Grammaires et analyseurs syntaxiques

Anne Abeillé & Philippe Blache

version v1.0e, 14 juin 1999

Table des matières

L'analyse syntaxique automatique

Anne Abeillé(*) & Philippe Blache(**)[1]

Résumé : La syntaxe joue un rôle déterminant dans le traitement automatique du langage naturel. Cependant, en dépit des progrès importants des trente dernières années, elle n'occupe pas toujours une position prépondérante dans les applications. Afin de mieux cerner le problème, nous dressons un bilan de l'existant (pour le français et dans une moindre mesure pour l'anglais) et du chemin parcouru. Nous distinguons à chaque fois ce qui concerne l'accumulation et la formalisation des ressources syntaxiques nécessaires (méthodes artisanales ou auto-apprentissage, formalismes linguistiques ou formalismes ad hoc) et ce qui concerne leur utilisation informatique (en désambiguisation, en analyse syntaxique ou en génération).

Mots-clés : Analyse syntaxique, algorithmes, désambiguïsation, étiquetage, évaluation, formalismes syntaxiques, génération, grammaires électroniques, parsage, corpus annotés.

1. Introduction

La syntaxe devrait être un composant essentiel de toute application en TALN. Ce n'est pas encore toujours le cas, bien que des progrès importants aient été effectués durant ces trente dernières années. Nous sommes en effet passés de systèmes reposant sur des informations syntaxiques très grossières à des approches permettant de décrire très finement des phénomènes syntaxiques non triviaux (e.g. extractions, dislocations, etc.). Le problème vient du fait que les formalismes autorisant ce type de description sont difficiles à implanter. La

[1](*) Université de Paris 7, TALANA et IUF, Case 7003, 2 Place Jussieu, 75005 Paris, `abeille@linguist.jussieu.fr`
(**) Université de Provence, LPL-CNRS, 29 Avenue Robert Schuman, 13621 Aix-en-Provence `pb@lpl.univ-aix.fr`
Cet article est une version remaniée de l'article "La syntaxe" paru dans la revue TAL ([DAN 97])

tentation est donc bien souvent de s'en passer, au risque de n'appréhender que superficiellement les phénomènes visés. La voie est étroite qui consiste à concilier finesse d'analyse et efficacité d'implantation. C'est pourtant à nos yeux la seule qui soit constructive à long terme pour le traitement automatique des langues. Elle n'exclut bien entendu pas que pour tel ou tel type d'application, en fonction du type d'information requis, on puisse se contenter d'une analyse très superficielle.

Nous tentons dans ce chapitre de rendre compte de tous ces aspects et de dresser un bilan de l'existant (pour le français et dans une moindre mesure pour l'anglais) et du chemin parcouru. Nous distinguons à chaque fois ce qui concerne l'accumulation et la formalisation des ressources syntaxiques nécessaires (méthodes artisanales ou basées sur corpus annotés, formalismes motivés par une théorie linguistique ou formalismes ad hoc) et ce qui concerne leur utilisation informatique (en désambiguïsation, en analyse syntaxique ou en génération).

En bref, nous sommes passés en 30 ans d'une simple application de techniques d'analyse des langages formels à l'émergence d'un véritable domaine de recherche : ce que nous appelons aujourd'hui la linguistique informatique (qui tend à devenir une discipline à part entière) provient essentiellement, même si ce domaine évolue aujourd'hui rapidement, de réflexions menées à l'intersection de la linguistique et de l'informatique sur la question de la syntaxe.

2. Le composant syntaxique et ses applications

2.1. *La notion de composant syntaxique en TALN*

Le traitement automatique des langues naturelles (TALN) a commencé avec les applications en traduction automatique. Un progrès dans l'automatisation de la traduction a été la décomposition de cette tâche complexe en trois sous-tâches :

– analyse (texte source),
– transfert,
– génération (texte cible).

La phase d'analyse automatique a été elle-même décomposée (cf. [WIN 83]) en sous-tâches, calquées sur les différents niveaux d'analyse linguistique :

– analyse morphologique (identification des mots ou des morphèmes),
– analyse syntaxique (identification des syntagmes et de leurs fonctions),
– analyse sémantique.

Chacune des ces tâches peut être à son tour une nouvelle fois décomposée. Pour l'analyse lexicale, on distingue ainsi :

- segmentation (identification des frontières de mots et des mots composés),
- lemmatisation (identification du mot sous sa forme canonique),
- étiquetage (identification de la bonne catégorie morpho-syntaxique pour une forme donnée, en contexte).

La prise en compte de la syntaxe des langues naturelles en tant que problème spécifique du TAL est apparue relativement tardivement. Au départ, les techniques d'analyse utilisées étaient celles mises au point pour les langages formels. On pensait que l'analyse des langues naturelles ne devait pas nécessiter de technologie particulière de ce point de vue. Les premières applications (systèmes de dialogue, traduction automatique) utilisaient en fait non pas une analyse syntaxique au sens propre du terme, mais des procédures de bas niveau (transfert lexical, ordonnancement, etc.) De la même façon, les premiers travaux portant sur l'accès à l'information reposaient sur une recherche de mots-clés. Il n'y avait aucune prise en compte des résultats fournis par les linguistes.

Pour l'anglais, l'extraction d'informations à partir de gros volumes textuels tels les dépêches d'agence a vu se constituer plusieurs générations de programmes, évalués dans le cadre des conférences MUC (*Message Understanding Conference* [2]). Dans des domaines restreints, on a vu des systèmes performants basés sur des moules (frames), avec très peu de connaissances syntaxiques, notamment ceux de l'école de Schank pour qui les relations conceptuelles doivent primer les connaissances linguistiques (par exemple le système ATRANS, voir [LYT 86]). La tendance actuelle est plutôt d'utiliser des systèmes d'analyse partielle popularisée sous le nom de "*skimming*" ("écrémage" de textes) ou "*shallow parsing*" (analyse superficielle). Mais comme souligné dans [JAC 90] : *"the ideal skimming system would need no less linguistic knowledge than a full syntactic parser. In fact, knowing what not to parse may require as much information as parsing itself* [3] *"*.

Plus récemment, la tendance a été à la décomposition de l'analyse syntaxique en sous-tâches distinctes :

- le parenthésage (identification des frontières syntagmatiques majeures),
- l'assignation de fonctions aux syntagmes distingués (ou à leur tête),
- la désambiguïsation syntaxique des têtes prédicatives (cadre de sous- catégorisation, actif/passif, etc.),

[2] Voir le site http ://www.muc.saic.com/

[3] *"Le système d'écrémage idéal devrait utiliser autant de connaissances linguistiques qu'un analyseur syntaxique complet. En fait, on a besoin d'autant d'information pour savoir ce qu'on peut ne pas analyser que pour analyser."*

– l'assignation d'une structure syntaxique globale (un arbre) à chaque phrase.

La décomposition de la tâche permet de se fixer à chaque fois un objectif moins ambitieux, jugé plus réaliste. Il est clair cependant, d'un point de vue linguistique, que les tâches ne sont pas indépendantes les unes des autres et qu'à chaque niveau des connaissances de niveaux différents sont nécessaires, ce qui peut entraîner une certaine redondance entre les modules. De plus, et selon le type d'application visé, tout ou partie de ces sous-tâches peut être utilisé.

Il est donc intéressant de préciser tout d'abord les applications du composant syntaxique avant d'examiner plus en détail l'état de l'art des grammaires et des analyseurs.

2.2. *Les applications du composant syntaxique*

La compréhension automatique de textes (ou la traduction automatique) suppose d'interfacer le composant syntaxique avec des composants de plus haut niveau : sémantique, discursif voire stylistique. Mais pour de nombreuses applications, connaissances lexicales et syntaxiques peuvent suffire. Nous ne parlerons pas ici du traitement de la parole, pour lequel il est maintenant clair que des informations syntaxiques sont nécessaires : étiquetage syntaxique pour aider la reconnaissance, frontières de syntagmes pour aider la synthèse de la prosodie (cf. [PIE 97], [BLA 97]).

On peut distinguer les applications demandant une analyse syntaxique complète (traduction) et celles pouvant se contenter d'une analyse partielle (indexation, extraction d'informations...) ; on peut également distinguer celles demandant un analyseur robuste (tolérant aux fautes), comme la traduction ou l'indexation de gros volumes de textes, et celles reposant sur un analyseur "exigeant" c'est-à-dire capable de détecter toutes les agrammaticalités (correction d'orthographe, logiciels d'apprentissage d'une langue). Nous nous limitons ci-dessous aux applications "directes" du composant syntaxique, qui peuvent faire l'impasse sur les composants de plus haut niveau.

2.2.1. *La correction d'orthographe*

Les premiers correcteurs n'incorporaient que des connaissances lexicales (mots inconnus ou connus) et corrigeaient essentiellement les fautes de frappe. La correction des fautes d'accord (30% des fautes en français) nécessite un module d'analyse syntaxique, incorporé généralement à la nouvelle génération des correcteurs commerciaux. On peut citer plusieurs exemples de correcteurs pour le français comme le Correcteur 101 (http ://www.machinasapiens.qc.ca)), Hugo Plus (http ://www.softissimo.com), Cordial (http ://synapse-fr.com) ou Antidote (http ://www.druide.com). Bien entendu, un correcteur parfait devrait également incorporer des connaissances sémantiques (pour la correction des homophones tels que *mer/mère* ou même de certains accords qui supposent d'identifier l'antécédent des pronoms (exemple : *la couleur du cartable que j'ai acheté/*ée*). Cependant, les correcteurs incorporant dictionnaires et grammaires donnent déjà de bons résultats.

