*TALN 2004, Fès, 19-21 avril 2004*

# Fusionner pour mieux analyser : Conception et évaluation de la plate-forme de combinaison

Francis Brunet-Manquat

GETA-CLIPS – Université Joseph Fourier (Grenoble 1)
BP 53 – 38041 Grenoble Cedex 9, France
Francis.Brunet-Manquat@imag.fr

## Résumé – Abstract

L'objectif de cet article est de présenter nos travaux concernant la combinaison d'analyseurs syntaxiques pour produire un analyseur plus robuste. Nous avons créé une plate-forme nous permettant de comparer des analyseurs syntaxiques pour une langue donnée en découpant leurs résultats en informations élémentaires, en les normalisant, et en les comparant aux résultats de référence. Cette même plate-forme est utilisée pour combiner plusieurs analyseurs pour produire un analyseur de dépendance plus couvrant et plus robuste. À long terme, il sera possible de "compiler" les connaissances extraites de plusieurs analyseurs dans un analyseur de dépendance autonome.

The goal of this article is to present our works about the combination of syntactic parsers to produce a more robust parser. We have built a platform which allows us to compare syntactic parsers for a given language by splitting their results in elementary pieces, normalizing them, and comparing them with reference results. The same platform is used to combine several parsers to produce a dependency parser, which is big construction broader and more robust than its component parsers. In the future, it should by possible to "compile" the knowledge extracted from several analyzers into an autonomous dependency parser.

## Mots Clés – Keywords

Analyse de dépendance, analyse syntaxique, combinaisons d'informations.

Dependency parsing, syntactic parsing, Information combination.

## 1 Introduction

Notre laboratoire est impliqué dans deux projets internationaux importants : CSTAR et son projet européen associé NESPOLE ! (http://nespole.itc.it) pour la traduction simultanée de l'oral et Universal Networking Language, UNL (http://www.unl.ias.unu.edu), pour la traduction de l'écrit. Ces projets se caractérisent notamment par la présence d'une représentation pivot des énoncés et par le fait que l'énoncé à traduire est susceptible d'être *bruité*, c'est-à-dire pas nécessairement conforme à la grammaire académique de la langue française. Dans un système à pivot, l'énoncé d'une langue source donnée est *analysé* dans la représentation pivot avant d'être *généré* vers une ou plusieurs langues cibles. La nécessité de

pouvoir traiter des entrées bruitées implique des outils robustes d'analyse de la langue, capables de fournir une analyse même partielle de la phrase.

L'objectif est de concevoir un analyseur prenant le « meilleur » d'un ensemble d'analyseurs. Le principe de notre travail est de combiner différents résultats d'analyse obtenus pour le même énoncé, puis de calculer les meilleures informations pour obtenir la ou les meilleures analyses possibles. Notre approche se base sur la méthode dite du « vote à la majorité », *plus une information sera commune aux différents analyseurs, plus le poids de cette information sera fort*, et sur un apprentissage permettant d'adapter les poids associés aux informations fournies en fonction de l'énoncé (par exemple, entrée bruitée ou non) et des analyseurs.

L'approche utilisée, dite par *combinaison*, a déjà connu de nombreux succès en reconnaissance de la parole (Fiscus 1997, Schwenck et Gauvain 2000), en étiquetage morpho-syntaxique (Halteren et al. 1998, Brill et al. 1998, Marquez et Padro 1998), en reconnaissance des entités nommés (Borthwick et al. 1998), en désambiguïsation du sens des mots (Pedersen, 2000) et, plus récemment, en analyse syntaxique (Henderson et Brill 1999, Inui et Inui 2000, Monceaux et Robba 2003). Ces travaux démontrent que combiner différents systèmes apporte le plus souvent une amélioration par rapport au meilleur système.

Nos travaux en analyse syntaxique se différencient de ceux de nos prédécesseurs par la méthode de combinaison employée. En effet, notre plate-forme se compose à la fois d'un mécanisme de re-construction et d'un traitement statistique. De plus, nous basons notre plate-forme d'analyse sur une représentation par dépendances, décrivant les relations syntaxiques entre mots. En effet, l'analyse antérieure suggère fortement que ce type de représentation est plus adapté pour une analyse robuste. Il nous permet, par exemple, de représenter clairement et simplement l'analyse partielle d'une phrase mal formée.

