Architecture logicielle de Context plate-forme d'ingénierie linguistique

Gustavo Crispino*, Slim Ben Hazez**, Jean-Luc Minel**

* Université de la République
J. Herrera y Reissig
565 11300 Montevideo - Uruguay
crispino@fing.edu.uy

** CAMS/LaLic, UMR du CNRS,
96 Boulevard Raspail
75 006 Paris - France
{benhazez,minel}@msh-paris.fr

Résumé

Nous présentons dans cet article l'architecture logicielle de Context, plate-forme d'ingénierie linguistique dédiée au filtrage sémantique. Nous avons défini un modèle conceptuel et un langage de description et de traitement des connaissances linguistiques. Ces connaissances sont gérées par un système dédié et indépendant des applications qui les utilisent. Les traitements sont spécifiés sous forme déclarative dans un langage formel que nous présentons.

1. Introduction

On assiste depuis plusieurs années à un renouveau des systèmes de résumé et de filtrage automatique des textes. Alors que les premières réalisations étaient fondées essentiellement sur l'utilisation de techniques statistiques, les recherches actuelles se caractérisent par une importance plus grande donnée aux connaissances linguistiques. Ainsi Marcu (1997), dans le cadre de la Rhetorical Structure Theory (Mann et alii 1988) s'appuie sur l'analyse des connecteurs pour construire des arbres rhétoriques qui hiérarchisent l'importance des parties textuelles, Berri (1996) attribue des étiquettes sémantiques aux phrases, Masson (1998) reconnaît partiellement des structures thématiques et Ellouze (1998) exploite différents types d'objets textuels pour produire des schémas de résumés. Les évaluations réalisées sur certains de ces systèmes (Minel et alii 1997, Jing et alii 1998) ainsi que les travaux menés en collaboration avec des résumeurs professionnels (Endres-Niggemeyer 1993) ont néanmoins montré la difficulté à réaliser des résumés standards, c'est à dire construits indépendamment de la formulation des besoins d'un utilisateur. C'est une des raisons pour laquelle nous nous sommes orientés vers la réalisation d'un système de filtrage de textes avec des critères sémantiques et selon un point de vue adapté. Ce système basé uniquement sur des connaissances linguistiques qui identifient certaines marques discursives, construit des extraits ciblés en réponse à certains profils prédéfinis. Nous pensons que ce type de système peut se généraliser à la reconnaissance de différents types d'actes de langage exprimés par des marques linguistiques et discursives explicites. Nous allons présenter dans cet article un modèle conceptuel de représentation des connaissances linguistiques orienté vers le filtrage sémantique ainsi qu'une plate-forme logicielle qui permet d'appliquer ces connaissances. En effet, les systèmes de résumé et de filtrage automatiques précédemment réalisés (Le Roux 1994, Berri 1996) présentaient d'une part, l'inconvénient de ne pas séparer les connaissances linguistiques du système informatique qui les mettait en œuvre et d'autre part d'être liés à des outils informatiques ce qui limitait leur portabilité.

2. Principes et objectifs de Context

Nous nous appuyons sur les acquis obtenus par la méthode d'exploration contextuelle (Desclés 1991, 1996, 1997) ; cette méthode identifie les connaissances linguistiques en les restituant dans leurs contextes et en les organisant en tâches spécialisées. Elle présente

l'avantage d'une part, de rendre le travail linguistique relativement indépendant de son implémentation informatique et d'autre part, d'articuler effectivement dans une même architecture logicielle les deux types de travaux. Dans cette approche, les linguistes analysent les textes en identifiant et en capitalisant tous les indices grammaticaux et lexicaux pertinents pour la résolution d'un problème, puis ils conçoivent et écrivent les règles d'exploration du contexte de ces indices identifiés dans un texte.

Nous avons défini un modèle conceptuel et un langage de description de ces connaissances que nous décrivons dans la section 3.1.1. Ces connaissances sont gérées par un système indépendamment des applications qui les utilisent, cela en vue d'assurer une pérennité, une capitalisation et une réutilisabilité de ces connaissances. Les règles d'exploration du contexte sont écrites dans un langage formel que nous présentons dans la section 3.1.2 .

Ce projet est le fruit d'une collaboration entre l'Université de la République (Uruguay) et l'équipe LaLIC(UMR 8557 du CNRS, EHESS, Université Paris-Sorbonne) et a reçu le soutien du programme ECOS-Uruguay (Action n°U97E01).