2.2.2. *L'indexation automatique*

C'est un des premiers débouchés de l'analyse syntaxique. En indexation libre (sans thésaurus), il s'agit essentiellement de repérer les groupes nominaux non adverbiaux, ce qui peut se faire avec une analyse syntaxique partielle. On peut trouver une application de ce type de technique dans [JAQ 99]. L'indexation automatique constitue la base élémentaire de l'aide au résumé (cf. [MAN 99]).

2.2.3. *L'interrogation de bases de données*

C'est également dans ce domaine, en particulier pour l'interfaçage de base de données en langue naturelle, que la nécessité d'une véritable analyse syntaxique a été clairement identifiée (cf. [SAB 87]). La génération d'une requête nécessite en effet l'analyse de la structure de la question posée. Plus précisément, cette génération se fait généralement en associant des rôles sémantiques aux fonctions syntaxiques. Une analyse complète est donc nécessaire.

2.2.4. *La simplification de textes*

C'est une application semble-t-il récente, qui consiste à découper en phrases, à supprimer les ajouts et les subordonnées ou faire de celles-ci des principales. La réduction de longueur et de complexité attendue permet de satisfaire des normes de langage contrôlés (ou simplifiés) ou de faciliter les traitements ultérieurs comme le résumé, la traduction, etc. (cf. [CHA 96]). Une analyse complète de la phrase est préférable mais peu de connaissances supra syntaxiques sont requises.

2.2.5. *L'alignement automatique de textes*

La disponibilité de corpus bilingues dans les 10 dernières années a favorisé les recherches en alignement automatique. Ces recherches s'appuient toujours sur des textes découpés en phrases et fournissent de meilleurs résultats à partir de textes étiquetés syntaxiquement (cf [LAN 95]). L'alignement automatique de phrases donne de bons résultats, l'alignement de mots nettement moins : on peut penser qu'un alignement automatique de syntagmes constituerait un objectif plus raisonnable. Un projet d'évaluation de systèmes d'alignement traitant le français est en cours dans le cadre du projet ARCADE (consulter le site http ://www.lpl.univ-aix.fr/projects/arcade).

2.2.6. *L'extraction de connaissances linguistiques à partir de textes*

Comme cela commence à se faire pour l'anglais, les textes taggés et arborés permettent de bien meilleurs résultats pour l'extraction de termes, de collocations, de cadres de complémentation, de restrictions de sélection, etc. (cf. Bourigault et Jacquemin, ce volume, mais également [DAI 95], [GAU 95], on peut également consulter le site http ://www.univ-st-etienne.fr/eurise/gi/gi.html traitant de l'inférence grammaticale).

2.2.7. *La génération de phrases*

Enfin, n'oublions pas l'importance des connaissances syntaxiques en génération automatique, qu'il s'agisse d'une application autonome (génération de bulletins météo) ou de la sortie d'un programme plus complexe (de dialogue ou de traduction). La génération automatique de textes suppose planification et connaissances discursives, qui sont prises en charge par le composant stratégique du générateur (*quoi dire* ?). La génération de phrases a besoin d'un composant linguistique (*comment le dire* ?), qui repose sur de nombreuses informations syntaxiques. Les programmes de génération les plus riches à cet égard sont ceux qui incorporent plus qu'une grammaire "jouet", et qui s'appuient sur des formalismes utilisés aussi en analyse, comme par exemple [DAN 85] (lexiques-grammaires), [MKE 86] (grammaires fonctionnelles), [KIT 91] (Théorie Sens-texte), [DAN 98] (GTAG)...[4] A la grammaire de phrase, commune avec l'analyseur, doit être ajoutée une grammaire de texte.

En clair, l'analyse syntaxique automatique (production de toutes les analyses complètes pour chaque phrase) n'est ni la seule ni la plus intéressante application d'une grammaire électronique. Avec la stabilisation des formalismes,

[4]On dit parfois que, du point de vue syntaxique, la tâche de génération est plus aisée que celle d'analyse, puisque le problème de l'ambiguïté (analyses multiples) ne se pose pas. Ceci n'est pas juste, si l'on a pour objectif une génération de qualité avec en sortie un texte non ambigu pour le lecteur humain : éviter de générer des phrases ambiguës demande l'incorporation de toutes les connaissances disponibles en analyse.

on assiste à un effort de construction de grammaires à large couverture ces dernières années (projets GRAAL ou FTAG pour le français, XTAG et LINGO pour l'anglais, grammaires multilingues parallèles du XRCE (Xerox) ...) qui a pour objectif précisément la "réutilisabilité" (et la réversibilité) de ces grammaires pour de multiples tâches.

3. Les connaissances syntaxiques

S'il est clair que le TALN a besoin des travaux des linguistes en syntaxe, la dialectique entre théorie et données, caractéristique des sciences expérimentales, est parfois délicate et peut conduire à des malentendus. L'implantation fidèle d'une théorie ne peut se limiter aux seuls exemples cités par les auteurs de la théorie dans leurs argumentations. Si la théorie est correcte, elle doit pouvoir être étendue à l'ensemble de la langue. Une grammaire électronique doit être basée sur un formalisme pour être cohérente et extensible, mais aussi sur des données pour être "couvrante". Les chercheurs en TALN sont plus attentifs depuis 10 ans aux données des corpus (que les sociolinguistiques étudient surtout à l'oral) et à la diversité des textes ([HIN 83], [CHN 93]) ce qui les amène à prendre en compte des phénomènes souvent mal décrits : parenthèses, nombres, dates. Certains formalismes sont bien adaptés pour accueillir ces phénomènes de micro-syntaxe (e.g. grammaire des dates de [MAU 91] ou [FLI 96], analyse des locutions de [ABE 95], etc.), d'autres moins. La notion de langue de spécialité (cf. [KIT 82]), avec des constructions syntaxiques spécifiques au domaine plus qu'à la langue, est également importante.

3.1. *La représentation des connaissances syntaxiques*

A la différence des connaissances lexicales pour lesquelles le mode de représentation est stable depuis 2000 ans (une liste de formes canoniques rangées par ordre alphabétique), la question de la représentation des connaissances syntaxiques est un problème en soi : les grammaires "pour humains" sont anciennes (cf. [AUR 94], [CHE 94]) mais leur présentation varie d'un auteur à l'autre, n'est jamais formalisée de façon simple (liste ou table), et le couplage n'est pas fait avec un dictionnaire de référence. Les tables des matières des livres de grammaire sont très disparates, et incluent en général autant de morphologie (conjugaison, mots dérivés) et de sémantique ("valeur" des suffixes, "valeur" des compléments circonstanciels, etc.) que de syntaxe proprement dite.

Les années 60 ont vu l'adoption enthousiaste par certains informaticiens des travaux en grammaire générative, qui donnaient enfin au mot "grammaire" un sens calculatoire ([CHO 57]). On a vu des analyseurs très directement inspirés des grammaires transformationnelles ([KUN 62]) qui ont permis de définir un formalisme ad hoc : les ATN ([WOO 70]). Les difficultés rencontrées par ce type de formalisme (peu efficaces, non réversibles, difficiles à maintenir et à mettre à jour...) ont directement inspiré la nouvelle génération de modèles regroupés sous le nom de grammaires d'unification qui sont nés d'une collaboration entre syntacticiens et psycholinguistes insatisfaits du modèle transformationnel d'une part, logiciens et informaticiens travaillant en TALN d'autre part. Comme le note [SHI 86], il convient de distinguer les formalismes grammaticaux, comme les grammaires de métamorphose de [COL 75] ou les grammaires de clauses définies (DCG, [PER 80]), qui proposent dès les années 70 de nouveaux métalangages pour les analyses syntaxiques, et les modèles qui proposent une théorie linguistique à part entière, c'est-à-dire des principes qui, indépendamment du modèle formel, contraignent le type d'analyse possible ou attendu dans la théorie.

Si l'on s'en tient aux théories linguistiques[5], la première grammaire d'unification est la grammaire Lexicale Fonctionnelle (LFG) de J. Bresnan et R. Kaplan ([BRE 82]) qui réhabilite les fonctions grammaticales (redéfinies comme des structures de traits), dans le cadre d'une grammaire universelle et psychologiquement "réaliste", c'est-à-dire mathématiquement contrainte. La seconde, dans l'ordre chronologique, est la grammaire Syntagmatique Généralisée (GPSG) de G. Gazdar, E. Klein, G. Pullum et I. Sag ([GAZ 85]) qui enrichit les règles syntagmatiques en décomposant les catégories en traits et en exprimant des principes de partage de valeurs de traits entre syntagmes, tout en restant dans la classe des grammaires hors contexte. La troisième est la grammaire d'arbres adjoints (TAG) d'A. Joshi ([JOS 87], [JOS 91]), qui met la grammaire "dans le lexique" sous la forme de structures arborescentes élémentaires (arbres lexicalisés) enrichies par des traits, combinées par substitution ou adjonction. La dernière est la grammaire syntagmatique guidée par les têtes (ou endocentrique : HPSG) de C. Pollard et I. Sag ([POL 87], [POL 94]), qui, en tant que grammaire syntagmatique enrichie, se situe dans le prolongement direct de GPSG, mais qui intègre également les innovations d'autres théories. Dans ce modèle, la grammaire se ramène à un système statique de classification et exprime des contraintes sur les informations linguistiques, représentées comme des structures de traits hiérarchisées et typées, et interprétées comme des descriptions partielles des objets linguistiques.

