants dans un composé du fichier-dictionnaire,
- *Nb_Traits_Morpho(t)* est le nombre de traits morphologiques dans le type de flexion *t*,
- *nb_mots_composés* est le nombre de tous les composés dans le DELAC,
- *nb_formes_mc* est le nombre de formes fléchies possibles d'un mot composé,

– *max_long_formes_simples* est la longueur maximale de formes fléchies des composants simples des mots composés.

Pour les trois langues considérées, les mots composés se fléchissent, dans la grande majorité, de façon régulière et sont binaires ou ternaires. Pour l'anglais seulement 0,2% du DELAC sont des mots à flexion irrégulière (voir Tab.8). Le coût de flexion de ces mots irréguliers devient négligeable par rapport au reste. Nous pouvons donc admettre la complexité suivante :

$$O(\sum_{(t : transducteur\ de\ flexion)} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) + nb\_fichiers\_dicos + nb\_mots\_ composés * \prod_{(t \in T)} Nb\_Traits\_Morpho(t) + nb\_formes\_mc^4 * max\_long\_formes\_simples * nb\_mots\_ composés)) =$$

$$O(\sum_{(t : transducteur\ de\ flexion)} (nb\_trans_t + nb\_états_t + nb\_chemins_t * max\_long\_chemins_t) + nb\_fichiers\_dicos + nb\_mots\_ composés * (\prod_{(t \in T)} Nb\_Traits\_Morpho(t) + nb\_formes\_mc^4 * max\_long\_formes\_simples))$$, où

– *nb_fichiers_dico* est le nombre de fichiers-dictionnaires.

En admettant les mêmes limites pour les transducteurs de flexion que pour Fig.11, pour un DELAC anglais nous obtenons la complexité :

$$O(nb\_transd + nb\_fichiers\_dicos + nb\_mots\_ composé * max\_long\_formes\_simples)$$, où

– *nb_transd* est le nombre de tous les transducteurs de flexion nécessaires pour la flexion de tous les composants simples du DELAC.

Le nombre de fichiers-dictionnaires étant négligeable (e.g. 24 pour le DELAC Anglais – voir section 5.5) par rapport au nombre des mots composés, nous obtenons :

$$O(nb\_transd + nb\_mots\_ composé * max\_long\_formes)$$

avec les constantes multiplicatives (pour *nb_transd* et *nb_mots_ composé * max_long_formes* respectivement) égales à 18 et 16 dans le cas de l'anglais, 36 et 256 dans le cas du français, et 63 et 70^4 dans le cas du polonais.

La génération des formes fléchies des mots composés, telle qu'elle est décrite ci-dessus, est donc faite en temps linéaire en fonction du nombre de transducteurs de flexions et en fonction du produit du nombre de ces composés et de la longueur maximale d'une forme fléchie d'un composant simple.

Les nombres de transducteurs de flexion, nécessaires pour la flexion des DELAC des trois langues considérées, sont limités par les nombres de tous les transducteurs flexionnels apparaissant dans les DELAS. Pour le français cette limite est d'environs 500, pour l'anglais de 380, pour le polonais elle d'environs 900.

Obtenu ainsi, le dictionnaire des mots composés fléchis DELACF, peut être compacté en un transducteur fini unique (voir section 2.6). Nous rappelons que ceci se fait par un algorithme linéaire en fonction du nombre de formes fléchies contenues dans le DELACF. Le transducteur obtenu est ensuite minimisé par un programme linéaire en fonction du nombre d'états. La consultation de dictionnaire minimal résultant est, elle aussi, linéaire en fonction de la longueur de l'entrée recherchée.

4.6 Conclusion

Résumons les règles qui régissent le classement des mots composés selon leurs modèles de flexion :

Règle 1. Si dans l'entête d'un fichier-dictionnaire il n'existe aucune ligne concernant un type de flexion, cette flexion est régulière.

Règle 2. Si l'entête contient une ligne décrivant une flexion irrégulière, la flexion régulière correspondante est bloquée, sauf si elle aussi est donnée explicitement dans l'entête.

Règle 3. Une description concernant toutes les formes d'un certain type de flexion peut être abrégée en utilisant le code de ce type de flexion.

La méthode de la description des mots composés pour la flexion automatique, présentée ci-dessus, a été élaborée suite à l'étude des trois langues : le français, l'anglais et le polonais. Cette dernière, pour laquelle nous avons recensé environs 1 100 noms composés, possède une flexion riche et les exceptions dans la flexion des composés y sont plus répandues qu'en français et en anglais. Grâce à cela nous avons pu proposer une méthode suffisamment puissante pour être utilisable en d'autres langues (son adéquation à l'allemand et à des langues agglutinantes reste à vérifier).

Il existe toujours plusieurs façons différentes de distribuer les mots composés dans des fichiers-dictionnaires. On peut effectuer des regroupements différents et/ou modifier les entêtes tout en produisant les mêmes résultats : les formes fléchies obtenues sont les mêmes pour tous les composés. Par exemple, en français la classe des *NomNom* a été décrite comme irrégulière toute entière avec la tête comportant le premier élément et les deux éléments subissant la flexion (exemple [249]). Or, on pourrait aussi bien dire qu'il y a deux classes différentes des *NomNom*, l'une régulière avec les deux noms caractéristiques :

[277] *#+/+*
armoire(armoire.N21:fs)-penderie(pendrie.N21:fs),N+NN:fs/+N
moissonneuse(moissonneuse.N21:fs)-lieuse(lieuse.N21:fs),N+NN:fs/+N
...

et l'autre irrégulière avec seul le premier nom caractéristique et le deuxième non caractéristique mais subissant la flexion :

[278] *#+/-*
#p:p/p
allocation(allocation.N21:fs)-chômage(chômage.N1:ms),N+NN:fs/+N
bateau(bateau.N3:ms)-mouche(mouche.N21:fs),N+NN:ms/+N
coton(coton.N1:ms)-tige(tige.N21:fs),N+NN:fs/+N

...

De la même façon, l'entête de l'exemple [273] du polonais contenant des numéraux peut être remplacée par :

[279] *#+/-*
#n:n/g
#g:g/g
#d:d/d
#a:a/g
#i:i/i
#l:l/l
#v:v/g
czterdziestu(czterdzieści.NU18:Dop)
rozbójników(rozbójnik.N1111:Dop),N+NumN:Dop/+C-n-v (quarente voleurs)
...

où l'on dit que seul le premier composant est caractéristique. Le nominatif (ligne 2), l'accusatif (ligne 5) et le vocatif (ligne 8) exigent la mise au génitif du deuxième élément. Les lignes 3-4 et 6-7 signifient que pour tous les autres cas, les deux éléments s'accordent. Cette information est obligatoire, sans elle on ne permettrait de fléchir que le premier composant.

La description de la flexion doit être la plus commode possible pour le lexicographe. Dans l'exemple [273] nous avons choisi l'entête la plus courte.

Notre algorithme de flexion automatique ne tient pas compte des composés décrits sous forme d'automates ou transducteurs finis. Pour ceci il faudrait que dans un tel automate les constituants caractéristiques soient accompagnés de leurs étiquettes grammaticales comme dans un DELAC. Un algorithme approprié « fléchirait » alors cet automate en produisant un autre automate qui contienne toutes les formes fléchies. Le problème majeur serait le risque de l'explosion de la taille l'automate (même si elle est toujours plus petite ou égale à celle d'un DELACF correspondant, grâce à quelques factorisations), surtout pour les langues comme le polonais où par exemple les adjectifs possèdent 70 formes flexionnelles.

Chapitre 5 Construction d'un dictionnaire électronique des mots composés anglais

5.1 Introduction

La construction d'un dictionnaire électronique à large couverture est un travail à long terme, minutieux et seulement partiellement automatisable. Dans les sections suivantes nous discutons les problèmes auxquels nous nous sommes heurtée lors de la création d'un dictionnaire électronique des mots composés (DELAC/DELACF) de l'anglais.

5.2 Dictionnaires usuels et dictionnaires électroniques pour le traitement automatique du langage naturel

Le premier problème est lié aux ressources que nous utilisons pour dresser des listes des mots qui deviendront les entrées de notre dictionnaire électronique. Typiquement, les premiers exemples sont tirés des dictionnaires usuels qui sont censés contenir les unités les plus employées de la langue. M. Gross (1989a) et Vivès (1990) ont présenté une discussion sur les différences entre les dictionnaires usuels et électroniques où ils montrent l'impossibilité du passage automatique de l'un à l'autre. Nous allons exemplifier cette problématique pour les mots composés anglais.

Le recensement des composés à partir d'un dictionnaire usuel n'est pas automatisable pour trois raisons :

1) Nous ne savons pas quels éléments d'une entrée d'un dictionnaire usuel ont le **statut d'un mot composé**. Même si une police spéciale est employée pour mettre en relief certaines séquences, l'emploi d'une telle police est ambigu. Prenons un extrait d'une entrée du *Oxford Advanced Learner's Dictionary of Current English* (OALDCE 1989).

> **pot** /pot/ *n* **1** [C] **(a)** round vessel made of earthenware, metal, etc for cooking things in: *pots and pans* ¤ *a chicken ready for the pot.* **(b)** (esp in compounds) any of various types of vessel made for a particular purpose: *a teapot* ¤ *a coffee-pot* ¤ *a flowerpot* ¤ *a chamber-pot* ¤ *a lobster-pot.* **(c)** amount contained in a pot: *They have eaten a whole pot of jam!* ¤ *Bring me another pot of coffee.* **2** [C esp *pl*] (*infml*) large sum; a lot of money: *making pots of money.*

   Les séquences en italique peuvent être soit des mots composés (*pots and pans, coffee-pot, chamber-pot, lobster-pot*), soit des expressions figées (*making pots of money*), soit des exemples d'utilisation du mot simple avec un sens particulier (*They have eaten a whole pot of jam!*). Une extraction automatique des séquences en italique doit donc être suivie d'un tri manuel.

2) Les dictionnaires usuels contiennent beaucoup d'**informations implicites** qu'un utilisateur humain est censé connaître ou pouvoir déduire. Ceci n'est pas acceptable pour les dictionnaires électroniques dont les utilisateurs directs sont des programmes informatiques - les données dont ils ont besoin doivent être codées d'une façon explicite, cohérente et conséquente. Voici d'autres entrées du même dictionnaire OALDCE (1989) :

deal[2] /di:l/ *n* (idm) **a good/great deal** (**of sth**) much; a lot: *spend a good deal of money* ¤ *take a great deal of trouble* ¤ *be a great deal better* ¤ *see sb a great deal*, ie often

deal[4] /di:l/ *n* (...) **3** (idm) (...) **a fair/square deal** fair treatment in a bargain: *We offer you a fair deal on furniture*, ie We sell it at fair prices.

Nous n'avons ici aucun indice fiable pour déterminer les **catégories** des mots composés en gras. Ils ont tous la même structure *a Adjectif deal*, c'est seulement grâce à la traduction et aux contextes que nous lisons *a good deal* et *a great deal* comme déterminants composés, tandis que *a fair deal* et *a square deal* comme noms composés.

La **flexion** des composés est aussi souvent implicite dans un dictionnaire usuel. Ceci pose un problème dans des exemples (toujours dans OALDCE 1989) comme

half-brother *n* brother with only one parent in common with another

où le constituant caractéristique *brother* possède deux variantes du pluriel - *brothers* ou *brethren* - en fonction de sa signification. Néanmoins, la mise au pluriel correcte du composé - *half-brothers* - n'est pas indiquée.

L'**existence des formes fléchies** est une autre propriété qu'il faut « deviner » dans un dictionnaire usuel. Par exemple, nous trouvons dans OALDCE (1989) les articles suivants :

the armed forces, **the armed services** a country's army, navy and air force

animal spirits natural enjoyment of life

Siamese twins twins born with their bodies joined together in some way

brothers in arms soldiers serving together, esp in wartime.

Ces 5 noms composés sont au pluriel et nous ne savons pas s'ils possèdent une forme du singulier. On pourrait considérer que là où l'entrée apparaît au pluriel la non-existence du singulier est sous-entendue. Ceci serait correct dans les cas de *armed forces, armed services* et *animal spirits* qui perdent leur sens idiomatique au singulier (i.e. deviennent des groupes nominaux libres). Mais les deux derniers exemples, *Siamese twins* et *brothers in arms*, admettent le singulier *Siamese twin* et *brother in arms*.

Considérons un autre extrait du OALDCE (1989)

four-in-hand *n* coach or carriage pulled by four horses and driven by one person

Cet article ne mentionne pas la formation du pluriel pour *four-in-hand* qui est pourtant particulière comme nous l'avons vu dans l'exemple [191].

3) Les dictionnaires usuels ne sont **pas exhaustifs,** ce qui est dû entre autres à des contraintes de prix. Quand aux mots composés, nous sommes convaincus que leur présence dans différents dictionnaires usuels n'est pas liée à leur fréquence d'apparition dans des textes. Les composés dont le sens est facilement déductible (pour un lecteur humain d'un certain niveau de culture) à partir de ses composants, tels *chemical process* (processus chimique), *collective responsibility* (responsabilité collective)*, market structure* (structure du marché)*, product development, communication agreement*, ne figurent pas dans les dictionnaires de référence comme NSOED (1996), HO (1994) et OALDCE (1989). Il s'agit pourtant de séquences fréquentes dans les textes, et dont la quantité est beaucoup plus importante que celle des séquences idiomatiques. Un travail considérable doit alors être effectué pour les rechercher par d'autres moyens qu'en étudiant ces dictionnaires (par exemple par l'extraction terminologique sur un corpus). Remarquons aussi que les dictionnaires qui

utilisent la technologie du CD-Rom, et qui ne sont donc plus bornés au niveau de la taille, ont plus d'entrées, mais contiennent toujours des informations morphologiques limitées et ne contiennent pas de composés dont le sens est jugé évident.

Il ne s'agit pas ici de critiquer les dictionnaires traditionnels. Les trois types d'incomplétude d'information qui ont lieu dans ces ouvrages sont, pour la plupart, justifiés par l'utilisateur à qui elles s'adressent - un lecteur humain. Nous voulons seulement montrer qu'en utilisant un dictionnaire usuel comme base pour un dictionnaire électronique des mots composé il faut s'attendre à un surplus considérable de travail manuel nécessaire pour vérifier et compléter de nombreuses entrées.

Le gros des mots composés qui se trouvent actuellement dans notre DELAC anglais provient des deux dictionnaires de l'anglais : *New Shorter Oxford English Dictionnary* (NSOED 1996) sur CD-Rom, et *Le dictionnaire Hachettes-Oxford anglais-français* (HO 1994). Le premier a été dépouillé par M. McCarthy-Hamani, ainsi que Michael Walsh et David Harte de l'Université de Dublin, et le deuxième par Katia Zellagui de l'Université de Besançon. Les composés y étaient recherchés « à la main » et recopiés dans des listes. Nous avons formaté ensuite ces listes pour obtenir des entrées du DELAC. Lors de ce travail nous avons constaté des différences remarquables quand à la couverture du lexique dans les deux dictionnaires. Près de 22 400 noms composés provenaient du NSOED (1996) et près de 21 000 du HO (1994). L'union (sans doublons) des deux listes compte 39 000 entrées, dont seulement 4 200 composés (11%) sont présents dans les deux ouvrages. Ceci prouve encore une fois que, dans la perspective de l'analyse automatique de grands corpus, un dictionnaire usuel ne peut pas être considéré comme représentatif du lexique d'une langue.

5.3 Recensement et description des formes lemmatisées

Nous avons mentionné que le DELAC anglais actuel a été créé à partir des listes de mots composés trouvés par des lexicographes dans des dictionnaires usuels. Ces listes contenaient plus ou moins d'informations morphologiques (le pluriel, les variantes, etc.), syntaxiques (la catégorie du composé, les catégories des composants) et distributionnelles (humains, collectif, etc.). Certaines n'en contenaient pas du tout. Les formats choisis par les auteurs pour leurs listes étaient variables et, de plus, la plupart des listes contenaient des incohérences avec le format admis. La mise en forme de ces données a consisté en la programmation de filtres textuels qui, en fonction du format, convertissaient les lignes des fichiers en des entrées DELAC (dans un premier temps sans étiquettes grammaticales pour les composants simples). Un certain nombre de lignes ont été rejetées par les algorithmes si leur format ne correspondait pas à celui qui avait été supposé. Il a fallu alors les corriger à la main et joindre au résultat. Mais certaines erreurs du format ne se laissaient pas découvrir lors du traitement automatique, et elles ne sont ressorties que lors de l'analyse manuelle ou lors des stades ultérieurs du traitement. A côté de ce problème de format des listes, nous avons rencontré des difficultés linguistiques dont nous donnons un aperçu.

5.3.1 Séparation des catégories

La notion de la catégorie grammaticale est loin d'être précise et les frontières entre différentes catégories sont floues. Par exemple, Quirk et al. (1972) ont proposé un critère pour distinguer l'adjectif du nom anglais. Selon eux, un adjectif est celui qui peut apparaître aussi bien en position attributive qu'en position prédicative. Ainsi, *head* peut être adjectif car il se déplace de sa position attributive dans

[280] head teacher

vers la position prédicative dans

[281] This teacher is head.

Ceci n'est pas le cas du

[282] bow window

car il serait incorrect de dire

*[283] *This window is bow.*

Bauer (1983, p. 228) argumente que cette distinction n'est pas pertinente pour certains aspects de la syntaxe. Il admet aussi que d'autres critères comme l'existence du comparatif et superlatif, possibilité de modification par *so* et *very*, etc. ne sont pas satisfaisants non plus car très peu d'adjectifs les respectent simultanément.

Aussi, Bauer (1983) en décrivant le phénomène de conversion (voir section 4.4) en anglais constate que c'est un processus libre dans le sens où tout lexème peut être soumis à la conversion vers toute classe « ouverte » (i.e. substantif, verbe, adjectif ou adverbe) si cette conversion est motivée. Dans ce contexte, les étiquettes grammaticales telles que définies pour les dictionnaires électroniques des mots simples et composés (chapitre 2), perdent leur sens. Nous les gardons pour les raisons de convention et pour les buts pragmatiques de l'organisation et maintenance des données. Ainsi, nous allons dire que la catégorie d'un mot composé est celle que ce mot a dans les emplois bien établis de la langue. Dans notre cas « emplois établis » voudra dire ceux qui sont décrits dans les dictionnaires usuels utilisés par les lexicographes.

Dans les listes de composés fournies à l'entrée des filtres informatiques, les catégories des composés n'avaient pas toujours été explicitement marquées. Ainsi, il était parfois nécessaire de reconsulter les dictionnaires desquels ils provenaient, car les essais pour deviner la catégorie d'une séquence en fonction de sa structure syntaxique échouaient trop souvent pour être satisfaisants. La difficulté majeure est venue des composés contenant des gérondifs. Les séquences comme

[284] financially rewarding, afore-coming, inward-looking (introverti)*, all-embracing* (global)

doivent être classées en tant qu'adjectifs grâce à l'occurrence d'un adverbe à la première position. Plus difficiles sont les exemples suivants qui ont tous la structure d'une phrase nominale :

[285] God-fearing (pieux)*, bone chilling, cancer-causing* (cancérigène)*, card-carrying* (militant)*, cost-saving, hair(-)raising* (à vous faire dresser les cheveux sur la tête)*, free standing* (indépendant)*, easy-going* (accommodant), *double(-)acting* (à double effet)

[286] energy consuming (consommation d'énergie, qui consomme de l'énergie)*, life(-) giving* (vital, le don de vie)*, arse-licking* (flagornerie, flagorneur)*, air swallowing, air-conditioning* (climatisation)

[287] second coming (second avènement)*, absolute blocking, action painting* (peinture gestuelle)*, architectural engineering,*

Ceux du dernier groupe sont sûrement des noms. Ceux du premier fonctionnent très bien comme adjectifs et semblent moins acceptables en tant que noms (*cost-saving =? the action of*

saving cost). Ceux du milieu peuvent être comptés aussi bien parmi les adjectifs que parmi les noms.

5.3.2 Elimination des doublons

Comme les mots composés étaient recensés par plusieurs auteurs il était naturel que les différentes listes contiennent des doublons. L'élimination automatique des doublons est une chose facile si l'on ne tient compte que de la forme graphique du mot. Mais, un doute apparaît là où différentes occurrences de la même séquence sont accompagnées de différentes informations syntacticto-sémantico-distributionnelles, car ceci peut signifier que la séquence possède plusieurs sens qu'il convient de séparer. Entre autres, pour la description du comportement syntaxique des noms il est important de repérer ceux parmi les non-humains qui peuvent avoir des emplois en tant qu'humains. Par exemple, l'extrait suivant du NSOED (1996) :

> **number one** (a) oneself, one's own person and interests; (b) *nursery & euphem.* an act of urination; (c) *colloc* the finest quality, the best obtainable (freq. attrib.); **number ones** a best dress uniform worn esp. in the Navy

définit un mot à trois sens différents du point de vue distributionnel. Ce mot doit être dédoublé dans le DELAC. Au sens (a) ci-dessus correspond une entrée avec la marque *+Hum*, aux sens (b) et (c) une autre entrée sans cette marque, et *number ones* fait une troisième entrée à cause des restrictions sur le nombre (inexistence du singulier). Les sens (b) et (c) ne peuvent pas être séparés, car la différence entre eux ne peut pas être exprimée par l'ensemble d'étiquettes distributionnelles que nous avons choisies - *Hum* (humain)*, Anl* (animal)*, HumColl* (humain collectif)*, AnlColl* (animal collectif)*, Conc* (concret) et la marque vide.

Selon un autre extrait du NSOED (1996) :

> **left wing** *n. & a.phr.* **A** *n.phr.* (1) the division on the left side of an army or fleet in battle array. (2) *Football & Hockey etc.* (The position of) a player on the left side of the centre; the part of the field in which a left wing normally plays. (3) The radical or socialist section of a group or political party; the more liberal or progressive section of a right-wing or conservative group or political party.

Les premier et troisième articles ci-dessus sont représentés dans le DELAC avec la marque *+HumColl*, tandis qu'au deuxième correspondent deux entrées du DELAC - avec et sans la marque *+Hum.*

5.3.3 Marquage de la structure syntaxique et des composants caractéristiques

Dans le contexte de ce que nous avons dit dans la section 5.3.1, nous avons renoncé à l'idée de marquer la structure syntaxique de nos noms composés, car ceci impliquerait un travail difficile de levée d'ambiguïté des catégories des composants, souvent non pertinent. La connaissance de cette structure n'étant pas indispensable pour pouvoir créer le dictionnaire DELACF, nous nous sommes limitée à distinguer les composants caractéristiques (*C*) des non caractéristiques (*X*). Ainsi, dans notre notation l'étiquette grammaticale contenant par exemple la marque *N+XXC* signifie que le composé est un nom contenant trois constituants dont le troisième est caractéristique, par exemple *white-collar worker* (employé de bureau).

Nous rappelons (voir sections 2.2.3, section 3.2, et chapitre 4) que le marquage des constituants caractéristiques et de leurs irrégularités est crucial pour la flexion automatique des mots composés.

La règle générale de la mise au pluriel en anglais est celle de ne modifier que le dernier composant dans le cas des structures germaniques *Nom Nom* (*greenhouse effect -> greenhouse effects* (effet de serre)), *Adj Nom* (*golden wedding -> golden weddings* (noces d'or)), *Adj Nom Nom* (*real-time system -> real-time systems* (système à temps réel)) etc., et le composant précédant directement la préposition dans les structures prépositionnelles *Nom of Nom* (*man-of-war -> men of war* (navire de guerre)), *Adj Nom of Nom* (*electronic point of sale -> electronic points of sale*), etc. La détermination de la tête de ces noms composés serait donc automatisable si ce ne l'était pas pour deux raisons :

1) Les irrégularités de la flexion présentées dans 3.2-3.4 et dans 4.4.2. Leur repérage est difficile s'il n'a pas été fait tout de suite, c'est-à-dire lors de la consultation du dictionnaire source.

2) L'ambiguïté de structure dans les composés du type *Nom Prep Nom Nom.* Analysons deux exemples :

   [288] weapon of mass destruction

   [289] prisoner of war camp

   Dans le premier exemple la tête contient le premier nom. Dans le deuxième les trois premiers composants constituent un groupe nominal modifiant le dernier nom qui est caractéristique. Le tout pourrait donc être reformulé par *camp of prisoners of war*.

   La séparation de ces deux cas a été faite manuellement, le nombre des composés quaternaires étant relativement limité dans nos listes.

5.3.4 Existence du pluriel

En général, le seul type de flexion des noms en anglais est la flexion en nombre. La plupart de noms composés ont leurs formes lemmatisées au singulier et se mettent au pluriel par la modification du constituant caractéristique. Mais, pour certains exemples, l'existence du pluriel est problématique.

Si le constituant caractéristique est un nom simple sans pluriel, le composé entier n'en n'a pas non plus. C'est le cas des exemples comme (les noms simples sans pluriel sont soulignés) :

[290] street furniture (mobilier urbain), *cable news, excess luggage, pidgin English*[36]

De nombreux noms simples abstraits ont un pluriel qui n'est envisageable que dans des contextes très particuliers, comme *madness(?es)* (folie), *negligence(?s)* (négligence), *freedom(?s)* (liberté), *pink(?s)* (le rose) et les gérondifs non concrets : *engineering(?s)* (ingénierie), *screening(?s)* (projection, sélection), *thinking(?s)* (réflexion). Leur emploi au pluriel signifie d'habitude *different types of freedom* (différents types de liberté), *several cases of negligence* (plusieurs cas de négligence) etc. Ainsi, des mots composés dont les constituants caractéristiques sont de ce type ont la possibilité de flexion en nombre marquée dans le DELAC (+*N*) même si leurs pluriels sont problématiques :

[291] ?cancer screenings (dépistages du cancer), *?call queuings* (mises en file d'attente des appels), *?architectural engineerings, ?convergent thinkings* (pensées convergentes), *?canine madnesses* (rage), *?contributory negligences* (fautes de la victime entraînant un partage de la responsabilité), *?non-existences* (inexistances), *?car businesses, ?amaranth pinks*

[36] Mais il existe *the new Enlishes*.

Un autre cas concerne les noms composés qui semblent, par leur nature, difficiles au pluriel car ils désignent des objets a priori uniques. Souvent ce sont des noms propres ou des composés « communs » contenant des noms propres. Leurs constituants caractéristiques sont très souvent des noms simples communs possédant incontestablement le pluriel. Par exemple, *town, time, cape* ou *theorem* ne posent aucun problème du point de vue de la mise au pluriel, tandis que le pluriel est problématique pour :

[292] Cape Town, Greenwich Mean Time,Cape of Good Hope, Gauss' theorem

Néanmoins, les emplois particuliers comme ci-dessous sont admissibles :

[293] What he remembered from before the war and what he saw now were two different Cape Towns. (Ce dont il se souvenait d'avant la guerre et ce qu'il a vu maintenant étaient deux Cape Town différents.)

donc ces composés reçoivent dans le DELAC la marque d'existence du pluriel.

5.4 Etiquetage des composants simples

Une phase importante de construction du DELAC est de reconnaître les constituants simples de tous les composés. Ceci se fait en deux étapes. Premièrement, nous effectuons l'étiquetage des entrées du DELAC par le DELAF anglais entier. Alors, un composant non reconnu peut être soit une faute de frappe qu'il faut corriger, soit un mot correct manquant dans le DELAF. Dans ce dernier cas, le DELAS doit être complété et un nouveau DELAF reconstruit afin d'assurer la cohérence du système de dictionnaires.

La deuxième étape est celle où l'on fournit les codes flexionnels pour les constituants caractéristiques et, dans le cas d'une flexion irrégulière, pour d'autres constituants qui subissent la flexion. Ceci est nécessaire, comme nous l'avons vu dans le chapitre 4, pour la génération automatique du DELACF. Ici, nous effectuons l'étiquetage à l'aide d'un DELAF qui ne contient que les noms et les mots à catégorie hypothétique *X* (voir ci-dessus), car dans les structures nominales seuls les noms et les mots convertis en noms sont des éléments qui peuvent se fléchir. Ainsi, nous évitons de nombreuses ambiguïtés pour les constituants caractéristiques, surtout ceux qui peuvent être à la fois des verbes et des noms (*an answer, to answer*, etc.). Si un constituant est un nom ambigu, i.e. s'il reçoit plusieurs étiquettes avec la catégorie *N* mais avec des codes flexionnels différents (e.g. *brother - brothers, brethren*), la désambiguïsation se fait à la main.

Examinons quelques exemples de constituants que nous avons trouvés dans le DELAC et qui n'existaient pas dans notre DELAF d'origine.

5.4.1 Nouveaux mots simples communs

Lors de l'étiquetage du DELAC par le DELAF nous avons repéré près de 500 nouveaux mots simples communs, inexistants dans le DELAF. Pour la plupart, ceux-ci étaient des noms (*structurist, occupier, rhesus, lych, mistle, lamper, élan, goofer, cribble, diadem, zoot*, *viscosity,* etc.) ou des adjectifs (*nitty, arterial, salicylic, reeky, assertory, bally, coequate, cosmical, greaseproof, hypercomplex, structurist, underactive* etc.). Un certain nombre de nouveaux noms simples sont obtenus par l'effacement de séparateurs à l'intérieur d'un mot

composé (par exemple *blackwood, ironbark, wallbanger, waveband, aftermarket, servicewoman, leftwing*).[37]

Les nouveaux mots qui n'ont pas le statut de mots simples, i.e. qui ne fonctionnent qu'en tant que constituants de mots composés, ont été ajoutés au DELAS avec le code *XI* s'ils ne subissent pas la flexion, ou avec *X* suivi du nombre identique à celui pour les noms simples se fléchissant de la même façon. Par exemple, à partir des nouveaux mots (soulignés) dans les composés suivants :

[294] *walkie-talkie, hurdy-gurdy* (orgue de Barbarie)*, ne'er-do-well* (bon à rien)

nous avons ajouté les entrées suivantes au DELAS :

[295] *walkie.XI*
ne.XI
er.XI
talkie.X1
gurdy.X5

5.4.2 Noms propres

Le DELAS/F anglais ne contient pas de noms propres. Ainsi, l'orthographe des nombreux composés qui contiennent des noms propres ne peut être vérifié qu'à la main, e.g.

[296] *Saint Andrew's Cross, cedar of Lebanon* (cèdre du Liban)

De plus, le repérage de ces séquences selon le critère de la majuscule initiale dans le mot inconnu passe sous silence des cas où un nom propre s'écrit en minuscule quand il est dans des composés, par exemple (selon NSOED 1996) :

[297] *charley horse* (courbature)*, pitot tube* (*=Pitot tube* ; un tub utilisé pour mesurer la pression)*, trudgen stroke* (un style en natation)

D'autre part, certains composés ont un nom propre en position de tête qu'il faut donc fléchir, comme pour :

[298] *doubting Thomas(es)* (Saint Thomas)*, black-eyed Susan(s)*

Pour ceci nous avons été obligés d'introduire un petit dictionnaire constitué de tous les noms propres apparaissant dans les entrées du DELAC :

[299] *Thomas.N3*
Susan.N1 etc.

5.4.3 Emprunts

Un nombre important d'entrées de notre DELAC sont des composés étrangers qui ont été adoptés tels quels dans l'anglais. Comme le montrent les exemples [167]-[170](section 3.4.6), la mise au pluriel dans ces cas se fait soit comme dans la langue d'origine de l'emprunt (*nouveaux riches*), soit « à l'anglaise » (*beau ideals, beaus ideal*) soit plusieurs variantes sont admises (*operas buffa, opera buffas, opere buffe*), soit le pluriel reste égal au singulier (*petit bourgeois*).

