UN OUTIL MULTIDIMENSIONNEL DE L'ANALYSE DU DISCOURS

J. CHAUCHÉ

Laboratoire de Traitement de l'Information

I.U.T. LE HAVRE Place Robert Schuman - 76610 LE HAVRE FRANCE

& C.E.L.T.A. 23, Boulevard Albert 1er - 54000 NANCY FRANCE

RESUME :

Le traitement automatique du discours suppose un traitement algorithmique et informatique. Plusieurs méthodes permettent d'appréhender cet aspect. L'utilisation d'un langage de programmation général (par exemple PL/1) ou plus orienté (par exemple LISP) représente la première approche. A l'opposé, l'utilisation d'un logiciel spécialisé permet d'éviter l'étude algorithmique nécessaire dans le premier cas et de concentrer cette étude sur les aspects réellement spécifiques de ce traitement. Les choix qui ont conduit à la définition du système SYGMART sont exposés ici. L'aspect multidimensionnel est analysé du point de vue conceptuel et permet de situer cette réalisation par rapport aux différents systèmes existants.

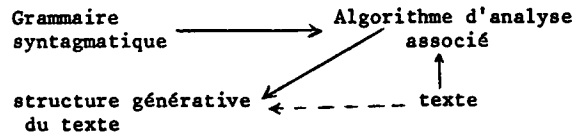
INTRODUCTION :

Un logiciel spécifique de traitement automatique du discours comporte plusieurs éléments : en premier lieu la description des objets manipulés permet de définir l'univers de travail du réalisateur. En second lieu la manière de manipuler ces objets rend compte des potentialités de réalisation d'application diverses. Il est nécessaire au préalable de définir la nature du modèle sous-jacent par rapport aux théories existantes. Dans le présent article on exposera donc successivement une approche du modèle théorique, une description des objets manipulés et enfin, les outils de manipulations. L'exemple du système SYGMART montre une réalisation concrète des choix précédemment exposés.

Le modèle transformationnel.

Du point de vue formel les outils utilisés pour le traitement automatique des langues naturelles peuvent se diviser en deux grandes catégories :

- le modèle génératif définissant un processus formel engendrant un langage. L'analyse consiste alors à retrouver le processus déductif conduisant à la phrase ou au texte étudié. C'est dans ce cadre que sont effectuées la plupart des réalisations actuelles. L'exemple le plus important est sans doute la définition des grammaires syntagmatiques et des analyseurs associés. Nous pouvons schématiser une réalisation par le graphe suivant :



Beaucoup de points s'opposent à cette démarche. Les principales difficultés sont :

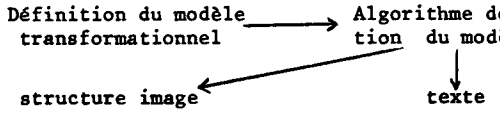
Existe-t-il une grammaire complète des textes à traiter ?

Quel algorithme d'analyse mettre en oeuvre si les restrictions formelles sont trop contraignantes ?

Dans le cas du traitement des langues naturelles, l'algorithme utilisé est-il suffisamment souple pour permettre une adaptabilité constante ?

- Le modèle transformationnel qui définit une fonction d'un espace (textuel) dans un autre espace (relationnel) ou une fonction de l'espace relationnel sur lui-même.

Le schéma est alors le suivant :



Les principales questions sont alors les suivantes :

Analyse : comment définir un accepteur d'un langage donné ?

Preuve que la fonction transformationnelle est partout définie.

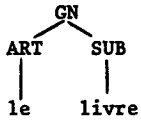
Existe-t-il un algorithme transformationnel acceptable et comment le décrire ?

Des réalisations ont déjà été effectuées suivant cet aspect formel, notamment les systèmes Q, CETA puis ROBRA. Le but du présent article est d'exposer une évolution de cette approche et en particulier l'approche multirelationnelle ou multidimensionnelle.

La séparation relation étiquette ou structure et signification.

Lorsque l'on utilise un modèle pour une application donnée, on projette une signification sur un objet formel. Pour cette raison chaque élément de la structure est affecté d'une étiquette ayant un sens particulier.