Après avoir détaillé la conception de notre plate-forme d'analyse, nous décrivons notre processus de construction de structures de dépendance. Ensuite, nous présentons les évaluations obtenues suivies par une conclusion et une discussion sur les perspectives.

# 2 Conception de la plate-forme d'analyse

La plate-forme d'analyse ne doit pas intégrer les analyseurs, mais elle doit être capable d'extraire les informations linguistiques des résultats qu'ils produisent, de les interpréter, de les fusionner et enfin de produire un arbre de dépendance (ou plusieurs) combinant le meilleur des informations extraites.

## 2.1 Étapes du traitement

Le processus supporté par la plate-forme comporte deux étapes : la *normalisation* des résultats d'analyse et la *construction* des analyses de dépendance.

L'étape de normalisation des résultats d'analyse se compose de deux phases[1] :

- La phase d'*extraction* permet de récupérer les informations linguistiques des résultats obtenus à partir des analyseurs linguistiques. Ces analyseurs se répartissent en trois catégories en fonction des résultats qu'ils fournissent (Monceaux et Robba 2002) : les analyseurs fondés sur les constituants qui retournent une segmentation en

[1] L'étape de normalisation a été décrite et évaluée dans (Brunet-Manquat 2003).

groupes, les analyseurs fondés sur les dépendances qui retournent les dépendances entre mots d'une phrase et les analyseurs fondés sur les constituants et les dépendances qui retournent une segmentation en groupes et des dépendances entre ces groupes et entre les mots.

- La phase de *projection* traite les informations extraites pour obtenir un ensemble de structures de dépendance dites normalisées. À chaque information contenue dans ces structures est associé un indice exprimant la confiance relative dans cette l'information (catégories, variables grammaticales ou relations syntaxiques) en fonction de l'analyseur l'ayant produite. Ces indices sont précalculés lors d'une phase d'apprentissage (voir 2.3 Apprentissage des indices de confiance). Une structure de dépendance est décrite par une représentation matricielle qui présente de nombreux avantages : maniabilité, efficacité, etc. (voir 2.2 Matrice de dépendances (MD)).

L'étape de construction d'un ensemble d'analyses de dépendance se compose de trois phases :

- La phase de *correspondance* qui permet de lier les nœuds des différentes structures normalisées fournies par l'étape précédente. Pour ce faire nous créons une structure, appelée *réseau de segmentation*, représentant différentes segmentations de la phrase (treillis) et permettant de lier les nœuds des structures normalisées. Ce réseau peut être vu comme un « *pivot de liaison* » entre ces structures (voir 3.1 Correspondance des structures de dépendance).
- La phase de *fusion* des informations linguistiques qui permet, à l'aide des correspondances établies précédemment, d'obtenir une unique représentation de dépendance contenant toutes les informations linguistiques produites, même les informations contradictoires. Toutes ces informations ainsi fusionnées verront leurs indices de confiance recalculés (voir 3.2 Fusion des informations).
- La phase de *production* qui permet de construire les nouvelles structures de dépendance en fonction des informations fusionnées, des indices de confiance recalculés, et de contraintes linguistiques et structurelles de production (voir 3.3 Production des structures de dépendance).

## 2.2 Matrice de dépendances (MD)

Notre plate-forme d'analyse est *fondée sur les dépendances*, c'est-à-dire qu'elle retourne les dépendances entre les mots d'une phrase. Une structure de dépendance est décrite, dans notre plate-forme, par une représentation matricielle. Notre représentation, nommée *matrice de dépendance (MD)*, est un couple *<L, M>* composé de :

- Une *liste de nœuds (L)*, un nœud étant composé d'informations linguistiques relatives aux mots qu'il décrit ;
- Une *matrice carrée (M)* permettant de décrire les dépendances entre nœuds. La case (i, j) contient l'ensemble des dépendances entre le nœud i et le nœud j de la liste de nœuds.