[37] B. Courtois, lors du projet de recherche quotidienne de néologismes par Internet dans les journaux « Washington Post » et « New York Times », a aussi constaté cette tendance générale à souder des mots composés anglais pour en créer des mots simples.

Ainsi, les nouveaux mots simples à introduire dans le DELAS obtiennent des codes permettant la génération des pluriels respectifs :

[300] *nouveau.X6*
riche.X1
beau.X1
ideal.X1
opera.X25 (le même mot avec le code N1 existe déjà dans le DELAS)
buffa.X5
buffa.X25
petit.XI
bourgois.X2

Nous leur avons attribué la catégorie *X* car ils ne fonctionnent pas en anglais en dehors des composés (voir section 5.4.1).

5.4.4 Conversions et dérivations

Nous avons aussi trouvé des exemples (sections 3.4 et 3.5) de composés anglais contenant des mots simples qui, en principe, ne sont pas des noms mais qui sont modifiés lors de la mise au pluriel (*battle royals, johny-come-latelies, take-aways, has-beens* etc.) - voir les exemples [140]-[142], [159], [162], [191]. La première étape de l'étiquetage par le DELAF ne saura pas repérer ces cas car les mots simples en question sont des adjectifs (*royal*), adverbes (*lately*), prépositions (*away*), verbes (*been*) ou autres parties de discours existant dans le DELAF anglais. Ils seront par contre trouvés lors de la deuxième étape (qui n'utilise qu'un DELAF des noms et des *X*) et ensuite codés manuellement en tant qu'appartenant à la catégorie *X* comme ceci a été le cas dans les sections précédentes :

[301] *royal.X1*
lately.X5
away.X1
been.X1

La première étape de l'étiquetage du DELAC par le DELAF repère les cas de dérivations (voir exemples [190]) où certains constituants simples n'ont pas le statut de mots simples indépendants : *up-to-dateness, square-toedness, forty-niner, captain-generalcy* etc. Ils entrent dans le DELAS aussi avec le code *X* suivi d'un nombre indiquant la flexion :

[302] *dateness.X3*
toedness.X3
niner.X1
generalcy.X5

Finalement, les participes obtenus par conversion des noms en verbes, et non reconnus par le DELAF dans les exemples [192] : *better-humored, bowler-hatted, ill-omened,* etc. sont codés comme des *XI* car ils ne subissent pas la flexion :

[303] *humored.XI*
hatted.XI
omened.XI

5.5 Génération automatique du DELACF

Après l'étiquetage du DELAC par le DELAF, la correction des fautes de frappe, le codage des nouveaux composants, et la levée d'ambiguïtés pour les noms caractéristiques ambigus, comme décrit plus haut, nous obtenons un DELAC final prêt à être soumis à la flexion. Pour générer le dictionnaire DELACF à partir du DELAC, nous avons employé la méthode de flexion automatique des mots composés décrite dans le chapitre 4. Nous avons traité les noms composés séparément des autres catégories - adjectifs, adverbes, prépositions et conjonctions composées - qui ne subissent pas la flexion et donc leurs DELACF respectifs (voir les annexes C.2, C.3, C.4, C.5) sont obtenus par l'insertion d'un point après la virgule dans chaque entrée du DELAC, par exemple l'adverbe composé du DELAC :

[304] *as soon as possible,ADV*

a donné lieu à l'entrée suivante du DELACF :

[305] *as soon as possible,.ADV*

Les noms composés ont dû être divisés en classes selon la façon dont ils se fléchissent. Ainsi, nous avons 9 classes à flexion régulière (dans chaque nom de classe *C* représente un constituant caractéristique et *X* un composant non caractéristique, à ne pas confondre avec la catégorie *X*) : *XC, XXC, CXX, XXXC, CXXX, CXC, XCXX, CX, CC*. Chaque classe est contenue dans un sous-fichier DELAC avec l'entête d'une seule ligne, où *X* est représenté par un « - » et *C* par un « + ». Les premières entrées des sous-fichiers respectifs se trouvent dans l'annexe B.1.

Parmi les noms composés à flexion irrégulière nous avons distingué 15 classes. Des extraits sont donnés dans l'annexe B.2. Finalement, l'annexe C.1 contient les premières entrées du DELACF des noms.

5.6 Tailles et typologies du dictionnaire des mots composés anglais

Actuellement, le DELAC anglais contient 59 652 formes lemmatisées avec la distribution suivante :

Catégorie	**Nombre d'entrées**		**Exemples**
	DELAC	**DELACF**	
noms	51 651	102 000	*black market, no-man's land, center of gravity*
adverbes	4 047	4 047	*to some extent, open-mindedly, so-so, by force*
adjectifs	3 470	3 470	*light-hearted, Indo-african, absent without leave*
prépositions	361	361	*out of, in front of, immediately after, with relation to*
conjonctions	123	123	*as well as, as soon as, in case, immediately after*
TOTAL	59 652	110 001	

Tab.7 Nombres d'entrées du DELAC et du DELACF anglais

Les noms composés se divisent en 11 classes dont les tailles sont énumérées dans le tableau ci-dessous. La flexion automatique des entrées du DELAC a résulté en un DELACF de près de 110 000 formes fléchies, dont environ 102 000 formes de noms composés.

Classe	Nombre d'entrées	%	Exemples
XC	45 290	88 %	*black hole, aircraft carrier, light-heavyweight, bye-bye*
XXC	3 069	6 %	*atomic mass unit, binary coded decimal, students' union*
CXX	1 917	3,8 %	*way of life, ambassador-at-large, brother-in-law*
XXXC	348	0,7 %	*long range ballistic missile, blind man's buff*
CXXX	328	0,6 %	*court of common pleas, settlement out of court, lily of the valley*
CXC	311	0,6 %	*law and order, bits and pieces, part and parcel*
XCXX	79	0,2 %	*atomic order of magnitude, electronic point of sale*
CX	52	0,1 %	*passer-by, hanger-on, notary public*
CC	8	0,01 %	*man-servant, woman-writer, knight-templar*
Irréguliers	102	0,2 %	*court martial, head of state*
Autres	147	0,3 %	*separation of church and state, marshal of the royal air force, inspector of weights and measures*
TOTAL	51 651		

Tab.8 Classes typologiques des noms composés anglais

5.7 Conclusion

Le DELAC anglais que nous avons construit nécessite une augmentation importante du nombre d'entrées, pour obtenir une bonne couverture de la langue générale. Néanmoins, nous considérons que sa taille actuelle a été suffisante pour systématiser le recensement et la description des composés et pour repérer les difficultés majeures auxquelles se heurte leur traitement. Nous avons démontré que les phénomènes de la composition nominale et de la mise au pluriel des noms composés en anglais sont plus complexes que l'on croit d'habitude.

La création d'un dictionnaire électronique de mots composés est un travail long et méticuleux. Des programmes informatiques sont indispensables pour effectuer certaines conversions textuelles. Néanmoins, la plupart des phases de traitement ne sont que partiellement automatisables, car leurs résultats doivent toujours être vérifiés et corrigés par un lecteur humain. Il s'agit ici des entrées rejetées par les algorithmes comme incohérentes avec les formats admis, et que ces algorithmes ne peuvent donc pas traiter. Plus difficiles sont les cas

qui passent inaperçus par les programmes et peuvent seulement être relevés « par hasard » lors de l'application du dictionnaire électronique, ou par une relecture systématique de toutes les entrées par un lexicographe. Seule cette dernière démarche permet de fournir un dictionnaire électronique de haute qualité. Certaines parties de notre DELAC anglais ont encore besoin d'une telle relecture, d'autant plus que l'anglais n'est pas notre langue maternelle et nous n'avons pu que rarement consulter un anglophone en cas de doute.

Chapitre 6 Description des déterminants numéraux anglais par des outils à états finis

6.1 Introduction

Nous avons vu dans la section 2.5 que le système INTEX permet de créer et d'appliquer aux textes des automates finis et des expressions rationnelles, ou des transducteurs finis, où chaque chemin entre l'état initial et l'état final représente un mot, simple ou composé. Ces outils de description sont les plus adaptés aux mots qui ont un nombre important de variantes, qui ont un degré relativement élevé de productivité, ou bien qui utilisent un petit nombre de mots simples pour en créer un grand nombre de combinaisons. L'analyse combinatoire exhaustive de toutes les unités simples appartenant aux séquences à décrire est dans ces cas beaucoup plus facile à réaliser et à maintenir par des outils a états finis que par des listes textuelles.

Nous présentons l'exemple de déterminants numéraux cardinaux et ordinaux anglais, où l'intérêt d'emplois des outils à états finis est évident : les numéraux constituent un ensemble a priori infini construit à partir d'une classe fermée de seulement quelques dizaines de mots simples (la conjonction *and* et 67 numéraux simples : *zero, one, two, ..., ten, eleven, ..., twenty, thirty, forty, ..., ninety, hundred, thousand*, *million, billion, trillion, quadrillion, first, second, third, ..., tenth, ..., billionth, trillionth, quadrillionth*[38]). Pour cette raison, et aussi grâce au fait que les numéraux correspondent à des séquences contiguës dans des textes, ils peuvent être décrits indépendamment du reste des déterminants.

Nous décrivons les numéraux d'une part par des automates finis proprement dits (i.e. sans alphabet de sortie), et d'autre part par des transducteurs finis. Cette double description est motivée par deux méthodes différentes, disponibles dans le système Intex, d'application des graphes de composés à l'analyse lexicale d'un texte. La bibliothèque d'automates est plus simple et ainsi plus facilement extensible à des sous-ensembles plus larges de déterminants composés, ce que nous démontrons dans la section 6.6. La bibliothèque de transducteurs permet d'attacher aux séquences reconnues (en toutes lettres) leur valeur numérique, et ainsi d'exprimer certaines ambiguïtés et équivalences interprétatives présentes entre différents numéraux.

6.2 Déterminants numéraux cardinaux

Les déterminants numéraux peuvent s'écrire en chiffres (*25*), en toutes lettres (*twenty-five*), ou encore comme une combinaison de deux possibilités (*25.6 million*). Dans les nombres en chiffres (Fig.23), les fractions sont séparées des entiers par un point, et les groupes de trois chiffres avant le point peuvent être séparés soit par un blanc (*251 554.05*), soit par une virgule (*251,554.05*).

Les numéraux en toutes lettres demandent un ensemble beaucoup plus élevé de graphes. Pour les cardinaux de 0 à 10^{18}-1 nous en avons construit 11. Nous utilisons le mécanisme

[38] Nous avons arrêté notre description à 10^{18}-1 car les nombres plus grands sont d'habitude exprimés par des puissances de 10.

d'imbrication des graphes (boites grisées) qui donne aux automates finis une puissance descriptive équivalente à celle des RTN (recursive transition networks). Néanmoins, grâce au fait que nos imbrications ne sont jamais récursives, le langage que nous décrivons est régulier[39].

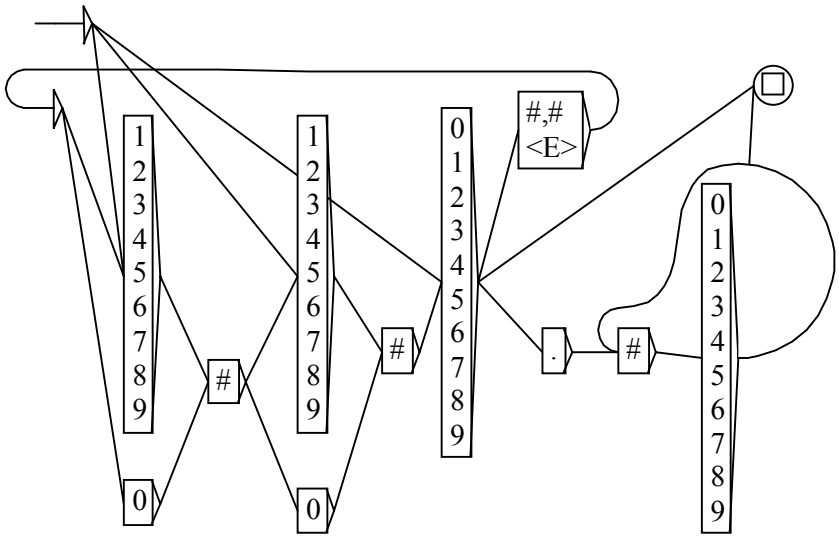


Fig.23. Graphe *DnumDig* des numéraux cardinaux en tous chiffres

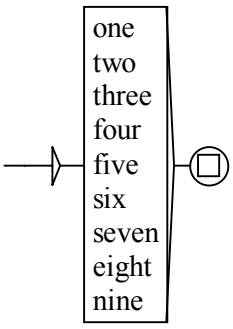


Fig.24. Graphe *DnumTxt1_9* : cardinaux de 1 à 9.

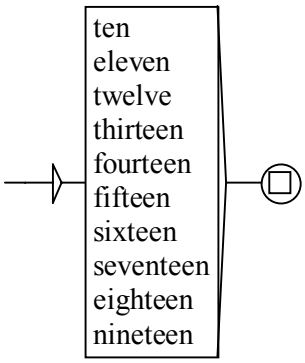


Fig.25. Graphe *DnumTxt10_19* : cardinaux de 10 à 19

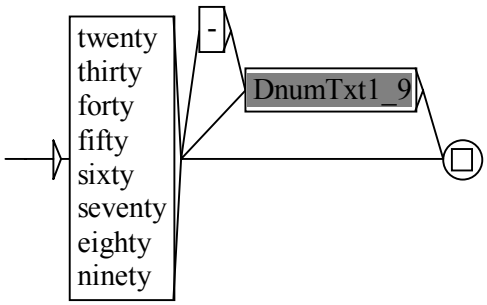


Fig.26. Graphe *DnumTxt20_99* : cardinaux de 20 à 99

[39] En effet, considérons un graphe où toutes les imbrications sont non récursives. Nous pouvons remplacer chacun de ses nœuds grisés par le graphe qu'il représente. Par un nombre fini de telles substitutions nous pouvons obtenir un graphe sans nœuds grisés (et donc équivalent à un automate fini) qui reconnaît le même langage que le graphe de départ.

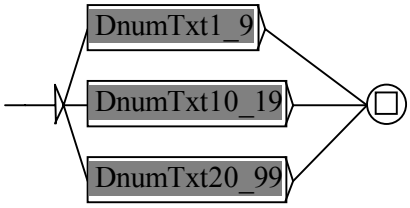


Fig.27. Graphe *DnumTxt1_99* : cardinaux de 1 à 99

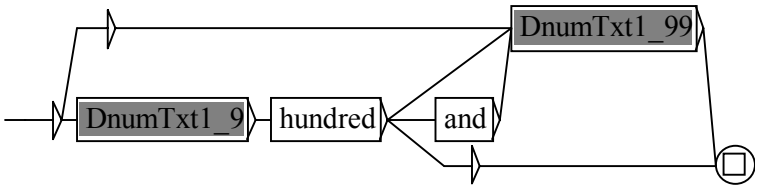


Fig.28. Graphe *DnumTxt1_999* : cardinaux de 1 à 999.

Les numéraux simples entre 1 et 20 sont listés dans les graphes *DnumTxt1_9* (Fig.24) et *DnumTxt10_19* (Fig.25). Ce premier est imbriqué dans le graphe *DnumTxt20_99* (Fig.26) pour les numéraux simples et composés de 20 à 99. Tous les trois graphes font partie du graphe *DnumTxt1_99* (Fig.27) des numéraux entre 1 et 99. Celui-ci est ensuite inclus dans celui des cardinaux inférieurs à 1000 (Fig.28), et ainsi de suite.

Analysons plus en détail le graphe *DnumTxt10^3_10^6-1* (Fig.29) qui décrit en particulier les déterminants à double lecture tels que :

[306] two thousand seven hundred and ten = twenty-seven hundred and ten (*= 2710*)

La première lecture ci-dessus est réalisée par la deuxième branche du graphe et la deuxième par l'une des branches inférieures. Remarquons aussi que le nombre des milliers et des centaines qui les suivent peut être exprimé par un nombre en tous chiffres possédant éventuellement une partie non entière comme :

[307] 7.6 thousand (*= seven thousand six hundred*)

ce qui est exprimé par le chemin supérieur du graphe.

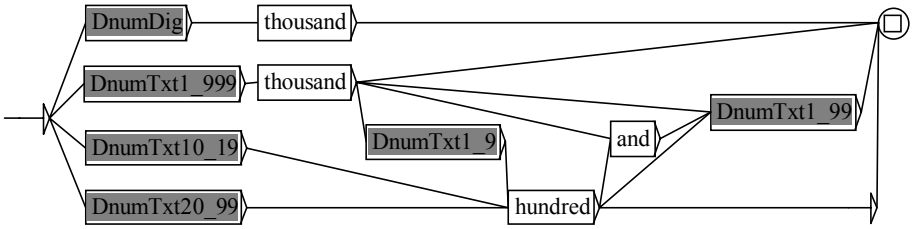


Fig.29. Graphe *DnumTxt10^3_10^6-1* : cardinaux entre 1 000 et 999 999.

Le graphe des cardinaux entre 1 et 10^6-1 (Fig.30) est obtenu par la concatenation de ceux des nombres inférieurs (Fig.28) et égaux/supérieurs (Fig.29) à 1 000.

Pour les nombres plus grands le schéma des graphes est répetitif, avec trois branches dont une contenant des chiffres (*6.8 million*), la deuxième toujours contenant *DnumTxt1_10^6-1* suivi de *million, trillion*, etc., et la troisième contenant le graphe construit à l'étape précedente. Ainsi, *DnumTxt1_10^6-1* (Fig.30) est inclus dans *DnumTxt1_10^12-1* (Fig.31), et ce dernier, à son tour, dans *DnumTxt1_10^18-1* (Fig.32), etc.

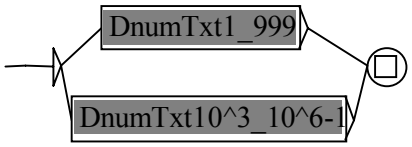


Fig.30. Graphe *DnumTxt1_10^6-1* : cardinaux positifs inférieurs à 10^6.

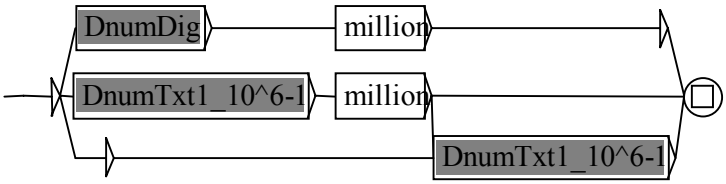


Fig.31. Graphe *DnumTxt1_10^12-1* : cardinaux positifs inférieurs à 10^{12}.

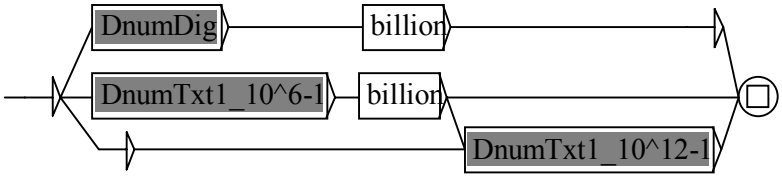


Fig.32. Graphe *DnumTxt1_10^18-1* : cardinaux positifs inférieurs à 10^{18}.

Le graphe principal *DnumTxt* (Fig.33) contient le dernier graphe créé (ici nous nous sommes arrêtés à 10^{18}-1) ainsi que le déterminant *zero* qui n'a pas été pris en compte dans d'autres graphes car il correspond au chiffre *0* omis dans la lecture des cardinaux positifs.

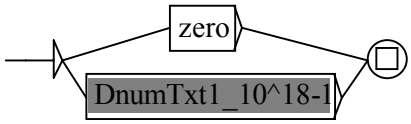


Fig.33. Graphe *DnumTxt* des déterminants numéraux cardinaux inférieurs à 10^{18}.

6.3 La description des cardinaux par transducteurs finis

Il est souhaitable que l'analyse des déterminants par graphes permette d'attribuer aux séquences reconnues des étiquettes grammaticales contenant le lemme, la catégorie, les marques distributionnelles et les traits flexionnels, de la même façon que ceci a lieu pour les mots contenus dans les dictionnaires DELAF et DELACF (cf Silberztein 1997). Ici, nous allons introduire des lemmes particuliers - chaque numéral reconnu sera accompagné de son équivalent écrit en chiffres, par exemple :

*[308] two thousand one hundred and ten,**2110**.DET+Num:p*

Nous pouvons obtenir ce type d'étiquettes par l'introduction de symboles de sortie (productions), marqués en dessous des nœuds des graphes. Lors du passage par l'un des chemins d'un graphe, les symboles de sortie sont concaténés et l'étiquette ainsi obtenue est rattachée à la séquence reconnue. Le rajout des telles productions augmente la complexité de la description pour trois raisons[40] :

– les numéraux simples contenus auparavant dans le même nœud d'un graphe doivent être séparés à cause de leurs productions individuelles (par exemple Fig.34 en comparaison avec Fig.26),

[40] L'augmentation de la complexité de description par transducteurs par rapport à celle par automates finis se reflète dans le nombre de graphes créés dans les deux cas. Pour les cardinaux inférieurs à 10^{18} nous avons obtenu 11 automates, mais les mêmes cardinaux représentés par transducteurs ont nécessité 23 graphes.

– la lecture des grands nombres (à partir de 10^9) n'est pas la même en anglais britannique qu'en anglais américain, ce que nous décrivons dans la suite,

– compte tenu du fait que le chiffre *0* est omis dans la lecture des cardinaux positifs (*201 = two hundred *zero one*), il faut introduire des ε-transitions avec sortie « 0 » pour permettre l'insertion de ce symbole à des positions non initiales. Ainsi, la création progressive des graphes décrivant les numéraux de plus en plus grands ne suit pas les mêmes règles que dans le cas des automates finis (section précédente).

Pour expliquer ce troisième phénomène, prenons l'exemple du graphe *Dnum100_999* (Fig.36) qui décrit les cardinaux de 100 à 999. Pour exprimer le nombre des centaines, nous y imbriquons le graphe *Dnum1_9* (Fig.35) créé auparavant. Mais pour décrire les dizaines et les unités nous ne pouvons pas nous servir des graphes déjà existants comme *Dnum20_99* (Fig.34), car nous devons admettre les dizaines et/ou des unités nulles, i.e. des insertion du « 0 ». Le cas où à la fois les dizaines et les unités sont nulles, est reconnu par l'ε-transition avec sortie « 00 ». De plus, nous créons un graphe auxiliaire *Dnum1_99bis* (Fig.37) décrivant les nombres entre 1 et 99 qui sont précédés des centaines. La branche supérieure de ce graphe nous permet d'insérer le chiffre « 0 » à la positions des dizaines.

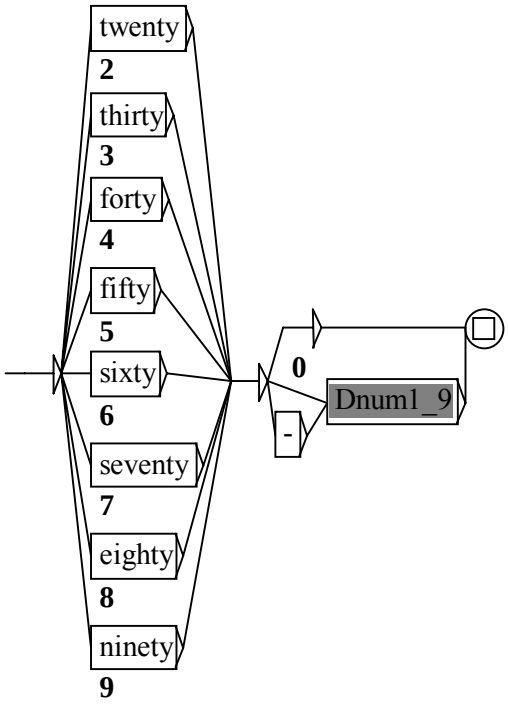


Fig.34. Graphe *Dnum20_99* : cardinaux de 20 à 99

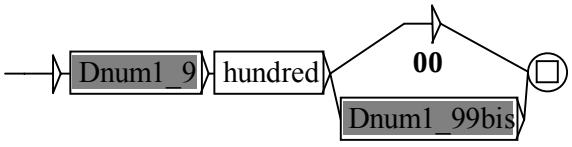


Fig.36. Graphe *Dnum100_999* : cardinaux de 100 à 999.

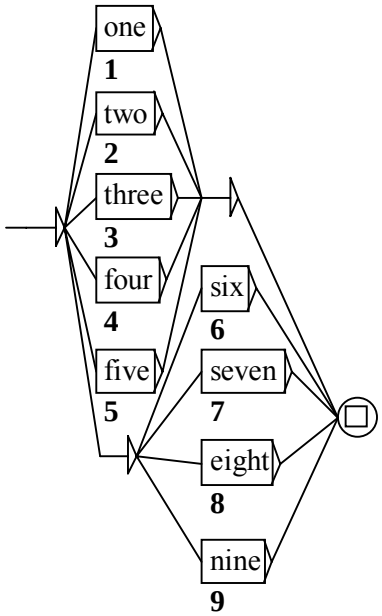


Fig.35. Graphe *Dnum1_9* : cardinaux de 1 à 9.

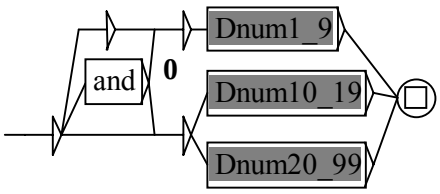


Fig.37. Graphe auxiliaire *Dnum1_99bis* : cardinaux entre 1 et 99.

Nous avons mentionné que la lecture des nombres en anglais britannique (AB) est différente de celle en anglais américain (AA) à partir de 10^9. D'après OALDCE (1989), on passe en AB d'un million à un billion, puis à un trillion, à un quadrillion etc. par rajout de 6 zéros, donc par multiplication par un million. En AA ce passage à lieu déjà après le rajout de 3 zéros, donc après la multiplication par un mille (voir Tab.9).

Nombre	**Anglais britannique**	**Anglais américain**
1 000 000 000	one thousand million	one billion
1 000 000 000 000	one billion	one trillion
1 000 000 000 000 000	one thousand billion	one quadrillion
1 000 000 000 000 000 000	one trillion	one quintillion

Tab.9 Lecture de grands nombres en anglais britannique et en anglais américain.

C'est pourquoi le graphe principal *Dnum* (Fig.38) des cardinaux inférieurs à 10^{18} contient trois parties. La première, qui représente les nombres inférieurs à 10^9 est commune pour l'AB et l'AA. Nous y distinguons les déterminants *zero* et *one* car ce premier est omis dans la lecture des cardinaux positifs et ce deuxième, reconnu seul, doit obtenir le trait morphologique *s* du singulier.

La deuxième partie du graphe *Dnum* correspond à la lecture britannique des nombres supérieurs ou égaux à 10^9. Suite au passage par cette branche, une séquence reconnue obtient une étiquette contenant le trait du dialecte *+GB*. La troisième partie reconnaît les versions américaines des mêmes cardinaux, et leur attribue le trait *+US*.

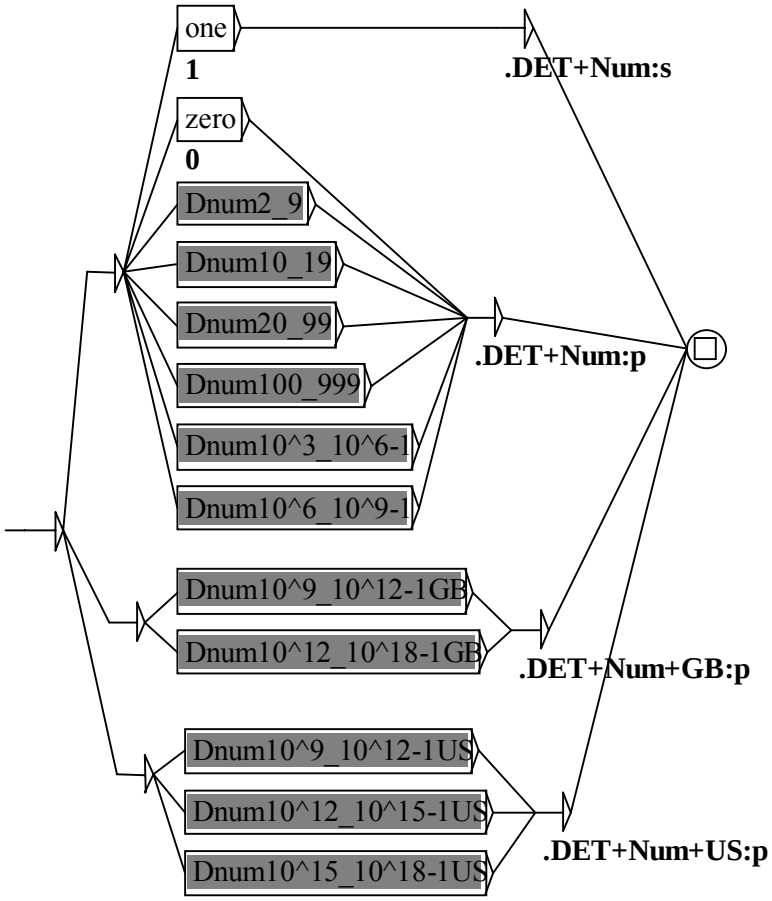


Fig.38. Graphe *Dnum* : cardinaux inférieurs à 10^{18}.

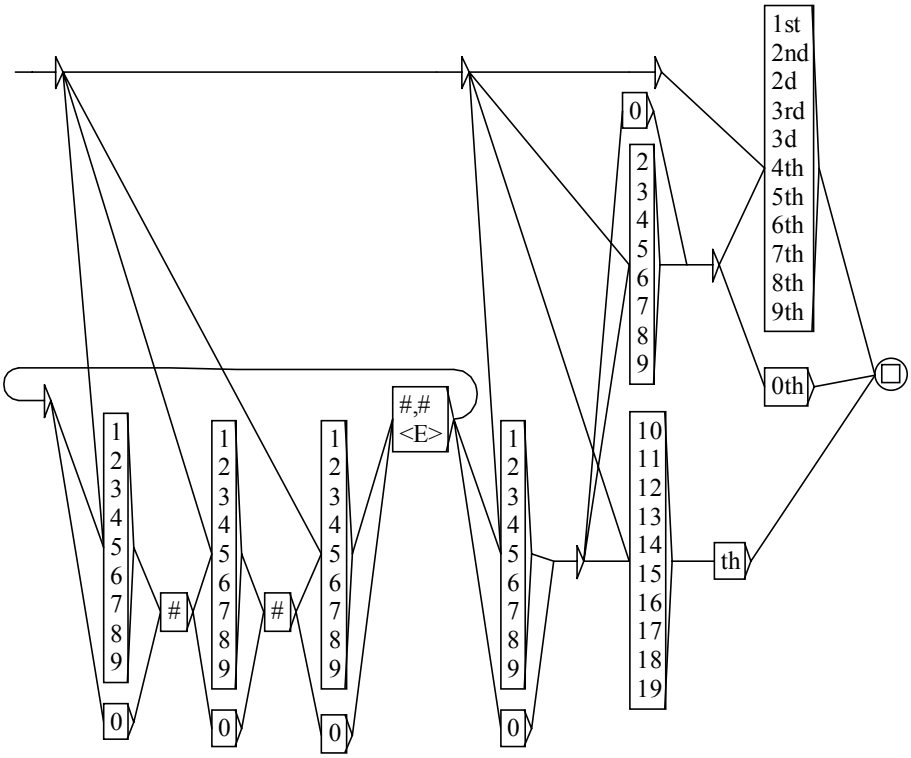


Fig.39. Graphe *DnumOrdDig* des numéraux ordinaux en tous chiffres

6.4 Déterminants numéraux ordinaux

La description des numéraux ordinaux, aussi bien par automates finis que par transducteurs, est parallèle à celle des cardinaux, mais il est nécessaire de traiter spécialement le dernier constituant qui porte la marque *st, nd, rd, d* ou *th*. C'est pourquoi le graphe *DnumOrdDig* (Fig.39) des ordinaux en tous chiffres est considérablement plus complexe que celui des cardinaux (Fig.23), même s'il ne contient pas la partie après le point correspondante à des fractions.