Après l'explosion des années 80, où chaque année voyait éclore de nouveaux modèles, on a assisté à une salutaire stabilisation, qui permet un retour

[5]Pour plus de détails sur chacune de ces théories, voir notamment [ABE 93]. On peut citer aussi les grammaires catégorielles d'unification qui décomposent les catégories en traits, et pour qui, comme pour les grammaires catégorielles classiques, la description des catégories lexicales suffit à déterminer leurs propriétés combinatoires et l'analyse syntaxique se ramène à un problème de déduction logique ([DES 90], [LEC 96]).

aux données et à l'analyse de langues ou de phénomènes encore mal décrits. Actuellement sont surtout actifs LFG, HPSG et TAG. Leurs caractéristiques communes sont le rejet des transformations, la réhabilitation des descriptions de surface et une formalisation à base de structures de traits. Plus précisément, les grammaires d'unification adoptent le même type de représentations (structures de traits) pour le lexique, la syntaxe et la sémantique et manipulent des descriptions partielles spécifiées au fur et à mesure des combinaisons. Elles reposent sur des formalismes déclaratifs (les règles ne sont pas ordonnées) et monotones (chaque règle ne peut qu'ajouter de l'information, sans modification destructrice). Elles sont récemment allées plus loin dans l'abandon de la notion de règle, en supprimant non seulement les dérivations transformationnelles mais aussi les règles syntagmatiques (pour les structures de surface) et les règles lexicales (utilisées en morphologie ou pour les changements de valence). Les règles dynamiques de production sont en effet remplacées par des conditions statiques (positives ou négatives) de bonne formation. D'où l'appellation plus juste de ces théories comme basées sur les contraintes et non seulement sur l'unification (qui est une opération formelle parmi d'autres)[6].

Leur intérêt pour le TALN est triple : d'une part, en tant que systèmes formels, elles facilitent la définition d'algorithmes d'analyse (et ce d'autant plus qu'il y a équivalence avec un modèle mathématique de grammaire formelle)[7] ; d'autre part, elles permettent l'intégration de ressources hétérogènes, à la fois parce qu'elles utilisent le même type de représentation (les structures de traits) et parce qu'elles permettent l'accumulation harmonieuse de contraintes sans leur imposer d'ordre d'application[8] ; enfin par la séparation stricte qu'elles établissent entre données linguistiques et programmes de traitement, elles sont neutres quant à leur application (elles peuvent être utilisées en analyse comme en génération, ou en désambiguïsation)[9].

Dans le cadre des grammaires de constituants[10], on assiste à une unifor-

[6]Ce parti pris en faveur de la déclarativité se traduit en TAG et en HPSG par une réflexion sur la classification des objets linguistiques. Un réseau hiérarchisé pour le lexique, comme pour les syntagmes, permet d'exprimer les généralisations comme des contraintes associées à des superclasses dans la hiérarchie. Les arbres élémentaires d'une grammaire TAG sont vus comme le résultat d'une classification croisée de descriptions lexicales et syntaxiques partielles ([CAN 99]). En HPSG, les schémas de dominance immédiate sont remplacés par une classification croisée des types de syntagmes et des types de clauses ([SAG 97]).

[7]Mise à part HPSG, dont la complexité dans le cas général est celle d'une machine de Turing ([CAR 91]), les autres modèles sont équivalents à des formalismes contraints : grammaires contextuelles pour LFG, hors contexte pour GPSG et "légèrement" sensibles au contexte" pour TAG.

[8]On peut avoir besoin d'accéder à une information phonétique (élision possible) avant une information syntaxique (V ou N) ou le contraire, selon la phrase, dans une application de synthèse vocale par exemple.

[9]Sur la question (non triviale) de la réversibilité des grammaires, voir notamment [APP 87], [EST 93] et Danlos, ce volume.

[10]Peu d'implantations sont basées sur la grammaire générative chomskyenne ; on peut

misation des formalisations : règles de réécriture, règles lexicales, structures de traits, règles de précédence, typage, etc. La tendance à la lexicalisation de la syntaxe permet une meilleure interface avec les dictionnaires. On peut citer également les travaux sur les équivalences entre formalismes (équivalence forte ou faible, cf. [JOS 91]), qui peuvent permettre un partage de ressources linguistiques, par exemple entre HPSG et TAG ([KAS 95], [BEC 98]) ou entre TAG et grammaires catégorielles combinatoires ([DOR 96]), ou la réutilisation de mêmes algorithmes ([VIJ 93]).

On assiste également depuis quelques années à un retour des grammaires de dépendance ([TES 59], [MEL 88]), bien adaptées au TAL car lexicalisées, dans des systèmes qui soit exploitent l'équivalence formelle avec une grammaire de constituants ([GAI 65]) soit adoptent une stratégie originale ([KAR 95], [GIG 97]). Cette théorie est aujourd'hui fréquemment utilisée pour des applications recourrant à une analyse superficielle comme l'étiquetage syntaxique de corpus (cf. [CAL 99]) ou le guidage de la génération prosodique dans les systèmes de phonétisation (cf. [VAN 99])[11].

On peut par ailleurs signaler des travaux mariant grammaires de dépendances et grammaires de constituants (cf. [BES 99]). Dans cette approche, les informations syntaxiques sont organisées en plusieurs types de propriétés permettant de caractériser les catégories. Cette approche, totalement déclarative, permet d'envisager le même type d'applications que les grammaires de dépendance aussi bien en termes d'analyse superficielle (qui se traduit par une vérification d'un sous-ensemble de propriétés), qu'appofondie.

3.2. *Le développement de grammaires à grande échelle*

Les besoins spécifiques du TAL ont contribué au développement d'études linguistiques ou plus particulièrement à une mise en perspective particulière de ces travaux. En effet, les grands projets de description générale d'une langue sont exceptionnels en linguistique, au moins pour ce qui concerne la syntaxe. Un des résultats du TAL a été de fixer comme objectif l'élaboration de grammaires couvrantes. Cela s'est fait dans un premier temps en "récupérant" des descriptions de différentes constructions faites dans des cadres plus ou moins variés et en les intégrant au sein d'un même système. Nous assistons aujourd'hui pour certaines langues comme le français à une telle unification dans un cadre formel homogène. Ce type d'approche permet d'augmenter la couverture des systèmes en intégrant au fur et à mesure la description de nouvelles constructions.

---

citer les travaux de [BER 91], [LAE 91], ou [STA 92] (qui traduit en fait le modèle GB en grammaire d'unification).

[11] Elle fait également l'objet de nombreuses recherches portant sur son implantation (cf. [KAH 98]).

On peut dire qu'on a abouti à un relatif consensus sur ce qu'il faut comme couverture syntaxique, c'est-à-dire qu'une grammaire doit au moins décrire (pour une langue comme le français) :

- les dépendances locales : accord, sous-catégorisation des prédicats, expressions semi-figées, restrictions modifieur-modifié, clitiques, ...
- les dépendances "moyennes" : pronominalisation, contrôle des infinitives, association négative, quantifieurs flottants, ...
- les dépendances à distance : questions, relatives, constructions disloquées...
- les alternances syntaxiques (passif, impersonnel, causatives...)
- les phénomènes de coordination et de comparaison.

D'autres phénomènes (la ponctuation, les relations pronom-antécédent, les ellipses...) peuvent également être pris en compte mais ne relèvent que partiellement de la syntaxe. On peut juger que l'accumulation des connaissances syntaxiques est encore insuffisante (e.g. coordination), mais il est clair que la balle est dans le camp des linguistes, qui sont généralement tributaires de méthodes artisanales (cf. [LAP 97]).

Aucun système ne pourra jamais traiter une langue naturelle dans sa totalité. Comme l'enseigne la sociolinguistique, les usages ne sont pas homogènes et évoluent ; une grammaire ne peut qu'en figer un sous-ensemble à un moment donné. Il serait important, notamment dans le cadre de la communication homme-machine, de prendre mieux en compte les usages du français parlé (cf. [BLN 91]). Il faut pour cela non seulement disposer de descriptions de phénomènes caractérisant cet usage (par exemple les dispositifs en bribes), mais également disposer de mécanismes d'analyse particulièrement robustes, en tout cas capables de s'adapter à différents usages.

Par ailleurs, la "couverture" d'une grammaire n'est pas un objectif purement quantitatif (nombre de règles, nombre de phénomènes décrits...). Son accroissement peut avoir des effets contre-productifs pour l'analyseur (ou le générateur), l'ajout de nouveaux phénomènes entraînant souvent un facteur de complexité supplémentaire et créant de nouvelles "fausses" ambiguïtés[12]. Il est donc important de pouvoir adapter à chaque fois la grammaire au type de corpus à traiter, comme cela commence à se faire pour les dictionnaires électroniques.

[12] Ainsi ajouter aux adverbes de fin de phrase, une possibilité d'insertion entre le verbe et ses compléments entraîne une double analyse (attachement à V ou à Phrase) pour les phrases se terminant par la séquence Verbe-Adverbe ; de même si on incorpore les phrases clivées comme constructions autonomes (pour des cas comme *C'est à midi qu'il est parti*) on aboutit à une double analyse (relative ou clivée) pour une bonne part des phrases en *c'est...qui* ou *c'est...que*.

Le développement de grammaires à grande échelle pose enfin des problèmes non triviaux de partage de tâches, de maintenance et de mise à jour, d'interfaçage avec d'autres ressources (notamment lexicales). Pendant longtemps, les seules grammaires électroniques de qualité représentaient le travail d'un ou deux individus pendant 10 ans et étaient étroitement dépendantes de l'application (par exemple la traduction automatique) et des programmes de traitement (ce qui obligeait à modifier les grammaires chaque fois qu'on modifiait les programmes). On peut citer les grammaires en chaîne de [SAL 73] et [SAE 81], et, pour l'anglais, les grammaires industrielles de Logos, d'IBM ou de Metal.