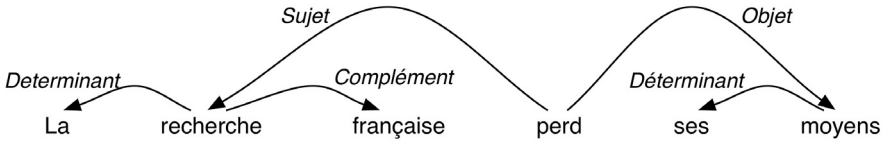



Figure 1 : Structure de dépendance syntaxique

La MD correspondant à la structure de dépendance syntaxique ci-dessus est :

L =

| | | |
|---|---|---|
| la | :: | cat=déterminant |
| recherche | :: | cat=nom |
| française | :: | cat= adjectif |
| perd | :: | cat=verbe |
| ses | :: | cat=déterminant |
| moyens | :: | cat=nom |

M =

relation →

| | la | recherche | française | perd | ses | moyens |
|---|---|---|---|---|---|---|
| la | | | | | | Déterminant |
| recherche | Déterminant | | Complément | | | |
| française | | | | | | |
| perd | | Sujet | | | | Objet |
| ses | | | | | | |
| moyens | | | | | | |

Une représentation matricielle des données présente deux avantages pour le traitement informatique :

- *Maniabilité* : de nombreux outils mathématiques sont associés aux matrices : ajout, suppression, comparaison, etc. Tous ces outils permettent un traitement simple de l'information contenue dans une matrice.
- *Efficacité* : les méthodes utilisant les matrices comme structures de données, telle que la reconnaissance de motifs ou la fusion de matrice, se montrent très efficaces et très simples à mettre en place.

## 2.3 Apprentissage des indices de confiance

Dans (Brunet-Manquat 2003), nous présentons les règles de projection permettant de transformer les informations extraites en un ensemble d'informations normalisées. À chaque nouvelle information normalisée I (catégories, variables grammaticales ou relations syntaxiques) est associé un indice de confiance. Cet indice exprime la confiance relative de l'information I en fonction de l'analyseur l'ayant produite.

Les indices sont calculés à l'aide des évaluations effectuées sur chaque analyseur. Pour chaque analyseur Ai, nous calculons les taux de rappel et de précision de chaque information linguistique I pouvant être produite par Ai :

$$Rappel_{Ai}(I) = Nombre\ d'informations\ I\ correctes\ /\ Nombre\ d'informations\ I\ de\ référence$$

$$Précision_{Ai}(I) = Nombre\ d'informations\ I\ correctes\ /\ Nombre\ d'informations\ I\ proposées$$

Un indice de confiance correspond au calcul de la F-mesure (efficacité globale) qui combine précision et rappel en une mesure unique :

$$Indice_{Ai}(I) = ( Précision_{Ai}(I)\ \beta\ Rappel_{Ai}(I) ) / ( Précision_{Ai}(I) + Rappel_{Ai}(I)) )$$

Dans le cadre de nos travaux sur la construction d'un analyseur robuste et couvrant, ni le rappel ni la précision ne sont à privilégier. La F-mesure permet de représenter une moyenne harmonique entre ces deux mesures.

# 3 Construction des structures de dépendance

À la fin de l'étape de normalisation, un ensemble de structures de dépendance est associé à chaque phrase. La phase suivante consiste à fusionner toutes ces structures pour obtenir une unique représentation de dépendances contenant toutes les informations linguistiques présentes dans ces

structures (catégories, variables grammaticales, relations syntaxiques). Pour réaliser cette fusion, il faut, dans un premier temps, mettre en *correspondance* ces structures.

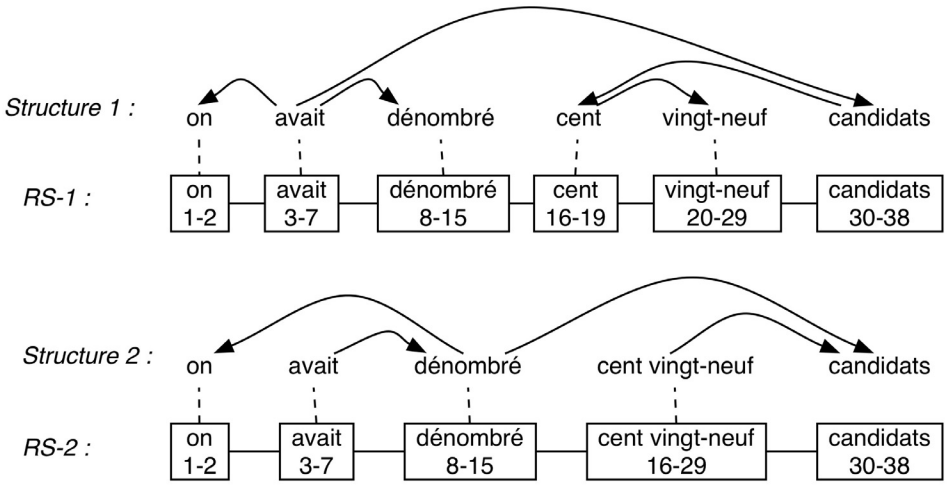
## 3.1 Correspondance des structures de dépendance