3. Architecture générale de Context

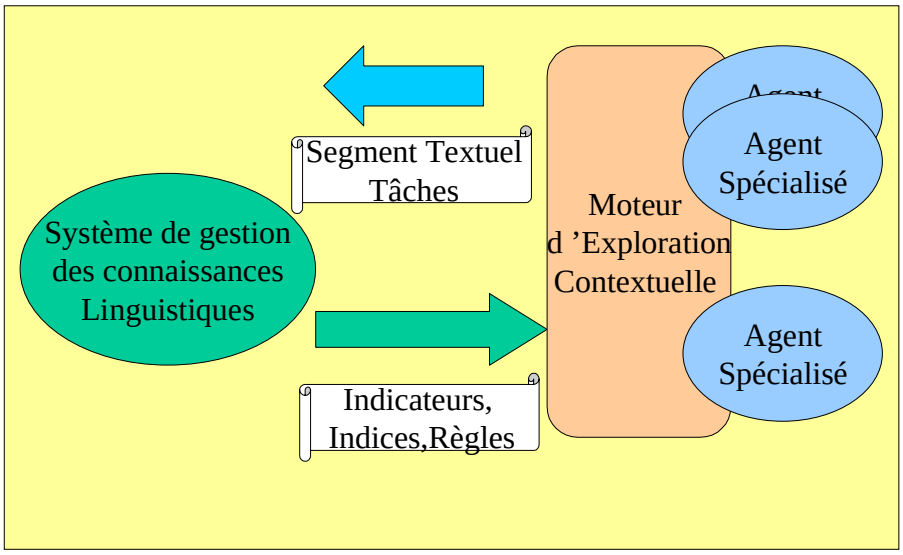

Figure 1 : Architecture générale

Nous avons développé une plate-forme logicielle capable d'accueillir ces connaissances et de les exploiter, relativement indépendamment de la plate-forme matérielle puisque nous avons choisi de développer l'ensemble en utilisant langage JAVA. L'architecture générale (fig. 1) de CONTEXT est la suivante : en réponse à des appels d'agents spécialisés, le moteur d'exploration contextuelle déclenche, pour une ou plusieurs tâches spécialisées, le processus de reconnaissance des indicateurs et des indices présents dans un segment textuel. Ce processus est réalisé par le système de gestion des connaissances linguistiques qui, en retour, fournit au moteur d'exploration contextuelle les règles potentiellement déclenchables.

3.1. Le système de gestion des connaissances linguistiques

La méthode d'exploration contextuelle postule qu'il est possible de repérer certaines informations sémantiques à partir de marques de surface en réponse à des besoins spécifiques d'utilisateurs cherchant à sélectionner des informations importantes, comme par exemple :

• repérer les actions dans des textes techniques (Garcia 1998) ;

• repérer les relations causales entre des situations (Jackiewicz 1998) ;
• identifier les définitions des termes proposés explicitement ou implicitement par un auteur ainsi que les annonces thématiques(Cartier 1998).

Le travail préalable du linguiste consiste alors à étudier systématiquement un corpus de textes pour y rechercher des régularités lexicales et discursives dont l'emploi est représentatif de la catégorie sémantique considérée. D'après l'hypothèse de travail qui jusqu'ici s'est révélée féconde, les modes d'expressions associés à ces catégories discursives dans les corpus, sont en nombre fini. Par conséquent, cela n'exige ni le repérage de structures syntaxiques spécifiques, comme peuvent le faire (Rebeyrolle et alii 1998) qui s'appuient sur les travaux de Z. Harris, ni la construction d'ontologies du domaine en vue d'en énumérer les concepts ou plus modestement les éléments thématiques. Pour appliquer la méthode d'exploration contextuelle, le linguiste doit accumuler les unités textuelles qui sont les expressions linguistiques – les marques des catégories discursives explorées - ; ces expressions peuvent être de simples unités lexicales comme le verbe *présenter* ou des unités composées comme *les lignes suivantes*. Ces marqueurs linguistiques lorsqu'on les identifie dans un texte sont des indices (au sens de C.S. Peirce) de ces catégories. Certains indices sont plus importants que d'autres, ce sont les indicateurs. Le linguiste doit ensuite spécifier les dépendances contextuelles entre ces indices.

3.1.1 Le langage de description des connaissances linguistiques

Nous avons défini un langage de description qui permet au linguiste de constituer sa base de connaissance en spécifiant les tâches, les indicateurs, les indices et les règles d'exploration contextuelles associées (voir Desclés 1997 pour une explication détaillée de ces concepts). Plutôt que de présenter ce langage sous la forme d'une grammaire formelle nous allons illustrer son utilisation.