Exemple :



Cette approche a l'inconvénient de rassembler deux éléments distincts par leurs natures et leurs significations : la structure et les étiquettes.

structure

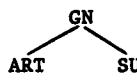
GN ART SUB le livre

étiquettes

Sans cette séparation chaque point possède une seule identité et la structure doit alors répondre à au moins deux objectifs :

- les liaisons ou relations syntaxiques
- les liaisons ou relations qualitatives

Nous aurons dans le premier cas :

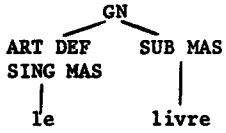


définissant le groupe nominal composé d'un article et d'un substantif

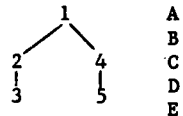
dans le second cas :

définissant l'article comme défini singulier

La plupart des modèles transformationnels ont été définis avec un multi-étiquetage.



Cette approche importante détermine les objets qui seront manipulés de façon abstraite (théorique) ou concrète (programme). Ainsi les systèmes Q par exemple opèrent sur des Q-graphes dont chaque branche est étiquetée par une arborescence simplement étiquetée. Le système CETA opère sur des arborescences multi-étiquetées. Dans ces deux cas l'analyse du discours consiste à rechercher une structure qui représentera alors la compréhension du système pour ce texte. L'exploitation de cette structure définira alors l'application. Une étude approfondie conduit à définir comme objet de base un triplet : structure, multi-étiquette, fonction d'association.



structure

A : GN
B : ART DEF SING MAS
C : le
D : SUB MAS
E : livre

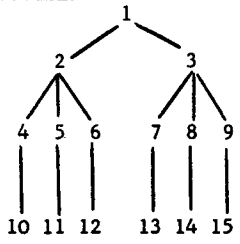
multi-étiquettes

1 → A
2 → B
3 → C
4 → D
5 → E

fonction d'association

La fonction d'association n'est pas nécessairement injective. Cette propriété permet de mieux dissocier structure et contenu :

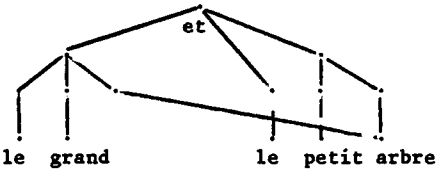
Exemple : Le grand et le petit arbre.



A : COORD
B : GN
C : GN
D : DEF
E : GA
F : SUB
G : le
H : grand
I : DEF
J : GA
K : SUB
L : le
M : petit
N : arbre

1 → A
2 → B
3 → C
4 → D
5 → E
6 → F
7 → I
8 → J
9 → K
10 → G
11 → H
12 → N
13 → L
14 → M
15 → N

L'ellipse du mot 'arbre" n'existe pas dans la structure et existe par la définition de la fonction d'étiquetage. Ce qui correspond schématiquement au graphe suivant :



La définition précédente permet de définir des algorithmes de traitements simples et efficaces alors que pour ce dernier type de graphe les traitements comporteront des algorithmes complexes.

Eléments structurés.

Un élément structuré est par définition un objet multidimensionnel ou multichamp. La structure précédente est issue de l'étude syntaxique des textes. Elle permet de définir une forme élaborée du texte et d'avoir un accès à ses différentes composantes en rapport avec leurs fonctions. Pour le traitement des langues naturelles il est bien sûr évident que cette analyse ne suffit pas. Cela ne signifie pas que tous les problèmes liés à cette analyse soient résolus mais que la levée des obstacles, de l'analyse syntaxique ou autre, suppose une étude plus approfondie. Lorsqu'une réalisation utilise le même espace définitionnel pour représenter le sens et la forme les problèmes évoqués précédemment sur les difficultés liées à la confusion structure-étiquette se multiplient et se transportent au niveau structurel. Comment représenter deux structures d'un texte donné sous forme arborescente si ces deux arborescences sont contradictoires ? Ce problème est insoluble dans le cadre arborescent classique. On peut bien sûr définir plusieurs types d'analyses, obtenir plusieurs arborescences du même texte. Dans ce cas la liaison entre ces différentes arborescences sera très difficile sinon impossible à formaliser et à mettre en oeuvre. Il est donc nécessaire d'avoir un modèle de représentation qui permette de définir plusieurs structures sur le même ensemble de points, chacun de ces points étant associé à une multi-étiquette suivant une fonction quelconque. Cette définition correspond à la définition des éléments structurés dont l'approche formelle est la suivante :