La correspondance de structures consiste à regrouper les nœuds représentant la même segmentation dans la phrase (information commune minimale). Mais elle consiste également à représenter les discordances issues des différentes segmentations des structures et dues, par exemple, aux mots composés, aux entrées des dictionnaires (États Unis ou États_Unis), etc.

Pour ce faire, nous créons une structure, appelée *réseau de segmentation (RS)*, représentant les différentes segmentations de la phrase et permettant de lier les nœuds des structures normalisées. Ce réseau peut être vu comme un « *pivot de liaison* » entre ces structures. Ce réseau est un treillis, chaque nœud du réseau représentant une segmentation possible d'un mot et servant de liaison entre les nœuds des structures de dépendance. Concrètement, un nœud $N_{rs}$ d'un RS contient deux informations :

- SNODE($N_{rs}$) : intervalles représentant la sous-chaîne dans la phrase correspondant au nœud $N_{rs}$, Par exemple, les mots de la phrase « On avait dénombré cent vingt-neuf candidats » auront pour intervalles : *On[1-2]*, *avait[3-7]*, *dénombré[8-15]*, etc. Cette information est basée sur la proposition de (Boitet et Zaharin, 1988) *Structured String-Tree Correspondences (SSTC).*
- L : un ensemble contenant les nœuds des structures normalisées liés au nœud $N_{rs}$.

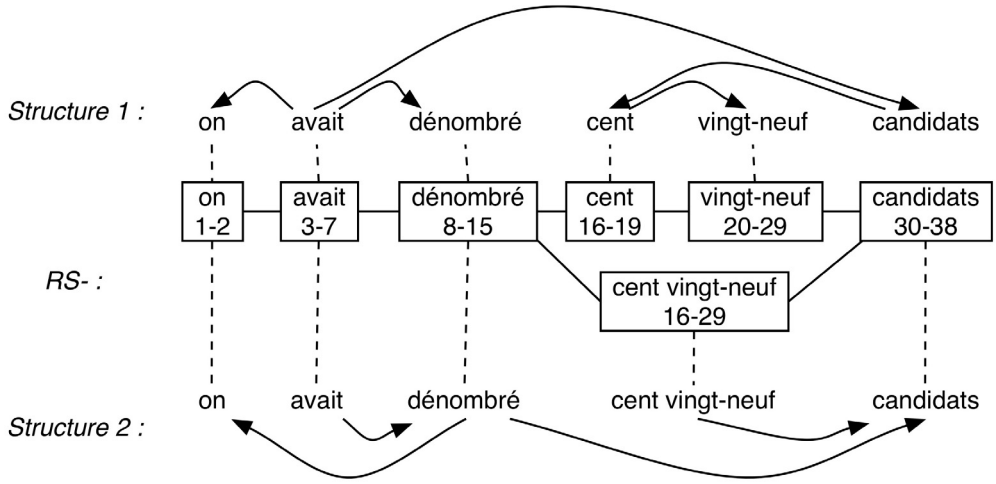
La première étape consiste à créer un réseau de segmentation initial pour chaque arbre de dépendance. Chaque nœud $N_{rs}$ d'un RS initial est créé en fonction d'un nœud $N_i$ de la structure de dépendance $S_k$ : SNODE($N_{rs}$) = SNODE($S_k.N_i$) et L(Nrs) = {$S_k.N_i$}. Les nœuds du RS seront insérés dans le treillis selon l'ordre d'apparition dans la phrase (en fonction du SNODE). Dans la suite du traitement, nous prendrons comme exemple les deux structures de dépendance et leurs RS initiaux suivants :




Faisons la correspondance entre RS-1 et RS-2. Le premier réseau initial RS-1 est désigné comme le réseau de base RS-base, qui servira tout au long du traitement. La suite consiste à introduire les particularités des autres RS initiaux dans le réseau de base. Pour ce faire, nous utilisons deux règles de construction :