- Table 1 : Déclaration des formes :

Forme	Nom de Classe
maintenant	&partiedoc1
chapitre	&partiedoc2
lignes	&partiedoc2
&être + &importance	&soulignement

- Table 2 : Déclaration des indices et indicateurs :

Nom de Classe	Type	Catégorie	Fiabilité	Nom de Tâche
&partiedoc1	indice	adverbe		résumé
&verbe-présentatif	indicateur			résumé
&partiedoc2	indicateur	nom		résumé

- Table 3 : Déclaration du ou des noms de règles :

Nom de la Règle	Etiquette attribuée	Segment Textuel	Nom de Tâche	Nom de Classe
RCenthe1001	Thematique_2	phrase	résumé	&partiedoc2
RCenthe112	Thematique_2	phrase	résumé	&verbe-présentatif

Dans la table 1, le linguiste déclare soit des formes significatives (des indices et des indicateurs) qu'il organise en classes non nécessairement disjointes, soit des combinaisons de classes (dernière ligne du tableau). Ces combinaisons permettent de déclarer des lexies ; ainsi la déclaration de *&être + &importance* permet au linguiste de déclarer des lexies du type *il est primordial, il est particulièrement important, il est, essentiel, etc.* Des outils d'aide à l'acquisition permettant de produire automatiquement les formes fléchies ou dérivées sont en cours de développement. Néanmoins il convient de remarquer que le travail du linguiste consiste justement à ne retenir que certaines formes fléchies, car pour une tâche donnée seules certaines flexions d'un verbe sont significatives. Dans la table 2, ces formes sont discriminées

en indices ou en indicateurs et associées à une ou plusieurs tâches. Une tâche a pour finalité de regrouper des règles d'exploration contextuelle comme le montre la table 3 et correspond généralement à un processus d'étiquetage sémantique d'un segment textuel précisé. Mais d'une manière plus générale, une règle peut déclencher différents type de décision(voir section 3.1.2). Des outils d'aide à la gestion de la cohérence et à l'intégration des connaissances issues d'autres travaux linguistiques permettent de répondre à l'objectif d'une acquisition incrémentale et capitalisable des connaissances.

3.1.2 Le langage des règles d'exploration contextuelle

Les règles d'exploration contextuelle sont exprimées dans un langage formel de type déclaratif. Chaque règle comprend une partie *Déclaration d'un Espace de Recherche E,* une partie *Condition* et une partie *Action* qui n'est exécutée que si la partie *Condition* est vérifiée.

- La partie *Déclaration d'un Espace de Recherche E* permet de construire un segment textuel, l'espace de recherche, en appliquant différentes opérations sur la structure du texte construite par le moteur d'exploration contextuelle (voir section 3.2.1). Il est possible de construire plusieurs espaces de recherche dans une même déclaration. Une dizaine d'opérations ont été définies pour construire un espace de recherche à partir de la structure d'un texte.
- La partie *Condition* explicite les conditions que doivent vérifier les indicateurs et les indices. Le langage actuel permet d'exprimer différentes conditions, comme l'existence, la position et l'agencement des indices. D'autres conditions permettent d'exprimer des contraintes sur les attributs des unités lexicales ou sur les morphèmes qui les composent.
- La partie *Action* indique le type d'actions réalisées par la règle. Actuellement les actions possibles sont : attribuer une étiquette à un segment textuel ou déclencher une tâche.

Tâche déclenchante : thématique ;
Commentaire : capte un schéma du type : *il semble .. crucial*
Classe de l'Indicateur : &modal3;
E1 := Créer_espace(PhraseParent_de Indicateur);
L1:= &verbe-etat3 ;
L2 := &adjectif-necessité
Condition: Il_existe_un_indice y appartenant_à E1 tel_que classe_de y appartient_a (L1) ;
Condition : Il_existe_un_indice z appartenant_à E1 tel_que classe_de z appartient_a (L2) ;
Actions :
1 : Attribuer(PhraseParent_de I, « Soulignement_Auteur »)

Figure 2 : Un exemple de règle écrite dans le langage formel.

3.2. Le moteur d'exploration contextuelle

Le moteur d'exploration contextuelle exploite les connaissances linguistiques pour une ou plusieurs tâches choisies par l'utilisateur. Il est composé de deux systèmes qui coopèrent.