Un élément structuré est défini par un quadruplet (P,S,E,F) où :

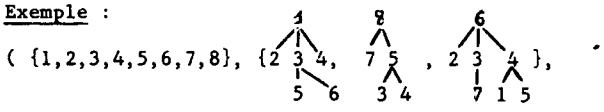
P : est un ensemble fini de points

S : est un ensemble fini de structures arborescentes sur les points de P et tel que chaque point de P appartient à au moins une structure de S.

E : est un ensemble fini de multi-étiquettes.

F : est une application surjective de P sur E.

Exemple :

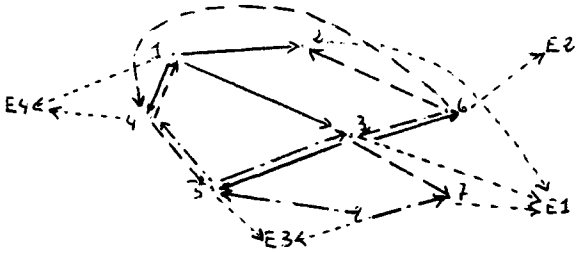


({1,2,3,4,5,6,7,8}, {2 3 4, 7 5, 2 3 4 }, 5 6 3 4 7 1 5

{E1,E2,E3,E4}{1→E4,2→E1,3→E1,4→E4,5→E3,6→E2,

7→E1,8→E3})

la représentation graphique d'un tel objet est plus facile lorsque l'on regarde une seule structure (une seule dimension ou champ). La synthèse graphique de cet exemple donne la figure suivante :



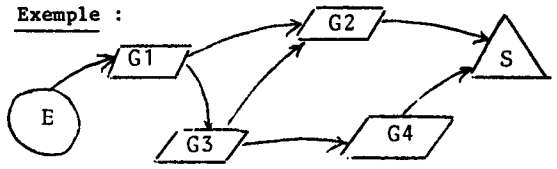
Le problème classique de l'analyse textuelle, (définir une grammaire syntagmatique engendrant un langage), est transformé et devient : définir pour chaque élément du langage un élément structuré associé. Le problème qui se pose alors est similaire à celui obtenu dans le cadre des grammaires syntagmatiques : la définition de l'image structurelle recouvre-t-elle l'ensemble du langage ? On peut remarquer que le cas des grammaires syntagmatiques est un cas particulier de cette approche. L'association est alors la suivante : on affecte à chaque élément du langage engendré par la grammaire la structure syntaxique de cet élément.

Cette approche permet de définir une association plus complexe par la multiplicité des structures associées au même ensemble de points. On aura donc associé à chaque texte ses structures syntaxiques, sémantiques, logiques, etc... En pratique le nombre de champs ou dimensions est limité (par exemple 16 dans le cas du système SYGMART).

Réseau transformationnel :

Un objet formel est intéressant dans la mesure où il existe un moyen de le manipuler. Cet aspect algorithmique est nécessaire à toute réalisation et limite la complexité des objets définis. Le modèle opératoire pour les éléments structurés définis ci-dessus est réalisé par un réseau transformationnel. Chaque point du réseau est constitué d'une grammaire transformationnelle et chaque arc partant d'un point de ce réseau est étiqueté d'une condition basée sur la présence d'un schéma.