- **Règle 1)** *Correspondance* : Si le nœud $N_i$ de RS-k est *équivalent* à l'un des nœuds $N_{rs}$ de RS-base (l'équivalence est vraie si SNODE($N_i$) == SNODE($N_{rs}$)), $N_{rs}$ sera lié aux nœuds de la structure que représente $N_i$ : $L(N_{rs}) = L(N_{rs})\ \beta\ L(N_i)$.
- **Règle 2)** *Insertion* : Si le nœud $N_i$ de RS-k n'est équivalent à aucun nœud de RS-base, le nœud $N_i$ est inséré dans RS-base en fonction de son SNODE (ordre d'apparition dans la phrase).

Les premiers nœuds *on[1-2], avait[3-7], dénombré[8-15]* de RS-2 vérifient la première règle, ils correspondent aux nœuds *on[1-2], avait[3-7], dénombré[8-15]* de RS-base. Le quatrième nœud *cent vingt-neuf[16-19]* de RS-2 vérifie la seconde règle, il est donc inséré dans RS-base. Le dernier nœud de RS-2 vérifie la première règle. Nous obtenons donc le réseau suivant :




Le réseau de segmentation final obtenu représente les segmentations possibles et lie les nœuds des structures entre eux[2]. Maintenant que la correspondance entre les nœuds des structures est établie, nous pouvons fusionner ces structures pour fournir une unique représentation de dépendance combinant toutes les informations linguistiques relatives aux structures.

## 3.2 Fusion des informations

Les correspondances entre les différentes structures étant établies, la phase de fusion des informations linguistiques peut débuter. La méthode utilisée lors de cette phase est basée sur la méthode dite de « vote à la majorité » : *plus une information sera commune aux différents analyseurs, plus son poids augmentera*, dans notre cas plus son indice de confiance augmentera. Chaque indice pourra être vu comme le *vote pondéré* de l'analyseur pour l'information, ce vote étant *adapté* aux différentes capacités de l'analyseur en fonction de l'énoncé (par exemple, entrée bruitée ou non) lors de la phase d'apprentissage.

À la fin de la phase de correspondance, à chaque phrase sont associés un ensemble de structures de dépendance et un réseau de segmentation permettant de les lier. La suite du traitement consiste à créer pour chaque réseau de segmentation une structure de dépendance, nommée *matrice de fusion* (les nœuds du RS serviront de nœuds pour cette représentation), puis de la compléter en fusionnant toutes les informations linguistiques contenues dans les structures de dépendance associées.

---

[2] Nous proposons dans (Brunet-Manquat 2004) de parfaire la phase de correspondance en ajoutant des règles de correspondance permettant de traiter les mots composés, par exemple, en établissant une relation entre le nœud États_Unis et les nœuds États et Unis.

Certaines informations linguistiques seront équivalentes, d'autres contradictoires (par exemple la dépendance SUBJ(x,y) est contradictoire avec la dépendance OBJ(x,y), les catégories morpho-syntaxiques sont mutuellement contradictoires). Il ne s'agit pas simplement de regrouper toutes ces informations, il faut également calculer de nouveaux indices de confiance, *les indices de fusion*, pour chaque information, en fonction des indices de confiance fournis par l'étape de normalisation. Deux calculs sont possibles : le calcul normalisé ou le calcul corrigé.

**Calcul normalisé :** l'indice de fusion de l'information I est égal à la somme des indices de confiance divisé par le nombre d'analyseurs pouvant fournir cette information : (n : nombre d'analyseurs pouvant fournir l'information I)

$$\underset{\text{fusion}}{\text{Indice(I)}} = \frac{\left( \sum_{\substack{\text{i=analyseur fournissant} \\ \text{l'information I}}} \beta(I_i) \right)}{n}$$

Par exemple, calculons l'indice de fusion à associer à l'information OBJ(x,y) (relation OBJ entre les mots x et y), fournie à la fois par l'analyseur A1 et par l'analyseur A2. L'indice de fusion associé à OBJ(x,y) est égal à la somme des deux indices de confiance indice(OBJ::A1)=0,5 et indice(OBJ::A2)=0,7 divisée par le nombre d'analyseurs pouvant fournir ce type d'information (ici trois pour l'exemple), (0,5+0,7+0)/3 = 0,4. Si le troisième analyseur fournit une information de type SUBJ entre les mots x et y, et si l'indice de confiance relatif à cette information est de 0,8, l'indice de fusion associé à SUBJ(x,y) est égal à (0+0+0,8)/3 = 0,26.