Analysons aussi un exemple de graphe pour les ordinaux en toutes lettres. Le transducteur *DnumOrd100_999* (Fig.40) des ordinaux entre 100 et 999 est obtenu de celui pour les cardinaux (Fig.36) par rajout du symbole d'entrée « *hundredth »* au cas où les dizaines et les unités sont nulles (*seven hundredth*). Dans ce graphe le nombre des centaines est exprimé par un numéral cardinal (*DnumTxt1_9*, Fig.24), tandis que le nombre des dizaines et des unités est un numéral ordinal (*DnumOrdTxt1_99bis*). Voici des exemples de séquences reconnaissables par le graphe ci-dessous, avec les lemmes attachés au cours de l'analyse lexicale :

[309] seven hundredth,700th.DET+Num+Ord

[310] seven hundred and first,701st.DET+Num+Ord

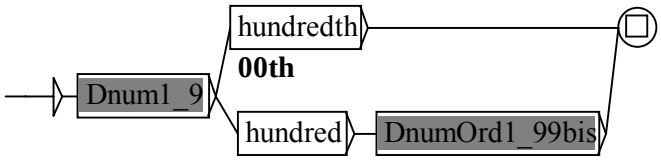


Fig.40. Graphe *DnumOrd100_999* : ordinaux de 100 à 999.

6.5 Emplois des étiquettes grammaticales des déterminants numériques

Le rattachement d'une forme canonique en chiffres, qui n'est pas ambiguë, permet de tenir compte des ambiguïtés des numéraux écrits en toutes lettres. Par exemple le nombre suivant reçoit, à raison, deux étiquettes différentes avec la marque du dialecte (*US* pour l'anglais américain, *GB* pour l'anglais britannique) :

[311] one billion,1000000000.DET+Num+US:p

[312] one billion,1000000000000.DET+Num+GB:p

D'autre part nous pouvons rendre compte des équivalences entre des numéraux écrits différemment. Les deux premiers exemples ci-dessous, ainsi que les deux derniers représentent les mêmes nombres, ce qui est explicité par leurs formes canoniques :

[313] one thousand nine hundred seventy-four,1974.DET+Num:p

[314] nineteen hundred seventy-four,1974.DET+Num:p

[315] one billion,1000000000.DET+Num+US:p

[316] one thousand million,1000000000.DET+Num+GB:p

6.6 Extension de la grammaire

Les déterminants numéraux constituent un sous-ensemble fermé et bien délimité de tous les déterminants composés. Grâce au mécanisme d'imbrication de graphes, nous pouvons

élaborer une grammaire plus générale de déterminants composés contenant des numéraux. Dans ce cas, la première bibliothèque de graphes, i.e. les automates sans sorties, doivent être utilisés. Par exemple dans la phrase suivante :

[317] [*Every one of the two hundred and twenty-five*] *participants received the proceedings of the conference.*

le déterminant composé *every one of the two hundred and twenty-five* contient le numéral composé souligné qui peut être vu comme unité remplaçable par tout autre numéral cardinal (sauf *one* et *two*). Ainsi, nous obtenons l'expression productive *every one of the <Dnum>*, où *Dnum* représente un déterminant numéral cardinal.

L'élargissement de la grammaire de déterminants numéraux exige entre autres l'analyse combinatoire complète de tous les déterminants simples (*all, no, such, the, these, what, whose, my,* etc.). Par exemple, le graphe *DetDetDnum* (Fig.41) représente les numéraux cardinaux (Fig.33) précédés d'un ou de deux déterminants simples (le graphe *Poss* contient les déterminants possessifs : *my, your, one's,* etc.). L'alignement de plus que trois déterminants simples avant un numéral, et de plus que deux après un numéral ne semble pas admissible. La description devient beaucoup plus compliquée pour les déterminants composés contenant des prépositions, comme *of* dans l'exemple [317], car les séquences à reconnaître peuvent être longues (e.g. *As many as twenty of those other fifty participants were absent.*).

Rappelons que l'étude complète de déterminants composés est complexe et ses limites sont difficiles à tracer, car il est nécessaire de considérer :

- les fractions (*three and two-third*),
- certains adverbes (*surprisingly many, almost every*, *certainly not all*, etc.),
- des possibilités d'insertions de modifieurs à l'intérieur d'un déterminant composé (e.g. *the United State's first third-country detention center*),
- les unités de mesures (*25 percent*, *350 miligrams*, etc.),
- d'autres déterminants dits nominaux, analysés en détails pour le français par Buvet (1994) (e.g. *un kilo de pommes, un verre de lait, un armée de touristes*).

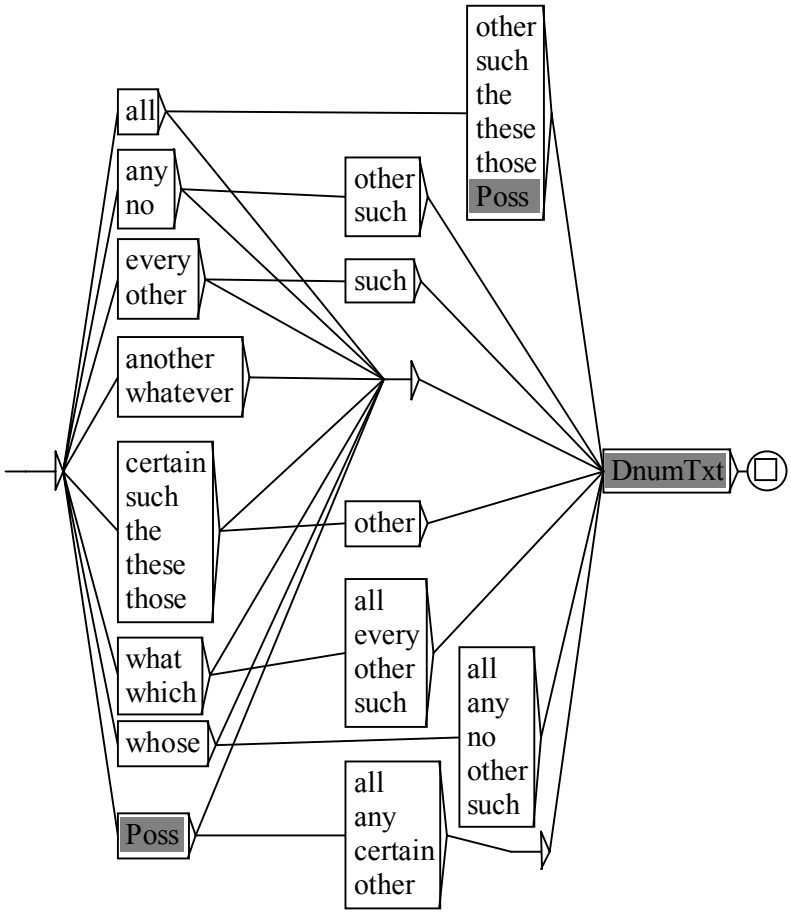


Fig.41. Graphe *DetDetDnum* : déterminants numéraux précédés d'un ou de deux déterminants simples.

6.7 Reconnaissance des numéraux par Intex

Les automates et transducteurs finis présentés dans cette étude peuvent être employés dans le système INTEX en tant que dictionnaires *non ambigus* (voir le manuel Intex, Silberztein 1999-2000, p. 123) de mots composés[41]. Cela veut dire que chaque séquence reconnue sera considérée comme étant effectivement une occurrence d'un numéral dans le texte. De plus, seulement les séquences les plus longues seront prises en compte, par exemple dans la phrase [317] seul le numéral maximal *two hundred and twenty-five* sera étiqueté, et non pas ses sous-séquences : *two hundred, two hundred and twenty,* et *twenty-five*.

Les séquences reconnues par les graphes apparaissent alors dans le dictionnaire des mots composés du texte[42].

Dans la version actuelle d'Intex (4.21) la définition du mot composé exige que son premier caractère soit obligatoirement une lettre de l'alphabet. Cette mesure, introduite pour des

[41] Menu *Text→Apply Lexical Resources*.

[42] Certaines de ces séquences sont pourtant des mots simples, car ils nous a semblé plus cohérent de décrire tous les numéraux par une seule bibliothèque de graphes. Si nous voulions séparer les numéraux simples (comme *ninety*, *hundredth*) des numéraux composés (comme *ninety-one, two hundredth*), les graphes deviendraient beaucoup plus complexes et moins intuitifs.

raisons opérationnelles, ne permet pas la reconnaissance, en tant que mots composés, de certains numéraux décrits plus haut. Par exemple, les graphes *DnumDig* (Fig.23) et *DnumOrdDig* (Fig.39) ne peuvent pas être employés sous Intex en tant que dictionnaires de mots composés. Aussi, les branches supérieures des graphes comme ceux des figures Fig.29, Fig.31 et Fig.32 ne sont jamais activées lors de l'analyse lexicale, donc les séquences comme *7.6 thousand* ne peuvent pas être reconnues.

6.8 Conclusion

Les outils à états finis sont très bien adaptés à la description des expressions numériques : 24 automates et 43 transducteurs, construits à partir de près de 70 mots simples, décrivent 10^{18} numéraux cardinaux et autant de numéraux ordinaux.

La taille des graphes obtenus est d'environs :

- 130 états et 1500 transitions pour les automates,
- 2700 états et 11000 transitions pour les transducteurs.

Le temps de leur application à un corpus de plusieurs mégaoctets, comme un mois du journal *Herald Tribune*, est de l'ordre d'une quinzaine de minutes.

Chapitre 7 Construction d'un dictionnaire électronique terminologique

7.1 Introduction

Dans le chapitre 5 nous avons vu que la création d'un dictionnaire électronique de mots composés à partir des entrées puisées dans des dictionnaires généraux traditionnels pose de nombreux problèmes dont certains ne trouvent pas de solution entièrement automatisable.

Le présent chapitre décrit la même tâche pour les dictionnaires techniques de traduction : disposant d'une grande banque de données terminologiques nous souhaitons l'adapter au traitement automatique de textes techniques. Ceci va poser de nouveaux problèmes car nous passons du traitement de la langue générale à celui du langage spécialisé, et de la reconnaissance des mots composés généraux à celle des termes techniques. Nous décrivons les étapes de création d'un dictionnaire électronique du domaine de l'informatique à partir de deux dictionnaires traditionnels de traduction et nous présentons les problèmes rencontrés.

7.2 Termes composés - mots composés du langage spécialisé ?

Selon la première intuition, un terme composé est pour le langage spécialisé la même chose qu'un mot composé pour la langue générale. Du point de vue orthographique, ce serait donc une suite contiguë de mots simples qui doit pour certaines raisons être traitée « en bloc ».

Nous ne nous attardons pas sur les notions de terme, de langage technique (i.e. spécialisé), de différents domaines techniques etc., et ceci pour deux raisons :

1) Il nous semble impossible de tracer une frontière nette entre la langue générale et spécialisée, ainsi qu'entre les différents domaines techniques et les langages qui leur sont propres.
2) Ce qui nous intéresse c'est l'approche pragmatique : la qualité des outils informatiques que nos proposons sera validée par l'utilisateur final, le traducteur.

Les notions de terme et de domaine technique ne sont donc pas prédéfinies - elles dépendent de l'application. Par exemple, l'ensemble des séquences figées que nous chercherons à reconnaître dans un texte ne sera pas le même dans le cas de la recherche documentaire et dans celui de l'aide à la traduction. De plus, dans ce deuxième cas, comme nous disons dans la section 8.3, une séquence de mots considérée par un utilisateur (traducteur, lexicographe, expert du domaine, etc.) comme terme peut ne plus l'être pour un autre utilisateur, un autre texte, un autre contrat de traduction, un autre domaine, etc. Ceci implique qu'un terme composé n'est pas forcément un mot composé, tel que nous l'avons défini dans la section 2.2.2. Par exemple, un nom propre comme *Vaclav Havel* n'est pas un mot composé, mais il est clairement un terme composé du langage spécialisé de la politique. Un message d'erreur comme *digit expected* n'est pas un mot composé, mais il est un terme dans le context de traduction technique (c'est une séquence qui doit toujours être traduite de la même façon). Des exemples de ce type sont nombreux.

En résumé, c'est donc l'utilisateur de nos logiciels qui détermine ce qu'est un terme, ce qu'est un langage spécialisé et quel est son domaine technique, par le choix des textes qu'il soumet au traitement, et des dictionnaires techniques qu'il utilise pour ce traitement.

Les outils informatiques décrits dans cette partie du mémoire ont été développés dans la société LCI Informatique, pour la troisième version de leur banque de données terminologique LexPro CD Databank[43] (appelé aussi LexPro par la suite).

7.3 Base terminologique LexPro CD Databank

LexPro CD Databank est une base de données terminologiques multilingue et multidomaine. Elle rassemble près de 120 dictionnaires techniques traditionnels du commerce sur un support informatique, ce qui représente plus de 10 millions de termes. Les dictionnaires sont très différents entre eux du point de vue :

1) de leur taille - de 200 à 100 000 entrées,
2) des langues concernées – 11 langues (le français, l'anglais, l'allemand, l'espagnol, l'italien, le portugais, le néerlandais, le danois, le russe, le suédois, l'arabe) sont présentes dans la base entière, le français, l'anglais, et le russe étant prédominants. Les dictionnaires sont bilingues ou multilingues et ne font pas la distinction entre une langue source et une langue cible.
3) des informations rattachées aux entrées – les dictionnaires étant convertis en des bases relationnelles, le nombre et le contenu des colonnes, donc des informations techniques et linguistiques accompagnant les entrées principales, sont très différentes d'un dictionnaire à l'autre ; certains ne contiennent que les termes principaux d'une langue avec leurs termes correspondants en une autre langue ; d'autres renseignent les variantes orthographiques et morphologiques (e.g. *active matrix*, *active-matrix*), les synonymes (*on-chip cache, primary cache, level 1 cache*), les domaines d'application (e.g. *Microsoft*), les dialectes (e.g. *US, GB*), les significations des abréviations, les catégories grammaticales, etc.
4) de la qualité des données – des erreurs (e.g. d'orthographe) et des incohérences (e.g. informations placées dans des mauvaises colonnes) sont plus ou moins fréquentes chez différents auteurs.

Pour que l'utilisateur puisse plus facilement choisir les dictionnaires qu'il utilise, ils ont été regroupés dans 26 thèmes dont les mieux représentés sont ceux de l'aéronautique, du monde des affaires, de l'électronique et de l'informatique, du juridique, et de la médecine et de la génétique.

7.4 Adaptation des dictionnaires techniques de traduction au traitement automatique du langage naturel

Dans notre application l'analyse lexicale des termes est fondée sur les mêmes principes que ceux décrits dans le chapitre 2 pour les mots composés de la langue générale. Nous utilisons pour ceci les fonctionnalités du système INTEX selon le schéma illustré dans l'organigramme Fig.7. La seule différence est celle du choix des dictionnaires DELAF et DELACF consultés lors de l'étiquetage des mots simples et composés : à la place des DELAF et DELACF de la langue générale, nous utilisons les dictionnaires DELAF et DELACF spécialisés obtenus à partir des dictionnaires techniques de LexPro CD Databank. Même si le passage d'un dictionnaire technique usuel à un dictionnaire électronique pour le TALN est facilité ici par le fait que les dictionnaires de LexPro se trouvent déjà sur le support informatique, nous nous

[43] La société LCI Informatique a arrêté son activité en 1999, la version 3.0 de LexPro CD Databank n'a pas été commercialisée jusqu'à présent.

heurtons à de nombreux problèmes, similaires à ceux des dictionnaires traditionnels généraux (voir section 5.2).

1) Dans LexPro il est relativement facile de reconnaître les unités qui ont le **statut de terme**. D'habitude, ce sont ceux qui sont contenus dans les colonnes des entrées principales et leurs traductions. Certains dictionnaires contiennent aussi des colonnes supplémentaires pour les variantes orthographiques et morphologiques des entrées, leurs abréviations, leurs antonymes, etc. Toutes ces « vraies » unités terminologiques, que nous appelons des **formes écrites**, sont assez faciles à repérer, mais le choix des colonnes pertinentes doit être fait individuellement pour chaque dictionnaire car les appellations et les contenus des colonnes sont différents d'un dictionnaire à l'autre.

2) Une fois les colonnes pertinentes extraites il faut procéder à **l'unification du format** des entrées, comme l'effacement des séparateurs et déterminants initiaux (e.g. *the cursive* → *cursive*). Certains auteurs insèrent plusieurs synonymes dans une seule colonne à l'aide d'une coordination. Ces cas-là doivent être divisés en plusieurs entrées et ceci n'est pas entièrement automatisable. Par exemple, la séparation des synonymes coordonnés par une virgule (e.g. *constrained handwriting, constrained handprinting*) pourrait être immédiate si elle ne risquait pas d'affecter les termes atomiques contenant une virgule (*multiple instructions, multiple data*). Encore plus difficiles sont les cas de remise en facteur où la partie commune n'est pas repérable automatiquement. Par exemple dans l'entrée

   [318] compression (expansion) of luminance

   le mot entre parenthèses peut remplacer le premier constituant, nous avons donc deux termes différents : *compression of luminance* et *expansion of luminance.* Mais dans une autre entrée :

   [319] passive (communications) satellite

   le mot entre parenthèses est facultatif, nous avons donc deux variantes du même terme : *passive satellite* et *passive communications satellite*.

3) Les dictionnaires techniques dont nous disposons contiennent un nombre non-négligeable **de fautes d'orthographe**. Leur repérage se fait automatiquement si l'on dispose d'un dictionnaire DELAF de la langue traitée. Pour ceci il suffit d'effectuer l'étiquetage des tous les termes simples et des constituants des termes composés par le DELAF[44] de la même façon que ceci est décrit pour le DELAC général (section 5.4). La correction peut être améliorée grâce à un correcteur orthographique comme celui présenté dans le chapitre 9. Néanmoins, il ne faut pas corriger de termes non reconnus par le DELAF avant d'analyser toutes les informations qui les accompagnent. Prenons trois exemples tirés du dictionnaire informatique de De Solliers (1998) :

[44] Nous ne disposons au début que du DELAF général qui est complété au fur et à mesure par des nouveaux termes simples que nous codons (voir section 7.5.1).

Entrée	Traduction
Inglish	Langage semblable à l'anglais utilisé pour les jeux d'aventure
turist	Influenced by Turing
donuts	Déformation de *doughnut* = any set of memory bits

Tab.10 Exemples de termes informatiques anglais non reconnus par le DELAF général.

Les trois termes simples anglais du domaine de l'informatique ont été repérés comme inexistants dans le DELAF général. Le correcteur orthographique a proposé des corrections – *English, tourist* – qui, vus rapidement, semblaient valables. Pourtant les définitions fournies par l'auteur indiquent clairement que ce sont des déformations volontaires des mots anglais. Il s'agit donc de nouveaux termes qui doivent entrer dans le DELAS spécialisé et non pas être considérés comme fautes d'orthographe.

4) Le **codage des nouveaux termes simples** et des constituants inconnus des termes composés est nécessaire pour la cohérence du système. Sans les codes flexionnels de tous les termes nous ne pourrions pas reconnaître les occurrences de leurs formes fléchies dans des textes, ni achever la construction du DELAC et du DELACF spécialisés. Mais ce travail est presque entièrement manuel et peut prendre un temps considérable pour certains dictionnaires. Nous en avons fait une estimation sur l'exemple du *Dictionnaire des Télécoms* français-anglais d'Estréméra. Parmi ses 17500 entrées (simples et composées) anglaises, nous avons rencontré près de 9000 mots simples non reconnus par le DELAF anglais général, dont 7300 sigles et 1700 fautes d'orthographe et nouveaux termes simples. La correction ou codage de ces 1700 mots a pris 6,5 heures de travail. Si l'on admet le même taux d'erreur et des nouveaux mots dans tous les autres dictionnaires, la correction et le codage des mots simples pour la partie anglaise (près de 5 millions de termes) de la base entière nécessitera 10 mois/homme. De plus, les sigles que nous avons écartés dans notre évaluation - en admettant qu'ils étaient pour la plupart des substantifs avec le pluriel qui s'obtient par rajout d'un *s* ou d'un *es* - peuvent présenter des particularités difficiles à repérer. Par exemple *IOT* (= *in other terms*) et *A.K.A.* (*also known as*) sont respectivement un adverbe et une conjonction.

5) Très peu d'auteurs de dictionnaires indiquent les **catégories grammaticales** de leurs termes. Nous ne pouvons compléter ces données que semi-automatiquement. Par rapport aux dictionnaires généraux, les dictionnaires de traduction présentent ici un nouvel aspect, car ils contiennent un nombre important d'unités phraséologiques. Analysons quelques exemples du dictionnaire informatique de De Solliers (1998) contenu dans la table Tab.11.

Les trois premières entrées ci-dessous sont des sigles nominaux qui peuvent, en particulier, apparaître au pluriel (*two FIFOs = two data structures of FIFO type, two CALLCs = two CALLC instructions*). Mais leurs termes complets correspondants sont difficiles à accepter en positions nominales au pluriel (?*two rotates left through carry, ?two rotate left through carries*). Les deux derniers exemples sont des messages provenant des logiciels informatiques. Leur degré de figement est problématique selon les critères présentés dans la section 2.2.2, mais du point de vue de notre application ce sont des éléments figés car ils sont traités par les traducteurs comme unités standardisées. On leur attribue la catégorie *Mess* qui ne possède pas de flexion. Rappelons que la catégorie d'un composé n'est pas

automatiquement déductible à partir de sa structure. Le dernier exemple de Tab.11 qui est une unité phraséologique, a la même structure que *building block system* (*système à blocks fonctionnels*) qui est un nom composé.

Entrée	=
FIFO	first in, first out
RLC	rotate left through carry
CALLC	Call if Carry
digit expected	ceci devrait être un chiffre
building file list	création de liste en cours

Tab.11 Exemples d'unités phraséologiques de traduction en anglais.

7.5 Construction d'un dictionnaire électronique anglais de l'informatique pour le TALN.

Afin de pouvoir utiliser les dictionnaires techniques du LexPro pour la reconnaissance automatique des termes par les algorithmes du système INTEX, tous ces dictionnaires doivent être convertis en des DELAS et DELAC spécialisés, et ensuite fléchis pour obtenir leurs DELAF et DELACF correspondants. Une équipe de lexicographes et informaticiens doit donc être amenée à confronter les problèmes présentés plus haut pour une très grande quantité de termes. Nous avons effectué la conversion en DELAS/DELAC des termes anglais contenus dans deux grands dictionnaires informatiques anglais-français : De Solliers (1998) et Hildebert (1998). Les résultats obtenus sont présentés dans le tableau ci-dessous.

	De Solliers (1998)	**Hildebert (1998)**	**Union**	**Termes communs**
Termes anglais (noms, adjectifs, adverbes, unités phraséologiques figées)	50 554	54 207	91 483	13 278 (14,5 %)
simples	19 946	14 685	27 568	7 063 (25,5 %)
composés	30 608	39 522	63 915	6 215 (9,7 %)

Tab.12 Données numériques sur les termes informatiques anglais

La dernière colonne montre à quel point les couvertures de ces deux dictionnaires de référence sont différentes – seulement 14,5% de termes communs - et donne une idée du nombre de

termes existants et pas encore recensés. Ces résultats se rapprochent de ceux pour les deux dictionnaires de la langue générale, NSOED 1996 et HO 1994, dont nous avons obtenu l'intersection des mots composés de 11% (voir section 5.2).

Ci-dessous nous donnons un aperçu des problèmes rencontrés lors de la conversion de l'ensemble des termes faisant l'objet de la table 12 en un système DELA (4 dictionnaires électroniques) de termes informatiques.

7.5.1 Construction d'un DELAS spécialisé de termes informatiques

Le DELAS spécialisé de termes informatiques a été construit en 2 étapes.

Lors de la première étape nous avons extrait des deux dictionnaires techniques mentionnés tous les termes simples (séquences sans séparateurs). Ensuite, nous avons recopié les catégories et les codes pour ceux qui se trouvaient déjà dans le DELAS, en leur attachant en plus le trait *+Spec* pour marquer leur appartenance au langage spécialisé. Ainsi, nous avons obtenu 12 875 entrées du futur DELAS spécialisé sous le format suivant :

[320] disk,N1+Spec

Ce travail a été fait automatiquement mais il a eu des effets de bord indésirables : chaque terme a obtenu toutes les étiquettes possibles indépendamment du sens qu'il avait dans son dictionnaire technique. Ceci n'est pas correct dans le cas général car, comme le remarque Lehrberger (1986), un mot du langage « standard » peut être concerné dans un langage spécialisé par des restrictions au niveau de catégories qui lui sont attribuées. Par exemple, le mot *ace* apparaissait dans les dictionnaires de De Solliers (1998) et Hildebert (1998) seulement comme substantif (*access control entry, a color expert, advanced computing environment, advanced CMOS ECL* etc.), mais il a obtenu trois étiquettes du DELAS général : *A0, N1* et *V4*, la première et la troisième ont été donc non pertinentes. D'autre part, parmi les homographes dans le DELAS général il n'y avait pas toujours celui qui correspondait au terme en question. Par exemple, le terme *so* avait dans les deux dictionnaires quatre significations nominales (*sort ; nom d'un virus PC ; synchroton orbital radiation ; Shift Out control character in ASCII ; Send-Only ; Serial Output ; Small Outline*), mais dans le DELAS général ce mot ne figure qu'en tant qu'adverbe et conjonction. Ainsi, ce terme reçoit deux étiquettes non pertinentes, *so.ADV+Spec* et *so.CONJ+Spec*, mais il ne reçoit pas d'étiquette correcte qui devrait être *so.N1+Spec*.

Les termes simples qui n'ont pas été reconnus par le DELAS – soit 12 156 mots (6 906 sigles et 5 250 autres mots simples communs et propres) – ont dû être codés manuellement. Ils se divisaient en les catégories suivantes :

1) noms propres

[321] Aberdeen, UniModem, Zenographics, etc.

2) sigles

[322] CAD (*computer aider design*), *CEP* (*compose edit processor*), *NCPSI* (*network control packed-switching interface*)

3) séquences soudées de mots simples connus

[323] bitmap, dataflow, filename, groupware, kilobit, etc.

4) mots simples connus avec des préfixes (1369 cas)

[324] antitrace, autoabstract, bistream, crossconnect, etc.

5) nouveaux mots obtenus par dérivation des cas 3) et 4) :

[325] microprogrammed, monospaced, bitmapped, etc.

6) conversions des participes connus vers des noms (la plupart des participes présents ont obtenu le code *N1*)

[326] addressing, answering, buffering, etc.

7) conversions des participes connus vers des adjectifs (la plupart des participes présents et des participes passés ont obtenu le code *A0*)

[327] assembling, answering, calling, magazined, drived, committed, decyphered, etc.

8) « vrais » nouveaux mots simples

[328] a pixel, a profiler, a flagger, a diff, an iterator, a keyer, a lite, a tuple

Remarquons que dans les catégories 6) et 7) ci-dessus se trouvent des mots simples fléchis de la langue générale (i.e. présents dans le DELAF général), mais ils ne sont pas des lemmes donc ils n'ont pas été trouvés dans le DELAS.

Après les traitements décrits ci-dessus, nous avons construit un premier DELAS informatique de 25 031 entrées qui a été fléchi automatiquement vers un DELAF informatique.

Lors de la deuxième étape de la création du DELAS, nous avons utilisé les noms du DELAF général et les noms du nouveau DELAF informatique pour étiqueter les constituants caractéristiques des termes composés. Les constituants qui n'ont pas été reconnus ont été codés manuellement. Pour la plupart ceux-ci étaient des :

9) noms obtenus par conversion des verbes connus

[329] an acknowledge (= *an acknowledge message*)*, an add* (= *an add instruction*)*, a decode,* etc.

10) noms obtenus par conversion des adjectifs connus

[330] a cellular (dans *analog cellular*)*, a compatible* (dans *sun-compatibles*), *a literal, a floppy*, etc.

Cette deuxième étape a permis de rajouter 1 871 nouveaux noms simples dans le DELAS de termes informatiques qui comptait ainsi 26 902 entrées, et son DELAF correspondant 73 163 entrées (voir la table Tab.13 pour les données numériques sur ce dictionnaire). Un extrait de ce DELAS se trouve dans l'annexe E.

Quant à l'existence du pluriel des noms simples informatiques inexistants dans le DELAS général, nous n'avons pas fait d'analyse précise et nous avons admis la flexion en nombre pour presque tous les noms, sauf certains cas isolés comme *Centronics,N2S* et *FFFFh,N2S.*

Catégorie	Nombre d'entrées		Exemples
	DELAS	DELAF	
noms	19 852	39 708	*acceptability, infrasound, inputting, DSR, ward, PPLambda, StarEthernet*
adjectifs	4 447	4 992	*absolute, abstract, infrared, turist, topological, toroidal, tracking, trafficked, trainable*
verbes	2 249	28 044	*transfer, trigger, troubleshoot, unify, unload, update, broadcast, cast, highlight*
adverbes	354	419	*instantaneously, linearly, numerically, recurrently, interactively, isotropically, softwarily*
TOTAL	**26 902**	**73 163**	

Tab.13 Nombres de termes simples anglais du domaine de l'informatique

7.5.2 Construction du DELAC de termes informatiques

Le DELAC de termes informatiques a été créé à partir des termes composés extraits des deux dictionnaires techniques mentionnés plus haut. Nous devions d'abord classer les termes par catégories. Premièrement, nous avons séparé ceux pour lesquels les catégories étaient déjà indiquées par les auteurs. Ceci a pu introduire des erreurs dans les cas où un terme anglais avait été traduit par un terme français d'une autre catégorie, par exemple *single step* (nom) = *pas à pas* (adverbe). Or, une seule catégorie, commune pour les deux termes, était sensée être indiquée.

Pour le reste nous cherchions à automatiser partiellement ce travail en analysant des séries de séquences susceptibles d'avoir toujours la même catégorie. Par exemple, nous avons remarqué que de nombreux adjectifs composés se terminaient par *bound, type, –free, -ble* (*peripheral bound, R-type, vibration-free, content-adressable, downward compatible*, *front-accessible*). Nous avons aussi analysé les séquences en *-ed* et *-ing* pour repérer les adjectifs composés avec un participe (*stream-oriented, tape-operated*, *free-running*). D'autres candidats pour des adjectifs étaient les termes binaires avec un tiret (*shock-proof, soon-to-be-released*). Les adverbes composés ont été recherchés, par exemple, parmi les termes binaires dont le premier composant était une préposition (*on error, on key down, at random*). Les messages (catégorie *Mess*) commençaient souvent par *no* (*no connection, no default printer, no-carbon-paper required*), etc. Néanmoins, le gros du travail a été fait manuellement.