Exemple :

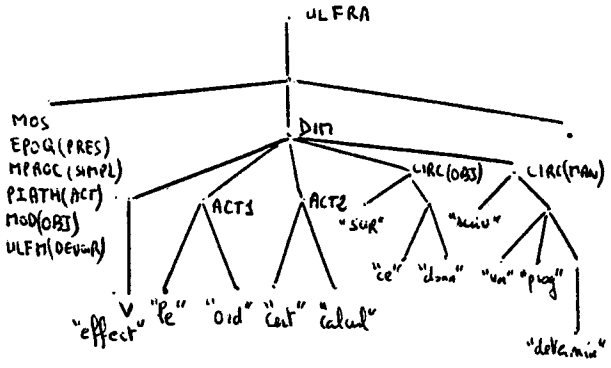


Le résultat de l'application du réseau transformationnel est défini par l'élément structuré obtenu après le parcours de ce réseau d'un point d'entrée E à un point de sortie S. Le réseau définit donc une application de l'ensemble des éléments structurés dans lui-même. Le parcours de ce réseau peut être simple ou récursif suivant la nature des règles appliquées dans les grammaires élémentaires. Une grammaire transformationnelle élémentaire a donc pour but de définir une transformation de l'élément structuré. Cette transformation est réalisée par un ensemble de règles transformationnelles ordonnées. Chaque règle définie un modèle de remplacement permettant une modification d'un élément structuré quelconque. Cette règle pouvant être simple ou récursive et dans ce dernier cas faire appel au réseau pour son exécution. Le point central d'une grammaire élémentaire est donc constitué par une règle élémentaire. Une règle élémentaire est définie par un ensemble de transformations d'arborescences, chacune de ces transformations devant s'appliquer sur un champ simultanément aux autres transformations des autres champs. Des contraintes correspondant à des points communs inter-champs peuvent être définies. On peut remarquer que le système CETA constitue dans ce cadre un cas particulier de traitement sur un seul champ. La transformation dans un champ est une extension des définitions de transformations d'arbre définies par Gladkij et Melcuk [7]. Une grammaire élémentaire possède également un mode d'application permettant de limiter l'applicabilité des règles, ceci afin de définir un processus transformationnel fini. L'ensemble des règles d'une grammaire élémentaire est ordonné et défini un algorithme de Markov [8] étendu aux éléments structurés. La définition d'un modèle de reconnaissance s'effectue suivant un processus analogue à la recherche d'un programme définissant une fonction donnée. Les objets traités sont des objets non classiques en programmation et les modifications de ces objets ne s'effectuent pas à travers un parcours de l'objet traité, mais par la définition de transformations ou modifications de sous-objets.

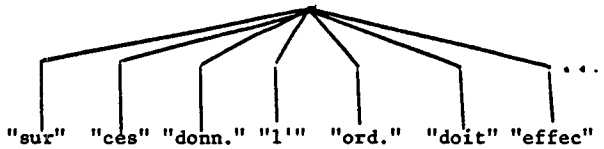
Soit par exemple la définition de l'analyse d'une phrase par Wang Huilin [9] :

phrase : "sur ces données, l'ordinateur doit effectuer certains calculs suivant un programme déterminé."

Structure recherchée :

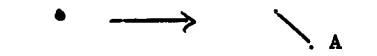


Par convention le texte est projeté suivant la forme d'élément structuré la plus proche du texte:

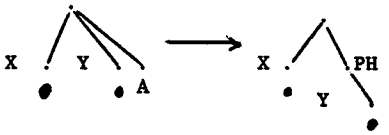


L'écriture du réseau de grammaire va définir un processus de transformations pour obtenir la structure souhaitée. Pour des raisons évidentes nous avons simplifié la représentation dans cet exemple en définissant sur chaque point une partie de l'ensemble des valeurs de l'étiquette associée et en ne considérant qu'un seul champ.
La première grammaire doit permettre une distinction entre phrase au cas où le texte en comporterait plusieurs (bien sûr également dans le cas où l'analyse a été choisie phrase par phrase). Ceci s'effectue en trois étapes :