3.2.1 L'analyseur de texte

L'analyseur de textes a pour objet de construire une première représentation qui reflète l'organisation structurelle du texte. Il s'appuie pour cela sur le texte balisé par un segmenteur développé par G. Mourad (Mourad 1999). Ce segmenteur applique des règles heuristiques pour reconnaître les sections, avec leurs titres, les paragraphes, les phrases et les citations. L'analyseur peut ainsi construire une structure hiérarchique qui est ensuite enrichie par les

agents spécialisés par des structures qui modélisent les chaînes de liage, segments textuels, cadres de discours (Charolles 1998, Adam 1990), etc., en vue d'améliorer la cohésion et la cohérence textuelle des extraits.

3.2.2 L'exécuteur

L'exécuteur déclenche pour toutes les tâches choisies par l'utilisateur les règles associées à celles-ci. Les règles sont considérées comme indépendantes par conséquent l'ordre de leur déclenchement, pour une tâche donnée, est indifférent. Nous pensons d'une part que ce principe est plus facilement maîtrisable lors de l'écriture des règles par un non-spécialiste, en évitant notamment les problèmes de manipulation d'arbres de décision, et d'autre part il correspond mieux à l'hypothèse que certaines marques sémantiques ne sont pas exclusives entre elles. En d'autres termes, la présence d'une négation dans une phrase conclusive n'implique pas que cette phrase n'est pas par ailleurs une information conclusive. L'exploitation de ces informations éventuellement conflictuelles est de la responsabilité des agents spécialisés. Toutes les déductions effectuées par les règles sont attribuées aux éléments qui composent la hiérarchie du texte et produisent ainsi une structure hiérarchisée « décorée »par des informations sémantiques. Enfin, il convient de noter que le langage formel de déclaration des règles ne présume en rien des outils informatiques utilisés pour les implémenter.

3.2.3 Les agents spécialisés

Les agents spécialisés ont pour tâche d'exploiter les « décorations sémantiques» du texte en fonction des besoins définis par l'utilisateur. Il existe ainsi un agent résumeur qui construit un résumé composé de phrases qui correspondent à un profil[1] type et un agent filtreur qui construit différents extraits de textes en fonction de profils choisis par l'utilisateur. Ces deux agents exploitent les connaissances des systèmes SERAPHIN (Le Roux et alii 1994) et SAFIR (Berri et alii1996) et proposent des interfaces qui permettent de naviguer entre le résumé ou les extraits, et le texte source. Behnami (1999) développe un agent orienté vers le repérage et exploite les interactions qu'entretiennent les figures, les tableaux et les images avec les descriptions textuelles du texte. Les agents spécialisés permettent ainsi de développer des traitements spécifiques pour un utilisateur tout en exploitant le modèle générique de traitement des connaissances linguistiques.

4. Conclusion

La plate-forme CONTEXT est actuellement opérationnelle et les connaissances linguistiques (Garcia 1998, Jackiewicz 1998, Cartier 1996, Jouis 1993) issues de systèmes antérieurs sont intégrées progressivement[2]. Nous pensons que son architecture qui privilégie le concept de composants logiciels et d'agents spécialisés la rend apte à accueillir différents types de traitement linguistique puisqu'il est possible de définir de nouvelles bases de connaissances pour de nouvelles tâches d'étiquetage sémantique. Elle vise ainsi à faciliter les étapes d'acquisition et de modélisation des connaissances linguistiques en proposant des formats, et des langages de représentation des données et des outils, de consultation, de manipulation, de recherche et d'analyse, etc. D'autre part, nous travaillons actuellement à l'intégration d'autres agents spécialisés ainsi qu'a la possibilité d'exploiter des textes annotés pour pouvoir exploiter les informations de type morpho-syntaxiques. Enfin le développement d'une palette de navigation textuelle qui permettra à l'utilisateur de naviguer entre les

[1] La notion de profil est présentée en détail dans (Berri 1996).
[2] La base de connaissances contient 11 000 marqueurs et 250 règles d'exploration contextuelle.

représentations (extraits, graphes) fournies par les agents spécialisés et le texte source est en cours de développement[3].

Remerciements : Nous remercions Jean-Pierre Desclés qui, en participant à de nombreuses séances de travail, nous a permis de mener à bien ce projet.

Références

Adam, Jean-Michel. (1990). *Éléments de linguistique textuelle*, Mardaga, Liège.

Behnami, Shannaz. 1999. Analyse des légendes de figures dans un processus de filtrage des informations. Rapport interne soumis à publication.

Berri, Jawad. (1996) Contribution à la méthode d'exploration contextuelle. Applications au résumé automatique et aux représentations temporelles. Réalisation informatique du système SERAPHIN. Thèse de doctorat, Université Paris-Sorbonne, Paris.