Les termes composés marqués comme adjectifs, adverbes ou messages n'ont pas subi d'autres traitements car ils sont invariables. Les substantifs ont dû être transformés en un DELAC étiqueté pour pouvoir être fléchis. Nous devions donc reconnaître à quelles positions se trouvaient les constituants caractéristiques. Pour ceci nous avons divisé automatiquement tous les termes nominaux composés en plusieurs fichiers en fonction de leur nombre de constituants. Les binaires ont été classés d'office comme des *XC*, i.e. ayant le deuxième constituant caractéristique et le premier invariable (certaines erreurs comme *load immediate* survenues ainsi ont pu être détectés plus tard lors de l'étiquetage). Les noms ternaires ont été classés soit comme des *CXX* s'ils contenaient une préposition à la deuxième position (*table of contents*, *assignement by name, return from subroutine*) soit comme des *XXC* pour tous les

autres cas (*optical disk drive, syntaxt-directed compiler*). La vérification des *CXX* a permis d'écarter les cas trompeurs comme *add-on hardware, bottom-up parsing*. Les quaternaires étaient des *CXXX* si une préposition se trouvait à la deuxième position (*Algebra of Communicating Processes*, *transfer on no zero*), des *XCXX* s'il y en avait une à la troisième position (*automatic request for repeat, virtual end of frame*), et des *XXXC* dans les autres cas (*advanced communications timekeeping technology, network-extensible windows system*). Une vérification était nécessaire pour les noms classés ainsi comme des *CXXX* car ils risquaient d'être des *XXXC* (*end of file label, out of order execution, speech to text conversion*). Les termes de plus de 4 constituants n'ont pas été étiquetés.

Pour chacun des types ci-dessus nous avons effectué l'étiquetage des constituants caractéristiques à l'aide du DELAF général des noms et du DELAF des noms informatiques comme ceci a été décrit dans la section 7.5.1 (deuxième phase de création du DELAS). Ainsi, nous avons obtenu un DELAC étiqueté de 55 799 noms informatiques dont la typologie est présentée dans la table Tab.15 et dont un extrait se trouve dans l'annexe F. La comparaison de la typologie du langage spécialisé (Tab.15) et de la langue générale (Tab.8) permet de constater que ce premier a la tendance à créer des unités lexicales plus complexes (63% contre 88% de termes binaires). Pour tous les noms composés informatiques, nous avons marqué la possibilité de la mise au pluriel, sauf dans les cas où la flexion était interdite pour le constituant caractéristique, comme :

[331] control/electronics(electronics.N2S:s),N+XC:s

La flexion automatique du DELAC informatique par l'algorithme présenté dans le chapitre 4 a produit le DELACF informatique de 109 389 entrées. Lors des traitements sur les noms composés décrits ci-dessus aucun cas de flexion irrégulière (voir section 3.3) n'a été repéré.

Catégorie	**Nombre d'entrées**		**Exemples**
	DELAC	**DELACF**	
noms	55 799	107 421	*information link, processing rate, physical recording density, assignment of copyright, fuzzy logic instruction per second*
adjectifs	1607	1 607	*all-digital, amplitude-modulated, object linking and embedding-aware*
adverbes	187	187	*at random, counter clockwise, in-situ, on double click*
unités phraséologiques	174	174	*access denied, bus priority out, enhanced mode required*
TOTAL	**57 767**	**109 389**	

Tab.14 Nombres de termes composés anglais de l'informatique

Classe	Nombre d'entrées	%	Exemples
XC	35 231	63 %	*card verifier, data integration, insert subroutine, numerical analysis, self-documenting, Z-buffer*
XXC	14 241	25,5 %	*absolute cell reference, five-bit byte, money back guarantee, surface emitting led*
XXXC	3 523	6,3 %	*active matrix color display, file transfer access control, shared media access LAN*
CnX[45]	996	1,8 %	*end of file, level of addressing, dial on demand, return from interrupt, Association for Computing Machinery, end of transmission block*
XCXX	182	0,3 %	*frequency modulation with feedback, Massachussets Institute of Technology, automatic request for repeat*
Autres	1 651	3 %	*Advanced peer-to-peer networking, electrically alterable programmable random access memory*
TOTAL	**55 799**		

Tab.15 Classes typologiques des noms composés anglais de l'informatique

7.5.3 Termes contenant des caractères spéciaux

Nous avons donné dans la section 2.2.2 les définitions du mot simple et composé et aussi mentionné les problèmes posés par les séquences figées contenant des séparateurs qui ne peuvent être classées ni comme mots simples, ni comme mots composés. Dans le langage technique de l'informatique nous avons rencontré un nombre important (4140) de telles séquences. Analysons les extraits du dictionnaire de De Solliers (1998) contenus dans la table Tab.16.

Aucun des termes ci-dessous ne peut être considéré comme mot simple, car tous contiennent des séparateurs (chiffres, caractères de ponctuation, etc.). D'autre part le système INTEX impose qu'un mot composé débute par une lettre, donc les termes de 1 à 7 ne peuvent pas être des mots composés non plus. Finalement, les termes 8 et 9 ne peuvent pas faire partie d'un DELACF pour l'instant, car ils contiennent des points et des virgules qui sont des métacaractères dans le système DELA – ils servent à séparer les formes fléchies, les lemmes et les codes flexionnels des entrées. Ceci prouve que le traitement de termes simples et composés ne peut pas toujours être effectué par les mêmes outils que le traitement de mots simples et composés du langage général.

[45] La classe *CnX* contient tous les composés avec le premier constituant caractéristique suivi de n'importe quel nombre de constituants invariables.

No	Terme anglais	Traduction
1	1963	Nom d'un virus de PC ; d'origine inconnue, il a été découvert en mai 1980
2	1DIR+	Logiciel interpréteur de commande, concurrent de NORTON Commander, rebaptisé WONDER PLUS dans les années 1980
3	.MOV	Format des fichiers vidéo QuickTime pour Windows
4	<bobbit>	Commonly used as a placeholder for omitted text in a followup message (not copying the whole parent message is considered goog form). Refers, of course, to the celebrated mutilation of John Bobbit
5	_1576	Noms d'un virus de PC ; il est connu comme indectant les fichiers .COM mais des variantes inconnues peuvent également infecter d'autres objets
6	0.75	Trois quart de pouce, environs 19 mm, standard, hauteur du facteur de forme pour les baies d'extension
7	32-bit value	
8	P.O.D.	Acronym for 'Piece Of Data' (as opposed to a code section). Usage : pedantic and rare. See also {pod}.
9	multiple instructions, multiple data	

Tab.16 Exemples de termes anglais contenants des séparateurs.

7.5.4 Recherche automatique des termes et de leurs traductions dans des textes

Si nous disposons, pour chaque dictionnaire technique de la base LexPro, d'un système DELA de ses formes écrites, nous pouvons reconnaître les occurrences des formes fléchies de termes de la base dans des textes.

L'utilisateur fournit le texte à analyser et il fait lui-même le choix des langues (source et cible) et des dictionnaires du LexPro qu'il veut utiliser pour analyser son texte. L'analyse se fait par un module fondé sur les fonctionnalités de l'étiquetage du système INTEX (voir Fig.7 section 2.9). Ici, les fonctions de l'étiquetage des mots simples et des mots composés utilisent en entrées les DELAF et les DELACF de tous les dictionnaires LexPro choisis par l'utilisateur. Si une séquence a été reconnue par l'un ou plusieurs des DELAF/DELACF ceci nous donne automatiquement l'accès à son (ses) lemme(s). Il suffit donc de rechercher ce(s) lemme(s) dans le(s) dictionnaire(s) correspondant(s) de la base pour trouver la (les) traduction(s) possible(s).

Remarquons qu'un dictionnaire électronique du type DELAF ou DELACF est toujours unilingue. Ceci est dû au choix de l'implémentation choisie qui est celle des automates finis. Elle permet d'obtenir un taux très élevé de compactage et le temps linéaire d'accès (voir Tab.6 page 38). Mais un tel compactage efficace par automate fini n'est possible que si beaucoup

d'entrées ont des préfixes et des suffixes communs. Ceci est le cas des mots simples et composés d'une langue comme l'anglais ou le français, mais si on rattache à ces mots leurs traductions en une autre langue (ou en plusieurs autres langues) il y a très peu d'entrées avec un suffixe commun. Ainsi le taux de compactage est très bas. Il est donc nécessaire d'effectuer ces 2 étapes :

1) reconnaître une séquence à l'aide d'un DELAF (DELACF) de la langue source pour attribuer à cette séquence son lemme et, de plus, indiquer le numéro du dictionnaire où la séquence a été trouvée (un DELAF/DELACF et son dictionnaire technique correspondant ont le même numéro),
2) rechercher le lemme dans le dictionnaire technique (qui est sous forme d'une base relationnelle) indiqué et fournir les traductions de ce lemme en la langue cible souhaitée par l'utilisateur.

7.6 Conclusion

L'adaptation d'un dictionnaire technique usuel au traitement automatique du langage naturel est un travail long et en grande partie manuel à cause des incohérences et lacunes présentes dans ces dictionnaires qui sont créés pour un utilisateur humain. Cette tâche pourrait devenir quasiment automatique si les auteurs de ces dictionnaires respectaient quelques règles de base, comme le marquage de catégorie grammaticale pour chaque terme et les commentaires sur les irrégularités de flexion. De plus, il serait important de comprendre la notion d'une « forme écrite » (section 7.4, premier point), c'est-à-dire d'un terme tel qu'il peut apparaître dans un texte, sans marque de variantes orthographiques, sans élements optionnels, sans commentaires ni précisions, sans déterminants initiaux, sans majuscules initiales si elles ne sont pas obligatoires, sans caractères de ponctuation n'appartenant pas au terme, etc. Ainsi, dans chaque colonne contenant des termes ne doivent apparaître que des formes écrites et toute information complémentaire doit être placée dans une autre colonne adaptée. Analysons quelques exemples problématiques trouvés dans la base.

Terme
calling or originating modem
compression (expansion) of luminance range
switching area (UK)
passive (communications) satellite
restoral, see also reproduction
end-of-medium (EOM)

Tab.17 Exemples d'entrées mal formées d'un dictionnaire technique

Leur codage correct, i.e. permettant la récupération automatique des formes écrites pour un dictionnaire électronique, demande la distribution des informations dans différentes colonnes, et peut parfois nécessiter la création d'une nouvelle colonne. Voici les corrections des exemples ci-dessus :

Terme	Sigle	Synonyme	Précision
calling modem		*originating modem*	
compression of luminance range			
expansion of luminance range			
switching area			*UK*
passive satellite		*passive communications satellite*	
restoral			*see also reproduction*
end-of-medium	*EOM*		

Tab.18 Correction des entrées mal formées

Pour la récupération des dictionnaires où les catégories des entrées ne sont pas indiquées, il serait intéressant d'étudier une possibilité de marquage automatique bilingue. Par exemple, ayant donné un terme simple d'une catégorie inconnue traduit par un terme simple d'une autre langue, on étiquetterait ces deux termes par les dictionnaires du type DELAF des langues correspondantes, et ensuite on rechercherait l'intersection des ensembles des catégories obtenues. Si l'intersection contient un seul élément, il serait admis comme la catégorie correcte des deux termes. Ces résultats seraient intéressant surtout pour des termes alignés en plus de deux langues.

Deuxième partie

Applications des dictionnaires électroniques des mots composés

Chapitre 8 Acquisition de termes

8.1 Introduction

Dans le chapitre précédent, nous avons décrit le travail nécessaire pour construire un système de dictionnaires électroniques d'une langue spécialisée afin de pouvoir effectuer l'analyse lexicale des textes techniques. Cette analyse, faite à l'aide d'un système comme INTEX, ne peut reconnaître que des termes déjà recensés. Mais la terminologie étant en croissance très rapide, nous voudrions disposer des outils d'enrichissement de nos dictionnaires.

Dans ce chapitre nous proposons une application particulière de certaines fonctionnalité du système INTEX, appelée **LexProTerm**, en tant que méthode d'extraction et enrichissement terminologique en anglais fondée sur l'utilisation de ressources linguistiques et terminologiques déjà existantes qui sont les dictionnaires DELA généraux (voir sections 2.3 et 2.4) et la base LexPro CD Databank (voir section 7.3). La base LexPro est pour nous le point de départ pour la recherche de termes complexes d'un texte, qui sont soit déjà recensés dans la base, soit construits à partir du même matériau lexical que les termes déjà connus. Nous utilisons les termes simples significatifs de chaque domaine, i.e. les substantifs, les adjectifs, les adverbes et les participes apparaissant parmi les termes déjà répertoriés, pour rechercher de nouvelles séquences qui contiennent certains de ces termes simples significatifs, complétés éventuellement par des mots grammaticaux ou néologismes. D'autre part, les dictionnaires DELAF et DELACF de la langue générale nous permettent de proposer parmi les nouveaux candidats termes ceux qui contiennent éventuellement des néologismes[46], des noms propres, etc.

Nous présentons les résultats obtenus dans le domaine de l'informatique, pour lequel nous disposons du lexique de 85 000 termes anglais, simples et composés, décrit dans le chapitre précédent.

Au cours de l'acquisition, nous nous limitons à la recherche de termes complexes, i.e. contenant au moins deux blocs de lettres ou de séparateurs (voir définition 2a section 2.2.2). L'acquisition se fait sur un texte étiqueté (partiellement ambigu) par les dictionnaires DELAF et DELACF spécialisés et généraux, et elle est fondée sur la recherche de motifs syntaxiques dans le texte. Nous verrons que déjà des motifs assez simples peuvent donner des bons résultats (très bon rappel, précision relativement bonne) si les dictionnaires utilisés sont suffisamment riches.

Avant que l'extracteur puisse intervenir, une phase importante est celle de la préparation des ressources de LexPro comme ceci a été décrit dans le chapitre 7 : le nettoyage des dictionnaires, le classement des termes par catégories grammaticales et par la structure syntaxique, le codage des mots inconnus et la flexion automatique des termes simples et composés. Ce travail, seulement partiellement automatisable, a beaucoup d'importance pour la qualité du résultat final. Ceci peut être vu comme l'inconvénient principal de notre méthode. Remarquons néanmoins qu'une fois la phase préparatoire accomplie, nous pouvons utiliser les mêmes dictionnaires comme une base très fiable non seulement pour l'extraction

[46] Dans ce contexte, un néologisme sera pour nous un mot non reconnu ni par un dictionnaire spécialisé, ni par un dictionnaire général.

terminologique, mais aussi pour d'autres tâches, telles que l'automatisation de la traduction, l'indexation documentaire, etc.

Dans la section suivante nous argumentons le choix de notre approche qui peut être classée comme fortement fondée sur des dictionnaires, et indépendante de la taille des corpus traités. La section 8.3 explique l'utilité d'un extracteur terminologique en tant qu'outil d'aide à la traduction. La section 8.4 décrit les principes et les propriétés de la méthode employée. Dans la section 8.5, nous montrons les phases du travail de l'extracteur, et la section 8.6 présente les résultats obtenus en anglais dans le domaine de l'informatique, traité avec les dictionnaires DELAF et DELACF spécialisés. Dans la section 8.7, nous effectuons une analyse comparative de notre outil avec un autre extracteur de référence, Acabit, et dans la section 8.8 nous analysons les aspects novateurs de notre approche. Finalement, dans la section 8.9, nous montrons les perspectives de notre logiciel : les possibilités d'affinement des patrons de recherche, l'adaptation au français et à d'autres langues, la prise en compte de différents formats de textes, etc.

8.2 Pourquoi cette approche ?

De nombreuses sociétés, centres scientifiques et corps administratifs effectuent des travaux visant le recensement et l'unification des terminologies propres à leurs domaines d'activités. En conséquence, des dictionnaires spécialisés et des listes terminologiques de taille souvent très importante sont accessibles à la vente, ou bien fonctionnent comme outils internes, et leur maintenance est parfois confiée à une équipe de terminologues.

D'autre part, des scientifiques en ingénierie linguistique, comme Daille (1994), Bourigault (1994), David et Plante (1990), Smadja (1993) proposent des outils d'extraction automatique de termes, qui dans la plupart des cas admettent le corpus traité, et éventuellement des outils linguistiques généraux (un étiqueteur, un analyseur syntaxique ou une grammaire locale), comme le seul point de départ. Ceci est idéal pour le traitement de nouveaux domaines techniques ou pour des utilisateurs ne possédant pas de ressource terminologique. Néanmoins, ceux qui ont déjà fait un effort de constitution de bases terminologiques, ou bien ceux qui utilisent des dictionnaires de domaines bien définis, n'ont pas la possibilité, avec ces logiciels, de réutiliser leurs ressources déjà disponibles.

ANA (Enguehard et Pantera 1994) est l'un des systèmes qui répondent à ce besoin. Il permet d'introduire un ensemble initial (*bootstrap*) de termes caractéristiques du domaine qui sont le point de départ dans la recherche de nouveaux termes. Cette liste initiale contient d'habitude quelques dizaines de termes, mais il n'y a pas de limite formelle pour la taille du *bootstrap* : ANA a pu fonctionner avec une liste initiale qui contenait les mêmes termes que ceux utilisés dans notre approche (section 8.5.1). Une autre méthode importante fondée sur un lexique spécialisé existant et permettant l'*enrichissement* terminologique plutôt que l'acquisition initiale, est FASTER (Jacquemin 1997). Ses résultats sont de très bonne qualité du point de vue de la pertinence des candidats termes proposés, mais il est nécessaire d'utiliser un corpus de taille importante (l'auteur travaille sur un texte de 1,6 million de mots) afin d'obtenir un rendement satisfaisant (3300 nouveaux termes à partir d'un lexique de 70 000 entrées).

Malheureusement, un très grand corpus du domaine traité n'est pas toujours disponible. En particulier dans le cadre d'aide à la traduction technique, dans lequel nous nous plaçons, les traducteurs ont rarement affaire à des documents qui dépassent 1 mégaoctet de texte. Ils disposent par contre presque toujours d'un ou plusieurs dictionnaires techniques, et éventuellement de lexiques personnels, ou fournis par le client. Il est donc intéressant de

proposer un extracteur terminologique qui permette de réutiliser des listes de termes disponibles, et dont la qualité de résultats ne dépende pas de la taille du corpus traité.

8.3 Extraction terminologique au service d'un traducteur technique

Dans le travail d'un traducteur technique un rôle important est attribué à la constitution d'un *glossaire* du document à traduire. Le traducteur, pas toujours expert du domaine traité, lit le texte en langue source et répertorie tous les termes simples et complexes inconnus ou difficiles à traduire, accompagnés d'exemples de leurs occurrences dans le texte. Cette liste est ensuite envoyée au client qui fournit ses propres traductions ou valide celles proposées par le traducteur. Gouadec (1997) propose une méthodologie très précise de création d'un tel glossaire, appelé chez lui un *concordancier*, et explique son rôle en tant que garant de l'homogénéité terminologique, ainsi que sa valeur contractuelle entre le traducteur et son client.

La constitution et la validation du glossaire du texte devraient en principe être effectuées avant le début de la phase de traduction. Ceci peut entraîner des délais importants, surtout pour des documents volumineux. C'est ici qu'un programme d'extraction automatique de candidats termes peut intervenir. Il analyse le texte et produit instantanément une liste de candidats termes, classés selon leurs fréquences d'apparitions, parmi lesquels le traducteur choisira ceux qu'il voudra inclure dans son glossaire.

Dans ce cadre, l'extracteur doit viser le maximum de rappel possible, car, pour la constitution du glossaire, le traducteur ne travaillera plus sur le texte entier, mais sur la liste de candidats. Il n'aura donc aucune possibilité de « rattraper » les termes qui ont échappé à l'extracteur. En même temps, la liste des candidats ne peut pas être excessivement longue (précision relativement bonne), en particulier le temps de sa consultation ne peut pas dépasser celui de la création du lexique « à la main ». Remarquons néanmoins la difficulté de définir les notions de rappel et de précision dans notre contexte, liée à la question de ce qui doit être considéré comme un bon terme. Pour un terminologue qui dépouille de grandes quantités de textes, un bon terme est celui avec un statut établi constaté dans différentes sources et chez plusieurs auteurs. En revanche, pour un traducteur, un terme qu'il faut retenir est celui qui pose un problème de traduction dans le texte traité. Un candidat retenu par le traducteur pour le glossaire d'un texte donné peut ne plus l'être pour un autre texte, un autre client ou un autre contrat de traduction.

8.4 Principes de la méthode

A partir des ressources lexicographiques très riches dont nous disposons - les dictionnaires de la langue générale du LADL et la base terminologique LexPro - nous avons mis en forme LexProTerm, le prototype d'un système de recherche de termes dans des textes spécialisés, basé sur les algorithmes du système INTEX. LexProTerm peut être classé comme outil d'enrichissement (plutôt que d'extraction) terminologique car il exploite les ressources déjà existantes et les enrichie.

Voici les caractéristiques de notre méthode de recherche de nouveaux termes :

– Elle est fortement fondée sur les ressources lexicographiques, dans la mesure où leurs qualité et complétude ont une importance fondamentale pour les résultats de l'extraction.

– Elle est un outil d'aide à la traduction car son intérêt principal est d'assister un traducteur technique dans la tâche de création du glossaire d'un texte à traduire.

– Elle est indépendante de la taille du corpus (donc non statistique). Ceci est lié à la spécificité du domaine de la traduction, où les textes à traiter sont de taille très variable. Un corpus de très grande taille étant rarement disponible, l'emploi de toute méthode fondée sur un calcul statistique est problématique.

– Elle se limite à la recherche de termes complexes, c'est-à-dire constitués d'au moins deux mots simples séparés par un blanc ou un caractère de ponctuation.

– Elle a été élaborée et testée pour l'anglais (avec un DELAS et un DELAC généraux de près de 90 000 et 60 000 mots respectivement – pour ce deuxième voir chapitre 5) dans le domaine de l'informatique (avec un DELAS et un DELAC spécialisés d'environs 27 000 et 57 000 termes respectivement – voir chapitre 7), mais son adaptation est envisageable pour toute langue pour laquelle on dispose d'un dictionnaire DELAF général, et pour tout domaine couvert par les dictionnaires techniques disponibles.

– Elle utilise la technique de la recherche de motifs syntaxiques dans un texte étiqueté. Une grande partie des algorithmes employés (l'identification des items du texte, l'indexation, l'étiquetage du texte, la recherche du patron) ont été récupérés du système INTEX.

L'idée qui se cache derrière la méthode proposée peut être exprimée par l'hypothèse suivante :

> *La création d'un nouveau terme se fait le plus souvent par une combinaison grammaticalement correcte de termes simples et composés déjà existants.*

Cela signifie que le vocabulaire terminologique d'un domaine peut croître significativement mais son noyau lexical reste à peu près stable. Autrement dit, pour nommer de nouveaux concepts on emploie souvent les mêmes mots, dans de nouvelles combinaisons. Prenons deux exemples de nouveaux termes trouvés dans un corpus informatique :

[332] notification message timeout

[333] user free memory

Dans le premier cas les trois composants simples du terme complexe étaient déjà recensés dans nos dictionnaires spécialisés. Dans le deuxième cas, le mot *free* ne constituait pas une entrée indépendante du dictionnaire de l'informatique mais aussi bien *user* que *free memory* s'y trouvaient. Dans les deux cas, des nouveaux termes ont été donc créés par combinaison de termes existants.

8.5 Phases de l'extraction

Nous ramenons le problème de l'extraction de termes à celui de la recherche de patrons syntaxiques dans un texte, comme ceci a lieu dans le système INTEX. Le texte est d'abord soumis à l'analyse lexicale qui attribue à chaque unité atomique une ou plusieurs étiquettes syntaxiques provenant des dictionnaires utilisés. Ensuite, dans le corpus ainsi étiqueté, nous cherchons toutes les séquences qui correspondent au patron syntaxique donné. Ces séquences seront les candidats que l'utilisateur va pouvoir valider, i.e. décider s'ils sont ou non des termes.

Le schéma de fonctionnement de l'extraction, qui est une variante du schéma général du traitement par INTEX (Fig.7, page 43), est montré sur la figure Fig.42 (les dictionnaires DELAF et DELACF entourés des rectangles dessinés en gras sont consultés en mode

prioritaire – voir section 2.9). Nous pouvons voir que le nombre de données en entrée est le plus important dans l'étape de la recherche des mots simples et composés du texte. En effet, les résultats de cette étape sont décisifs pour l'efficacité de la méthode.

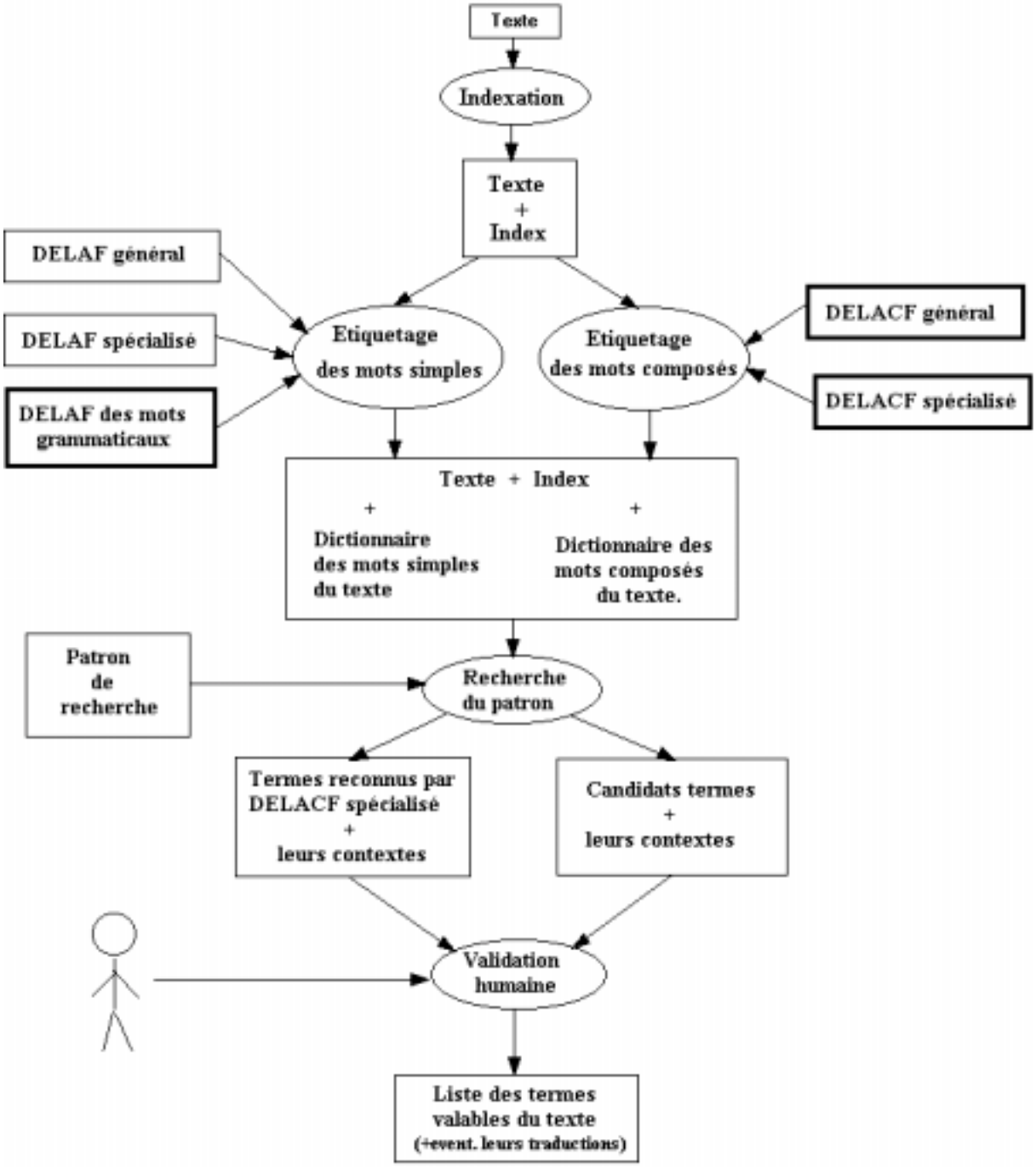


Fig.42. Schéma de fonctionnement de l'extraction.

Quatre algorithmes les plus importants utilisés - l'indexation (i.e. la création du fichier inverse du texte), l'analyse lexicale des mots simples, l'analyse lexicale des mots composés, et la recherche de patrons - ont été récupérés du système INTEX. Les patrons syntaxiques de l'extraction sont représentés, comme les dictionnaires, sous format d'automates finis. Pour les détails de l'implémentation, consulter Silberztein (1997).

8.5.1 Etiquetage du texte

Deux phases préliminaires du traitement sont celles de l'identification des items (suites contiguës soit de lettres, soit de séparateurs) du texte, et de la constitution de l'index (fichier inverse) qui, pour chaque item, donne la liste de toutes ses occurrences. L'analyse lexicale s'occupe ensuite de la reconnaissance des mots simples et composés du texte selon 5 dictionnaires DELAF/DELACF et deux niveaux de priorité.

Les mots simples sont recherchés dans 3 dictionnaires : le DELAF général et le DELAF spécialisé, décrits dans les chapitres 5 et 7, ainsi qu'un dictionnaire des mots grammaticaux, qui a le même format que le DELAF, et qui contient près de 500 prépositions (*about*, *after*, *through…*) déterminants (*the*, *no*, *one*…), conjonctions (*though*, *and*, *or*…), adverbes (*above*, *almost*, *yet*…), pronoms (*who, another, few*…) et certains verbes (*are, can, have, may, will*…). A ce petit lexique est attribuée une priorité supérieure aux deux autres dictionnaires DELAF. Cela signifie que chaque mot simple du texte est d'abord recherché parmi les mots grammaticaux, et seulement s'il n'y figure pas, sa recherche est poursuivie dans les DELAF général et spécialisé. Cette mesure a été introduite pour éviter le superflu des séquences incorrectes extraites plus tard par le patron syntaxique : il n'est pas rare qu'un auteur décide de fournir dans son dictionnaire spécialisé les traductions de certains mots grammaticaux, même si elles sont, en principe, universelles. Ceci fait apparaître dans notre DELAF spécialisé des entrées non significatives du domaine. Le dictionnaire prioritaire nous garantira que ces entrées ne participeront par à la recherche de patrons, fondés essentiellement sur l'étiquette *+Spec* comme nous le verrons dans la section suivante.

Il arrive qu'un mot, qui dans la plupart des cas a une fonction grammaticale, puisse avoir dans un langage spécialisé une signification spécifique. Par exemple, en informatique, *or* désigne une opération logique, donc il est, en principe, incorrect d'interdire à l'analyseur lexical l'accès à l'étiquette *or,.N+Spec:s*. Ceci ne permettrait pas la reconnaissance de certains bons candidats termes informatiques, comme *or gate*. Nous espérons néanmoins, que le silence ainsi introduit est minimal, au moins pour les domaines bien couverts par les dictionnaires LexPro, car les termes « curieux » comme *or gate* figurent déjà dans le DELACF spécialisé, et ils participeront éventuellement à la recherche des candidats plus larges comme *exclusive-or gate.*

Parallèlement à la reconnaissance des mots simples, l'analyseur lexical consulte les deux dictionnaires des mots composés : DELACF général et DELACF spécialisé. Les deux DELACF ont priorité sur tous les autres dictionnaires. Ceci veut dire, que si une séquence contiguë d'items du texte est reconnue en tant qu'entrée du DELACF spécialisé ou général, elle est dans la suite traitée *en bloc,* i.e. on ne cherche plus à étiqueter ses sous-séquences par les autres dictionnaires, et dans la recherche de patrons elle sera équivalente à un mot simple.

L'analyse lexicale produit deux dictionnaires associés au texte de départ - le dictionnaire des mots simples et celui des mots composés reconnus dans le texte – dont le format est identique à celui des DELAF et DELACF généraux et spécialisés. Les unités d'un ou plusieurs items reçoivent une ou plusieurs étiquettes grammaticales, pour lesquelles nous n'effectuons aucune désambiguïsation, mis à part l'utilisation des dictionnaires prioritaires. Ainsi un mot se retrouve souvent avec 2, 4 ou 6 étiquettes, dont certaines identiques au trait *Spec* près, par exemple le dictionnaire des mots simples du texte peut contenir des entrées suivantes :

[334] registers, register.N+Spec:p

[335] registers, register.N:p

[336] registers, register.V+Spec:P3s

[337] registers,register.V:P3s

L'ambiguïté entre les étiquettes [334] et [335], ainsi que [336] et [337] ne pose pas de problèmes pour la reconnaissance du patron que nous avons choisi. En revanche, l'attribution par le DELAF spécialisé de catégories différentes pour le même mot (ambiguïté entre [334] et [336]), peut être à l'origine d'un certain nombre de candidats termes incorrects.