**Calcul corrigé :** L'indice de fusion de l'information I est égal à la somme des indices de confiance des informations I moins la somme (multipliée par un coefficient de correction β) des indices de confiance des informations contradictoires à l'information I, le tout divisé par le nombre d'analyseurs pouvant fournir l'information I :

$$\underset{\text{fusion}}{\text{Indice(I)}} = \frac{\left( \sum_{\substack{\text{i=analyseur fournissant} \\ \text{l'information I}}} \beta(I_i) - \beta \sum_{\substack{\text{p=analyseur fournissant une} \\ \text{information contradictoire à I}}} \beta(I_p) \right)}{n}$$

Pour l'exemple précédent, les relations syntaxiques entre les mots x et y sont contradictoires : soit OBJ(x,y), soit SUBJ(x,y). L'indice de fusion associé à OBJ(x,y) est égal à ((0,5+0,7) – (0,4 * 0,8))/3 = 0,29 (en prenant comme coefficient de correction 0,4) et l'indice de fusion associé à SUBJ(x,y) est égal à (0,8 – 0,4 * (0,5+0,7))/3 = 0,1.

Ces calculs favorisent les informations fournies par le plus grand nombre d'analyseurs. Les nouveaux indices de fusion ainsi calculés serviront lors de la phase de production.

## 3.3 Production des structures de dépendance

Cette dernière phase permet de construire les nouvelles structures de dépendance grâce aux informations recueillies précédemment. Ces structures sont produites à partir des indices de fusion associés à ces informations et de contraintes linguistiques et structurelles. Le mécanisme de production est basé sur une méthode de satisfaction de contraintes comportant 3 règles :

Soit une information I de type catégorie morpho-syntaxique ou variable grammaticale d'une matrice de fusion $MD_{fusion}$ :

- Pour chaque nœud N de $MD_{fusion}$, elle est conservée si son indice de fusion est supérieur aux indices de fusion des informations contradictoires.

Pour les dépendances syntaxiques de $MD_{fusion}$ :

- Une seule dépendance syntaxique entre deux nœuds $N_i$et $N_k$ est conservée, à savoir l'information de dépendance ayant l'indice de fusion le plus fort de la case M(i,k) de $MD_{fusion}$ ;
- Un nœud $N_i$ ne dépend que d'un seul nœud $N_k$ : de toutes les informations syntaxiques désignant Ni comme dépendant (colonne M(i) de $MD_{fusion}$), seule l'information ayant l'indice le plus fort est conservée.

En ce qui concerne le traitement des nœuds discordants issus des différentes segmentations, nous choisissons de conserver seulement les nœuds issus du « meilleur » segmenteur parmi nos analyseurs (l'analyseur ayant la segmentation en unité lexicale la plus proche du corpus de référence). Nous introduirons prochainement un mécanisme associant à chaque nœud un indice de confiance sur sa segmentation exactement comme les indices de confiance sur les informations linguistiques et permettant d'introduire une contrainte sur la segmentation lors de notre phase de production.

# 4 Expérimentation et mesures

## 4.1 Corpus et analyseurs

Nous disposons pour cette évaluation de trois analyseurs syntaxiques : l'analyseur IFSP (Incremental Finite-State Parser) (Ait-Mokhtar et Chanod 1997) qui construit les groupes syntagmatiques noyaux des phrases en entrée, puis utilise la structure ainsi construite pour extraire des relations syntaxique entre mots, l'analyseur syntaxique du GREYC (Vergne 1998) qui combine des techniques d'étiquetage grammatical pour construire des segments non-récursifs et un algorithme de calcul de dépendances pour calculer la structure fonctionnelle et l'analyseur XIP (Ait-mokhtar et al. 2002) qui disposent de différents processus linguistiques organisés de façon incrémentale (annotation morphologique, découpage en syntagme, extraction de dépendances) pour obtenir une analyse de dépendance.