Berri, Jawad, Emmanuel Cartier, Jean-Pierre Desclès, Agata Jackiewicz, Jean-Luc Minel. (1996). SAFIR, système automatique de filtrage de textes, *Actes du colloque TALN'96*, Marseille.

Cartier, Emmanuel. (1998). Analyse automatique des textes : l'exemple des informations définitoires. *RIFRA'98, Rencontre Internationale sur l'Extraction le Filtrage et le Résumé Automatiques*. Sfax, Tunisie.

Charolles, Michel. (1988). Les plans d'organisation textuelle ; période, chaînes, portées et séquences , *Pratiques*, n° 57, pp 3-13.

Desclés, Jean-Pierre, Christophe Jouis, Hum-Ghum Oh, Danièle Maire Reppert. (1991). Exploration Contextuelle et sémantique : un système expert qui trouve les valeurs sémantiques des temps de l'indicatif dans un texte. In *Knowledge modeling and expertise transfer*, pp.371-400, D. Herin-Aime, R. Dieng, J-P. Regourd, J.P. Angoujard (éds), Amsterdam.

Desclés, Jean-Pierre, Emmanuel Cartier, Agata Jackiewicz, Jean-Luc Minel. (1997). Textual Processing and Contextual Exploration Method. In *CONTEXT'97*, Rio de Janeiro, Brésil.

Desclés, Jean-Pierre. (1996). Systèmes d'exploration contextuelle. Actes du colloque sur le Calcul du sens et contexte. Université de Caen.

Ellouze, Mariem, Abdelmajid Ben Hamadou. (1998). Utilisation de chémas de résumés en vue d'améliorer la qualité des extraits et des résumés automatiques. *RIFRA'98, Rencontre Internationale sur l'Extraction le Filtrage et le Résumé Automatiques*. Sfax, Tunisie.

Endres-Niggemeyer, Brigitte. (1993). An empirical process model of abstracting. In *Workshop on Summarizing Text for Intelligent Communication*, Dagsthul, Germany.

Garcia, Daniela. (1998). Analyse automatique des textes pour l'organisation causale des actions. Réalisation du système informatique COATIS. Thèse de Doctorat, Université Paris-Sorbonne.

Jackiewicz, Agata. (1998). L'expression de la causalité dans les textes. Contribution au filtrage sémantique par une méthode informatique d'exploration contextuelle. Thèse de Doctorat, Université Paris-Sorbonne.

Jing, Hongyan, Regina Barzilay et Kathleen McKeown. (1998). Summarization evaluation methods : Experiments and analysis. In *Symposium on Intelligent Text Summarization*, Stanford, CA.

Jouis, Christophe. (1993). Contribution à la conceptualisation et à la modélisation des connaissances à partir d'une analyse linguistique de textes. Thèse de doctorat, EHESS, Paris.

Leroux, Dominique, Jean-Luc Minel, Jawad Berri. (1994). SERAPHIN project. First European Conference of Cognitive Science in Industry. Luxembourg.

Mann, William C, Sandra A. Thompson. (1988). Rhetorical Structure Theory : Toward a functional theory of text organization. *Text*, 8(3):243-281.

Marcu, Daniel. (1997). From discourse structures to text summaries. In *Workshop Intelligent Scalable Text Summarization*, Madrid, Spain.

Masson, Nicolas. (1998). Méthodes pour une génération variable de résumé Automatique : Vers un système de réduction de textes. Thèse de Doctorat, Université Paris-11.

Minel , Jean-Luc, Sylvaine Nugier, Gérald Piat. (1997). How to appreciate the Quality of Automatic Text Summarization. *Workshop Intelligent Scalable Text Summarization*, Madrid, Spain.

Mourad, Ghassam. (1999). La segmentation des textes par l'étude de la ponctuation. 2° Colloque International sur le Document Électronique, CIDE'99, Damas, Syrie.

Pazienza, M.T. (1997) (éd.). Information extraction (a multidisciplinary approach to an emerging information technology*), International Summer School, SCIE'97*, Springer Verlag (Lectures Notes in Computer Science).

Rebeyrolle, Josette et Marie–Paule Pery-Woodley. (1998). Repérage d'objets textuels fonctionnels pour le filtrage d'information : le cas de la défintion. *RIFRA'98, Rencontre Internationale sur l'Extraction le Filtrage et le Résumé Automatiques*. Sfax, Tunisie.

[3] Par J. Couto sous la direction de G. Crispino.