Les entrées du dictionnaire des mots composés du texte peuvent aussi être ambiguës, si elles figurent à la fois dans le DELACF général et dans le DELACF spécialisé, ou bien si un terme complexe a réellement plusieurs emplois avec des catégories différentes. Là aussi seul ce dernier type d'ambiguïté peut influencer les résultats de la recherche du patron syntaxique.

8.5.2 Recherche de patrons

Nos patrons de recherche se présentent sous forme d'automates à états finis, dont l'alphabet sera celui des étiquettes grammaticales attribuées aux unités du texte dans la phase de l'analyse lexicale. Nous allons toujours chercher à extraire les séquences contiguës et maximales décrites par les patrons, sans nous préoccuper de l'existence de sous-termes ou insertions éventuelles à l'intérieur de ces séquences.

Nous avons mis au point un patron qui est représenté par le graphe sur l'illustration Fig.43. Analysons quelques exemples de séquences extraites par les différents chemins du graphe. La branche centrale contenant l'étiquette <*N+Spec:s*> permet de trouver toutes les suites de noms spécialisés. En effet, comme le montrent nos dictionnaires informatiques, les termes complexes du schéma $N_1N_2...N_k$ (avec k>=2), e.g. *access frequency, program reference table, data transmission control unit*, sont de loin les plus nombreux. Nous admettons l'insertion éventuelle de la marque « *'s* » ou « *'* » du génitif, ainsi que du séparateur « / », entre deux noms, pour ne pas manquer les candidats du type *Windows NT User's Guide, server's hostname, matrix' mode, I/O activity*. La contrainte sur le nombre dans l'étiquette <*N+Spec:s*> a été introduite pour éviter le bruit trop important provenant des ambiguïtés entre les verbes à la troisième personne du singulier et les noms au pluriel, comme dans les contextes suivants (les séquences extraites à tort sont soulignées) : *the analyzer displays information on the following..., an array supports write caching if it has ..., the hit ratio shows cache efficiency and ...* Cette contrainte risque d'introduire un certain silence, car il est en principe possible, qu'un terme du type $N_1N_2...N_k$ contienne un nom au pluriel sur une des positions 1…k-1, comme ceci a lieu dans des termes déjà connus, e.g. *active contents type, advanced communications system, american national standards institut.*

Les deux chemins supérieurs du graphe, contenant les étiquettes <*ADV+Spec*>, <*A+Spec*> et <*V+Spec:K*>, assurent la prise en compte des adjectifs, participes passés et adverbes spécialisés à l'intérieur des séquences du type *AN, ANN, NAN, AdvVN etc.*, comme *long return, parallel access array, storage system's physical disks, locally attached arrays.*

La partie du graphe, utilisant les étiquettes <*MOT*> (n'importe quel forme simple) et <*NB*> (nombre), permet d'extraire les mots liés par le trait d'union ou le soulignement, qui marquent souvent le caractère figé des séquences concernées, à condition qu'il n'y ait pas d'espace autour de ces séparateurs (ceci est exprimé par le signe #). Cette branche du patron correspond à des candidats comme *DAE-to-DAE interconnection, dual-initiator/dual-bus configuration, 512-byte data block.*

Le nœud du graphe étiqueté par le symbole *<!DIC>* est celui qui donne la possibilité de prendre en compte les néologismes, i.e. les mots (communs et propres) non reconnus ni par le DELAF général ni par le DELAF spécialisé. Parmi les exemples de ce type trouvés dans nos corpus se trouvent (les néologismes sont soulignés) : *OpenManage Data Administrator, midplane connectors, nonmirrored write, powerup initialization sequence*.

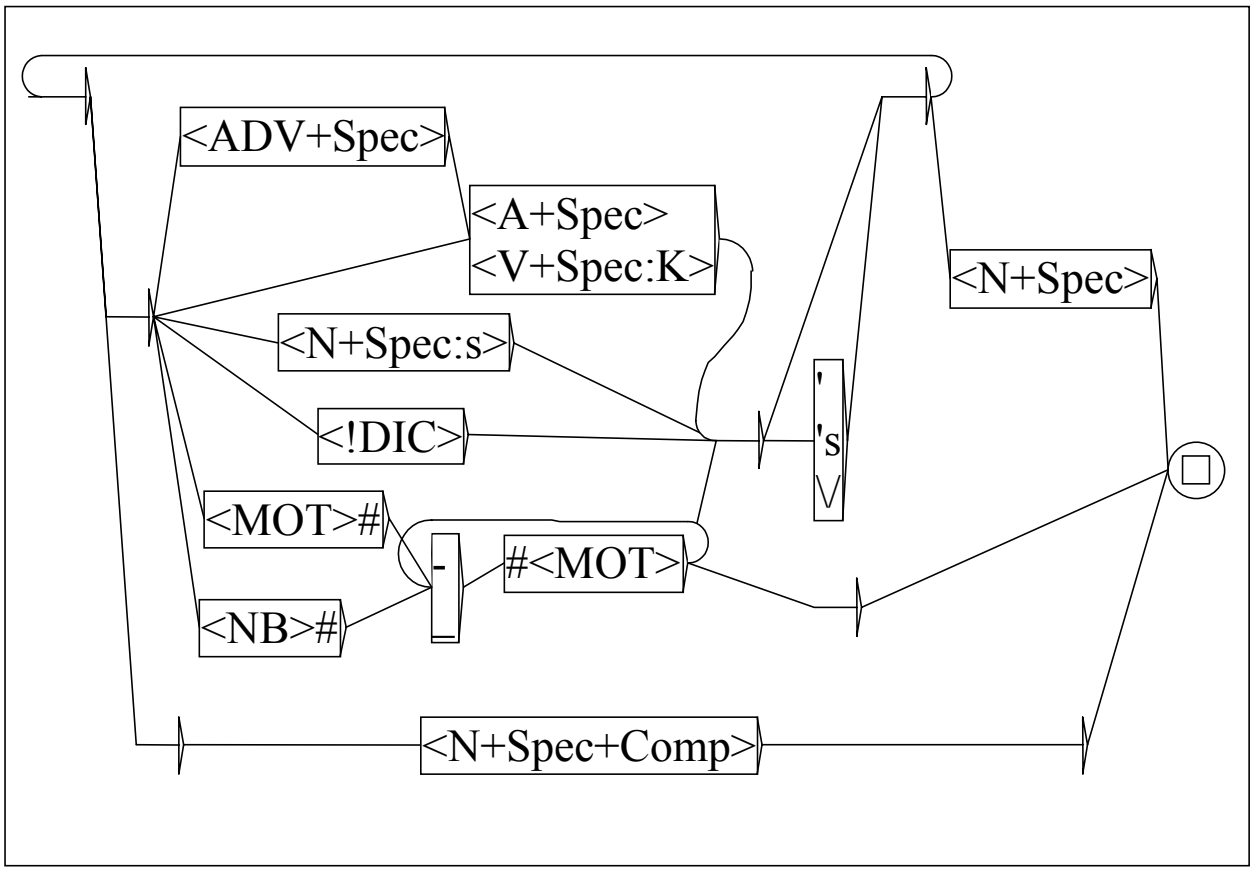


Fig.43. Patron de recherche de nouveaux termes.

Finalement, le chemin inférieur du graphe permet, grâce aux traits *+Spec+Comp,* d'extraire les noms composés terminologiques reconnus déjà par notre DELACF spécialisé au cours de l'analyse lexicale, qui n'ont pas encore été inclus dans des fréquences plus longues extraites par les autres parties du patron. Ces composés, étant des termes établis du domaine, ont une grande chance d'être retenus par l'utilisateur pour le texte donné.

Remarquons que toutes les étiquettes du graphe contenant le trait *+Spec*, à part celle avec en plus le trait *+Comp*, peuvent correspondre non seulement à des mots simples spécialisés, mais aussi à des composés, ce qui permet la reconnaissance des surcompositions obtenues par ajouts de nouveaux modifieurs ou têtes à des termes déjà connus, comme le montrent les candidats suivants (les composés existants déjà dans le DELACF spécialisé sont soulignés) : *ac power distribution, disk-based application, user free memory*.

La mise au point du graphe ci-dessus a été faite d'une façon expérimentale, grâce à de nombreux allers-retours entre le patron de recherche et un corpus informatique de 700 kilo-octets fourni par un traducteur technique. Nous avons essayé de trouver un juste milieu entre le rappel et la précision introduits, que nous essayons d'estimer dans la section 8.6 et 8.7.

8.5.3 *Validation*

L'illustration Fig.44 présente l'interface de la phase de validation. Cette validation est effectuée par le traducteur qui utilise le logiciel pour créer son glossaire de traduction.

Après l'ouverture du texte, a lieu la phase d'extraction décrite dans la section précédente. Ensuite, les séquences extraites sont regroupées par variantes orthographiques : deux séquences sont considérées comme variantes, si elles sont égales à l'emploi des minuscules et des majuscules près. Pour chaque ensemble de variantes du même candidat, celle qui emploie le moins de majuscules est choisie comme la forme représentative. Toutes les formes représentatives sont triées selon les fréquences de leurs variantes dans le texte et divisées en deux listes : « Termes connus » et « Nouveaux termes », selon si elles figurent ou non dans le DELACF spécialisé (et sont donc des termes connus). La sélection d'un candidat de chaque liste entraîne l'affichage de toutes ses occurrences avec leurs contextes gauche et droit de longueurs réglables. L'utilisateur peut consulter la liste de candidats soit dans l'ordre alphabétique (option « Toutes » sur la liste « Fréquences »), soit selon leurs fréquences. Dans le deuxième cas, seuls les candidats ayant la fréquence sélectionnée s'affichent.

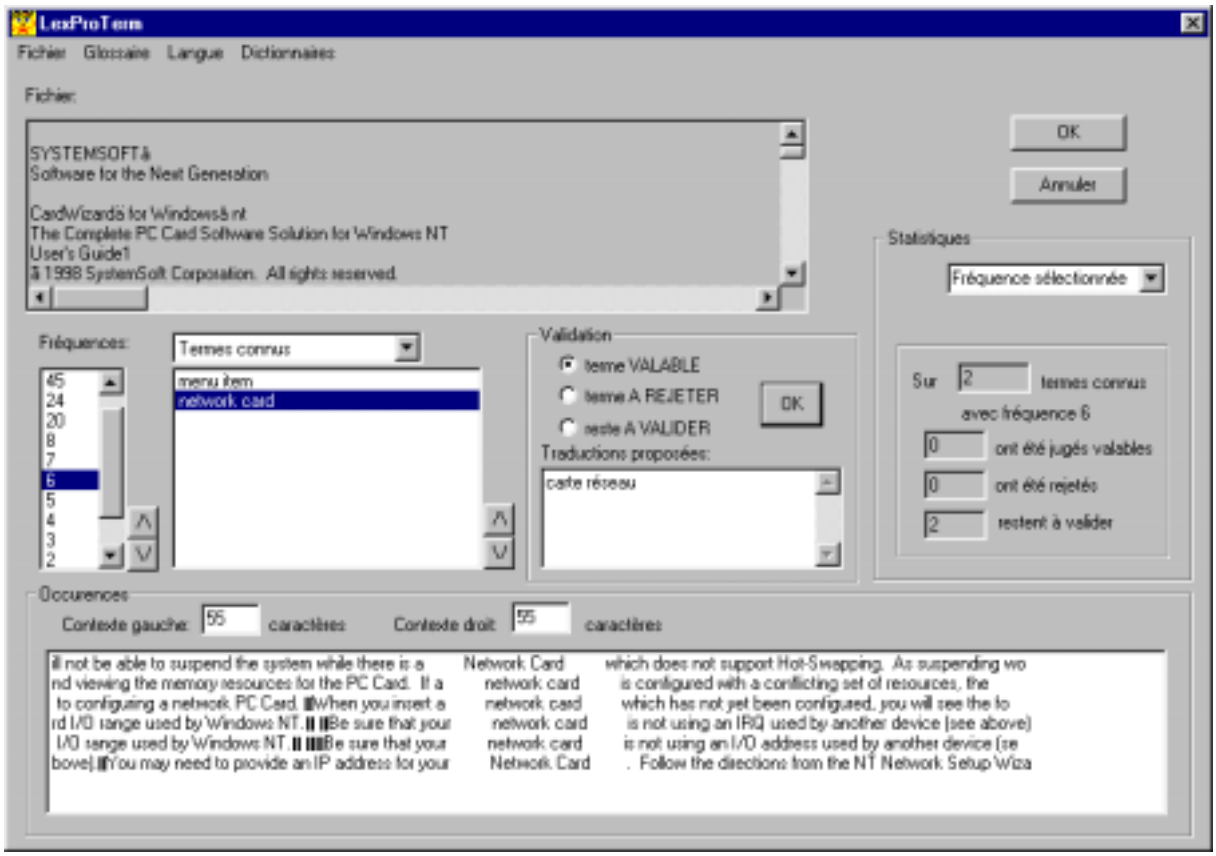


Fig.44. Interface de validation.

Le rôle principal de l'utilisateur est de valider les candidats termes en activant l'un des trois boutons du milieu de l'écran (on choisit le bouton intitulé « reste A VALIDER », si l'on n'est pas sûr du statut d'un candidat), et éventuellement de proposer une ou plusieurs traductions pour chaque terme jugé valable. Les statistiques à droite de l'écran indiquent le nombre de candidats déjà retenus ou rejetés et de ceux qui restent à valider. On peut interrompre la validation à tout moment. Typiquement, on ne va examiner que les fréquences les plus

élevées, mais cette stratégie n'est pas toujours la bonne : de nombreux candidats corrects n'apparaissent qu'une seule fois, même dans des corpus volumineux.

Les résultats finaux de la validation peuvent être exportés dans un fichier lisible par un logiciel permettant la consultation et la création de bases terminologiques (comme Access etc.) afin de mettre en forme le glossaire de traduction du texte (voir section 8.3).

8.6 Premiers résultats

Pour estimer l'efficacité de notre logiciel, nous nous servons des deux critères habituels, ceux de *précision* et de *rappel*. La précision est définie comme la proportion de bons termes parmi tous les candidats termes proposés par l'extracteur. Le rappel signifie la proportion de bons termes proposés par l'extracteur parmi tous les termes existant dans le texte traité.

Nous avons déjà mentionné à la section 8.3, qu'il était difficile de décider si une séquence est ou non un terme dans le contexte de création du glossaire d'un texte à traduire. Néanmoins, nous avons fait un test pouvant nous donner des premières indications quant à la qualité de notre outil. Il a été réalisé sur un petit corpus de 52 kilooctets (8500 mots) de texte anglais sur le domaine informatique, fourni par un traducteur technique.

Nous avons d'abord effectué un prétraitement du corpus, qui consiste à marquer manuellement toutes les occurrences de termes. Ce choix a dû être parfois arbitraire, car, à part les termes informatiques connus, comme *AC power*, *hard disk*, *power management,* nous avons sélectionné certaines séquences sans statut terminologique établi, mais devant être, à notre avis, traitées comme unités de sens au cours de la traduction, par exemple : *cleanup feature, easy-to-read displays, non-network function, active termination device*. Nous prenons en compte la séquence maximale, sans rechercher ses sous-termes éventuels. Le glossaire du texte ainsi obtenu, contenant 839 occurrences[47], a été comparé aux listes de séquences extraites du même texte par notre extracteur. Parmi les 839 termes, 240 ont été reconnus par le DELACF spécialisé, et 450 ont été extraits par le patron syntaxique, ce qui donne le rappel égal à 82%.

Cette valeur, pas assez élevée du point de vue de notre application, est due en grande partie aux limites de l'analyse linguistique dans notre système, car nous n'effectuons ni de désambiguïsation locale ni d'analyse syntaxique globale. David et Plante (1990) démontrent que les problèmes du bruit et du silence dus à l'ambiguïté catégorielle des mots et à une mauvaise reconnaissance de frontières de syntagmes sont inévitables dans tout système qui n'effectue pas de analyse syntaxique complète de la phrase. En effet, pour diminuer le bruit trop important, nous sommes obligée, d'introduire des limites dans le patron de recherche. Les termes non reconnus sont entre autres ceux qui contiennent des prépositions (46% de cas), comme *PC Card support for Windows NT*, contiennent des noms au pluriel sur des positions non terminales (15%), comme *options menu*, sont contenus dans des séquences plus longues (20%), comme *PC Card information screen displays* (le bon terme est souligné, *displays* a été extrait à tort). Ce dernier exemple montre le problème très important en anglais d'ambiguïtés entre les noms et les verbes, renforcé encore dans le domaine de l'informatique par le phénomène fréquent de conversion (voir section 5.4.4) de noms en verbes ou de verbes en noms. Par exemple, le mot *network,* fonctionnant dans la langue générale en tant que nom, gagne un sens verbal dans la langue spécialisée : *to network*, avec les formes fléchies associées : *networks*, *networked, networking*. Le phénomène inverse est encore plus courant :

[47] Ce nombre se réduit à 417, si l'on compte une seule fois les différentes occurrences et versions orthographiques du même terme.

de nombreux verbes deviennent des noms désignant l'action de ces verbes. Ainsi l'on obtient *an interrupt, a merge, a reset, an assert,* qui peuvent aussi se mettre au pluriel, ambigu avec la troisième personne du singulier des mêmes verbes.

Pour évaluer le taux de précision de l'extraction, il faut comparer le nombre de candidats termes corrects avec celui de tous les candidats proposés. Puisque l'utilisateur ne consultera chaque candidat qu'une seule fois, nous faisons ce calcul, contrairement à celui du rappel, sur les listes sans doublons. Le nombre de toutes les séquences uniques extraites par le patron est égal à 644 (100 séquences proviennent du DELACF spécialisé). Dans cet ensemble, 339 candidats (dont 87 entrées du DELACF) sont pertinents, donc le taux de précision est égal à 53%. L'utilisateur retiendra donc à peu près 1 candidat sur 2, ce qui nous semble raisonnable pour le travail de constitution du glossaire de texte à traduire.

8.7 Comparaison avec Acabit

Pour mieux rendre compte de l'efficacité de notre outil d'extraction terminologique, nous avons effectué un test comparatif avec un extracteur de référence, **Acabit** de B. Daille (1994), dont certains principes de fonctionnement s'approchent de ceux que nous avons admis.

Acabit effectue l'extraction initiale, i.e. il n'admet aucune connaissance a priori de la terminologie du domaine traité. Le corpus, l'étiqueteur grammatical, et les grammaires locales de syntagmes nominaux de la langue sont ici les seuls points de départ pour l'extraction. L'algorithme est fondé sur des méthodes à la fois linguistiques et statistiques :

- des patrons linguistiques, sous forme de grammaires locales à états finis, semblables à ceux que nous utilisons, sont employés pour présélectionner des syntagmes nominaux bien formés;
- les séquences présélectionnées par les patrons sont ensuite soumises à un calcul statistique, afin de n'en retenir que celles qui satisfont un certain seuil de fréquence (fixé à deux) et de les ordonner selon une mesure appelée le coefficient de vraisemblance.

Afin de pouvoir comparer les performances des deux extracteurs terminologiques, nous avons choisi un corpus (IBM 1997-99) du domaine de l'informatique, et plus précisément de l'architecture des ordinateurs. Il est constitué des textes du *IBM Journal of Research and Development* accessible par Internet, et il contient 280 mille formes simples, soit plus de 400 pages (1,68 mégaoctets) de texte.

8.7.1 Résultats de LexProTerm

Nous avons analysé manuellement la liste de candidats termes extraits par notre outil et la liste de leurs occurrences dans le corpus de test IBM (1997-99). Le nombre de candidats était de 22 766 et le nombre total d'occurrences de 35 118. Un candidat était classé comme pertinent s'il apparaissait au moins une fois dans le corpus en tant que terme valable. Le tableau Tab.19 contient un résumé des résultats obtenus par notre algorithme. Nous remarquons que la précision atteinte aussi bien au niveau de candidats différents (47%) qu'au niveau de leurs occurrences (59%) confirme approximativement les résultats du premier test (53%) décrit dans la section 8.6. Quant au taux de rappel, nous n'avons pas pu effectuer son calcul pour un si grand corpus (pour ceci nous aurions dû d'abord marquer manuellement toutes les occurrences de termes sur 400 pages de texte). Nous pouvons néanmoins donner un indice sur la densité des séquences extraites par LexProTerm : le nombre d'occurrences pertinentes extraites étant de 20 632, et le nombre de ligne du corpus de 27 000, il y a une occurrence pertinente sur trois-quarts de lignes du corpus.

	Candidats extraits	Candidats pertinents			Précision
		Inconnus	Connus	Total	
Occurrences	35 118	15 875	4 757	20 632	59%
Formes différentes	22 766	9 297	1 408	10 705	47%
Formes différentes avec fréquence 1	18 216	6830	771	7601	42%
Formes différentes avec fréquence supérieure à 1	4551	2467	637	3104	68%

Tab.19 Résultats de l'extraction par LexProTerm

Plus de 22 000 candidats à analyser manuellement constitue une tâche assez lourde dans le procès d'extraction de termes. Il est possible qu'un utilisateur, qui dispose d'un temps trop restreint, choisisse de ne tenir compte que des candidats les plus fréquents. Si l'on fixe le seuil de fréquence à 2, comme ceci est le cas dans Acabit, le taux de précision atteint 68%. Mais le prix à payer est une baisse considérable de rappel, car les termes apparaissant une seule fois sont plus de deux fois plus nombreux (7601) que les autres (3104).

Parmi les candidats pertinents extraits par LexProTerm :

- un grand pourcentage est constitué de séries de termes composés construits autour du même terme simple, par exemple 134 termes extraits ont pour modifieur le terme simple *cache*, 103 autres le terme *chip*, etc. :

  [338] cache cache associativity, cache coherence protocols, cache-hit latency,…

  [339] chip-level test case, chip delays, chip-to-chip interconnection,…

- 44% de la totalité de termes contiennent plus de deux constituants, et la longueur maximale obtenue est de 7 constituants :

  [340] architectural-level instruction stream test-case generator

- certains termes valables ont été extraits au pluriel alors qu'ils devraient être proposés au singulier ; ceci a été dû à l'inexistence de traitement de la coordination par notre système, par exemple dans la phrase suivante :

  [341] Additional packet exchanges allow I/O channel adapters to manipulate interrupt and busy bits in the hub chip.

  le mot *bits* est la tête du syntagme coordonné *interrupt and busy bits*, mais le patron d'extraction ne prévoyant pas d'occurrence d'une conjonction, seule la séquence *busy bits* a été extraite – et elle est proposée, à tort, au pluriel.

- certains termes apparaissent dans le corpus aussi bien au singulier qu'au pluriel, comme :

  [342] ABIST engine, ABIST engines

  et ces deux cas sont considérés comme termes indépendants, car nous n'avons pas développé d'outil de lemmatisation de candidats termes complexes.

L'analyse des candidats non pertinents extraits par LexProTerm permet d'observer plusieurs phénomènes étant à l'origine du bruit et du silence :

- Certaines occurrences de termes connus ne sont pas pertinentes, par exemple le terme *new ligne* signifie un caractère à un code particulier, mais dans le contexte ci-dessus cette séquence apparaît en tant que syntagme libre :

  [343] …in order to bring the new line into the BCE and operand buffers…

- La qualité de ressources lexicographiques utilisées a une grande influence sur la qualité des résultats de l'extraction. Les auteurs (Hildebert 1998, De Solliers 1998) des dictionnaires terminologiques que nous avons convertis en des dictionnaires DELAF et DELACF (chapitre 7) pour les buts de l'extraction, ont introduit des mots simples de la langue générale qui n'ont pas de sens particulier dans le domaine de l'informatique. Ils s'agit entre autres des noms « neutres » comme :

  [344] problem, issue, criteria, impact, contribution, portion, reason, exemple, situation, effort, etc.

  ainsi que de « vrais » adjectifs (ou adjectifs prédicatifs), dans le sens de Levi (1978, pp. 15-17), comme

  [345] successfull, necessary, special, typical, appropriate, important, etc.

  Ces mots provoquent soit l'extraction des syntagmes libres (°*important aspects*[48], °*performance issue*) et des mots composés généraux (°*person-month*), soit le marquage incorrect de frontières des termes (°*appropriate ABIST macros*, °*host adapter portion*) où seules les séquences soulignées devraient être extraites.

- Il n'est pas toujours correct de s'appuyer sur les codes flexionnels du DELAS général pour automatiquement étiqueter et fléchir les termes simples. Ceci concerne particulièrement les adjectifs, comme *long* qui désigne un type de format de données, et apparaît dans les termes composés extraits à raison comme :

  [346] long divide, long double type, long operand

  Cet adjectif, qui est aussi un mot simple général, existe dans le DELAS général avec le code flexionnel A2, ce qui indique sa possibilité de gradation par rajout des terminaisons *–er* pour le comparatif, et *–est* pour le superlatif. Le code A2 a été automatiquement recopié pour le terme simple *long* dans le DELAS spécialisé. Ainsi, dans le DELAF spécialisé apparaissent *long, longer* et *longest* alors que les deux dernières formes n'ont pas d'emploi particulier dans le domaine de l'informatique. Ceci entraîne l'extraction de candidats non pertinents comme :

  [347] °longer LRU code field, °longest decimal representation.

- Par le choix de méthode d'extraction nous ne sommes pas à l'abri des cas ou un mot commun fréquent en anglais a un sens spécialisé dans le domaine technique traité. Par exemple, le nom *key* apparaît en position de modifieur dans 30 syntagmes nominaux extraits à tort, et dans 48 syntagmes extraits à raison. Dans ce premier cas, par exemple dans :

[48] Le symbole « ° » signifie que la séquence qui le suit n'est pas un candidat terme pertinent, indépendamment du fait qu'elle est ou non un syntagme correct de l'anglais.

[348] °key role, °key member, °key decision, °key internal cache management concepts

key a son sens commun - « le plus important, le principal » - indépendamment du sens du nom qu'il modifie. Dans le deuxième cas il concerne le concept d'une clef dans le domaine de la cryptographie :

[349] key generation, key translation, key attribute, key enabler

– Il existe de nombreux cas de syntagmes qui sont non figés, mais qui sont synonymiques avec certains termes. Par example :

[350] °format changes, °formal way to verify

sont des syntagmes libres utilisés dans le corpus en alternance avec les termes connus :

[351] format conversion, formal verification.

Pour les buts de recherche documentaire, les exemples [350] devraient être considérés comme variantes de termes et donc comme candidats pertinents. Nous, dans le contexte de l'extraction terminologique et d'aide à la traduction, les classons comme non pertinents. D'autres nombreux exemples de ce type sont des séquences non figées comme :

[352] °fault-tolerant design approach, °LSAR register pair, °full-chip checking job, °field solver tool

qui contiennent des termes en position de modifieur et dont la tête pourrait être effacée sans modification du sens :

[353] fault-tolerant design, LSAR registers, full-chip checking, field solver

– Le manque de désambiguïsation de mots est à l'origine d'un grand ensemble de candidats non pertinents. Souvent la frontière d'un terme est mal repérée à cause d'un verbe à la troisième personne du singulier ambigu avec un nom spécialisé au pluriel, comme dans les exemples ci-dessous :

[354] The S/390 G5 system controller makes those views obsolete by providing unparalleled system recovery and data fault tolerance…

[355] The control flow includes command/status buses and finite-state machines.

Finalement, il est important d'admettre que seul un expert du domaine précis concerné par le corpus est capable de déterminer avec assurance le statut d'une séquence extraite. Nous-même avons eu de nombreux doutes lors de l'analyse des candidats, surtout au sujet de certains mots simples qui semblent intermédiaires entre la langue générale et la langue spécialisée, comme :

[356] mechanism, coverage, element, performance, failure, component, single

Leurs occurrences dans les séquences extraites semblaient affaiblir sensiblement le degré de figement de ces séquences (alors que les sous-séquences soulignées sont sûrement des termes) :

[357] single exception summary bit, bus-snooping mechanism, dc stuck-at-fault test coverage, coupled-system performance, host hardware component

En résumé des observations faites ci-dessus, nous pouvons envisager au moins trois possibilités d'amélioration de notre méthode d'extraction :

- l'attachement d'un outil de désambiguïsation locale pour pouvoir repérer plus fidèlement les frontières de syntagmes, et ainsi éviter les cas du type [354]-[355],
- une meilleure préparation des ressources terminologiques (DELAS, DELAF, DELAC et DELACF spécialisés) utilisées pour l'étiquetage des corpus, entre autres au niveau des entrées du type [344]-[345]; cette tâche doit être faite manuellement,
- l'introduction d'un module de lemmatisation de termes complexes afin de rattacher différentes formes fléchies du même terme (voir exemple [342]) et ainsi diminuer le nombre de formes extraites sans faire baisser le taux de rappel.

8.7.2 Résultats d'Acabit

L'outil de B. Daille (1994) a trouvé 2133 candidats termes différents dans le corpus de test IBM (1997-99). Nous les avons analysés manuellement et nous en avons retenu 1423 comme termes valables, d'où le taux de précision égal à 67%. Nous n'avons pas effectué le même calcul au niveau des occurrences de ces candidats termes, mais nous savons que chacun d'eux est apparu au moins deux fois dans le corpus.

Parmi les candidats pertinents extraits la grande majorité est constituée de termes contenant deux unités lexicales pleines (noms, adjectifs, adverbes, verbes). En effet Acabit se concentre sur l'extraction de ce type de termes appelés « termes de base », et son grand avantage est de relier différentes variantes des mêmes termes. Par exemple, les séquences dans chacun des ensembles suivants :

[358] integration levels, level of integration, levels of integration

[359] permanent failure, permanent failures, permanent physical failure, permanent single-bit failure, failure is permanent

donnent lieu à l'extraction d'un seul terme, respectivement *integration level* et *permanent failure.* Certaines séquences sont néanmoins, pour des raisons que nous ne connaissons pas, extraites deux fois, et proposées en tant que deux candidats différents, l'un au pluriel, l'autre au singulier, comme *control macro, control macros*, ou l'un écrit en minuscules et l'autre en majuscules, comme *computer science, Computer Science*.

Le rattachement de variantes terminologiques est entre autres basé sur l'hypothèse qu'un terme de longueur supérieure à 2 est souvent obtenu à partir des termes de longueur 1 ou 2 par une des deux opérations, selon la terminologie admise par Daille (1994) : l'insertion (insertion de modifieurs ou substitution) ou la juxtaposition (surcomposition, placement de modifieurs en position non initiale). Dans l'exemple [359], *permanent physical failure* peut être obtenu à partir de *permanent failure* soit par insertion de *physical* :

[360] permanent failure + physical ⇒ permanent physical failure

soit par la substitution de *failure* par *physical failure* :

[361] permanent failure (*failure ← physical failure*) *⇒ permanent physical failure*

Acabit permet aussi d'extraire certains candidats termes qui ont plus de deux unités lexicales pleines car une séquence de mots reliés par un tiret est automatiquement considérée comme une seule unité. Parmi les candidats pertinents extrait par Acabit nous retrouvons 203 termes de ce type :

[362] test-case generator, Cache-to-cache latency, data-pattern-dependent jitter

En analysant les candidats non pertinents extraits par Acabit, nous retrouvons :

– des syntagmes libres, tels que :

[363] °great deal, °correct result, °brief description, °bus utilization, °customer satisfaction, °current responsibility

– des syntagmes non nominaux :

[364] °be braodcast, °at the end, °available through IBM, °in electrical engineering, °integrated cryptographic

– des séquences placées à la frontière de deux syntagmes (le candidat extrait est souligné) :

[365] Mr. Wile's previous verification °experiences included storage controller element simulation…

[366] …tester-based °diagnostics work best when the failure is in the scan chain…

– un grand nombre de séquences binaires qui sont des sous-séquences de termes valables plus larges (le candidat extrait est souligné). Ceci est dû à deux phénomènes. Premièrement, un terme composé de longueur supérieure à 2 n'est pas forcément toujours obtenu par une insertion ou une juxtaposition de termes binaires. Par exemple, les termes suivants :

[367] most significant bit, direct attached crypto

sont obtenus, respectivement, par un placement d'un adjectif non terminologique au superlatif *most significant* devant un terme unaire *bit*, et par une élision du terme simple *operations* dans le terme quaternaire *direct attached crypto operations*. Dans les deux termes ternaires ci-dessus Acabit extrait donc, à tort, °*significant bit* et °*attached crypto* comme candidats termes de base.