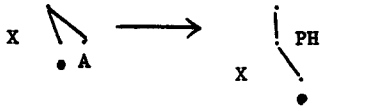
initialisation



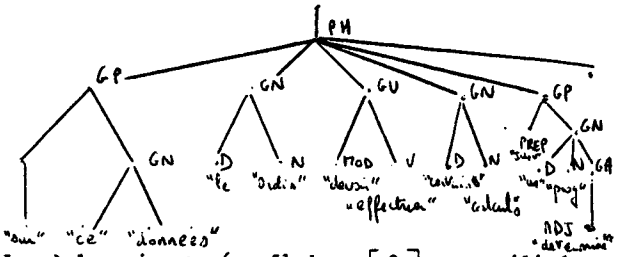
règle générique



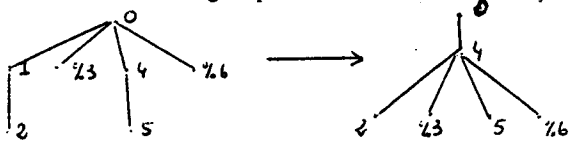
règle finale



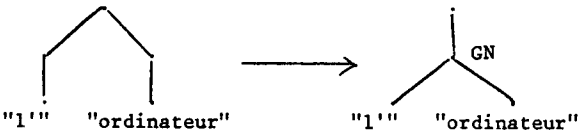
La structure recherchée est déduite de la structure syntaxique qui dans ce cas est la suivante :



La règle suivante (rgnf1 dans [9] est utilisée pour obtenir les regroupements GN :



Cette règle appliquée sur le texte précédent donne par exemple :



Cet exemple utilise deux réseaux de grammaires enchainées, le premier correspondant à la recherche de la structure syntaxique, le second, à la construction de la structure choisie (grammaire F12 et F13 dans [9]).

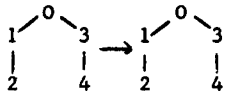
La séparation structure-étiquette induit une propriété importante par rapport à la puissance de définition d'une règle :
La généralité des transformations peut se définir en deux étapes : définition structurelle et définition sémantique. La définition structurelle est très générale et la définition sémantique très spécifique. La règle est alors applicable si la définition sémantique adaptée à la définition structurelle correspond à une réalisation effective dans l'élément structuré traité. Nous avons le schéma fonctionnel suivant :

base de connaissance

définition structurelle ⟶ règle produite

Si par exemple on veut définir la transformation : apprendre quelque chose à quelqu'un → enseigner quelque chose à quelqu'un.
la base de connaissance précisera : apprendre à → enseigner à

et la règle structurelle :



dans ce cas précis il n'y a pas de modification structurelle, la structure est néanmoins nécessaire

Avec la même règle nous pouvons avoir dans la base de connaissance la transformation :
offrir à → donner à

permettant la transformation :

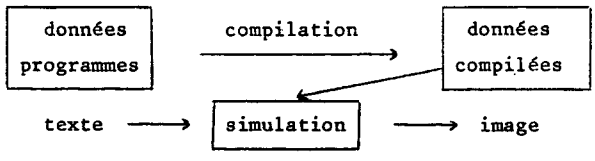
offrir quelque chose à quelqu'un → donner quelque chose à quelqu'un.

Nous avons ainsi avec une seule règle structurelle défini deux règles potentiellement applicables. L'avantage d'une telle définition est évident : factorisation des règles, indépendance de la grammaire par rapport aux lexique, possibilité de définir un comportement spécifique pour chaque élément du lexique sans avoir à définir une grammaire de transformations structurelles trop importante.

Le système SYGMART :

Le système SYGMART est un système opérationnel simulant un modèle transformationnel d'éléments structurés. Il est composé de trois sous-systèmes OPALE, TELESI et AGATE, chacun de ces sous-systèmes correspondant aux différentes fonctions essentielles de traitement d'un texte :

OPALE effectue le passage texte élément structuré.
TELESI effectue la transformation d'éléments structurés.
AGATE effectue le passage d'élément structuré texte.
La forme générale de l'application d'un sous système est la suivante :



Les données programmes comportent deux éléments : un dictionnaire définissant la base de connaissance et une grammaire définissant le processus transformationnel.

Le sous-système OPALE :

Ce sous-système permet de définir un élément structuré à partir d'un texte. Chaque champ comportera la même structure et chaque point de cette structure sera associé à une étiquette correspondant au résultat d'une analyse d'un mot suivant ce sous-système. Cette analyse est basée sur un automate d'états finis permettant une lecture d'un dictionnaire avec segmentation. Au cours de cette segmentation différents renseignements sont évalués et mémorisés dans l'étiquette résultante de l'analyse.