La stabilisation des modèles syntaxiques basés sur l'unification a permis le développement de plates-formes d'implantation (workbench) multilingues, librement accessibles, telles que XTAG ([PAR 92]), ALE ([CAR 92]), ConTroll ([GOE 97]), PAGE etc. qui incorporent des outils de test et de mise à jour et permettent le développement de grammaires neutres quant à leur application. On assiste depuis une dizaine années au développement de grammaires électroniques "réutilisables" à grande échelle, surtout pour l'anglais, (cf. [DOR 94] pour XTAG, projets LINGO et Verbmobil pour HPSG, cf. [KAY 94]) grâce à ces outils[13]. Pour le français, on peut citer les lexiques-grammaires de l'équipe de M. Gross (cf. [GRO 75]) qui ne sont pas directement formalisées pour l'analyse syntaxique mais constituent une base de connaissances lexico-syntaxiques irremplaçable, ainsi que la grammaire FTAG du français développée à TALANA ([ABE 91], [CAN 99]) utilisée en analyse et en génération ([DAN 98]). Il reste que le coût de développement de telles grammaires est élevé, ce qui explique pourquoi certains se tournent vers des techniques d'auto-apprentissage basées sur des corpus annotés.

3.3. *Les corpus syntaxiquement annotés*

Un autre type de ressources syntaxiques sont les corpus de référence annotés, domaine en plein essor (cf. [ABE 99]). Le développement de l'informatique linguistique ayant commencé plus tôt aux Etats Unis, et avec plus de moyens, il n'est pas étonnant qu'il existe pour cette langue une avance considérable en matière de corpus. Le principal, pour la syntaxe[14], est le *Penn Treebank*

[13]Les projets basés sur des grammaires d'unification peuvent s'écarter sensiblement des thèses ou des analyses de leurs modèles : pour des questions d'efficacité on peut restreindre le métalangage disponible (cf. le système ALEP, [NAM 93]).

[14]Le corpus Suzanne ([SAM 94]), annoté pour la morphologie, la syntaxe et la sémantique est de petite taille (120 000 mots extraits du Brown corpus), le *Bank of English* (200 Millions de mots annotés par une grammaire de dépendance, [JAR 94]) n'est pas publiquement disponible. Les autres corpus, comme le British National corpus (1 Million de mots) sont annotés pour la morpho-syntaxe et en cours d'annotation pour la syntaxe.

([MAR 93])[15] qui contient environ 4 millions de mots (*Brown Corpus*, corpus ATIS, *Dow Jones News service* et *Wall Street Journal*) et a mis en jeu une dizaine d'hommes- années à l'Université de Pennsylvanie sur 4 ans. L'analyse syntaxique a été décomposée en deux phases : étiquetage morpho-syntaxique (tagging) et analyse syntagmatique (parsing), avec à chaque étape une phase automatique et une phase de correction et d'enrichissement manuels. On peut reprocher au résultat de minimiser le poids des locutions et de privilégier une approche syntagmatique trop strictement "chomskyenne" (avec des arbres de surface décorés de catégories vides) même si les auteurs ont progressivement ajouté la notion de fonction grammaticale (mais pas celle de trait syntaxique). Compte tenu de la richesse de ses annotations et de sa taille, ce corpus a des utilisations multiples en TALN : pour l'évaluation des étiqueteurs et des parseurs ([BLK 91]), pour l'induction de grammaires ou de préférences syntaxiques ([BLK 93]), pour la construction automatique d'étiqueteurs ou de parseurs probabilistes ([BRI 92], [MAG 95], ...), pour l'enrichissement des dictionnaires électroniques (extraction de collocations, extraction de cadres de complémentation). On peut souhaiter que l'équivalent soit prochainement disponible pour le français[16].

4. Les analyseurs syntaxiques

Depuis 20 ans, plusieurs objectifs essentiels ont été poursuivis et notamment la finesse de l'analyse, la couverture des analyseurs et leur efficacité (temps d'exécution, robustesse). Dans l'état actuel des choses, le développement d'un analyseur reste un compromis entre finesse et efficacité, ces deux aspects étant inversement proportionnels. Ainsi, les analyseurs à description syntaxique superficielle peuvent être très rapides, propriété plus difficile à obtenir pour les analyseurs reposant sur des grammaires à large couverture proposant des analyses détaillées.

Un progrès important, réalisé avec le développement des grammaires d'unification, a été la séparation radicale des données et des programmes, ce qui permet de réutiliser le même analyseur pour des langues ou des grammaires différentes (ce qui n'était pas le cas des analyseurs basés sur les ATN, ce qui n'est toujours pas le cas pour les analyseurs basés sur les grammaires chomskyennes).

[15] Distribué par le Linguistic Data Consortium, treebank@unagi.cis.upenn.edu.

[16] Seuls existent pour l'instant des corpus annotés pour la morpho-syntaxe (projet GRACE : 1 M mots ; projet Multext : 200.000 mots annotés, projet PAROLE : 200.000 mots annotés, projet CLIF ; 300.000 mots annotés).

4.1. *Les algorithmes d'analyse syntaxique*

On peut distinguer deux types d'approches selon qu'elles s'appuient sur des formalismes "simples" ou complexes. Dans le premier cas, de nombreuses techniques d'analyses sont proposées (cf. [AHO 72], [HOP 79]) et les algorithmes sont de complexité polynomiale[17]. L'algorithme de CYK constitue une référence fondatrice dans ce domaine (cf. [KAS 65], [YOU 67]). Cependant, l'algorithme le plus productif est celui de Earley (cf. [EAR 70]). Il a connu en plus de 20 ans d'existence de nombreuses adaptations (cf. [SCH 94]). On doit également citer Tomita qui a proposé un certain nombre d'algorithmes d'analyse efficaces dans le cas de formalismes simples (cf. [TOM 87], [TOM 91]). On peut par ailleurs citer des heuristiques particulièrement adaptées au TAL, comme la technique du coin-gauche (cf. [ROS 97], [PER 87]) qui consiste à prédire les frontières gauches des constituants ou celle du coin-tête (cf. [KAY 89], [VNO 97]) qui consiste à propager les dépendances syntaxiques à partir des têtes. D'une façon générale, la mise au point de ces algorithmes classiques, l'étude de leurs propriétés, etc. s'est faite dans un cadre proche de celui de l'analyse des langages formels (cf. [AHO 72]) et a surtout concerné les grammaires hors contexte ou une sous-classe (grammaires LR).

Le problème est sensiblement différent pour les formalismes linguistiques plus évolués, comme les grammaires d'unification. Ceux-ci peuvent en effet entraîner le problème de l'analyse dans la catégorie des problèmes NP- complets[18]. Il s'agit dans ce cas de trouver des algorithmes adaptés, mais également des heuristiques permettant une analyse efficace dans les cas moyens (cf. par exemple pour le cas des formalismes DI/PL [SHI 84]). Nous disposons à l'heure actuelle d'un certain nombre de solutions : adaptation des algorithmes classiques ([SHI 85]), compilation vers des formalismes simples (cf. [PHI 85] qui traduisent une grammaire GPSG en une grammaire de réécriture hors contexte), analyse par contraintes (cf. [BLA 96]), etc. A la limite, pour un modèle comme HPSG, le problème de l'analyse peut se ramener à celui de la satisfaction de contraintes, sans règles de production et donc sans parseur proprement dit ; c'est la stratégie adoptée par le système TFS ([EME 90]), élégant et réversible mais peu convaincant en termes d'efficacité. Une approche reposant totalement sur un système de résolution de contraintes nécessite en fait le développement de «solveurs de contraintes» adaptés au problème (cf [BLA 99b]). D'une façon générale, la plupart des algorithmes définis par les chercheurs ont des temps

[17]Les problèmes posés par l'analyse syntaxique en termes de calculabilité sont très intéressants (cf. [BART 87]) mais à prendre avec précaution. Quand on parle de complexité (en temps) polynomiale (au pire des cas) c'est en fonction de la longueur de la phrase à analyser, mais la taille de la grammaire (facteur constant) peut en pratique jouer un rôle beaucoup plus important, de même que les temps d'accès au lexique.

[18]C'est-à-dire qui ne peuvent être résolus que par une machine de Turing non déterministe en temps polynomial.

de traitement incompatibles avec les exigences des industriels, et ceux-ci ont de plus en plus recours à des techniques de pré-traitement, afin de limiter la tâche du composant syntaxique proprement dit. Dans cette optique, il convient de noter le retour récent à des systèmes syntaxiques basés sur les automates à états finis, ou sur les transducteurs, à complexité linéaire, traditionnellement utilisés en analyse lexicale ou en morphologie, pour des tâches qui relèvent de la syntaxe ([ROC 93], [APP 93], [KOR 97], [ROC 97]).

4.2. *Les analyseurs probabilistes*

On peut distinguer, en analyse syntaxique, les approches probabilistes (e.g. [CHU 88], [BOD 95], [RAJ 95]) et les méthodes symboliques (ou linguistiques). Les premières distinguent généralement deux phases d'analyse : l'étiquetage (tagging) et le parenthésage (parsing) : la décomposition de la tâche d'analyse doit permettre la réalisation à moindre coût de sous-programmes indépendants les uns des autres. Elles s'appuient généralement sur des grammaires apprises automatiquement sur corpus (préalablement annoté ou non) : l'étiqueteur morpho-syntaxique ne dispose que d'un dictionnaire indiquant pour chaque forme sa (ou ses) catégorie(s) possible(s), ordonnées selon leur probabilité, et d'une centaine de bigrammes (ou trigrammes) pour les séquences de catégories possibles. L'analyseur maximise la probabilité d'une dérivation, pour une phrase, comme produit des probabilités des règles (ou structures élémentaires) qu'elle met en jeu. Dans une optique sans grammaire préalable, on peut par exemple générer automatiquement toutes les règles binaires possibles à partir d'un vocabulaire non terminal et leur associer une probabilité (qui peut être nulle) sur corpus. Les méthodes probabilistes ont des avantages connus (auto-apprentissage des règles sur corpus, hiérarchisation automatique des analyses), mais elles ne débouchent généralement pas sur des grammaires réutilisables et ne sont pas toujours aisément interfaçables avec des traitements de plus haut niveau. De plus, le coût d'annotation du corpus d'apprentissage est loin d'être négligeable.