Deuxièmement, un terme peut être une insertion ou une juxtaposition de termes binaires, mais Acabit n'a pas correctement déterminé quels composants constituent le terme binaire d'origine. Ainsi, dans la surcomposition suivante :

[368] operation-graphe + finite-state machine ⇒ operation-graphe finite-state machine

le candidat °*operation-graphe machine* a été extrait à tort. Egalement, des substitutions comme :

[369] dynamic recovery (*recovery ← CPU recovery*) ⇒ *dynamic CPU recovery*

[370] multiple-input register (register ← signature register) ⇒ *multiple-input signature register*

ont été à l'origine des séquences non pertinentes °*dynamic CPU* et °*multiple-input signature.*

Aussi, le terme obtenu par le placement d'un modifieur binaire devant un terme simple :

[371] fiber optic + connector ⇒ fiber optic connector

a donné lieu à un mauvais candidat °*optic connector.*

Nous avons pu constater, pour certains termes valables de longueur 2 extraits par Acabit, que l'intérêt de leur repérage était mineur par rapport aux termes plus longs qui les contenaient. Par exemple, les deux candidats suivants :

[372] °integrated cluster, cluster bus

ont été extraits, l'un à tort et l'autre à raison. Toutes les 16 occurrences pour chacune de ces deux séquences ont lieu à l'intérieur du même terme ternaire :

[373] integrated cluster bus

dont le fort statut terminologique est confirmé par l'existence de l'abréviation *ICB*. Puisque aucune de ces sous-séquences binaires n'apparaît indépendamment l'une de l'autre, nous croyons qu'il serait plus convenable de considérer le terme ternaire comme terme de base, et non pas comme une insertion ou juxtaposition de termes binaires. De la même façon devraient être considérés comme termes de base les séquences suivantes :

[374] Integrated Cryptographic Facility (*ICRF*), *Integrated Cryptographic Feature* (*ICRF*), *random number generator* (*RNG*), *pseudorandom number generator* (*PRNG*), *absolute address history table* (*AAHT*), *key agreement protocol* (*KEP*), *system assurance kernel* (*SAK*)

et non pas leurs sous-séquences, extraites par Acabit, qui n'apparaissent qu'à l'intérieur des ces premières :

[375] °Cryptographic Facility, °Cryptographic Feature, °number generator, °address history, °history table, °agreement protocol, °assurance kernel

En résumé de l'analyse des candidats extraits par Acabit, nous pouvons constater que cet outil :

- obtient de bons résultats pour les termes valables binaires, car il peut repérer leurs différentes variantes ;
- ne trouve que peu de termes de longueur supérieure à 2, qui sont pourtant très nombreux dans le corpus (témoigne de ceci le fait que 44% de candidats pertinents extraits par LexProTerm sont composés de 3 composants ou plus).

8.7.3 Comparaison

Le tableau Tab.20 présente un résumé sur la précision des résultats d'Acabit et de LexProTerm. Remarquons, que LexProTerm, après l'introduction du seuil de fréquence fixé à deux, arrive à extraire deux fois plus de termes qu'Acabit, avec la même précision que ce dernier.

Nous avons déjà mentionné qu'il serait difficile de déterminer le taux de rappel de l'extraction pour les deux outils que nous avons testés, à cause de la grande taille du corpus. Néanmoins, nous pouvons obtenir certains indices sur le silence propre aux deux méthodes si nous analysons les candidats pertinents qui ont été extraits par l'un de ces outils et non pas par l'autre.

Extracteur	Candidats extraits	Candidats pertinents	Précision
LexProTerm	22760	10677	47%
LexProTerm (avec seuil de fréquence 2)	4551	3104	68%
Acabit	2133	1425	67%
Candidats communs	1432	1163	81%

Tab.20 Précision des deux outils

D'abord, nous constatons que le nombre de séquences extraites par les deux outils atteint 1432, dont 1163 (81%) sont des termes valables. Il y a 262 termes extraits par Acabit et non pas par LexProTerm. Près de la moitié de ce nombre sont des termes binaires qui apparaissent dans des termes plus larges extraits par LexProTerm. Il reste 135 termes que LexProTerm n'a pas extraits pour diverses raisons :

- ils apparaissent dans des candidats non pertinents extraits par LexProTerm, comme *emulation code* (LexProTerm a extrait °*control unit emulation code take* et °*emulation code reduces*), *guard-band range* (dans °*specified guard-band ranges*),
- ils se terminent par des noms qui ne figurent pas dans le DELAF spécialisé, et qui donc ne peuvent pas être reconnus par le nœud final du patron de l'extraction (Fig.43) ; c'est le cas de *multifiber ferrule, jitter budget, leading-edge penalty*, *external randomization, legacy AVPs* etc.
- ils contiennent des mots qui n'apparaissent pas dans le DELAF spécialisé, mais ils apparaissent dans le DELAF général, et de ce fait ils ne peuvent pas être reconnus pas le nœud du patron contenant l'étiquette *<!DIC>,* comme par exemple dans le cas de *dwelling time, decoupling capacitance*, *correctable error, internode bus,*
- leur structure syntaxique n'est pas prévue dans le patron de recherche de LexProTerm, par exemple dans *mode of operation* apparaît une préposition, dans *fencing command* le premier composant est un participe présent du verbe spécialisé *fence*, or le patron de recherche n'admet que la forme du participe passé pour les verbes ; dans *modulo reduction* le premier composant existe dans le DELAF spécialisé en tant que préposition.

Les termes extraits par LexProTerm et non pas par Acabit sont très nombreux (environs 9 500) et il est difficile de les classer tous. Nous en mentionnons seulement quelques types les plus fréquents. Compte tenu des principes de fonctionnement d'Acabit, il est normal que l'ensemble de termes extraits par LexProTerm et non pas par Acabit contienne :

- presque tous les termes de longueur supérieure à deux,
- presque tous les termes qui apparaissent une seule fois dans le corpus.

Quant aux termes binaires avec la fréquence[49] supérieure à 1, LexProTerm en a extrait plus de mille de plus qu'Acabit. Nous y trouvons par exemple *array macro*, *array chip, access key,*

[49] Remarquons qu'un terme n'a souvent pas la même fréquence pour LexProTerm que pour Acabit, notamment dans les cas où un terme binaire peut faire partie d'un terme plus long, ou bien quand il apparaît sous différentes variantes.

active configuration, address conflict, etc. Nous les avons examinés de plus près, car Acabit se concentre sur ce type de termes. Par exemple, la séquence *address conflict* apparaît telle quelle deux fois dans le corpus, et les deux fois Acabit la lemmatise comme suite de deux noms (*address/NN/address/441 conflict/NN/conflict/9061*). Ces termes n'ont probablement pas été retenus à cause de leurs faibles valeurs du coefficient de vraisemblance.

En résumé de ce test comparatif, le tableau Tab.21 rassemble les avantages (marqués «+») et les inconvénients (marqués «-») des deux méthodes d'extraction.

Acabit		**LexProTerm**	
+	bonne précision	+	bon rappel
+	indépendant du domaine (i.e. n'exige pas de liste initiale de termes)	-	exige une liste initiale de termes du domaine, les résultats de l'extraction dépendent de la qualité de cette liste
-	exige un grand corpus	+	indépendant de la taille du corpus
-	n'effectue que l'extraction initiale	+	permet de repérer les termes connus d'une façon sûre et d'enrichir leur liste
+	bons résultats pour les termes binaires	-	beaucoup de bruit au niveau des candidats binaires
-	mauvais résultats pour les termes avec plus de 2 constituants	+	extrait les termes maximaux
+	extrait des termes avec une préposition	-	n'extrait pas de termes avec une préposition
+	lemmatise les constituants des candidats termes et rattache différentes formes fléchies du même terme	-	différentes formes fléchies du même terme sont considérées comme termes indépendants
+	relie les termes avec leurs variantes	-	seules les séquences contiguës sont recherchées dans le texte ; différentes variantes sont considérées comme termes séparés

Tab.21 Comparaison des deux outils d'extraction

8.8 Aspects novateurs

L'originalité de notre méthode d'extraction n'est pas dans les algorithmes employés, car :

- La recherche de patrons dans un texte étiqueté est une technique souvent appliquée dans la tâche d'extraction (par exemple chez Daille (1994) et Auger et al. (1996) en français, ou chez Justeson et Katz (1995) en anglais).
- La méthodologie de construction et d'utilisation des dictionnaires électroniques est celle employée au LADL (voir Courtois et Silberztein (1990)).
- Les principaux programmes informatiques ont été repris du système INTEX.

– L'analyse lexicale du texte n'effectue qu'un minimum de désambiguïsation des mots.

L'originalité de notre approche est de fournir à ces algorithmes des données de haute qualité et complétude (cet aspect est présent aussi dans le système de reconnaissance de termes et leurs variantes chez Jacquemin, Klavans et Tzoukermann (1997)). Nous avons exploité les résultats des travaux d'experts en lexicographie, terminologie et traduction. Leurs dictionnaires, généraux et spécialisés, étant l'effet de l' « extraction » humaine, sont de très bonne qualité du point de vue de la pertinence des mots et séquences qu'ils contiennent. De plus, nous nous sommes penchée sur la préparation de ces ressources, nécessaire pour le traitement automatique : la correction orthographique, le marquage des catégories et des traits flexionnels, la génération des formes fléchies, etc. Ainsi, nous disposons d'un noyau lexical très fiable que nous pouvons ensuite enrichir par une méthode automatique, standard du point de vue algorithmique, mais originale et efficace grâce à la qualité des ressources.

Les autres aspects novateurs de notre méthode sont à voir dans les points suivants :

1) Application de l'extraction dans le domaine de traduction assistée par ordinateur, qui présente des caractéristiques et exigences particulières, telles que :
   - La nécessité de traiter des textes de taille très variée, rarement aussi gros que les corpus auxquels sont traditionnellement appliqués les outils d'extraction. Cette contrainte exclut l'application efficace de toute méthode d'extraction qui comprend des calculs statistiques, telles que Daille (1994), Justeson et Katz (1995), Nakagawa et Mori (1998) et autres, dont un panorama est présenté chez Jacquemin (1997, pp. 24-29).
   - L'importance pour la qualité de traduction d'un très bon rappel des termes extraits. La même condition est prise en compte par Ladouceur et Cochrane (1996), mais leur article ne précise malheureusement pas les algorithmes employés.
   - La spécificité de la notion du terme (voir section 8.3).
2) La variété des ressources utilisées et de leurs rôles dans le processus d'extraction :
   - Le dictionnaire des mots composés terminologiques (le DELACF spécialisé) isole les séquences qui ont un statut terminologique déjà reconnu.
   - Les dictionnaires des mots simples et composés terminologiques (le DELAF et le DELACF spécialisés) fournissent les étiquettes qui sont à la base du patron de recherche (trait +*Spec*).
   - Les dictionnaires des mots composés généraux et terminologiques (le DELACF général et le DELACF spécialisé) permettent de traiter « en bloc » certaines séquences figées (i.e. nous ne cherchons pas de nouveaux termes à l'intérieur des mots composés connus).
   - La complétude du dictionnaire des mots simples généraux (le DELAF général) permet de considérer les mots simples non reconnus comme néologismes du domaine traité et de les prendre en compte dans le patron de recherche.
3) L'utilisation d'un analyseur lexical qui tient compte des mots composés. Cet aspect est absent ou très limité dans les étiqueteurs employés par les extracteurs existants.
4) L'hypothèse que le matériau lexical à l'intérieur d'un domaine est relativement stable par rapport à la croissance très importante de la terminologie. Ainsi, nous admettons que la création d'un nouveau terme se fait le plus souvent par une combinaison grammaticalement correcte de termes simples et composés déjà existants. Cette hypothèse est reflétée dans le

patron de recherche utilisé et confirmée par les résultats des testes. Elle apparaît aussi chez Nakagawa et Mori (1998) , mais les mots simples caractéristiques du domaine y sont recherchés non pas dans un dictionnaire mais dans le corpus par une méthode statistique.

8.9 Perspectives

La méthode d'extraction présentée ci-dessus n'est que le début de notre travail. Il nous reste à mettre en forme tous les dictionnaires LexPro, comme nous le décrivons pour le dictionnaire informatique dans le chapitre 7. Beaucoup de ces dictionnaires sont peu volumineux, et donc le nombre de termes simples, sur lesquels est fondée une grande partie du patron de recherche, peut s'avérer trop bas. Dans ce cas, nous pouvons entreprendre, avant de commencer l'extraction, une mesure supplémentaire d'auto-enrichissement de dictionnaires spécialisés par récupération des termes simples significatifs de chaque domaine, i.e. des substantifs, adjectifs, adverbes et participes apparaissant en tant que composants simples des termes complexes déjà répertoriés. Ces nouvelles entrées, soumise ensuite à la flexion, peuvent compléter les DELAF existants.

Il sera aussi nécessaire d'ajouter, au cours de l'extraction, la lemmatisation des candidats termes, afin de ne plus proposer la même séquence au singulier et au pluriel (e.g. *disk module* et *disk modules*) comme deux candidats indépendants. Remarquons que cette lemmatisation est à faire sur les termes complexes entiers, i.e. seuls leurs constituants caractéristiques doivent être lemmatisés, et non pas tous leurs composants, comme ceci est fait par Daille (1994).

Un problème important qui reste à résoudre est celui des ambiguïtés des mots simples et composés apparues suite à l'étiquetage du texte. Le rattachement à notre système d'un des étiqueteurs disponibles, par exemple à celui de Brill (1994), peut s'avérer difficile à cause de la spécificité des dictionnaires que nous utilisons, entre autres ceux des mots composés. Néanmoins, nous envisageons de tester cette possibilité pour augmenter la précision de notre logiciel.

Nous souhaitons aussi élaborer de nouveaux patrons de recherche. Premièrement, des nouvelles structures syntaxiques, entre autres celles contenant des prépositions, comme *arrangement of slots,* doivent faire l'objet d'une étude détaillée. Deuxièmement, nous voudrions élaborer des méthodes fondées sur l'observation que les listes de termes connus contiennent de nombreuses séries, e.g. *access control group, access control list, access control machine, access control profile, access control register*, etc. Il est probable qu'une séquence contenant le même affixe qu'une des séries recensées soit un bon candidat terme.

Actuellement, notre logiciel ne traite que des fichiers textes au format ANSI. La prise en compte des formats enrichis, e.g. de l'emploi d'une police spéciale pour certaines parties du document, ainsi que le traitement des données textuelles inclues dans des tableaux, illustrations etc., sera un affinement important du point de vue d'un traducteur technique.

Finalement, l'adaptation du logiciel à d'autres langues de LexPro est à envisager. Nous sommes consciente que ceci représente un travail considérable, car les patrons de recherche pour une langue seront rarement utilisables dans une autre langue. Nous espérons néanmoins que, le point fort de la méthode étant la taille et la qualité de nos dictionnaires généraux et terminologiques, nous allons pouvoir fournir des résultats intéressants pour les utilisateurs multilingues.

8.10 Conclusion

LexProTerm est l'un des rares outils qui permettent l'utilisation d'une base initiale de termes et son enrichissement. Mais la préparation de ressources terminologiques nécessaires pour cette méthode est un processus coûteux (voir chapitre 7). Il serait très intéressant d'effectuer une étude sur les possibilités d'automatisation de ce processus.

Nous voyons l'un des avantages importants de notre méthode d'extraction de termes dans le fait que ses résultats[50] ne dépendent pas de la taille des documents, sur lesquels elle est effectuée. En effet, si l'on soumet à l'extraction seulement une partie d'un corpus, les candidats termes proposés seront exactement les mêmes que ceux qui dans la même partie ont été trouvés lors de l'extraction sur le corpus entier. Seule la fréquence des candidats (donc leur ordre de présentation) varie, mais ceci n'affecte pas le contenu de la liste[51].

Le test comparatif de LexProTerm avec le système Acabit a confirmé l'hypothèse sur la création de nouveaux termes à partir de termes connus, admise pour notre méthode (section 8.4), et a prouvé l'utilité de cette hypothèse pour l'extraction terminologique. Les résultats obtenus ont aussi soulevé deux problèmes :

- celui des termes hapax (à fréquence 1) dont le nombre dans des textes est deux fois plus élevé que d'autres termes ; ceci implique que les méthodes statistiques d'extraction, bonnes au niveau de la précision, ne peuvent pas être satisfaisantes au niveau du rappel,
- celui des termes contenant plus que deux constituants qui sont presque aussi nombreux dans des textes que les termes binaires, contrairement à ce que constate B. Daille (1996, p. 127), et qu'il faut donc traiter au même titre que ces derniers.

Notre méthode permet de remédier à ces deux problèmes, mais elle demande aussi des améliorations, telles que l'introduction d'un étiqueteur grammatical, et d'un module de rattachement des formes fléchies du même terme. Néanmoins, dès maintenant nous pouvons obtenir des résultats d'extraction très riches si nous disposons d'un dictionnaire couvrant précisément le domaine traité, et si la terminologie disponible dans ce dictionnaire est assez complète et de bonne qualité.

[50] Nous comprenons ici par résultat d'extraction la liste des candidats retenus par le logiciel avant toute intervention humaine.

[51] Nous avons mentionné que, dans le cadre de la traduction, il est important de ne pas négliger les termes d'une fréquence basse d'occurrences. Si néanmoins l'utilisateur choisit de ne valider que les candidats de fréquences élevées, son résultat final, i.e. la liste validée, dépendra évidemment de la taille du corpus.

Chapitre 9 Correction orthographique

9.1 Introduction

Nous présentons une méthode de correction orthographique développée pour le logiciel LexPro CD Databank en tant qu'outil de consultation de la base terminologique. La méthode est basée sur le format du dictionnaire utilisé – celui de l'automate fini. Elle imite l'algorithme standard de recherche de mots dans un automate, mais elle admet des modifications dans le mot recherché si la recherche normale est bloquée.

9.2 Opérations élémentaires sur des lettres

La méthode élaborée est indépendante de la langue concernée, c'est-à-dire qu'elle fonctionne pour toute langue pour laquelle on dispose d'un dictionnaire DELAF.

Nous admettons, comme ceci a lieu dans les travaux de référence du domaine de la correction automatique de l'orthographe (Damerau 1964, Oflazer 1996), que chaque faute de frappe résulte d'un mot correct par application d'une ou plusieurs opérations élémentaires sur des lettres. Nous distinguons 4 opérations élémentaires :

1) l'insertion d'une lettre : *certificat* > **certificlat* ;

2) l'omission d'une lettre : *immunisation* > **immunisatio* ;

3) le remplacement d'une lettre : *cornichon* > **kornichon* ;

4) l'inversion de deux lettres voisines : *enveloppe* > **evneloppe*.

9.3 Exemple

L'interprétation de l'origine d'une faute peut être ambiguë. Par exemple, pour le mot erroné anglais :

*[376] *apte*

il existe au moins 7 corrections possibles :

[377]	*apter*	: omission de *r*
	apt	: insertion de *e*
	ape	: insertion de *t*
	apse	: remplacement de *s* par *t*
	ate	: insertion de *p*
	ante	: remplacement de *n* par *p*
	pate	: inversion de *p* et *a*

L'algorithme de correction est fondé sur celui de la recherche d'un mot dans un DELAF sous forme d'automate fini. D'abord nous recherchons le mot tel quel dans l'automate et, s'il n'a pas pu être trouvé, nous faisons le retour en arrière (*back-tracking*). A chaque fois que nous retournons à un état visité précédemment, nous essayons de trouver une autre continuation du chemin en admettant l'une des quatre opérations mentionnées ci-dessus. Examinons un extrait du dictionnaire DELAF anglais, Fig.45, contenant entre autres les sept corrections [377] du mot erroné [376].

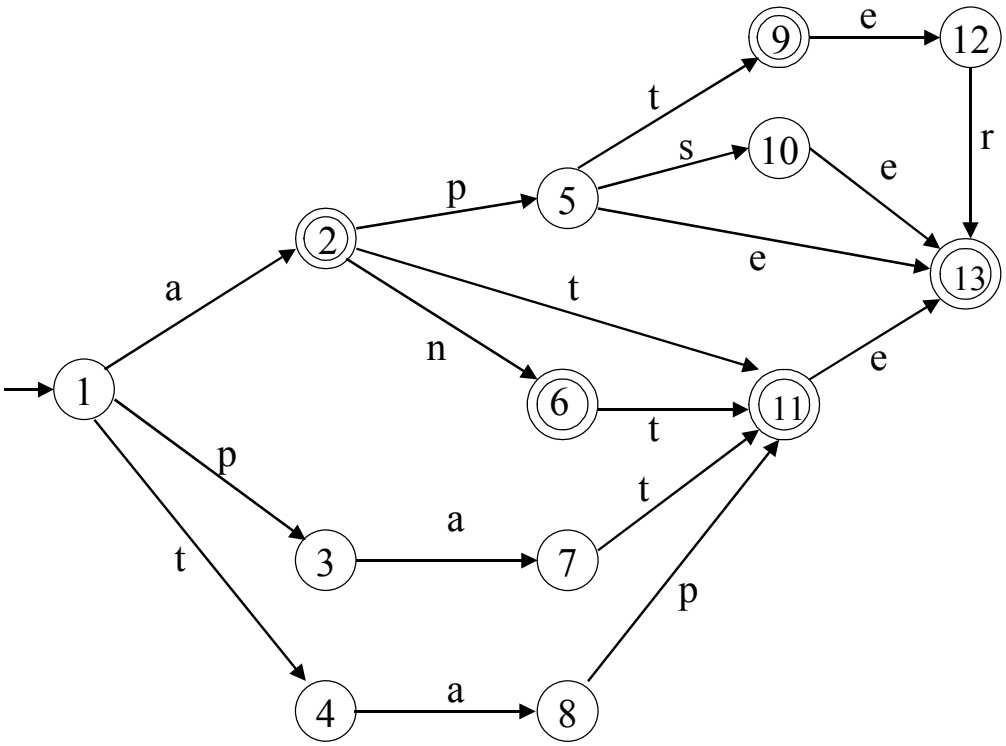


Fig.45. Extrait d'un automate DELAF anglais

Les états terminaux sont marqués par un double cercle. La consultation commence dans l'état initial numéro 1. La recherche du mot *apte* nous amène à l'état 12 qui n'est pas terminal et où aucune transition n'est plus possible puisque la séquence entière a été lue. L'automate étant déterministe le retour en arrière n'est pas nécessaire pour s'assurer que la séquence recherchée n'existe pas dans le dictionnaire. A ce moment-là, nous commençons la « consultation modifiée » qui consiste à chercher des mots semblables en admettant l'une des quatre opérations élémentaires directement après l'une des cinq positions possibles dans le mot d'origine (le début du mot inclu) :

	a	*p*	*t*	*e*
0	1	2	3	4

A la fin de la séquence d'entrée (position 4) une seule hypothèse de source de l'erreur est possible : la dernière lettre, i.e. celle qui devrait se trouver juste après le *e* a été omise. Nous essayons toutes les transitions sortant de l'état courant 12 et menant à un état terminal. Il y en a une seule : (12,*r*,13) et elle donne le premier candidat pour la correction : *apter*. Pour continuer la recherche il faut retourner de l'état 12 à l'état 9 (et de la position 4 dans le mot à la position 3), où nous testons trois interprétations possibles de l'erreur :

- La lettre *e* a été insérée à tort. Puisque le suffixe de *apte* suivant la lettre *e* est vide et l'état courant est terminal, nous obtenons un nouveau candidat : *apt*.
- Une lettre a été omise à tort entre *t* et *e*. Nous essayons toutes les transitions partant de l'état courant 9 et arrivant à un état à partir duquel il est possible de reconnaître le suffixe *e*. Puisque la seule transition débutant dans l'état 9 et celle par laquelle nous venons de retourner, nous n'obtenons aucun nouveau candidat.

– La lettre correcte à la position 4 a été remplacée à tort par la lettre *e*. Nous essayons toute transition qui mène de l'état 9 à un état terminal. Il n'y en a aucune, nous n'obtenons aucun nouveau candidat.

A l'étape suivante nous retournons de nouveau en arrière jusqu'à l'état 5 (position 2 dans le mot) et nous testons 4 hypothèses :

– La lettre *t* a été insérée à tort. Nous cherchons toutes les transitions de l'état 5 à un état terminal en lisant le suffixe *e*. Il y en a une : (5,*e*,13), elle donne un nouveau candidat : *ape*.

– Les lettres *t* et *e* ont été inversées à tort. Nous essayons de reconnaître le suffixe *et* en partant de l'état courant 5, ce qui n'est pas possible.

– Une lettre a été omise à tort entre les lettres *p* et *t*. Nous cherchons toutes les transitions qui mènent de l'état courant 5 à un état à partir duquel il est possible de reconnaître le suffixe *te*. Une telle transition n'existe pas.

– La lettre correcte à la position 3 a été remplacée à tort par la lettre *t*. Nous cherchons tout état accessible de l'état 5 à partir duquel il est possible de reconnaître le suffixe *e*. L'état 10 remplit cette condition et le candidat obtenu est *apse*.

De la même façon, en retournant de l'état 5 à l'état 2 (position 1 dans le mot) nous testons les quatre hypothèses possibles dont deux sont vérifiées :

– La lettre *p* a été insérée à tort. En passant par les transitions (2,*t*,11) et (11,*e*,13) nous obtenons un nouveau candidat *ate*.

– La lettre correcte à la position 2 a été remplacée à tort par la lettre *p*. En passant par les transitions (2,*n*,6), (6,*t*,11) et (11,*e*,13) nous obtenons le candidat *ante*.

Finalement, en retournant de l'état 2 à l'état 1 (position 0 dans le mot) nous obtenons *pate* par l'hypothèse de l'inversion des lettres *a* et *p* vérifiée par le chemin (1,*p*,3), (3,*a*,7), (7,*t*,11), (11,*e*,13).

9.4 Algorithme

Soit `search(word,state)`la procédure standard de recherche dans un automate de la séquence `word` à partir de l'état `state`. Au début cette procédure est appelée avec le paramètre `word` égal au mot recherché et le paramètre `state` égal à l'état initial `s`. Après chaque transition, l'appel récursif de `search` prend comme premier paramètre le suffixe restant et comme deuxième paramètre l'état d'arrivée de la transition précédente. La procédure se termine avec succès si l'on arrive à un état terminal après avoir lu entièrement la séquence d'entrée. La procédure échoue dans l'un des deux cas :

– la séquence d'entrée a été lue entièrement mais nous ne nous trouvons pas dans un état terminal;

– la séquence d'entrée n'a pas été lue entièrement mais aucune transition de l'état courant n'est possible pour le suffixe restant.

La recherche modifiée commence au moment où la recherche standard a été bloquée sans que la séquence d'entrée soit reconnue. Soit $[l_1\ l_2\ \ldots\ l_{wl}]$ le mot d'origine et `wl` sa longueur. Soit `wp = 0,1,...,wl` la position courante dans le mot (i.e. la position de la dernière lettre qui a été lue avec succès). Soit `st` l'état courant (celui à partir duquel aucune

transition n'est plus possible). Nous supposons alors que l'une des quatre opérations possibles a eu lieu :

- L'insertion incorrecte d'une lettre à la position `wp` (si `wp`≤`wl`). Nous omettons la lettre l_{wp} et essayons de reconnaître le suffixe $[l_{wp+1} \ldots l_{wl}]$ à partir de l'état courant `st`, c'est-à-dire nous appelons `search(`$[l_{wp+1} \ldots l_{wl}]$`,st)` au lieu de `search(`$[l_{wp}\ l_{wp+1} \ldots l_{wl}]$`,st)` qui aurait été appelé dans la recherche standard. Si cet appel se termine avec succès le candidat proposé pour la correction est la séquence $[1 \ldots l_{wp-1}\ l_{wp+1} \ldots l_{wl}]$.
- L'inversion incorrecte des lettres l_{wp} et l_{wp+1} (si `wp<wl`). Nous essayons de reconnaître le suffixe inversé $[l_{wp+1}\ l_{wp} \ldots l_{wl}]$ à partir de l'état courant `st` , c'est-à-dire nous appelons `search(`$[l_{wp+1}\ l_{wp} \ldots l_{wl}]$`,st)`. Si ceci se termine avec succès nous proposons le candidat $[1 \ldots l_{wp-1}\ l_{wp+1}\ l_{wp} \ldots l_{wl}]$.
- L'omission incorrecte d'une lettre à la position `wp`. Pour chaque transition qui mène de l'état `st` à l'état st_s par une lettre `l`, nous essayons de reconnaître le suffixe $[l_{wp}\ l_{wp+1} \ldots l_{wl}]$ à partir de l'état st_s, c'est-à-dire nous appelons `search(`$[l_{wp}\ l_{wp+1} \ldots l_{wl}]$`,`$st_s$`)`. Si la tentative se termine avec succès, nous proposons le candidat $[1 \ldots l\ l_{wp}\ l_{wp+1} \ldots l_{wl}]$.
- Un remplacement incorrect d'une lettre à la position `wp` (si `wp`≤`wl`). Pour chaque transition qui mène de l'état `st` à l'état st_s par une lettre `l`, nous essayons de reconnaître le suffixe $[l_{wp+1}\ l_{wp+2} \ldots l_{wl}]$ à partir de l'état st_s, c'est-à-dire nous appelons `search(`$[l_{wp+1}\ l_{wp+2} \ldots l_{wl}]$`,` st_s`)`. Si la tentative se termine avec succès, nous proposons le candidat $[1 \ldots l_{wp-1}\ l\ l_{wp+1} \ldots l_{wl}]$.

9.5 Erreurs multiples dans un mot

Même si dans la plupart des fautes de frappe (de 69% à 94% selon les auteurs cités par Kukich 1992) une seule opération élémentaire sur des lettres entre en jeu, il arrive qu'il faut en supposer deux ou plus, comme dans **ingeneer* (2 remplacements dans le mot correct *engineer*), **comitee* (2 omissions dans *committee*), **authentification* (2 insertions dans *authentication*), **inflexion* (l'omission et 1 remplacement dans *inflection*). Remarquons que tout mot fini peut être ramené à tout autre mot fini par un nombre fini d'opérations élémentaires (la prise en compte seulement des insertions et des effacements serait également suffisante). L'algorithme décrit ci-dessus est adaptable à tout nombre d'opérations admises, mais nous pensons qu'il n'est pas pratique de dépasser le seuil de deux opérations, pour deux raisons :

- nous risquons d'obtenir des candidats trop éloignés du mot d'origine;
- à partir de 3 opérations la recherche risque de durer trop longtemps pour l'application interactive telle que LexPro.