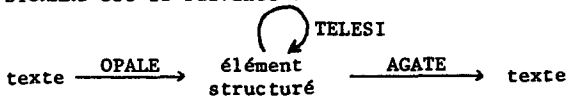
Le sous-système TELESI :

Ce sous-système définit le processus central du système SYGMART. Il permet de définir un réseau transformationnel. Ce réseau est composé de grammaires comportant un ensemble (éventuellement vide) de règles. Chaque grammaire définit une transformation d'éléments structurés et le résultat de cette grammaire définit le parcours du réseau. Chaque grammaire possède un mode d'application, le plus complexe étant le mode récursif qui permet de définir un parcours de l'objet transformé. Le réseau définit lui-même une transformation d'éléments structurés. L'entrée du système est composé soit du résultat du sous-système OPALE soit du résultat de l'application de ce sous-système lui-même. Le dictionnaire associé au sous-système TELESI définit la base de connaissances à associer aux règles de transformations. Cette application du contenu du dictionnaire par rapport aux règles de transformations, s'effectue de manière dynamique.

Le sous-système AGATE :

Ce dernier sous-système définit la transformation élément structuré texte. Cette transformation est nécessaire dans beaucoup d'application et s'effectue par le parcours canonique d'une arborescence d'un champ déterminé. Chaque étiquette associée à un point de ce parcours permet de définir un mot à l'aide d'un automate d'états finis de synthèse, miroir du sous-système OPALE.

La forme générale de l'application du système SYGMART est la suivante :



Du point de vue pratique, le système SYGMART existe en trois versions. Deux versions PL/1 et une version C. Les versions PL/1 sont définies sous les systèmes IBM OS/MVS et Honeywell Multics. La version C est définie sous le système UNIX et fonctionne sous un système à base du microprocesseur MC68000. Une réalisation sur une traduction automatique Espagnol-Français effectuée au CELTA avec le système SYGMART donne un exemple du temps d'exécution nécessaire : la traduction d'un texte de 800 mots traités ensembles (et non phrase par phrase, ce qui implique la manipulation d'arborescences et d'éléments structurés de plus d'un millier de points) a été réalisée sur un Amdahl 470/V7 en 33 mn 38 s (soit 14 10^6 opérations/mots) La version micro-ordinateur nécessite une mémoire d'au moins 756 Ko et un disque dur d'au moins 20 Mo. Les trois exemples suivants sont extraits de trois réalisations distinctes et représentent des parties de grammaires TELESI :

1) extrait de la grammaire d'analyse de l'espagnol C. VIGROUX CELTA France.
2) extrait de la grammaire d'analyse du Chinois WANG HUIN LIN Institut de Linguistique Pekin Chine.
3) extrait de la grammaire d'analyse du Néerlandais P. ROLF Université Catholique de Nimègue Hollande.

-=-=-=-=-=-=-=-=-=-

REFERENCES :

[1] : BOITET C., GUILLAUME P., QUEZEL-AMBRUNAZ M
Manipulation d'arborescences et parallèlisme : système ROBRA, COLING 1978.

[2] : CHAUCHE J.
Transducteurs et arborescences
Thèse, Grenoble 1975.

[3] : CHAUCHE J.
Le Système SYGMART
Document privisoire, Le Havre 1980.

[4] : CHAUCHE J., CHEBOLDAEFF V., JATTEAU M., LESCOEUR R.
Spécification d'un système de traduction assistée par ordinateur.

[5] : COLMERAUER A.
Les systèmes Q, Université de Montréal 1970.

[6] : EUVRARD A, BOURQUIN MC, ATTALI A., LECOMTE J.
Les problèmes liés au passage de la structure de surface vers la structure d'interface.
CELTA Nancy, 1981.

[7] : GLADKIJ A.V., MEL'CUK I.A.
Tree grammars, Linguistics Mouton 1975.

[8] : MENDELSON
Introduction to mathematical logic
VAN NOSTRAND 1964

[9] : WANG H.
La place de la modalité dans un système de traduction automatique trilingue Français-Anglais-Chinois.
Thèse, NANCY 1983

-=-=-=-=-=-=-