Les méthodes linguistiques ont pour avantage de permettre l'utilisation de grammaires lisibles, réutilisables, réversibles, et pour inconvénient le coût de construction de la grammaire au départ (même si on voit se développer de nombreux outils d'aide, cf. [CAN 96], [CAN 99]) et les nombreuses ambiguïtés non résolues (en termes de structures arborescentes ou d'étiquettes fonctionnelles), même si on voit se développer des systèmes de préférence basés sur les travaux en psycholingistique (cf. [KYN 99]). L'avenir est sans doute aux méthodes mixtes et de nombreux travaux vont désormais dans ce sens (cf. [KLA 96]) par exemple en introduisant un "backbone" context-free aux analyseurs probabilistes (cf. [CAL 97], [MAG 97]). Ces approches consistent à la base à associer des poids permettant de contôler la sélections des règles de réécriture.

4.3. *Désambiguïsation ou analyse ?*

Devant la double difficulté de l'analyse syntaxique formée par le taux d'échec lié au fait qu'une grammaire ne peut être totalement "couvrante" et le taux d'ambiguïté artificielle lié au fait que les critères de choix ne sont pas toujours d'ordre syntaxique (voire linguistique), certains ont mis en doute l'utilité d'une analyse syntaxique complète, qui, à l'instar des algorithmes utilisés pour les langages formels, cherche (généralement en parallèle) toutes les analyses possibles pour une phrase bien formée. Deux types d'approches ont été poursuivies : celle d'une analyse partielle, et celle d'une analyse en deux temps avec une première phase de désambiguïsation syntaxique.

Dans le cadre de l'analyse partielle, on peut citer, pour l'anglais, le programme probabiliste de [CHU 88] qui se limite à la détection de groupes nominaux non récursifs, et le système Fidditch [HIN 89], basé sur une grammaire hors contexte, qui ne résout pas tous les attachements syntaxiques et se base sur une stratégie déterministe (une seule analyse par phrase), et pour le français le système à états finis de [ROC 93].

Dans le cadre des systèmes en deux étapes, il s'agit d'adapter à la syntaxe les recettes qui ont fait leurs preuves pour la désambiguïsation morpho-syntaxique. On peut citer l'analyseur ENGCG de [KAR 95] et celui de [AIT 97], basés sur les grammaires de dépendance, ou l'analyseur de [SRI 97] basé sur les TAG. Après une étape de désambiguïsation morphosyntaxique, il s'agit de désambiguïser l'étiquette syntaxique de chaque mot (sa fonction dans les deux premiers systèmes, son arbre élémentaire dans le troisième) et de procéder à des regroupements syntagmatiques ou au calcul des relations de dépendance (dans le cas de [SRI 97])[19].

Les avantages de ces analyseurs partiels, par rapport aux analyseurs classiques, sont leur rapidité et leur robustesse. Leurs inconvénients sont de ne pouvoir diagnostiquer les malformations (ou l'agrammaticalité), et de s'appuyer sur des grammaires souvent peu lisibles et difficiles à mettre à jour (problème d'ordonnancement des contraintes).

[19]Cette approche en deux étapes n'est pas sans rappeler le modèle de la "Sausage machine" proposée en psycholinguistique par [FRA 78]. De façon générale, l'intégration des travaux des psycholinguistes aux analyseurs syntaxiques, notamment avec l'analyseur déterministe de [MAR 80] qui échoue devant les phrases labyrinthes (*garden path*) même bien formées, à l'instar de l'humain, est un autre aspect prometteur des recherches actuelles en analyse syntaxique (cf. [KYN 99]).

Il faut également souligner de nombreuses techniques de plus haut niveau permettant non pas de désambiguïser, mais plutôt de contrôler la désambiguïsation. Il s'agit de techniques s'appuyant sur les propriétés formelles observées dans les structures ambiguës (cf. [MAX 91]). Il peut s'agir également de retarder les choix fait par l'analyseur en les factorisant au sein d'une même représentation. Cette solution peut reposer sur la sous-spécification : plutôt que d'indiquer toutes les solutions de rattachement par exemple, on n'en signale qu'une en indiquant qu'elle n'est pas impérative. C'est l'approche utilisée dans le cadre des TAG par [CHN 97]. Un autre type de solution consiste à utiliser des opérateurs particuliers dans la relations de dominance (cf. [BLA 99a]) de façon à factoriser sans perte d'information.

5. Resultats et evaluation

L'analyse lexicale a fait de grands progrès ces dix dernières années. Pour le français, les entreprises spécialisées en TALN comme les principales équipes de recherche du domaine ont toutes des dictionnaires morphologiques de taille réelle (plus de 300 000 formes fléchies), et des outils, plus ou moins performants, permettant d'appliquer ces dictionnaires sur n'importe quel texte. Plusieurs projets européens (GENELEX, MULTEXT) et français (GRACE) ont permis ou permettront dans un avenir proche de standardiser et de faire partager à toute la communauté TALN de tels dictionnaires et de tels outils[20].

On n'en est malheureusement pas encore là pour l'analyse syntaxique. Des progrès considérables ont été faits. Aujourd'hui, les travaux en linguistique jouent un rôle plus important dans l'identification des connaissances nécessaires (interaction lexique-syntaxe, formalismes morphologiques et grammaticaux...). Une avancée essentielle a également été réalisée dans le domaine des algorithmes d'analyse, inspirés de l'analyse des langages formels ou particuliers au TALN (coin gauche, guidé par les têtes, etc.). On dispose maintenant de techniques "classiques" d'analyse efficace (complexité linéaire ou polynomiale), qui tiennent compte du fait que les grammaires des langues naturelles sont des grammaires ambiguës, et qui évitent les problèmes de récursion infinie ou de retour-arrière (back-track) qui semblaient insurmontables il y a 20 ans (cf. le parseur tabulaire, ou chart parser, de [KAY 80]).

Actuellement, on a plusieurs analyseurs syntaxiques tout terrain, en libre accès, pour l'anglais, comme par exemple :

[20]Le LADL distribue le programme INTEX (cf. [SIL 93]) qui permet l'analyse lexicale de n'importe quel texte avec de gros dictionnaires.

- basés sur des grammaires de constituants : XTAG avec la grammaire anglaise de [DOR 94], un mode analyse classique et un mode désambiguïsation ("supertagging" [JOS 94], [SRI 97])[21], ou ALE avec la grammaire de G. Penn (implantation de [POL 94]).
- basés sur des grammaires de dépendance : le Link Grammar Parser ([SLE 93])[22], ou ENGCG ([KAR 95])[23] qui fournit des analyses partielles.

Pour le français, on a pour l'instant en libre accès sur le web, outre des étiqueteurs[24], deux analyseurs syntaxiques : celui du LATL ([LAE 91], [WEH 97])[25] basé sur une syntaxe chomskyenne (modèle du Gouvernement et du Liage simplifié et limité aux structures de surface), celui du GREYC ([GIG 97])[26] qui effectue des analyses partielles selon un modèle librement inspiré de L. Tesnière.

L'évaluation en analyse syntaxique, comme en TALN, est difficile et tend à devenir un domaine de recherche à part entière (cf. [SPA 96], [MAR 99]). Outre les corpus étiquetés et arborés, les bases de phrases tests constituent une ressource précieuse ([EST 97]). Si l'on veut évaluer automatiquement des résultats d'analyse, on est souvent conduit, même en se limitant aux grammaires de constituants, compte tenu de la diversité des choix d'analyse et des catégories, à ne considérer que le pourcentage de syntagmes bien reconnus et à assimiler le taux d'échec aux parenthèses qui "croisent" celles du corpus de référence (cf. [BLK 91], technique Perseval, critiquée par [CAL 99]). On aboutit, en fait, à un compromis entre le taux d'erreur (pourcentage de mots ou de phrases ne recevant pas la bonne analyse) et le taux d'ambiguïté (pourcentage de mots ou de phrases recevant plus d'une analyse). Pour les analyseurs (probabilistes ou symboliques) de l'anglais basés sur des grammaires de constituants, les résultats publiés font état pour des corpus de type journalistique d'un pourcentage maximum de 70% de phrases totalement (et correctement) analysées, avec, pour les analyseurs à base linguistique, 4 à 5 analyses par phrase en moyenne.

Pour les analyseurs basés sur les grammaires de dépendance ou sur une approche en termes de désambiguïsation, les résultats sont meilleurs, en partie parce qu'on peut compter en nombre de mots (et non en nombre de phrases) bien analysés : [JAR 94] fait état pour ENCGC d'un taux d'erreur de 2% des mots (ce qui fait 30% des phrases sur texte journalistique) avec environ 15% des mots recevant plus d'une étiquette fonctionnelle ; [SRI 97] fait état pour

[21] http ://www.cis.upenn.edu/ xtag/
[22] http ://bobo.link.cis.cmu.edu/grammar/html/
[23] http ://www.lingsoft.fi/, on peut envoyer un texte à engcg@ling.helsinki.fi
[24] Celui de l'INaLF (Lecomte et Paroubek 1996) et celui de Xerox ([CHN 95]), avec des résultats de 97-98% de mots simples correctement étiquetés, pour un jeu de moins de 40 étiquettes. Ces taggeurs prennent peu en compte les mots composés, et ne traitent pas la totalité des informations morphologiques, par exemple le mode des verbes, le genre des noms ou des adjectifs, sans parler des informations sur les lemmes.
[25] http ://latl.unige.ch/latl/fips.html
[26] http ://www.info.unicaen.fr/ giguet/

son "supertaggeur" (qui assigne un arbre élémentaire à chaque mot) d'un taux d'erreur de 8% des mots (sans ambiguïté) ou de 3% (en considérant pour chaque mot les 3 premiers "supertags") sur textes journalistiques[27].