L'adaptation de l'algorithme décrit dans la section précédente consiste à définir le nombre maximal d'erreurs admises (`max_err`) et la suivie du nombre (`prec_err`) d'opérations qui ont déjà été supposées sur le chemin menant de l'état initial à l'état courant. A chaque moment où la consultation est bloquée nous essayons d'admettre l'une des 4 opérations seulement si le nombre d'opérations déjà admises ne dépasse par le seuil (`prec_err < max_err`). Nous ne recherchons que les candidats les plus proches du mot d'origine. C'est-à-dire que nous ne réexploitons pas les chemins qui ont été parcourus avec `k` modifications pour rechercher des candidats avec le nombre de modifications supérieur à `k`. Dans notre

exemple (section 9.3) ceci signifie qu'après avoir trouvé 7 candidats avec une seule erreur nous ne cherchons plus à trouver les candidats avec 2 erreurs : *tape* (deux inversions de *t*)*, pat* (l'inversion de *p* et de *a*, et l'omission de *e*) ou *ant* (remplacement de *p* par *n* et l'omission de *e*).

9.6 Application à la reconnaissance de formes composées

La méthode de correction orthographique, présentée ci-dessus pour les mots simples, s'applique aussi aux mots composés, si l'on consulte un dictionnaire comme le DELACF (voir section 2.4). Elle permet entre autres de reconnaître certaines variantes orthographiques des mots composés (voir section 3.7). Par exemple un tiret optionnel sera reconnu comme opération de remplacement d'un caractère (le blanc) par un autre (le tiret), pour les cas comme *curling tongs* (*curling-tongs*), ou bien comme insertion d'un caractère, comme dans *bolthole* (*bolt-hole*). Le même sera le cas des mots qui s'écrivent alternativement avec une majuscule ou avec une minuscule comme *Boolean algebra* (*boolean algebra*). Certaines variantes du dialecte, comme *dialling code* (*dialing code*) et *armour plate* (*armor plate*) seront reconnus comme effacement d'une lettre.

Nous avons effectué un test de reconnaissance de formes composées par notre algorithme dans le corpus IBM (1997-99), le même que celui qui nous a servi pour tester la méthode d'extraction décrite dans le chapitre 8. Nous rappelons qu'il s'agit d'un texte de 280 000 formes simples (1,68 mégaoctets) concernant le domaine de l'architecture des ordinateurs. Nous avons tout d'abord soumis ce corpus à l'analyse lexicale par le système INTEX, à l'aide des mêmes dictionnaires que ceux utilisés pour l'extraction (voir section 8.5.1) : le DELAF et le DELACF généraux, le DELAF et le DELACF de termes informatiques, et le DELAF de mots grammaticaux. Parmi les résultats de cette analyse figuraient : 15 245 occurrences de mots composés, généraux et spécialisés, et 1649 formes simples non reconnues. Nous nous sommes demandée de combien ces résultats pourraient être améliorés (i.e. de combien le premier chiffre pourrait monter et le deuxième descendre) si nous utilisions notre algorithme de « consultation modifiée » d'un DELAF /DELACF à la place de l'algorithme standard.

Nous disposions d'une version de notre algorithme de « consultation modifiée » qui s'appliquait à une séquence de lettres à reconnaître et non pas à un texte entier. Nous avons donc préparé deux listes de séquences pour simuler le comportement d'un analyseur lexical. La première liste, appelée *ERR*, contenait les 1649 séquences non reconnues. La deuxième liste, appelée *CANDIDATS_MC*, a été obtenue par la recherche dans notre corpus de deux patrons syntaxiques, *NN_AN.grf* (Fig.46) et *ErrN_NErr_AErr.grf* (Fig.47). Le premier de ces graphes décrit des syntagmes nominaux potentiels constitués de deux noms (simples ou composés, généraux ou spécialisés) connus, ou d'un adjectif et d'un nom, les deux éléments étant séparés par un blanc ou par un tiret. Le deuxième graphe représente le même type de syntagmes où la place du premier ou du deuxième élément est prise par une forme simple inconnue. Ainsi, nous pouvons extraire la plupart des candidats qui peuvent être reconnus par la « consultation modifiée » comme variantes des mots composés connus.[52] La recherche des

[52] Nous effectuons ainsi une extraction qui ressemble à celle proposée dans le chapitre 8 (voir section 8.5.2 pour la description du patron syntaxique), sauf que les constituants admis cette fois sont des noms et des adjectifs non seulement spécialisés mais aussi généraux.

deux patrons dans le corpus IBM (1997-99) a donné la liste *CANDIDATS_MC* de 47 676 occurrences de formes composées.

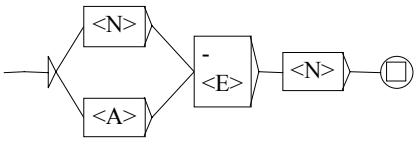


Fig.46. Graphe NN_AN.grf

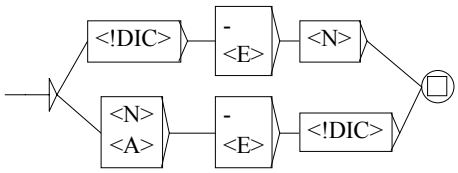


Fig.47. Graphe ErrN_NErr_AErr.grf

Nous avons ensuite soumis la liste *ERR* (9 555 occurrences, 1 649 formes) au programme de la « consultation modifiée » des dictionnaires DELAF et DELACF des termes composés informatiques. D'autre part, la liste *CANDIDATS_MC* (47 676 occurrences) a été soumise au même programme seulement pour le DELACF de termes informatiques. Les résultats de cette expérience sont résumés dans le tableau Tab.22. 7 828 occurrences (1 141 formes différentes) de formes simples ont été reconnues « proches » aux termes simples connus, i.e. chacune d'elles pouvait être obtenue par l'application d'une ou de deux opérations de base (section 9.2) à un ou plusieurs termes simples connus. 2 274 occurrences (211 formes différentes) ont été reconnues « proches » aux termes composés connus. 10 237 occurrences (3 917 formes différentes) de formes composées ont été reconnues « proches » aux termes composés connus.

Nous avons analysé « manuellement » ces séquences reconnues « proches » des termes simples et composés. Parmi les formes composées nous en avons retenu 1 569 (504 formes différentes), soit 15%, qui pouvaient être considérées comme étant effectivement des variantes des mots composés auxquels elles s'approchaient graphiquement. Grâce à l'application de l'algorithme de la consultation modifiée d'un DELACF spécialisé nous avons donc pu augmenter le taux de reconnaissance de mots composés de près de 10%. D'autre part, parmi les formes simples initialement non reconnues, nous avons pu reconnaître seulement 198 occurrences (52 formes différentes) comme variantes des mots simples connus, et 43 séquences (13 formes différentes) comme variantes des mots composés connus. L'augmentation du taux de reconnaissance était donc égale à 0,09% au niveau des occurrences, et à 0,9% au niveau des formes différentes.

La précision de notre algorithme s'exprime par la proportion du nombre de formes pertinentes parmi les formes reconnues par la consultation modifiée (non reconnues par la consultation standard). Cette proportion est très basse pour les mots simples. Ceci est dû au fait que les formes non reconnues sont dans une grande partie des noms propres et des sigles, souvent de petite longueur. Or, quand une forme courte est reconnue comme proche à un mot connu, deux, soit près de la moitié, de ses lettres peuvent différer de celles du mot d'origine, ce qui rend peu probable le fait que les deux formes soient effectivement apparentées.

Les résultats pour les mots composés sont plus intéressants, car nous avons pu augmenter le taux de reconnaissance des mots composés (le nombre de formes composées pertinentes reconnues par rapport aux mots composés trouvés par l'algorithme standard) de 10%. Mais pour ceci nous avons dû effectuer un dépouillage manuel où seulement une forme sur 7 a pu être retenue (précision 15%).

	Formes simples		Formes composées	
	occurrences	formes différentes	occurrences	formes différentes
Nombre de formes reconnues par la consultation standard	280 168	7 258	15 245	3 610
Nombre de formes non reconnues par la consultation standard	9 555	1 649	inconnu[53]	inconnu
Nombre de formes reconnues par la consultation modifiée	7 828 + 2 274	1 141 + 211	10 237	3917
Nombre de formes pertinentes reconnues par la consultation modifiée	198+43	52+13	1 569	504
Augmentation du taux de reconnaissance	0,09%	0,9%	10%	14%
Précision	2,4%	4,8%	15%	13%

Tab.22 Données statistiques sur l'algorithme de consultation modifiée

Parmi les formes retenues nous pouvions distinguer différents types de variantes :

1) variantes orthographiques (939 occurrences, 263 formes), où la séquence reconnue différait d'un mot composé uniquement par l'effacement ou l'insertion d'un blanc ou d'un tiret, par le remplacement d'un tiret par un blanc, ou par le remplacement des minuscules par des majuscules, par exemple (le composé connu figure entre parenthèses):

   [378] HyPerLink (Hyperlink), VisualAge (Visual Age), run-time library (runtime library), main-memory interface (main memory interface), op code (op-code), function level (Function Level),

2) variantes régionales (9 occurrences, 5 formes), par exemple :

   [379] behavioral model (behavioural model), *imbedded software (embedded software), wirability (wireability), interruptible (interruptable)*

3) fautes de frappe (6 occurrences, 4 formes) - dans tous les cas il s'agissait des fautes contenues dans le DELACF spécialisé et non pas dans le texte :

   [380] floating-point radixes (floating-point radixxes), symmetrical multiprocessors (symetrical multiprocessors)

4) variantes morphologiques et dérivationnelles qui ne changeaient pas le sens du terme (227 occurrences, 83 formes), où l'un des composants, le plus souvent le premier, pouvait changer de morphologie (singulier - pluriel) ou de catégorie (adjectif – nom, participe – adjectif, verbe – nom, etc.):

   [381] Communications Systems (communication systems), Costs Analysis (cost analysis), Operators Manual (operator manual), executables (executable)

   [382] automated call (automatic call), compiler options (compile options), cryptographic coprocessor (cryptography coprocessor), quiesce point (quiescent point), register names (registry names), table index (tab index),

[53] Afin de déterminer le nombre de mots composés existant dans le texte, et non existants dans le dictionnaire, il aurait fallu analyser manuellement le texte de plus de 200 pages.

[383] cache coherence (cache coherency), edge-detection (edge-detecting), sequential store (sequential storage), fault diagnostics (fault diagnosis),

5) variantes synonymiques (61 occurrences, 31 formes), où l'un des composants était remplacé par un synonyme et ainsi le sens de tout le composé était préservé :

[384] actual ratio (actual rate), printed circuit card (printed circuit board), compressed format (compressed form), dedicated wires (dedicated lines), key state (key status), multiple output (multiple outlet), near-end (head-end), slow performance (low performance), testing ability (testing facility)

6) variantes dérivationnelles (260 occurrences, 106 formes), où l'un des composants était dérivé de son correspondant dans l'autre terme, ou bien tous les deux étaient des dérivations différentes du même mot, et ainsi un composé obtenait un sens dérivé de l'autre composé, ceci pouvant concerner aussi bien le premier que le deuxième composant :

[385] asynchronous clock (synchronous clock), axial cable (coaxial cable), correctable errors (uncorrectable errors), higher density (high density), inactive states (active states),

[386] test data generation (test data generator), chip test (chip tester), configuration registers (configuration registry), local processors (local processes), microprocessor controller (microprocessor-controlled), process engineering (process reengineering), time dependence (time dependent),

7) variantes compositionnelles (67 occurrences, 12 formes), où grâce au rajout d'un ou de deux caractères nous obtenions des surcompositions de termes connus :

[387] STOSM (STOS), b-bit errors (bit errors), C bus multiplexing (bus multiplexing), CMOS technology (MOS Technology), Programmable DRAM (Programmable Ram), VLSI circuit (LSI circuit),

Remarquons que les variantes orthographiques (point 1 ci-dessus) constituent près de 60% (939 sur 1569) des formes pertinentes reconnues. Nous pourrions profiter de ce phénomène en modifiant notre algorithme de telle sorte que les opérations d'insertion de séparateur (blanc, tiret) et de remplacement d'un séparateur par un autre soient prioritaires par rapport aux autres opérations.

En résumé, nous pouvons constater que notre méthode de consultation modifiée, conçue en tant qu'outil de correction orthographique, a une précision trop basse pour être appliquée à des tâches comme l'extraction ou l'enrichissement terminologique, l'analyse lexicale, et la recherche documentaire. Notons néanmoins que le taux de précision est dans notre cas une notion relative liée à la complétude du dictionnaire utilisé :

- plus le dictionnaire est complet, plus il y a des mots composés reconnus par la consultation standard, et donc moins de formes composées correctes restent à reconnaître par la consultation modifiée,
- plus le dictionnaire est complet, plus il y a de chances qu'une séquence quelconque soit reconnue comme proche à un mot contenu dans le dictionnaire.

Ainsi, paradoxalement, il est possible que la précision indiquée dans le tableau Tab.22 soit plus élevée pour un petit dictionnaire que pour un dictionnaire bien complet, alors que c'est l'inverse pour la qualité globale de la reconnaissance.

9.7 Complexité de l'algorithme

La consultation modifiée de l'automate n'est pas déterministe car plusieurs corrections sont possibles pour un mot mal orthographié, comme ceci a été le cas de l'exemple ***apte*. Tous les chemins possibles avec une (ou deux) modifications doivent être explorés, alors la complexité du programme n'est pas linéaire, comme ceci est le cas de la consultation standard d'un dictionnaire-automate. Son calcul exact est difficile car elle dépend non seulement de la longueur du mot d'entrée et de la taille de l'automate, mais aussi du nombre de mots contenus dans l'automate qui ont des affixes communs avec le mot d'origine. Dans le pire des cas, i.e. dans la situation (théorique) où tous les mots contenus dans le lexique sont des candidats possibles pour la correction, la consultation modifiée exige l'exploration de l'automate entier.

Nous avons effectué des tests de performance pour différents jeux de formes composées (tableau Tab.23). Les temps indiqués ont été obtenus sur un PC avec un processeur Pentium 3 et 64 Mo RAM. Nous pouvons voir que le temps de consultation du dictionnaire est presque 4 fois plus élevé pour des séquences avec 2 modifications ou plus qu'avec une seule modification.

Temps de recherche d'une forme composée (**ms**)			
formes connues	formes avec 1 modification	formes avec 2 modifications ou plus	séquences libres[54]
16	53	197	171

Tab.23 Performances de l'algorithme de correction orthographique

9.8 Comparaison avec l'algorithme d'Oflazer

Oflazer (1996) a réalisé un algorithme de *error-tolerant finite–state recognition* basé sur la même idée de consultation d'un dictionnaire-automate par l'admission de 4 opérations élémentaires (section 9.2) qui peuvent être appliquées au mot d'origine en nombre admis en paramètre du programme. Il y a quelques différences majeures entre son algorithme et celui présenté ci-dessus :

- L'algorithme d'Oflazer effectue l'exploration de l'automate en largeur (pour l'état courant, tous les états joignables sont d'abord visités et empilés), tandis que dans notre algorithme l'exploration en profondeur est implémentée.
- Grâce à l'exploration en profondeur nous ne sommes pas obligée de calculer, à chaque transition suivie, le nombre de modifications admises sur le chemin entre l'état initial et l'état courant. Cette information est transmise directement à chaque nouvel appel récursif. Chez Oflazer cette information, appelée *cut-off edit distance* est calculée pour chaque transition suivie.
- Grâce à l'exploration en profondeur nous pouvons atteindre plus rapidement la première solution pour le mot recherché.
- L'algorithme d'Oflazer procède de gauche à droite. D'abord le mot exact est recherché par la consultation standard. Si le mot n'a pas été trouvé, la consultation recommence à

[54] Ensemble de 49 325 séquences (listes *ERR* et *CANDIDATS_MC*) extraits de notre corpus.

l'état initial et des suppositions sur des modifications possibles sont faites à partir de la première lettre du mot recherché. Notre algorithme procède de droite à gauche : il essaye d'abord d'accepter le mot d'origine sans modification, i.e. il cherche le plus long préfixe commun du mot recherché et de tous les mots existant dans le dictionnaire. Ensuite, si la séquence exacte n'a pas été trouvée, il effectue le retour en arrière sur le chemin d'arrivée. Grâce à cette solution le chemin parcouru n'est pas « perdu », i.e. il ne sera pas parcouru deux fois comme c'est le cas chez Oflazer.

– L'avantage majeur de l'algorithme d'Oflazer par rapport au nôtre est dans le fait que la distance entre deux séquences de lettres est mesurée par une fonction (*cuted*) indépendante de l'algorithme de consultation du dictionnaire. Cette fonction, telle qu'elle est décrite par Oflazer (1996), prend en compte les mêmes quatre opérations que celles admise par nous, mais elle peut aussi être plus spécialisée en fonction de la langue, du domaine, du corpus etc. Par exemple, Daciuk (1998) parle de son adaptation au polonais, où certaines lettres (souvent possédant des diacritiques) se prononcent de la même façon ou de façon proche de séquences de deux lettres sans diacritiques. C'est pourquoi elles sont très souvent à l'origine de fautes d'orthographe, comme *dzióra* (corr. *dziura* – « trou »), *żadki* (corr. *rzadki* – « rare »), *włanczać* (corr. *włączać* – « allumer »). Ce type de paires facilement remplaçables, comme *ó – u*, *ż – rz, an – ą*, etc. pourraient être considérées comme prioritaires parmi les opérations élémentaires sur les mots, ce qui permettrait de proposer en premier des corrections plus plausibles pour des mots erronés. En particulier, les échanges comme *ż – rz, an – ą* représentent 2 opérations élémentaires (respectivement 1 remplacement + 1 effacement, et 1 remplacement + 1 insertion). C'est pourquoi, dans notre algorithme, nous risquons d'obtenir des corrections avec une seule modification qui sont pourtant moins plausibles. Par exemple, pour *żadki* nous obtiendrions *gadki* (« bavardages », 1 remplacement), ce qui va bloquer la recherche pour *rzadki* (2 modifications).
  Un autre exemple de l'utilité de l'approche modulaire à la fonction *cuted* chez Oflazer a été donné dans la section 9.6 : si l'opération de remplacement d'un blanc par un tiret, ou de l'insertion d'un blanc ou d'un tiret est considérée prioritaire à d'autres opérations, nous pouvons augmenter la précision de reconnaissance de variantes orthographiques des mots composés.

9.9 Conclusion

Nous avons présenté une méthode de correction orthographique qui se base sur une consultation modifiée d'un dictionnaire-automate. Cette méthode – présentée et testée pour les mots simples - s'applique aussi aux mots composés, si l'on consulte un dictionnaire comme le DELACF, mais la précision obtenue est très basse. Cette méthode peut donc être employée à des tâches, où le tri manuel important peut être admis après l'intervention de l'algorithme. Pour les tâches comme l'extraction ou l'enrichissement terminologique, l'analyse lexicale, et la recherche documentaire, notre méthode n'est pas assez précise.

Le plus grand avantage de notre algorithme est l'indépendance de la langue traitée – il peut être appliqué tel quel à chaque langue pour laquelle nous disposons d'un dictionnaire du type DELAF ou DELACF. Les performances sont suffisantes pour les mots ne contenant aucune ou une seule faute. Avec l'admission d'un nombre plus élevé de fautes, le temps de recherche croit très rapidement.

Chapitre 10 Conclusion

Dans l'introduction de ce mémoire nous avons posé deux questions concernant l'étude des mots composés que nous avons entreprise :

1) Selon quelles méthodes pouvons-nous effectuer le recensement de mots composés à grande échelle ?

2) Est-il utile d'effectuer ce recensement ?

La réponse à la première question est donnée dans le contexte de la création de dictionnaires électroniques, sous formats disponibles dans le système INTEX. Nous avons analysé d'abord (chapitre 3) certaines propriétés linguistiques des mots composés, et plus particulièrement des noms composés, du point de vue de leur morphologie flexionnelle. Cette analyse nous a permis, dans le chapitre 4, de proposer une méthode formelle de description du comportement flexionnel des composés, et de mettre au point un algorithme qui génère leurs formes fléchies. Dans le chapitre 5, nous avons décrit la construction du dictionnaire électronique de mots composés (DELAC) anglais. Le format bien adapté au recensement des mots composés productifs étant celui d'automates et de transducteurs finis, nous avons illustré ceci par les déterminants numéraux cardinaux et ordinaux de l'anglais (chapitre 6). Finalement (chapitre 7), nous avons décrit la création d'un dictionnaire électronique terminologique du domaine de l'informatique.

Une fois que des ressources lexicographiques et terminologiques existent sous un format adapté au traitement automatique, leur emploi améliore en principe la qualités de nombreuses applications du TALN. Nous avons choisi deux types d'applications pour vérifier cette hypothèse. Premièrement (chapitre 8), nous avons élaboré une méthode d'extraction terminologique basée sur l'hypothèse que des séquences contiguës de termes connus ont de grandes chances d'être de nouveaux termes. Cette hypothèse pouvait être vérifiée grâce aux dictionnaires électroniques, généraux et spécialisés, décrits dans la première partie du mémoire. Les résultats obtenus s'avèrent, de certains points de vue, meilleurs de ceux obtenus par un extracteur terminologique de référence, qui est basé sur un calcul statistique et n'emploie pas de ressources terminologiques initiales. Deuxièmement (chapitre 9), nous avons élaboré un algorithme indépendant en principe de la langue de correction orthographique de mots simples et composés, basé sur la consultation d'un dictionnaire sous format d'automate fini. Les principes de fonctionnement de cet outil s'approchent de ceux appliqués dans un des correcteurs orthographiques de référence. D'habitude les outils de correction orthographique recherchent les mots simples inconnus d'un texte, et proposent leurs corrections étant aussi des mots simples. Nous avons voulu savoir si cette stratégie pouvait s'étendre à des mots composés, i.e. pour les mots simples inconnus nous avons cherché des mots simples et composés proches existant dans notre dictionnaire (*VisualAge – Visual Age*), puis pour des séquences de mots non reconnues comme mots composés, nous avons aussi cherché des mots composés proches (*compile options – compiler options*). Les résultats de cette expérience sont peu satisfaisants à cause de leur faible précision.

En conclusion, nous rappelons que le phénomène de composition est très présent dans les langues naturelles. Nous avons démontré que la description systématique, cohérente et fiable de mots composés doit se faire par leur recensement contrôlé (qui ne peut pas être totalement automatisé). Nous espérons avoir démontré qu'un tel travail et motivé par les résultats des applications informatiques qui en tiennent compte.

Références

ANSCOMBRE, J.-Cl. 1990. « Pourquoi un moulin à vent n'est pas un ventilateur », dans *Langue Française 86. Sur les compléments circonstanciels*, Paris, Larousse.

AUGER, P., DROUIN, P., AUGER, A. 1996. « Filtact : un automate d'extraction des termes complexes », dans Grarson, M., ed., *Terminologies nouvelles, Banques de terminologie, Actes de la table ronde, Québec, 18 et 19 janvier 1996, N°15*, Bruxelles, pp. 48-51.

BAUER, L. 1983. *English Word-Formation*, Cambridge University Press, Cambridge.

BAUER, L. 1988. *Introducing Linguistic Mophology*, Edingburgh University Press, Edinburgh.

BAT-ZEEV SHYLDKROT, H. 1996. « Analyse lexicale de l'ancien français », dans *Actes des Premières Journées INTEX 21-22 mars 1996*, Paris, LADL.

BENVENISTE, E. 1974. « Fondements syntaxiques de la composition nominale » et « Formes nouvelles de la composition nominale », dans *Problèmes de linguistique générale*, 2, pp. 145-176, Gallimard, Paris.

BIEŃ, K., SZAFRAN, K. 1996. « Analiza języka polskiego w Instytucie Informatyki Uniwersytetu Warszawskiego », dans Vetulani, Z., Abramowicz, W., Vetulani, G. (eds.) *Język i technologia*, Akademicka Oficyna Wydawnicza PLJ, Warszawa.

BLANCO, X. 1997. « Noms composés et traduction français-espagnol », dans *Linguisticæ Investigationes XXI : 2*, John Benjamins B. V., Amsterdam.

BOURIGAULT, D. 1994. *LEXTER un Logiciel d'Extraction de Terminologie. Application à l'extraction des connaissances à partir de textes.* Thèse de doctorat en Mathématiques, Informatique Appliquée aux Sciences de l'Homme, Paris, École des Hautes Études en Sciences Sociales.

BOWDEN, P., LINDSAY, E., HALSTEAD, P. 1998. « Automatic Acronym Acquisition in a Knowledge Extraction Program », dans *Preceedings from COMPUTERM, the First Workshop on Computational Terminology, August 15, 1988*, University of Montreal.

BRILL, E. 1994. *Supervised part of speach tagger*, http://www.cs.jhu.edu/~brill.

BUVET, P.-A. 1994. « Détermination : les noms », dans *Lingvisticae Investigationes 18(1)*, John Benjamins, Amsterdam.

CADIOT, P. 1992. « À entre deux noms : vers la composition nominale », dans *Lexique*, 11, P. U. L., pp. 193-240.

CHANOD, J.-P., TAPANAINEN, P. 1994. *Statistical and constraint-based taggers for French.* Technical report MLTT-016, Rank Xerox Research Centre, Grenoble, France.

CHANOD, J.-P., TAPANAINEN, P. 1995. « Creating a tagset, lexicon and guesser for a French tagger », dans *Proceedings from the ACL SIGDAT workshop on From Texts To Tags : Issues In Multilingual Language Analysis*, pp. 58-64, University College Dublin, Ireland.

CHROBOT, A. 1996. *Dictionnaire électronique des noms composés du polonais*, rapport de stage du DEA d'Informatique Fondamentale et Applications, Noisy-le-Grand, Université de Marne-la-Vallée.

CHROBOT, A. 1999. « Enrichissement terminologique en anglais fondé sur des dictionnaires généraux et spécialisés », dans Enguehard, Ch., Condamines, A. (eds.), *Terminologies Nouvelles*, *N° 19, Actes du Colloque « Terminologie et Intelligence Artificielle TIA-99 », Nantes 10-11 mai 1999,* décembre 1998 – juin 1999, Bruxelles, Agence de la francophonie et Communauté française de Belgique.

CHROBOT, A. 1999. « Flexion automatique des mots composés », dans Lamiroy, B., Klein, J., Pierret, J.-M., eds., *Cahiers de l'Institut de Linguistique de Louvain. Actes du XVI Colloque Européen sur les lexiques et la grammaire comparés des langues romanes, Louvain-la-Neuve, septembre 1997*, Louvain-la-Neuve.

CHROBOT, A. 2000. « Description des déterminants numéraux anglais par automates et transducteurs finis », dans les *Actes des 3es Journées Intex, 13-14 juin 2000,* Université de Liège, Belgique.

CLEMENCEAU, D. 1997. « Finite-State Morphology: Inflections and Derivations in a Single Framework Using Dictionaries and Rules », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

CORBIN, D. 1992. « Hypothèses sur les frontières de la composition nominale », dans *Cahiers de grammaire,* 17, novembre 1992, le MIR, Université de Toulouse.

COURTOIS, B. 1990. « Un système de dictionnaires électroniques pour les mots simples du français », dans Courtois, B., Silberztein, M. 1990. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990,* Paris, Larousse.

COURTOIS, B., SILBERZTEIN, M. 1990. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990,* Paris, Larousse.

DACIUK, J. 1998. *Incremental Construction of Finite-State Automata and Transducers, and Their Use in the Natural Language Processing*, thèse de doctorat, Politechnika Gdańska, Gdańsk.

DACIUK, J., MIHOV, S., WATSON, B., WATSON R. 2000. « Incremental Construction of Minimal Acyclic Finite State Automata »*, Computational Linguistics 26(1)*, pp. 3-16, MIT Press.

DAILLE, B. 1994. *Approche mixte pour l'extraction automatique de terminologie : statistique lexicale et filtres linguistiques*, thèse de doctorat en Informatique Fondamentale, Université Paris 7.

DAILLE, B. 1996. « ACABIT : une maquette d'aide à la construction automatique de banques terminologiques », dans Clas, A., Thoiron, P., Béjoint (eds.) *Lexicomatique et Dictionnairique*, FMA, Beyrouth, pp . 123-136.

DAMERAU, F. J. 1964. « A Technique for Computer Detection and Correction of Spelling Errors », dans *Communications of the ACM 7(3),* pp. 171-176.

DAVID, S., PLANTE, P. 1990. « De la nécessité d'une approche morpho-syntaxique en analyse de textes », dans 0*ICO* 2(3), pp. 140-155, Québéc.

DE SOLLIERS, F. 1998. *Dictionnaire Encyclopédique de l'Informatique*, Paris, La Maison du Dictionnaire.

DOMINGUES, C. 1998/9 « Traitement de la coordination à l'intérieur des groupes nominaux », dans Fairon, C. (éd.) *Linguisticae Investigationes, Volume spécial, Analyse lexicale et syntaxique : Le système INTEX*, John Benjamins, Amsterdam.

DOWNING, P. 1977. « On the Creation and Use of English Compound Nouns », dans *Language 53(4)*, Linguistic Society of America.

ENGUEHARD, Ch., PANTERA, L. 1994. « Automatic Natural Acquisition of Terminology », dans *Journal of Quantitative Linguistics 1994, Vol. 2, No. 1,* pp. 27-32, Netherland, Swets & Zeitlinger.

FABRE, C., SEBILLIOT, P. 1996. « Interprétation automatique des composés nominaux anglais hors domaine : quelles solutions ? », dans *Actes du 10ème Congrès Reconnaissance des Formes et Intelligence Artificielle (RFIA' 96)*, janvier 1996, Rennes.

FININ, T. 1986. « Constraining the Interpretation of Nominal Compounds in a Limited Context », dans Grishman, R., Kittredge, R. (eds.) *Analyzing Language in Restricted Domains*, Lawrence Erlbaum Associates, Hillsdale, New Jersey.

FOWLER. 1983. *A Dictionary of Modern English Usage*, Oxford University Press, Oxford.

GARRIGUES, M. 1997. « Une méthode de désambiguïsation locale nom/adjectif pour l'analyse automatique de textes », dans *Langages 126*, Larousse, Paris.

GARY-PRIEUR, M.-N. 1994. *Grammaire du nom propre*, Presses Universitaires de France, Paris.

GOUADEC, D. 1997. *Terminologie et Phraséologie pour Traduire – Le concordancier du Traducteur*, Paris, La Maison du Dictionnaire.

GREVISSE, M. 1993. *Le bon usage*, édition Duculot, Louvain-la-Neuve, Belgique.

GROSS, G. 1988. « Degré de figement des noms composés », dans *Langage 90*, pp.57-72, Larousse, Paris.

GROSS, G. 1990. « Définition des noms composés dans un lexique-grammaire », dans Courtois, B., Silberztein M. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990*, pp. 84-90, Larousse, Paris.

GROSS, G. 1994. « Classes d'objets et description des verbes », dans GIRY-SCHNEIDER, J. (ed.) *Langages 115*, *Sélection et sémantique, classes d'objets, compléments appropriés, compléments analysables,* Paris, Larousse.

GROSS, G. 1996. *Les expressions figées en français. Noms composés et autres locutions*, Ophrys, Paris.

GROSS, M. 1986. *Grammaire transformationnelle du français, III, Syntaxe de l'adverbe*, Cantilene, Paris.

GROSS, M. 1989a. « La construction des dictionnaires électroniques » dans *Annales des télécommunications*, tome 44, N°1-2.

GROSS, M. 1989b. « The Use of Finite Automata in the Lexical Representation of Natural Language », dans *Electronic Dictionaries and Automata in Computational Lingusitics*, Berlin - New York : Springer Verlag.

GROSS, M. 1997. « The Construction of Local Grammars », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

GUENTHNER, F. 1996. « Electronic Lexica and Corpora Research at the CIS », dans *Actes des Premières Journées INTEX 21-22 mars 1996*, Paris, LADL.

HABERT, B., JACQUEMIN, Ch. 1993. « Noms composés, termes, dénominations complexes : problématiques linguistiques et traitements automatiques », dans *Traitement Automatique des Langues, N° 2, Traitement automatique de la composition nominale*, Paris.