Le corpus utilisé est le corpus arboré de l'université Paris VII (Abeillé et Clément 1999). Ce corpus est constitué d'un million de phrases extraites du journal Le Monde. Les phrases sont segmentées en constituants et les mots sont annotés. Une petite partie de ce corpus a été normalisée pour correspondre à un corpus constitué de dépendances. Nous utilisons pour cette expérimentation un corpus de 200 phrases, choisies arbitrairement, constitué de phrases longues et complexes, 30 mots en moyenne par phrase (minimum 9 mots, maximum 73 mots). Par exemple :

« *La cessation de paiement constatée, le tribunal de commerce nomme un administrateur judiciaire, qui doit évaluer les dettes - alors gelées - et proposer soit un plan de continuation, soit la liquidation judiciaire.* »

## 4.2 Calcul des indices de confiance

Nous traitons dans un premier temps les 100 premières phrases pour la phase d'apprentissage (voir table 1). Notre expérimentation se restreint pour le moment à 5 informations linguistiques : 3 catégories morpho-syntaxiques (nom, verbe et adjectif) et 2 dépendances syntaxiques (sujet et complément de tous types).

Les mauvais résultats concernant les dépendances s'expliquent par le nombre moyen de mots par phrases et par leurs complexités. Les indices (F-mesures), ainsi calculés, nous permettent de produire nos résultats de combinaison sur les 100 phrases restantes.

| | Corpus | IFSP | | | Vergne | | | XIP | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Nb | R | P | F | R | P | F | R | P | F |
| Cat(Nom) | 806 | 84,8 | 78,4 | 78,4 | 78,6 | 77,1 | 77,8 | 86,1 | 74,5 | 79,9 |
| Cat(Verb) | 169 | 88,1 | 93,1 | 90,5 | 92,8 | 97,5 | 95,1 | 98,2 | 71,8 | 83,0 |
| Cat(Adj) | 189 | 76,7 | 72,8 | 74,7 | 87,8 | 57,6 | 69,6 | 75,1 | 57,4 | 65,1 |
| Sujet | 146 | 54,7 | 45,9 | 50,0 | 33,5 | 39,9 | 36,4 | 65,7 | 41,0 | 50,5 |
| Comp | 750 | 53,4 | 22,0 | 31,2 | 49,6 | 37,9 | 43,0 | 49,3 | 29,2 | 36,7 |

Table 1 : calcul des indices de confiance

### 4.3 Évaluation des résultats d'analyse

Nous évaluons maintenant nos résultats et également les résultats des autres analyseurs (voir table 2). Les résultats de combinaison sont obtenus en utilisant le calcul normalisé vu précédemment. Toutes les F-mesures concernant notre plate-forme de combinaison sont supérieures aux mesures effectuées sur les autres analyseurs. Ces mesures démontrent que combiner différents analyseurs apportent une amélioration par rapport aux autres analyseurs.

| | Corpus | IFSP | | | Vergne | | | XIP | | | Combinaison | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Nb | R | P | F | R | P | F | R | P | F | R | P | F |
| Cat(Nom) | 684 | 85,6 | 77,2 | 81,2 | 82,6 | 76,7 | 79,5 | 87,1 | 75,6 | 80,9 | 87,5 | 81,8 | 84,6 |
| Cat(Verb) | 181 | 85,6 | 93,9 | 89,5 | 91,7 | 91,2 | 91,4 | 97,7 | 65,0 | 78,1 | 90,6 | 94,2 | 92,3 |
| Cat(Adj) | 174 | 81,6 | 85,0 | 83,2 | 87,9 | 74,2 | 80,5 | 80,4 | 60,3 | 68,9 | 77,0 | 86,4 | 81,4 |
| Sujet | 148 | 58,1 | 45,9 | 51,3 | 33,1 | 35,7 | 34,3 | 70,9 | 39,9 | 51,0 | 67,5 | 43,8 | 53,1 |
| Comp | 671 | 26,4 | 58,7 | 36,4 | 49,0 | 35,8 | 41,4 | 53,0 | 30,6 | 38,8 | 59,1 | 32,7 | 42,1 |

Table 2 : évaluation des analyseurs et des résultats de combinaison

## 5 Bilan et perspectives

Notre plate-forme permet de comparer des analyseurs syntaxiques pour une langue donnée en découpant leurs résultats en informations élémentaires, en les normalisant, et en les comparant aux résultats de référence. Cette même plate-forme combine plusieurs analyseurs pour produire un analyseur de dépendance plus couvrant et plus robuste que ces composants. Les évaluations effectuées précédemment démontrent que notre méthode de combinaison, un mécanisme de re-construction associé à un traitement statistique, apportent une amélioration par rapport aux autres analyseurs.