Les résultats actuels en analyse syntaxique doivent être présentés de façon différenciée. En effet, l'analyse syntaxique ne peut être étudiée en dehors du contexte du TAL dans son ensemble. Elle est évidemment tributaire d'informations "extérieures" (notamment lexicales) et les performances obtenues dans ces secteurs ont sur elle une influence directe. Ainsi, une des sources d'erreurs en analyse syntaxique vient non pas de la syntaxe mais du fait que les connaissances de plus haut niveau (sémantique, discursif, encyclopédique) ne sont pas disponibles sous forme utilisable (formalisée) à grande échelle. Les exemples sont bien connus, s'agissant par exemple de trouver l'antécédent des pronoms pour les phénomènes d'accord (*je les ai mangés* ou *mangées*?), de rattacher les groupes prépositionnels (dans *la soupe de légumes du restaurant*, on doit préférer *soupe du restaurant* à *légumes du restaurant*), avec des interactions entre les deux (*les problèmes de cet homme qui t'intéresse* ou *qui t'intéressent*?)[28]. Un autre critère d'évaluation de l'analyse concerne, outre la robustesse et la couverture, sa "granularité", c'est-à-dire la finesse des descriptions obtenues. Ce n'est pas la même chose de repérer les syntagmes majeurs et de désambiguïser la construction précise d'un verbe ambigu en contexte (ce qui est un premier pas vers sa désambiguïsation sémantique). On peut dans le premier cas se limiter à une grammaire de réécriture hors contexte, voire à un automate à états finis (si on néglige les syntagmes récursifs), tandis que dans le second on doit mettre en œuvre un lexique syntaxique très riche et une description détaillée des constructions associées en surface (passif, causative etc.).

Il faut, pour terminer cette revue, citer un certain nombre de résultats obtenus par ce que nous pourrions appeler des "effets de bord". Ils correspondent à des avancées obtenues en linguistique ou en informatique grâce à une influence mutuelle de ces domaines. On peut citer ainsi l'influence directe qu'a eu la question de l'analyse syntaxique sur la programmation logique (origine du langage Prolog). Réciproquement, l'introduction de la notion d'unification dans les formalismes syntaxiques a joué un rôle fondamental dans l'apparition de nouvelles théories linguistiques et de descriptions syntaxiques plus fines. De la même façon, l'utilisation et la définition des structures de traits sont issues d'échanges entre la linguistique et l'informatique (représentation des connaissances, logique).

---

[27] Ces résultats ne sont pas si éloignés du taux précédemment cité pour les analyseurs, si l'on compte en termes de chemins d'étiquettes (ou d'arbres élémentaires) par phrase.

[28] On retrouve le même problème en analyse lexicale : les étiqueteurs n'ont généralement pas les moyens de désambiguïser *les cours de maths* (*cours* masculin ; lemme 1) et *les cours de récréation* (*cour* féminin ; lemme 2) et de nombreux contextes vont rester ambigus (*Je n'aime pas les cours*).

6. Conclusion

Le constat primordial établi par le TAL est celui de l'importance de la linguistique. Toutes les approches en TAL ne prenant pas en compte cette discipline dans sa globalité ont conduit à des impasses. On peut imaginer des systèmes s'intéressant à un seul des sous-domaines de la linguistique (e.g. lexicologie, syntaxe, sémantique). Mais cela n'est possible qu'en figeant les autres connaissances et en les rendant en quelque sorte implicites. La plupart des travaux s'intéressant à la sémantique jusqu'à une période relativement récente prétendaient par exemple se passer de syntaxe. Ceci est en fait rendu possible par le fait que ces études, d'une part, portent sur des domaines extrêmement limités et, d'autre part, utilisent des bases de connaissances très détaillées dans lesquelles les structures sémantiques associées aux objets contiennent implicitement les informations syntaxiques. On sait aujourd'hui qu'une démarche générale ne peut que prendre en compte l'ensemble des questions abordées par la linguistique.

L'objectif essentiel étant à terme l'analyse de différents usages de la langue, on peut donc proposer plusieurs axes de recherche pour l'analyse syntaxique : amélioration de la couverture et de la robustesse, définition de relations d'intégration entre différents modes de communications (écrit, oral, geste) mettant en jeu la syntaxe[29].

Il reste bien entendu un grand nombre de problèmes à régler. Il concernent plus particulièrement soit des aspects linguistiques, en particulier en termes de description, soit des aspects liés à l'implantation.

– *Linguistique* : d'une façon générale, l'un des problèmes fondamentaux de la syntaxe vient du fait qu'il n'est pas possible de définir une grammaire couvrant la totalité d'une langue. Cependant, l'augmentation de la couverture des analyseurs passe par la prise en compte de phénomènes plus variés. Le problème posé devient celui de la cohérence des grammaires développées. La description d'un phénomène particulier ne doit pas se faire à l'aide de mécanismes ad hoc. On constate cependant qu'il est très difficile de rester dans un cadre linguistique qui, sans être universel, serait toutefois suffisamment général.
– *Implantation* : on peut citer deux types de problèmes, à des niveaux différents, mais emblématiques du TAL. Tout d'abord, force est de constater que le traitement de l'ambiguïté offre une résistance particulière. Les solutions proposées ne sont généralement pas satisfaisantes ni d'un point de vue formel ni opérationnel. Parmi les voies actuellement explorées,

[29] Ainsi, la prise en compte de l'intonation peut fournir des indications précieuses en termes de limite d'unités. Dans la même perspective, les relations spatiales peuvent constituer un pivot essentiel entre le geste et l'écrit (notamment pour les problèmes de référence).

certaines peuvent constituer une réponse adaptée dans le cadre d'applications particulières (e.g. désambiguïsation interactive). D'autre sont plus générales, mais dans l'état actuel des choses, l'ambiguïté reste un des facteurs de complexité essentiels de l'analyse syntaxique.

L'autre problème essentiel concerne la généralité et la réutilisabilité des processus d'analyse. Pour répondre à cet objectif, l'implantation doit être aussi proche que possible du modèle théorique qui, lui, est par essence général (à vocation universel). Dans le même temps, il faut que les outils utilisés pour l'implantation soient également génériques : le développement de plateformes ad hoc ne permet en effet pas d'envisager une bonne évolution des analyseurs, ceux-ci devenant totalement dépendants de leur environnement. Il convient donc de trouver un équilibre entre ces deux impératifs. La question posée ici est celle de l'interprétation des théories linguistiques qui consiste à rechercher une mise en œuvre informatisée fidèle au modèle théorique tout en restant générique.

Références

[ABE 91] A. Abeillé *Une grammaire lexicalisée d'arbres adjoints pour le français : application à l'analyse automatique*, Thèse de Doctorat, Université Paris 7. (à paraître aux Editions du CNRS), 1991.

[ABE 93] A. Abeillé *Les nouvelles syntaxes : grammaires d'unification et analyse du français*, Armand Colin, Paris, 1993.

[ABE 95] A. Abeillé "The flexibility of French idioms: a representation with LTAG", in M. Everaert, E. Van der Linden, A. Schenk, Schreuder (eds) *Idioms: structural & psychological perspectives*, LEA: New York. pp. 15-42, 1995.

[ABE 99] A. Abeillé (ed) *ATALA Treebank Workshop*, http://talana.linguist.jussieu.fr/treebank99, 1999.

[AHO 72] A. Aho, J. Ullman *The Theory of Parsing, Translation and Compiling*, Prentice-Hall, 1972.

[AIT 97] S. Aït-Mokhtar, J-P. Chanod "Incremental finite-state parsing", in *Actes 5e ANLP*, Washington, pp. 72-79, 1997.

[APP 87] D. Appelt "Bidirectional grammars and the design of natural language generation systems", in *Actes 3rd Theoretical issues in NLP*, Las Cruces, pp. 185-191, 1987.

[APP 93] D. Appelt, J. Hobbs, J. Bear, D. Israel, M. Tyson "FASTUS : a finite state processor for information extraction from real-world texts", in *Actes IJCAI*, Chambéry, 1993.

[AUR 94] S. Auroux *La révolution technologique de la grammatisation*, Liège, Mandaga, 1994.

[BART 87] G. Barton , R. Berwick, E. Ristad *Computational Complexity and Natural Language*, MIT Press, 1987.

[BEC 98] T. BECKER ET AL. "An efficient kernel for multilingual generation in speech to speech dialogue translation", in proceedings of ACL-COLING, pp. 110.116, 1998.

[BER 91] R. BERWICK, S. ABNEY, T. TENNY *Principle-based parsing*, Dordrecht: Kluwer, 1991.

[BES 99] G. BÈS, P. BLACHE *Propriétés et analyse d'un langage*, in *Actes de TALN'99*, 1999.

[BLA 96] P. BLACHE, J.-L. PAQUELIN "Active Constraints for a Direct Interpretation of HPSG", in *proceedings of HPSG'96*, 1996.

[BLA 97] P. BLACHE, G. CAELEN (EDS) *Prosodie et Syntaxe*, T.A.L, 38 :1, 1997.

[BLA 99a] P. BLACHE "Shared Trees: Representing a Parse Forest with a Single Tree", in *proceedings of VEXTAL-99*, 1999.

[BLA 99b] P. BLACHE "L'analyse par contraintes : une nouvelle approche pour l'analyse linguistique", Rapport de recherche du LPL, http ://www.lpl.univ-aix.fr/ blache/publis.html, 1997.