HABERT, B., JACQUEMIN, Ch. 1995. « Constructions nominales à contraintes fortes et grammaires d'unifications », dans *Lingvisticae Investigationes* XIX:2, 401-427, John Benjamins B. V., Amsterdam.

HILDEBERT, 1998. *Dictionnaire des technologies de l'informatique*, Paris, La Maison du Dictionnaire.

HO. 1994. *Le dictionnaire Hachette-Oxford français-anglais anglais-français*, Oxford, Paris, Oxford University Press, Hachette Livre.

HOPCROFT, J., ULLMAN, J. 1979. *Introduction to Automata Theory, Languages and Computation,* Adison-Wesley Publishing Company, Reading, Massachusetts.

IBM, 1997-99. « IBM S/390 Server G3/G4 », dans *IBM Journal of Research and Development*, Vol.41, No. 4/5, 1997 et « IBM S/390 Server G5/G6 », dans *IBM Journal of Research and Development*, Vol.43, No. 5/6, 1999, *www.research.ibm.com/journal*

JACQUEMIN, Ch. 1997. *Variation terminologique : Reconnaissance et acquisition automatique de termes et de leurs variantes en corpus*, habilitation à diriger des recherches en informatique, IRIN, Université de Nantes.

JACQUEMIN, Ch., KLAVANS, J., TZOUKERMANN, E. 1997. « Expansion of Multi-Word Terms for Indexing and Retrieval Using Morphology and Syntax », dans *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics, Barcelona, 7-10 July 1997*, Association for Computational Linguistics.

JASSEM, K. 1996. *Elektroniczny słownik dwujęzyczny w automatycznym tłumaczeniu tekstu*, thèse de doctorat, Uniwersytet im. Adama Mickiewicza, Poznań.

JESPERSEN, O. 1965. *A Modern English Grammar on Historical Principles*, Allen & Unwin, London.

JUNG, R. 1990. « Remarques sur la constitution du lexique des noms composés », dans *Langue Française 87*, Paris, Larousse.

JUSTESON, J., KATZ, S. 1995. « Technical terminology : some linguistic properties and an algorythm for identification in text », dans *Natural Language Engineering, 1(1)*, pp. 9-27.

KAPLAN, R., KAY, M., 1994. « Regular Models of Phonological Rule Systems », dans *Computational Linguistics 20(3)*, MIT Press.

KARTTUNEN, L., WITTENBURG, K. 1983. « A Two-Level Morphological Analysis of English », dans *Texas Linguistics Forum 22*.

KLARSFELD, G., McCARTHY-HAMMANI, M. 1992. *Dictionnaire électronique du LADL pour les mots simples de l'anglais. DELAS v4*, rapport technique, LADL, Université Paris 7, Paris.

KORNAI, A. (ed.) 1999. *Extended Finite State Models of Language*, Cambridge University Press, Cambridge, UK – New York, USA - Melbourne, Australia.

KOSKENNIEMI, K. 1997. « Representations and Finite-State Components in Natural Language », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

KUKICH, K. 1992. « Techniques for Automatically Correcting Words in Text », dans *ACM Computing Surveys*, 24(4),

LADOUCEUR, J., COCHRANE, G., 1996. « Termplus, système d'extraction terminologique », dans Grarson, M., (ed.), *Terminologies nouvelles, Banques de terminologie, Actes de la table ronde, Québec, 18 et 19 janvier 1996, N°15*, Bruxelles, pp. 52-56.

LAPORTE, E. 1988. *Méthodes algorithmiques et lexicales de phonétisation de textes. Applications au français.* Thèse de doctorat, Paris, Université Paris 7.

LAPORTE, E. 1997. « Rational Transductions for Phonetic Conversion and Phonology », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

LAPORTE, E., MONCEAUX, A. 1997. « Grammatical disambiguation of French words using part of speech, inflectional features and lemma of words in the context », dans *GRAMLEX Deliverables. May-September 1997*, Laporte, E. (ed.), LADL, Paris, Université Paris 7.

LAPORTE, E., SILBERZTEIN, M. 1989. « Vérification et correction orthographiques assistées par ordinateur », dans *Actes de la Convention IA 89.*

LECLÈRE, Ch. 1990. « Organisation du lexique-grammaire des verbes français », dans Courtois, B., Silberztein, M. 1990. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990,* Paris, Larousse.

LEHRBERGER, J. 1986. « Sublanguage Analysis », dans Grishman, R., Kittredge, R. (eds.) *Analyzing Language in Restricted Domains*, Lawrence Erlbaum Associates, Hillsdale, New Jersey.

LEVI, J. 1978. *The Syntax and Semantics of Complex Nominals*, Academic Press, New York – London.

LYONS J. 1978. « Les lexèmes composés », dans *Sémantique linguistique*, pp. 164-178, Larousse, Paris, 1990, traduction française de l'édition anglaise, Cambridge University Press, 1978.

MATHIEU-COLAS, M. 1988. *Typologie des noms composés*, rapport technique, LLI, Université Paris 13.

MATHIEU-COLAS, M. 1990. « Orthographie et informatique : établissement d'un dictionnaire électronique des variantes graphiques », dans Courtois, B., Silberztein, M. 1990. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990,* Paris, Larousse.

MAUREL, D. 1989. *Reconnaissance de séquences de mots par automates. Adverbes de dates du français*. Thèse de doctorat, Université Paris 7, Paris.

MAUREL, D., LEDUC, B., COURTOIS, B. 1995. "Vers la construction d'un dictionnaire electronique des noms propres", dans *Lingvisticae Investigationes* XIX:2, John Benjamins B. V., Amsterdam.

MAUREL, D., PITON O. 1999. « Un dictionnaire de noms propres pour Intex : les noms propres géographiques », dans *Lingvisticae Investigationes* XXII, pp. 277-287, John Benjamins B. V., Amsterdam.

McILROY, M. D. 1982. « Development of a Spelling List », dans *IEEE Transactions on Communications*, COM-30(1), pp. 91-99.

MILLER, G. 1993. « Nouns in WordNet: A Lexical Inheritance System », *www.cogsci.princeton.edu/~wn.*

MILLER, G., BECKWITH, R., FELLBAUM, Ch., GROSS D., MILLER, K. 1993. « Introduction to WordNet: An On-line Lexical Database », *www.cogsci.princeton.edu/~wn.*

MOHRI, M. 1997. « On the Use of Sequential Transducers in Natural Language Processing », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

MONCEAUX, A. 1994. *La formation de noms composés de structure Nom Adjectif. Elaboration d'un lexique électronique*, thèse de doctorat, Université de Marne-la-Vallée, Noisy-le-Grand.

MONCEAUX, A. 1995. *Le dictionnaire des mots simples anglais : mots nouveaux et variantes orthographiques*, rapport technique IGM 95-15, Institut Gaspard Monge, Université de Marne-la-Vallée, Noisy-le-Grand.

MONTELEONE, M. 1997. « Synthesis of Results About Morphological Dictionaries in French and Italian », dans *GRAMLEX Deliverables. May-September 1997*, Laporte, E. (ed.), LADL, Paris, Université Paris 7.

NAKAGAWA, H., MORI, T., 1998. « Nested Collocation and Compound Noun For Term Extraction », dans *Preceedings from COMPUTERM, the First Workshop on Computational Terminology, August 15, 1988*, University of Montreal.

NOAILLY, M. 1989. « Le nom composé : us et abus d'un concept grammatical », dans *Cahiers de grammaire 14*, Université de Toulouse Le Mirail, Toulouse.

NSOED. 1996. *The New Shorter Oxford English Dictionary*, Oxford University Press, Oxford.

OALDCE. 1989. *Oxford Advanced Learner's Dictionary of Current English*, A.S. Hornby, Oxford University Press, Oxford.

OBRĘBSKI, T., VETULANI, Z. 1997. « Inflected form dictionary for Polish with codes for part of speech, lemma and inflection features », dans *GRAMLEX Deliverables. May-September 1997*, Laporte, E. (ed.), LADL, Paris, Université Paris 7.

OFLAZER, K. 1996. « Error-tolerant finite state recognition with applications to morphological analysis and spelling correction », dans *Computational Linguistics,* 22(1), pp. 73-89, MIT Press.

PEREIRA, F., RILEY, M. 1997. « Speech Recognition by Composition of Weighted Finite Automata », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

PIOT, M. 1978. *Etude trasformationnelle de quelques classes de conjonctions de subordination du français*, thèse de doctorat, Université Paris 7 , Paris.

QUIRK, R., GREENBAUM, S., LEECH, SVARTNIK, 1972. *A Grammar of Contemporary English*, Longman Group Ltd.

REVUZ, D. 1991. *Dictionnaire et Lexiques. Méthodes et Algorithmes,* thèse de doctorat, Paris, Université Paris 7.

ROCHE, E. 1992. « Text Disambiguation by Finite-State Automata, an Algorythm and Experiments on Corpora », dans *Proceedings of COLING-92, the XIV International Conference on Computational Linguistics,* Vol III, pp. 993-997, Nantes.

ROCHE, E. 1993. *Analyse Syntaxique Transformationelle du Français par Transducteurs et Lexique-Grammaire*, thèse de doctorat, Paris, Université Paris 7.

ROCHE, E., SCHABES, Y. 1997. « Deterministic Part-of-Speech Tagging with Finite State Transducers », dans Roche. E., Schabes, Y. (eds.) *Finite-State Language Processing*, MIT Press, Cambridge, Massachusetts.

SCHULZ, K., MIKOŁAJEWSKI, T. 1999. « Between finite state and Prolog: constrained-based automata for efficient recognition of phrases », dans Kornai, A. (ed.) *Extended Finite State Models of Language*, Cambridge University Press, Cambridge, UK – New York, USA, - Melbourne, Australia.

SCHWIND, C. 1990. « An intelligent language tutoring system », dans *Man - Machine Studies* 33, pp. 557-579.

SELKIRK, E. 1982. *The Syntax of Words*, MIT Press, Cambridge, Massachusetts.

SENELLART, J. 1996. « Statistique prudence » dans *Actes des Premières Journées INTEX 21-22 mars 1996*, Paris, LADL.

SENELLART, J. 1999. *Outils de reconnaissance d'expressions linguistiques complexes dans des grands corpus*, thèse de doctorat, Paris, Université Paris 7.

SILBERZTEIN, M. 1989. *Dictionnaires électroniques et reconnaissance lexicale automatique*, thèse de doctorat, Paris, Université Paris 7.

SILBERZTEIN, M. 1990. « Le dictionnaire électronique des mots composés », dans Courtois, B., Silberztein, M. 1990. (eds.) *Langue Française 87. Dictionnaires électroniques du français, septembre 1990,* Paris, Larousse.

SILBERZTEIN, M. 1993a. *Dictionnaires électroniques et analyse automatique de textes: le système INTEX,* Masson, Paris.

SILBERZTEIN, M. 1993b. « Les groupes nominaux productifs et les noms composés lexicalisés », dans Chevalier, J.-C., Gross, M., Leclère, Ch. (eds.) *Lingvisticae Investigationes, tome XVII:2,* John Benjamins B.V., Amsterdam.

SILBERZTEIN, M. 1997. *INTEX 3.4. Reference Manual*, LADL, Université Paris 7, Paris.

SILBERZTEIN, M. 1999-2000. *INTEX*, Paris, ASSTRIL, accessible aussi par : *www.ladl.jussieu.fr*.

SKLAVOUNOU, E. 1999. « Classes de déclinaison des adjectifs et des noms du grec moderne » dans Lamiroy, B., Klein, J., Pierret, J.-M., eds., *Théories linguistiques et applications linguistiques. Cahiers de l'Institut de Linguistique de Louvain, Actes du XVI Colloque européen sur la grammaire et le lexique comparés, Louvain-la-Neuve, septembre 1997,* Louvain-la-Neuve, Belgique.

SKORUPKA, S. 1967. *Słownik frazeologiczny języka polskiego*, Wiedza Powszechna, Warszawa.

SMADJA, F. 1993. « XTRACT: An Overview », dans *Computers and the Humanities*, 26, pp. 399-413, Kluwer Academic Publishers, Netherlands.

VETULANI, Z. 1996. « Dialog z komputerem w języku polskim », dans Vetulani, Z., Abramowicz, W., Vetulani, G. (eds.) *Język i technologia*, Akademicka Oficyna Wydawnicza PLJ, Warszawa.

VETULANI, Z., WALCZAK, B., OBRĘBSKI, T., VETULANI, G. 1989. *Jednoznaczne kodowanie fleksji rzeczownika polskiego i jego zastosowanie w słownikach elektronicznych – format POLEX,* Wydawnictwo Naukowe UAM, Poznań.

VIVES, R. 1990. « Les composés nominaux par juxtaposition » dans Courtois, B., Silberztein M. (eds.), *Langue française 87. Dictionnaires électroniques du français, septembre 1990*, pp. 98-103, Larousse, Paris.

WALKER, D., AMSLER, R. 1986. « The Use of Machine-Readable Dictionaries in Sublanguage Analysis », dans Grishman, R., Kittredge, R. (eds.) *Analyzing Language in Restricted Domains*, Lawrence Erlbaum Associates, Hillsdale, New Jersey.

WATSON, B. 1995. *Taxonomies and Toolkits of Regular Language Algorithms.* Ph.D. Thesis, Eindhoven University of Technology, the Netherlands.

WEBSTER, 1976. *Webster's Third New International Dictionary.* Webster, Springfield, Massachusetts.

ANNEXE A.

Exemple de l'analyse lexicale par INTEX

Nous analysons la phrase suivante :

There are now more than four million motor vehicles registered in Sydney.

à l'aide de trois dictionnaires anglais :

- DELAF général,
- DELACF général,
- transducteur de déterminants numéraux utilisé en mode prioritaire.

Suite à l'étiquetage, nous obtenons quatre listes d'étiquettes :

Les mots simples reconnus

are,are.N:s
are,be.V:P2s:P1p:P2p:P3p
four,four.DET:p
four,four.N:s
in,in.A
in,in.N:s
in,in.PART
in,in.PREP
million,million.DET:p
million,million.N:s:p
more,more.ADVA
more,more.DET:s:p
more,more.PRO:s:p
motor,motor.A
motor,motor.N:s
motor,motor.V:W:P1s:P2s:P1p:P2p:P3p
now,now.A
now,now.ADV
now,now.CONJ
registered,register.V:K:I1s:I2s:I3s:I1p:I2p:I3p
registered,registered.A
than,than.CONJ
than,than.PREP
there,there.ADV
there,there.INTJ
vehicles,vehicle.N:p
vehicles,vehicle.V:P3s

Les formes simples non reconnues

Sydney

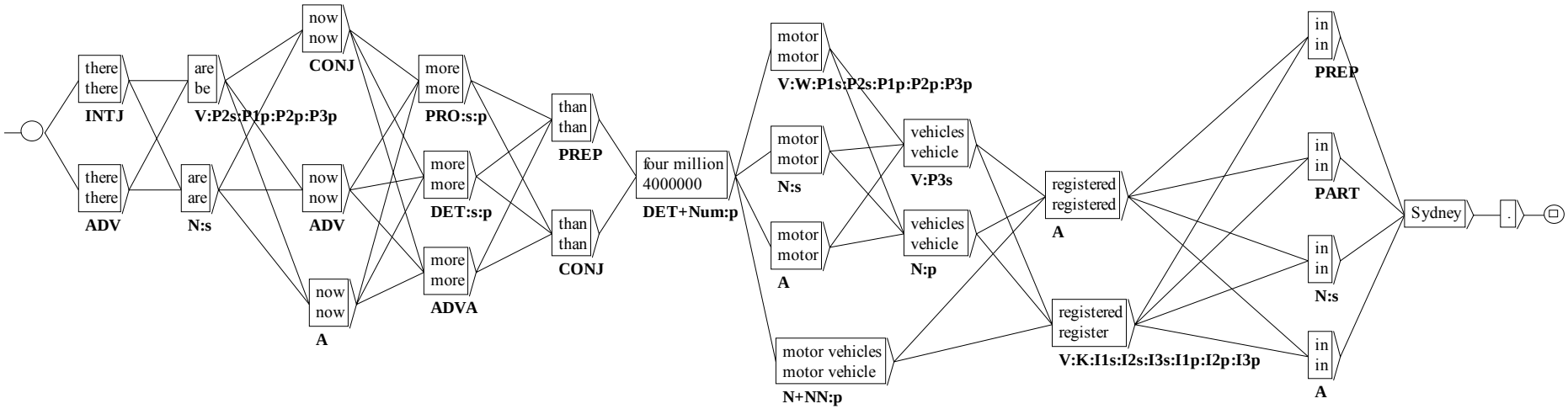
Les mots composés non ambigus

four million,4000000.DET+Num:p

Les mots composés ambigus

motor vehicles,motor vehicle.N+NN:p

Le transducteur du texte est présenté à la page suivante.


there
there
INTJ
there
there
ADV
are
be
V:P2s:P1p:P2p:P3p
are
are
N:s
now
now
CONJ
now
now
ADV
now
now
A
more
more
PRO:s:p
more
more
DET:s:p
more
more
ADVA
than
than
PREP
than
than
CONJ
four million
4000000
DET+Num:p
motor
motor
V:W:P1s:P2s:P1p:P2p:P3p
motor
motor
N:s
motor
motor
A
motor vehicles
motor vehicle
N+NN:p
vehicles
vehicle
V:P3s
vehicles
vehicle
N:p
registered
registered
A
registered
register
V:K:I1s:I2s:I3s:I1p:I2p:I3p
in
in
PREP
in
in
PART
in
in
N:s
in
in
A
Sydney
.

ANNEXE B.

Extraits du DELAC des noms anglais

B.1. Extraits des classes des noms composés réguliers

%-/+ (classe XC)
abortion clinic(clinic.N1:s),N+XN+Conc:s /+N\an;Med
about-turn(turn.N1:s),N+XC:s /+N\mil
absolute majority(majority.N5:s),N+XC+Hum:s /+N\pol
...
%-/-/+ (classe XXC)
Adam's apple(apple.N1:s),N+NsN:s /+N
advance booking office(office.N1:s),N+XXN+Conc:s /+N
apothecaries' measure(measure.N1:s),N+NN:s /+N
...
%+/-/- (classe CXX)
Act(act.N1:s) of Congress,N+NofN:s /+N\pol
angle(angle.N1:s) of reflection,N+NofN:s /+N
blessing(blessing.N1:s) in disguise,N+NPrepN:s /+N
...
%-/-/-/+ (classe XXXC)
All Fool's Day(day.N1:s),N+XXsN:s /+N
Chinese forget-me-not(not.N1:s),N+AVProAdv:s /+N
acquired immune deficiency syndrome(syndrome.N1:s),N+3XN:s /+N
...
%+/-/-/- (classe CXXX)
Acts(act.N1:p) of the Apostles,N+XXXN:p\rel%HO
court(court.N1:s) of common pleas,N+NofAN:s /+N
lily(lily.N5:s) of the valley,N+NofN+Conc:s /+N\a;Veg
...
%+/-/+ (classe CXC)
Alpha(alpha.N1:s) and Omega(omega.N1:s),N+NandN:s /+N
bold(bold.N2:s) and beautiful(beautiful.N2:s),N+NandN:s /+N
Stars(star.N1:p) and Stripes(stripe.N1:p),N+CXC+Conc:p
...
%-/+/-/- (classe XCXX)
cosmic order(order.N1:s) of magnitude,N+XNPX:s /+N\a;Unit
neutron time(time.N1:s)-of-flight,N+NNofN:s /+N
petty officer(officer.N1:s) third class,N+3XN+Hum:s /+N\a;Mil
...
%+/- (classe CX)
account(account.N1:s) current,N+NA:s /+N\%Jespersen
answer(answer.N1:s) back,N+CX:s /+N\%Jespersen
bole(bole.N1:s) Armeniac,N+XN:s /+N
...
%+/+ (classe CC)
lord(lord.N1:s) justice(justice.N1:s),N+CC:s /+N\%Webster ...

B.2. Extraits des classes des noms composés irréguliers

%-/-
%p:/ -
%p:-/p
attorney(attorney.N1:s) general(general.N1:s),N:s /+N\%Webster
battle(battle.N1:s) royal(royal.N1:s),N:s /+N\%Webster
beau(beau.X1:s) ideal(ideal.N1:s),N:s /+N\%Webster
...
%-/-
%p:p/-
%p:-/p
%p:p/p
cousin(cousin.N1:s)-german(german.N1:s),N+CX+Hum:s /+N\%Webster
journeyman(journeyman.N8:s) carpenter(carpenter.N1:s),N+NN+Hum:s /+N
...
%-/-
%p:- p
%p:p/p
Knight(knight.N1:s) Templar(templar.N1:s),N+NN:s /+N
...
%-/-
%p:-/-
cross-roads,N:s /+N\%Jespersen
shake-hands,N:s /+N\%Jespersen
...
%+/-/-
%p:p/-/-
%p:p/-/p
head(head.N1:s) of government(government.N1:s),N+NofN:s /+N
member(member.N1:s) of parliament(parliament.N1:s),N+NofN:s /+N
risk(risk.N1:s) of infection(infection.N1:s),N+NofN:s /+N
...
%+/-/-
%p:-/-/p
%p:p/-/-
%p:p/-/p
brandy(brandy.N5:s) and soda(soda.N1:s),N+CXX:s /+N
whisky(whisky.N5:s) and soda(soda.N1:s),N+CXX:s /+N

%-/-/-
%p:-/-/p
%p:p/-/-
four(four.X1:s)-in-hand(hand.N1:s),N+XXN+Conc:s /+N\equit

%+ /- /- /-
%p:p /- /- /-
%p:- /- /-/p
dog(dog.N1:s) in the manger(manger.N1:s),N+NPrepXX+Hum:s /+N\%HO
light(light.N1:s) o' love(love.N1:s),N+CXXX+Hum:s /+N\%Jespersen

ANNEXE C.

Extraits du DELACF anglais

C.1. Noms composés fléchis

A batteries,A battery.N+XN+Conc:p
A battery,.N+XN+Conc:s
A-bomb,.N+XN+Conc:s
A-bombs,A-bomb.N+XN+Conc:p
A-day,.N+XN:s
A-days,A-day.N+XN:p
A formation,.N+XN:s
A formations,A formation.N+XN:p
A-frame,.N+XN:s
A-frames,A-frame.N+XN:p
a-go-go,.N+DetNN:s
a-go-goes,a-go-go.N+DetNN:p
A horizon,.N+XN:s
A-horizon,.N+XN:s
A-horizons,A-horizon.N+XN:p
A horizons,A horizon.N+XN:p
A-level,.N+XC:s\school
A level,.N+XN:s
A-levels,A-level.N+XC:p\school
A levels,A level.N+XN:p
A-line,.N+XN:s
A-lines,A-line.N+XN:p
a priori assumption,.N+XXN:s
a priori assumptions,a priori assumption.N+XXN:p
...

C.2. Adjectifs composés

a button short,.A
able-bodied,.A -+
above-average,.A -+
above-cited,.A -+
above-found,.A -+
above-given,.A -+
above-mentioned,.A -+
above-named,.A -+
above-quoted,.A -+
above-reported,.A -+
above-said,.A -+
above-written,.A -+
absent-minded,.A +-
...

C.3. Adverbes composés

a bit more,.ADV+PAC+{do_N~}
a bit,.ADV+PDETC+{do_N~}
a case in point,.ADV+BPCPB+{be~}
a few times,.ADV+PAC+{happen~}
a generation ago,.ADV+PAC+{happen~}
a generation later,.ADV+PAC+{happen~}
a little later,.ADV+PAC+{happen~}
a little more,.ADV+PAC+{do_N~}
a little wee,.ADV
a little,.ADV+PDETC+{do_N~}
...

C.4. Prépositions composées

according to,.PREP+PCPN+{happen~}
across from,.PREP
against the will of,.PREP+BPCPN+{be~}
ahead of,.PREP
all for,.PREP
all the way to,.PREP+PCPN+{happen~}
as for,.PREP
as from,.PREP
...

C.5. Conjonctions composées

after the moment that,.CONJS/--+---
after the time that,.CONJS/--+---
any time,.CONJS/--+---
as a way of,.CONJS
as far as,.CONJS/--+---
as if,.CONJS/--+---
as long as,.CONJS/--+---
as soon as,.CONJS/--+---
as though,.CONJS/--+---
as to whether,.CONJS/--+---
assuming that,.CONJS/--+---
at a time when,.CONJS/--+---
at the thought of,.CONJS/----++
at the time when,.CONJS/--+---
at the time,.CONJS/--+---
before the moment that,.CONJS/--+---
...

ANNEXE D.

Fréquences des mots composés

Nous avons effectué l'analyse lexicale de 99 mégaoctets de texte du journal *Herald Tribune* (année 1994), à l'aide du système INTEX avec le DELACF anglais général (60 000 entrées) décrit dans le chapitre 5. Nous avons trouvé 524 539 occurrences de 25 717 formes composées différentes. La fréquence moyenne d'une forme composée était donc 20. L'analyse détaillée des fréquences montre que les mots très fréquents sont très peu nombreux et les mots peu fréquents sont très nombreux (voir section 2.7), ce qui est explicité par le tableau Tab.24 :

Fréquences	**Nombre de formes**	**Nombre d'occurences**
toutes	25 717	524 539
supérieures à 1000	55 (0,2%)	129 075 (25%)
inférieures à 20	22 051 (86%)	92 747 (18%)

Tab.24 Nombres de mots composés les plus et les moins fréquents.

Le tableau Tab.25 présente les exemples des formes composées les moins fréquentes. Remarquons que les composés comme *wine glass* ou *absolute zero,* qui semblent a priori plutôt fréquents, n'ont pas été trouvé dans le corpus plus de 2 fois :

Fréquence	**Nombre de formes composées avec cette fréquence**	**Nombre d'occurrences**	**Exemples**
1	6925	6925	*wine glass, X-ray photographs*
2	4305	8610	*absolute zero, account current*
3	2300	6900	*age of consent, blood bank*
4	1776	7104	*drug smuggler, easy-going*
5	1152	5760	*floppy disk, freedom of religion*
6	978	5868	*gold star, ground crew*
7	751	5257	*a little later, Achilles' heel*
8	675	5400	*in-between, in other respects*
9	493	4437	*abortion pill, leaning tower of Pisa*
10	469	4690	*main streets, ne'er-do-well*
11	375	4125	*primary colors, religious freedom*
12	333	3996	*second-rate, separation of church and state*
13	274	3562	*television broadcast, traffic lights*
14	288	4032	*to the bone, white wine,*
15	220	3300	*absentee ballots, American English*
16	217	3472	*after hours, alpine skiing*
17	191	3247	*bona fide, brave new world*
18	189	3402	*chewing gum, fairy-tale*
19	140	2660	*House of Lords, mother-in-law*
Total	22051	92747	

Tab.25 Formes composées peu fréquentes

La liste ci-dessous contient les formes composées les plus fréquentes avec leurs fréquences :

12 334	*United States*	2 025	*cold war*	1 340	*for the first time*
8 886	*prime minister*	1 903	*kind of*	1 300	*at the time*
5 184	*human rights*	1 849	*foreign trade*	1 299	*World War II*
5 170	*Hong Kong*	1 844	*as a result*	1 288	*in time*
4 665	*United Nations*	1 713	*stocks and bonds*	1 281	*nuclear weapons*
4 320	*at least*	1 710	*European Union*	1 271	*foreign minister*
4 225	*that is*	1 702	*not only*	1 232	*of course*
3 584	*White House*	1 648	*political parties*	1 216	*business and industry*
3 493	*Bosnia-Herzegovina*	1 640	*so much*	1 185	*health care*
3 487	*Associated Press*	1 618	*World Cup*	1 139	*a little*
3 026	*peace talks*	1 600	*so far*	1 130	*in fact*
2 882	*interest rates*	1 483	*as much*	1 128	*European Community*
2 778	*Los Angeles*	1 465	*for example*	1 105	*Soviet Union*
2 575	*a lot*	1 463	*foreign policy*	1 095	*in addition*
2 487	*armed forces*	1 443	*the day*	1 072	*Middle East*
2 456	*in the past*	1 422	*Democratic party*	1 059	*long-term*
2 396	*at home*	1 394	*stock exchange*	1 021	*foreign investment*
2 202	*as well as*	1 358	*central bank*		
2 140	*at the end*	1 344	*no longer*		

ANNEXE E.

Extrait du DELAS anglais de l'informatique

cab,N1
CABE,N1+Sig
cabinet,A0
cabinet,N1
cable,N1
cable,V4
cablecasting,N1
cableco,N1
CableLabs,N2P
cabletex,N3
cabling,A0
cabling,N1
CAC,N1+Sig
cache,N1
cache,V4
cacheable,A0
cached,A0
caching,A0
caching,N1
CACS,N1+Sig
cad,N1
CADAM,N1+Sig
CADAT,N1+Sig
CADD,N1+Sig
CADDIA,N1+Sig
caddie,N1
caddy,N5
caddy,V9
cade,A0
cade,N1
cadet,N1
CADKill,N1
cadmium,N1
CAE,N1+Sig
CAERE,N1+Sig
CAF,N1+Sig
cafe,N1
caffein,N1
CAFS,N1+Sig
CAGD,N1+Sig
cage,N1
cage,V4
CAGR,N1+Sig
CAI,N1+Sig

ANNEXE F.

Extraits du DELAC anglais de l'informatique

%-/+ (classe XC)
inner/macro(macro.N1:s),N+XC:s/+N
inner/macroinstruction(macroinstruction.N1:s),N+XC:s/+N
inner/plane(plane.N1:s),N+XC:s/+N
inner/product(product.N1:s),N+XC:s/+N
Innovative/Software(software.N1:s),N+XC:s/+N
inoperable/time(time.N1:s),N+XC:s/+N
input/acknowledgment(acknowledgment.N1:s),N+XC:s/+N
input/action(action.N1:s),N+XC:s/+N
input/amplification(amplification.N1:s),N+XC:s/+N
...
%-/-/+ (classe XXC)
out-band/signaling(signaling.N1:s),N+XXC:s/+N
out-turn/sheet(sheet.N1:s),N+XXC:s/+N
outer/macro/call(call.N1:s),N+XXC:s/+N
outgoing/line/circuit(circuit.N1:s),N+XXC:s/+N
outline/check/box(box.N3:s),N+XXC:s/+N
outline/number/format(format.N1:s),N+XXC:s/+N
outline/numbering/format(format.N1:s),N+XXC:s/+N
outline/view/options,N+XXC:p
...
%+/-, +/-/-, +/-/-/- (classe CnX)
acceleration(acceleration.N1:s)/of/tape,N+CXX:s/+N
access(access.N3:s)/in/series,N+CXX:s/+N
access(access.N3:s)/to/remote/function,N+CXXX:s/+N
access(access.N3:s)/to/supplementary/services,N+CXXX:s/+N
access(access.N3:s)/to/technical/support,N+CXXX:s/+N
access(access.N3:s)/to/the/Internet,N+CXXX:s/+N
...
%-/+/-/- (classe XCXX)
assembly/language(language.N1:s)/for/Multics,N+XCXX:s/+N
automatic/end(end.N1:s)/of/block,N+XCXX:s/+N
automatic/end(end.N1:s)/of/interrupt,N+XCXX:s/+N
automatic/request(request.N1:s)/for/repeat,N+XCXX:s/+N
automatic/request(request.N1:s)/for/repetition,N+XCXX:s/+N
average/information(information.N1:s)/per/character,N+XCXX:s/+N
...
(autres)
Advanced/Technology/Attachement/packet/interface,N+X:s
advanced/communication/technologies/and/services,N+X:p
advanced/communications/technologies/and/services,N+X:p
advanced/configuration/and/power/interface,N+X:s
advanced/data/communication/control/procedure,N+X:s
...