À court terme, notre plate-forme sera testée sur d'autres langues (une expérimentation est en cours de réalisation sur l'anglais). Nous comptons également combiner d'autres types d'analyseur (sémantique par exemple) aux analyseurs syntaxiques pour produire des structures de dépendance multi-niveaux, contenant plusieurs niveaux linguistiques : sémantique, logique, syntaxique, etc. À plus long terme, nous comptons apprendre de cette combinaison. Par exemple, il sera possible de « compiler » les connaissances extraites de plusieurs analyseurs dans un seul analyseur de dépendance autonome.

## Remerciements

Je tiens à remercier Xerox et Jacques Vergne pour m'avoir permis d'utiliser les analyseurs.

## Références

ABEILLE A. and L. CLEMENT (1999). A tagged reference corpus for French, LINC'99 Proceedings, EACL workshop, Bergen.

AIT-MOKHTAR S. and CHANOD JP. (1997), *Incremental finite-state parsing*, in Applied Natural Language Processing *1997*, April 1997, Washington.

AIT-MOKHTAR S., CHANOD JP. and ROUX C. (2002), *Robustness beyond Shallowness: Incremental Deep Parsing*, in Natural Language Engineering, 8 (2/3), pp 121-144, Cambridge University Press.

BOITET CH. and ZAHARIN Y. (1988), "*Representation trees and string-tree correspondences*", published in COLING-88, pp 59-64.

BRILL E. and WU J. (1998) *Classifier Combinaison for Improved Lexical Disambiguation*. In Proc. of the 17th COLING, pp. 191-195.

BROTHWICK A., STERLING J., AGICHTEIN E. and GRISHMAN R. (1998) *Exploiting diverse knowledge sources via maximum entropy in named entity recognition*. Proceedings of the sixth workshop on very large corpora, pages 152-160, Montreal.

BRUNET-MANQUAT F. (2003), "*Fusionner pour mieux analyser: quelques idées et une première expérience*", Proceedings of RECITAL'03, vol. 1/2, France, 10-14 juin 2003, pp. 429-438.

BRUNET-MANQUAT F. (2004), "*Description et conception d'une plate-forme robuste combinant des analyseurs d'énoncé*", journal on line ISDM, vol. 13, février 2004, 12 pages.

FISCUS J.G. (1997), "*A post-processing system to yield reduced error word rates: Recognizer output voting error reduction (ROVER)*", published in IEEE Workshop on Automatic Speech Recognizer and Understanding, pp 347-354.

HALTEREN H., J. ZAVREL and W. DAELEMANS (1998). *Improving data driven wordclass tagging by system combination*. In Proc. of the 17th COLING.

HENDERSON, J. C. and BRIL E. (1999). *Exploiting Diversity in Natural Language Processing: Combining Parsers*. In Proc. of the 1999 SIGDAT Conference on EMNLP and VLC, pp. 187-194.

ILLOUZ G. (1999), "*Méta-étiqueteur adaptatif: vers une utilisation pragmatique des resources linguistiques*", published in TALN'99.

INUI T. and INUI K. (2000), *Committee-based Decision Making in Probabilistic Partiel Parsing*, In Proc. of COLING-2000.

MARQUEZ and PADRO (1998). *On the evaluation and comparaison of taggers : the effect of noise in test corpora*. Actes COLING/ACL'98, Montreal, Canada.

MONCEAUX L. and ISABELLE ROBBA I. (2002), "*Les analyseurs syntaxiques : atouts pour une analyse des questions dans un système de question-réponse ?* ", Actes de *TALN'2003*, pp.195-204.

PEDERSEN T. (2000), *A Simple Approach to Building Ensembles of Naive Bayesian Classifiers for Word Sense Disambigusation* In Proc. of the NAACL, pp. 63-69, 2000.

SCHWENK H. and GAUVAIN J.L. (2000), "*Combining multiple speech recognizers using voting and language model information*", published in IEEE International Conference on Speech and Language Processing (ICSLP), pp. II:915-918.

VERGNE J. and GIGUET E. (1998), *Regards théorique sur le « Tagging »,* Actes de TALN'1998, pp 24-33.