[BLK 91] E. BLACK, S. ABNEY, D. FLICKINGER, C. GDANIEC, R. GRISHMAN, P. HARRISON, D. HINDLE, R. INGRIA, F.JELINEK, J. KLAVANS. M. LIBERMAN, M. MARCUS, S. ROUKOS, B. SANTORINI, T. STRZALKOWSKI "A procedure for quantitatively comparing the syntactic coverage of English grammars", *Actes ARPA Workshop*, IBM, New York, pp. 306-311, 1991.

[BLK 93] E. BLACK, R. GARSIDE, G. LEECH *Statistically-driven computer grammars for English: the IBM-Lancaster approach*, Amsterdam: Rodopi, 1993.

[BLN 91] C. BLANCHE-BENVENISTE, C. ROUGET, K. VAN DEN EYNDE *Le Français Parlé - Etudes Grammaticales*, Paris : Editions du CNRS, 1991.

[BOD 95] R. BOD *Enriching linguistics with statistics: performance models of natural language*, Thèse de PhD, Université d'Amsterdam, 1995.

[BRE 82] J. BRESNAN ED. *The mental representation of grammatical relations*, MIT Press, 1982.

[BRI 92] E. BRILL "A simple rule-based part of speech tagger", in *Actes Conférence ANLP*, Trento, Italie, pp. 151-155, 1992.

[CAN 96] M.-H. CANDITO "Un outil pour la génération semi-automatique de LTAGs" in *Actes TALN'96*, Marseille, 1996b.

[CAN 99] M.-H. CANDITO *Organisation modulaire et paramétrable de grammaires électroniques lexicalisées. Application au franĄçais et Ąà l'italien*, Thèse de doctorat, Université Paris 7, 1999.

[CAR 91] R. CARPENTER "The generative power of categorial grammars and HPSG with lexical rules", in *Computational Linguistics*, 17:3, pp. 301-314, 1991.

[CAR 92] R. CARPENTER *The Logic of typed feature structures*, Cambridge University Press, 1992.

[CAL 97] J. CARROLL, D. WEIR "Encoding Frequency Information in Lexicalized Grammars", in *IWPT'97*, 1997.

[CAL 99] J. CARROLL, G. MINNEN, T. BRISCOE "Corpus Annotation for Parser Evaluation", *Treebank Workshop*, 1999.

[CHA 96] M. CHANDRASEKAR, C. DORAN, B. SRINIVAS "Motivations and methods for text simplification", in *Actes 15ème COLING*, Copenhague, 1996.

[CHN 93] J.-P. CHANOD "Problèmes de robustesse en analyse syntaxique", in *Actes 2ème Conférence ILN*, Nantes, pp. 223-244, 1993.

[CHN 95] J.-P. CHANOD, P. TAPANAINEN "Tagging French: comparing a Statistical and a constraint-based method", in *Actes 7ème Conférence European ACL*, Dublin, pp. 149- 156, 1995.

[CHN 97] J. CHEN, K. VIJAY-SHANKER "Towards a Reduced Commitment, D-Theory Style TAG Parser", in proceedings of the *International Workshop on Parsing Technologies*, 1997.

[CHE 94] J.-C. CHEVALIER *Histoire de la grammaire française*, Paris : PUF, 1994.

[CHO 57] N. CHOMSKY *Syntactic structures*, MIT Press [trad. fr. 1965 Structures syntaxiques, Seuil], 1957.

[CHU 88] K. CHURCH "A stochastic parts program and noun phrase parser for unrestricted text", in *Actes 2de ANLP Conference*, Austin, pp. 136-143, 1988.

[COL 75] A. COLMERAUER *Les grammaires de métamorphose*, rapport interne, Département d'informatique, Université d'Aix-Marseille II. in [L. Bolc (ed) (1978) Natural language communication with computers, Springer Verlag], 1975.

[DAI 95] B. DAILLE "Repérage et extraction de terminologie par une méthode mixte statistique et linguistique", in *TAL*, 36 :1-2, pp. 101-118, 1995.

[DAN 85] L. DANLOS *Génération automatique de textes en langues naturelles*, Paris : Masson, 1985.

[DAN 96] L. DANLOS, F. MEUNIER "GTAG : présentation et applications industrielles", in *Actes 3e Conférence ILN*, Nantes, 1996.

[DAN 97] L. DANLOS, J. VÉRONIS *Etat de l'art*, revue TAL, 38 :2, 1997.

[DAN 98] L. DANLOS "G-TAG : un formalisme lexiclaisé pour la génération de textes inspiré de TAG", in revue TAL, 39 :2, 1998.

[DES 90] J.-P. DESCLÉS *Langages applicatifs, langues naturelles et cognition*, Paris : Hermès, 1990.

[DOR 96] C. DORAN , B. SRINIVAS "Bootstrapping a CCG from LTAG", in Abeillé, Rambow (eds), *Tree Adjoining grammars*, CSLI, Stanford (à paraître).

[DOR 94] C. DORAN, D. EGEDI, B. HOCKEY, B. SRINIVAS, M. ZAIDEL "The XTAG system : a wide coverage grammar for English", in *Actes 15e COLING*, Kyoto, pp. 922-928. (version mise à jour sous http://www.cis.upenn.edu/ xtag/), 1994.

[EAR 70] J. EARLEY "An Efficient Context-Free Parsing Algorithm, Communication of the ACM, 13:2, pp. 94-102. reprint in B. Grosz, K. Sparck Jones, B. Webber (eds) (1986) *Readings in Natural Language Processing*, Morgan Kaufman, pp. 25-34, 1970.

[EME 90] M. EMELE, R. ZAJAC "Typed unification grammars", in *Actes 13e COLING*, Helsinki, pp. 293-298, 1990.

[EST 93] D. ESTIVAL "Une grammaire pour l'analyse et la génération", *TAL*, 34 :1, pp. 83- 100, 1993.

[EST 97] D. ESTIVAL, S LEHMANN "TSNLP : des jeux de phrases test pour le TALN", *TAL*, 38 :1, pp. 155-172, 1997.

[FLI 96] D. FLICKINGER "Time expressions in an HPSG grammar", in T. Gunji (ed) *Studies on the Universality of Constraint-based phrase structure grammar*, Osaka University, pp. 1-8, 1996.

[FRA 78] L. Frazier, J. Fodor "The sausage machine : a new two-stage parsing model", *Cognition*, 6, pp. 291-325, 1978.

[GAU 95] E. Gaussier, J-M. Langé "Modèles statistiques pour l'extraction de lexiques bilingues", *TAL*, 36 :1-2, pp. 133-156, 1995.

[GAI 65] H. Gaifman "Dependency systems and phrase-structure systems", *Information and control*, 8, pp. 304-337, 1965.

[GAZ 85] G. Gazdar, E. Klein, G. Pullum, I. Sag *Generalized Phrase structure grammar*, Harvard University Press, 1985.

[GIG 97] E. Giguet, J. Vergne "Syntactic analysis of unrestricted texts", in *Actes Recent advances in NL processing*, Tzigov Chark, pp. 276-281, 1997.
Eng

[GOE 97] T. Götz, D. Meurers "The ConTroll System as Large Grammar Development Platform", Proceedings of the Workshop *Computational Environments for Grammar Development and Linguistic Engineering (ENVGRAM)*, 1997.

[GRO 75] M. Gross *Méthodes en syntaxe*, Paris : Hermann, 1975.

[HIN 83] D. Hindle "Deterministic parsing of syntactic non fluencies", in *Actes 21e ACL Meeting*, 1983.

[HIN 89] D. Hindle "Acquiring disambiguation rules from text", in *Actes 27e ACL Meeting*, Vancouver, 1989.

[HOP 79] J. Hopcroft, J. Ullman *Introduction to Automata Theory, Languages and Computation*, Addison-Wesley, 1979.

[JAC 90] P. Jacobs "To parse or not to parse : relation-driven text skimming", in *Actes 13e COLING*, Helsinki, vol. 2, pp. 194-198, 1990.

[JAQ 99] C. Jacquemin, E. Tzoukermann "NLP for Term Variant Extraction: Synergy between Morphologym Lexicon and Syntax", in *Natrual Language Information retrieval*, T. Strzalkowski (ed), Kluwer, 1999.

[JAR 94] T. Järvinen "Annotating 200 Millions words : the bank of English project", in *Actes 15e COLING*, Kyoto, 565-568, 1994.

[JOS 87] A. Joshi "Introduction to Tree Adjoining Grammar", in A. Manaster-Ramer (ed), *The Mathematics of Language*, J. Benjamins, 1987.

[JOS 91] A. Joshi, D. Weir, K Vijay-Shanker "The convergence of mildly context- sensitive formalisms", in P. Sells, S. Shieber, T. Wasow (eds), Foundational issues in NLP, MIT Press, 1991.

[JOS 94] A. Joshi, B. Srinivas "Disambiguation of super parts of speech (or supertags) : almost parsing", in *Actes 16th COLING*, Kyoto, pp. 154-160, 1994.

[KAH 98] S. Kahane, A. Polguère (eds) *Workshop on the Processing of Dependency Grammars*, 1998.

[KAR 95] F. Karlsson, A Voutilainen, J. Aikkilä, A. Antilla (eds) *Constraint grammar: a language independent System for parsing unrestricted texts*, Berlin: Mouton de Gruyter, 1995.

[KAS 65] T. Kasami "An efficient Recognition and Syntax Algorithm for Context-Free Languages", Scientific Report AFCRL-65-758, Air Force Cambridge Research Lab., Bedford, Mass, 1965.

[KAS 95] R. Kasper, K. Netter, K. Vijay-Shanker "Compiling a HPSG grammar into LTAG", in *Actes 33e ACL Meeting*, Santa-Cruz, 1995.

[KAY 80] M. KAY "Algorithm and data structure in syntactic processing", Actes Nobel Symposium on Text Processing, Gotheborg,. reprint in B. Grosz, K. Sparck Jones, B. Webber (eds) (1986) *Readings in Natural Language Processing*, Morgan Kaufman, pp. 35-70, 1980.

[KAY 89] M. KAY "Head-Driven Parsing", in *Actes International Workshop on Parsing Technologies*, Pittsburg, 1989.

[KAY 94] M. KAY, J.-M. GAWRON, P. NORVIG *Verbmobil : a Translation System for Face- to-Face Dialog*, CSLI, Chicago Press, 1994.

[KIT 82] R. KITTREDGE, J. LEHRBERGER (EDS) *Sublanguages. Structures of Language in Restricted Semantic Domains*, Walter de Gruyter, 1982.

[KIT 91] R. KITTREDGE, A. POLGUÈRE "Dependency grammars for bilingual text generation", in *International Conference on Current Issues in Computational Linguistics*, Penang, pp. 318-330, 1991.

[KLA 96] J. KLAVANS, P. RESNIK (EDS) *The Balancing Act - Combining Symbolic and Statistical Approaches to Language*, MIT Press, 1996.

[KOR 97] A. KORNAI (ED) "Finite state systems", *Journal of language engineering*, 2:4, 1997.

[KUN 62] S. KUNO, A. OETTINGER "A multiple path syntactic analyser", *Information processing*, 1962.

[KYN 99] A. KYNION " Hiérarchisation d'analyses basées sur des informations dépendencielles dans le cadre des LTAG", in *Actes de TALN'99*, 1999.

[LAE 91] C. LAEZLINGER, E. WEHRLI "Un analyseur interactif pour le français", *TA Informations*, 32 :2, pp. 35-50, 1991.

[LAN 95] J.-M. LANGÉ, E. GAUSSIER "Alignement de corpus multilingues au niveau des phrases", *TAL*, 36 :1-2, pp. 67-80, 1995.

[LAP 97] E. LAPORTE "Les mots. Un demi-siècle de traitements", *TAL*, 38 :2, 1997.

[LEC 96] A. LECOMTE (ED) Grammaires et théorie de la preuve, *TAL*, 37 :2, 1996.

[LEJ 96] J. LECOMTE , P. PAROUBEK *Le catégoriseur d'E. Brill : mise en œuvre de la version entraînée pour le français*, Rapport Interne, INALF : Nancy, 1996.

[LYT 86] S. LYTINEN, A. GERSHMAN "ATRANS automatic processing of money transfer messages", in *Actes 5e National Conference on artificial intelligence*, Philadelphia, 1986.

[MKE 86] K. MCKEOWN *Text generation using discourse strategies and focus constraints to generate natural language text*, Cambridge University Press, 1986.

[MAG 95] M. MAGERMAN"Statistical decison-tree models for parsing", in *Actes 23e ACL Meeting*, 1995.

[MAN 99] I. MANI, M. MAYBURY (EDS) *Advances in Automatic Text Summarization*, MIT Press, 1999.

[MAG 97] C. MANNING, B. CARPENTER "Probabilistic Left Corner Grammars", in *IWPT'97*, 1997.

[MAR 80] M. MARCUS *A theory of syntactic recognition for natural language*, MIT Press, 1980.

[MAR 93] M. MARCUS, B. SANTORINI, M.-A. MARCINKIEWICZ "Building a large annotated corpus of English : the Penn Treebank", *Computational Linguistics*, vol. 19, n ° 2, 313- 330, 1993.

[MAR 99] J. MARIANI, J. VÉRONIS "Evaluation en TAL", Numéro spécial de la revue *Langues*, 1999.

[MAU 91] D. MAUREL "Préanalyse des adverbes de date en français", *TA Informations*, 32 :2, pp. 5-18, 1991.

[MAX 91] J. MAXWELL, R. KAPLAN "A Method for Disfunctive Constraint Satisfaction", in M. Tomita (ed.) *Current Issues in Parsing Technologies*, Kluwer, 1991.

[MEL 88] I. MEL'ČUK *Dependency syntax : theory and practice*, SUNY Press, Albany, 1988.

[NAM 93] F. NAMER, P. SCHMIDT "Une grammaire du français dans un formalisme à structures de traits typées", in *Actes 2e Conference ILN*, Nantes, pp. 105-128, 1993.

[PAR 92] P. PAROUBEK, Y. SCHABES, A. JOSHI "XTAG: a graphical Workbench for developing TAGs", in *Actes 4 ° Conf. on Applied NLP*, Trento, pp. 223-227, 1992.

[PER 80] F. PEREIRA, D. WARREN "Definite clause grammars for Natural language analysis : a survey of the formalism and a comparison with ATN", *Artificial Intelligence*, 13, pp. 231-278, 1980.

[PER 83] F. PEREIRA, D WARREN "Parsing as deduction", in *Actes 21e ACL Meeting Cambridge*, pp. 137-144, 1983.

[PER 87] F. PEREIRA, S. SHIEBER *Prolog and Natural Language Analysis*, CSLI n ° 10, Stanford, 1987.

[PER 92] F. PEREIRA, Y. SCHABES "Inside-outside reestimation from partially bracketed corpora", in *Actes 30e ACL meeting*, 1992.

[PHI 85] J. PHILLIPS, H. THOMPSON "GPSGP - A Parser for GPSG", *Linguistics*, 23, 1985.

[PIE 97] J.-M. PIERREL "Bilan des grandes orientations de recherche en traitement automatique du langage parlé en France", *TAL*, 38 :2, 1997.

[POL 87] C. POLLARD, I. SAG *Information-based syntax and semantics, vol. 1*, CSLI series, Chicago Press, 1987.

[POL 94] C. POLLARD, I. SAG *Head-driven Phrase Structure Grammar*, CSLI series, Chicago Press, 1994.

[RAJ 95] M. RAJMAN "Approche probabiliste de l'analyse syntaxique", *TAL* 36-1-2. pp 157-200, 1995.

[ROC 93] E. ROCHE *Analyse transformationnelle du français par transducteurs et lexiques-grammaires*, Thèse de doctorat, Université Paris 7, 1993.

[ROC 97] E. ROCHE, Y. SCHABES (EDS) *Finite-State Language Processing*, MIT Press, 1997.

[ROS 97] D. ROSENKRANTZ, P. LEWIS "Deterministic Left-Corner Parser",*Annual Symposium on Switching and Automata Theory*, 1970.

[SAB 87] P. SABATIER *Contribution au développement d'interfaces en langage naturel*, Thèse d'Etat, Université Paris VII, 1987.

[SAG 97] I. SAG "English relative clause constructions", *Journal of Linguistics*, 1997.

[SAE 81] N. SAGER *Natural language information processing: a computer grammar for English and its applications*, Reading, Addison Wesley, 1981.

[SAL 73] M. SALKOFF *Une grammaire en chaîne du français*, Dunod, 1973.

[SAM 94] G. Sampson "Suzanne: a domesday book of English grammar", in *Corpus-based Research into language*, N. Oostdijk & P. de Haan (eds), Rodopi: Amsterdam, pp. 169-187, 1994.

[SCH 94] Y. Schabes "Left to right parsing of LTAG" in *Computational Intelligence*, 1994.

[SHI 84] S. Shieber "Direct Parsing of ID/LP Grammars", *Linguistics and Philosophy*, 7, 1984.

[SHI 85] S. Shieber "Using rectriction to extend parsing algorithms for complex feature-based formalisms", in *Actes 23e ACL Meeting*, Chicago, pp. 145-152, 1985.

[SHI 86] S. Shieber *Introduction to unification-based theories of grammar*, CSLI series, University of Chicago Press. [trad. fr. in Miller et Torris (eds) (1990), Formalismes grammaticaux pour le traitement automatique des langues naturelles. Hermès, pp. 27-86], 1986.

[SIL 93] M. Silberztein *Dictionnaires électroniques et analyse automatique de textes : le système INTEX*, Paris, Masson, 1993.

[SLE 93] D. Sleator et D. Temperley "Parsing English with a link grammar", in *Actes Internat. Workshop on Parsing Technologies*, 1993.

[SPA 96] K. Spark-Jones, J. Galliers *Evaluating Natural Language Processing Systems*, Lecture Notes in Artificial Intelligence, Springer, 1996.

[SRI 97] B. Srinivas *Complexity of lexical descriptions and its relevance for partial parsing*, Thèse de PhD, University of Pennsylvania, Philadelphia, 1997.

[STA 92] E. Stabler *The logical aproach to syntax*, MIT Press, 1992.

[TES 59] L. Tesnière *Eléments de syntaxe structurale*, Paris : Klincksieck, 1959.

[TOM 87] M. Tomita "An Efficient Augmented Context-Free Parsing Algorithm", *Computational Linguistics*, 13, 1987.

[TOM 91] M. Tomita "Parsing 2-Dimensional Language", in *Current Issues in Parsing Technologies*, Kluwer, 1991.

[VAN 99] G. van Noord "Pauses location and duration calculated with syntactic dependencies and textual considerations for t.t.s. system" in *Actes de ICPhS 1999*, 1999.

[VNO 97] G. van Noord "An Efficient Implementation of the Head-Corner Parsing", *Computational Linguistics*, 1997.

[VIJ 93] K. Vijay-Shanker, D. Weir "Parsing some constrained grammar formalisms", *Computational Linguistics*, 19:4, pp. pp. 591-636, 1993.

[WEH 97] E. Wehrli *L'analyse syntaxique des langues naturelles*, Masson, 1997.

[WOO 70] W. Woods Transition network grammars for natural language analysis", Communications of the ACM, 3:10, pp. 591-610, 1970.

[WIN 83] T. Winograd *Language as a cognitive process : Syntax*, Addison Wesley, 1983.

[YOU 67] D. Younger "Recognition and Parsing of Context-Free Languages in Time n3", *Information and Control*, 10:2, 1967.