OGMIOS : une plate-forme d'annotation linguistique de collection de documents issus du Web

Thierry HAMON, Julien DERIVIÈRE, Adeline NAZARENKO
LIPN – UMR CNRS 7030
99 av. J.B. Clément, F-93430 Villetaneuse, FRANCE
{Thierry.Hamon,Julien.Derivière,Adeline.Nazarenko}
@lipn.univ-paris13.fr

Résumé. L'un des objectifs du projet ALVIS est d'intégrer des informations linguistiques dans des moteurs de recherche spécialisés. Dans ce contexte, nous avons conçu une plate-forme d'enrichissement linguistique de documents issus du Web, OGMIOS, exploitant des outils de TAL existants. Les documents peuvent être en français ou en anglais. Cette architecture est distribuée, afin de répondre aux contraintes liées aux traitements de gros volumes de textes, et adaptable, pour permettre l'analyse de sous-langages. La plate-forme est développée en Perl et disponible sous forme de modules CPAN. C'est une structure modulaire dans lequel il est possible d'intégrer de nouvelles ressources ou de nouveaux outils de TAL. On peut ainsi définir des configuration différentes pour différents domaines et types de collections. Cette plateforme robuste permet d'analyser en masse des données issus du web qui sont par essence très hétérogènes. Nous avons évalué les performances de la plateforme sur plusieurs collections de documents. En distribuant les traitements sur vingt machines, une collection de 55 329 documents du domaine de la biologie (106 millions de mots) a été annotée en 35 heures tandis qu'une collection de 48 422 dépêches relatives aux moteurs de recherche (14 millions de mots) a été annotée en 3 heures et 15 minutes.

Abstract. In the context of the ALVIS project, which aims at integrating linguistic information in topic-specific search engines, we developed an NLP architecture, OGMIOS, to linguistically annotate large collections of web documents with existing NLP tools. Documents can be written in French or English. The distributed architecture allows us to take into account constraints related to the scalability problem of Natural Language Processing and the domain specific tuning of the linguistic analysis. The platform is developed in Perl and is available as CPAN modules. It is a modularized framework where new resources or NLP tools can be integrated. Then, various configurations are easy to define for various domains and collections. This platform is robust to massively analyse web document collections which are heterogeneous in essence. We carried out experiments on two different collections of web documents on 20 computers. A 55,329 web documents collection dealing with biology (106 millions of words) has been annotated in 35 hours, whereas a 48,422 search engine news collection (14 millions of word) has been annotated in 3 hours and 15 minutes.

Mots-clés : plateforme d'annotation linguistique, passage à l'échelle, robustesse.

Keywords: linguistic annotation, NLP platform, process scability, robustess.

1 Introduction

Si les moteurs de recherche actuels sont suffisants pour répondre aux requêtes les plus courantes sur Internet, il n'existe pas actuellement d'outils permettant la formulation de requêtes s'appuyant sur des techniques de recherche avancées (filtrage sur le sens, élimination d'ambiguïtés, exclusion des sites marchands, etc.) et spécialisées exploitant des connaissances du domaine. Par exemple, la plupart des publications dans le domaine de la biologie et de la bio-médecine sont enregistrées dans de grandes bases de données textuelles, plus ou moins spécialisées (Flybase pour l'espèce Drosophilia Menogaster, Medline pour la biologie et la médecine). Ce type de bases documentaire est aujourd'hui essentiel au travail des scientifiques mais ceux-ci sont confrontés à la masse de textes, sans pouvoir y faire face. Les outils disponibles sont trop généraux, ils renvoient des centaines ou des milliers d'articles pour la moindre requête. Pour juger de la pertinence d'un document dans ce contexte, il faut en analyser le contenu (reconnaissance des entités, reconnaissance des termes techniques).

Le projet ALVIS[1] vise à développer un moteur de recherche *open source* incluant des techniques de recherche avancées et d'analyse du contenu textuel, notamment du point de vue sémantique. Par rapport aux moteurs de recherche actuels, ALVIS cherche à prendre en compte à la fois le thème et le contexte de la recherche pour affiner l'analyse de la requête et du document. Le projet s'appuie sur une architecture *peer-to-peer*. Le système est constitué d'un réseau de « nœuds » assurant l'infrastructure de recherche globale, auxquels sont adjoints des nœuds spécialisés pour un domaine donné. Les nœuds spécialisés proposent une véritable analyse du contenu textuel pour améliorer l'accès au document. A terme, des tâches d'extraction d'informations structurées et leur fusion avec des informations déjà enregistrées au sein de bases de données devraient pouvoir être prises en charge par ce type de moteur spécialisé.

L'accès au contenu sémantique des documents issus du web ou de grandes bases documentaires nécessite une première phase d'enrichissement linguistique des documents en un temps suffisamment court. Il s'agit ici de réduire le goulet d'étranglement que constituent généralement les outils de TAL lorsqu'ils sont intégrés dans des applications de recherche d'information. L'architecture logicielle que nous avons développé permet de satisfaire cette contrainte. Cette plate-forme, OGMIOS, est à la fois générique et spécialisable. Elle est conçue pour analyser de manière robuste des collections de taille variées et hétérogènes du point de vue de la langue (pour l'instant le français et l'anglais[2]), de la longueur et du type de leurs documents . Elle peut aussi être spécialisée pour un domaine particulier. Dans le cadre du projet ALVIS, les expériences ont porté en priorité sur le domaine de la biologie, mais nous avons également pu tester la plate-forme sur un corpus de dépêches relatives aux moteurs de recherche.

Cet article présente notre approche permettant de répondre aux contraintes de performances, de généricité et d'adaptabilité à un domaine de spécialité, qu'impose l'utilisation du TAL dans une application de recherche d'information (RI) spécialisée. Dans la section 2, nous donnons un aperçu de l'état de l'art des plates-formes d'annotation de documents. La plate-forme est décrite dans la section 3 avec les modules de traitement qu'elle intègre. L'évaluation des performances de la plateforme est présentée à la section 4.

[1] ALVIS Superpeer semantic Search Engine, projet IST / STREP n° 002068, voir `http://www.alvis.info/alvis`.

[2] Des versions slovène et chinoise ont également été développées dans le cadre du projet mais avec une ambition moindre pour le slovène et avec une architecture un peu différente pour le chinois.

2 État de l'art

Lors de cette dernière décennie, plusieurs architectures d'ingénierie du texte ont été développées pour articuler les traitements linguistiques (Cunningham *et al.*, 2000) sans toutefois se placer dans un contexte de recherche d'information. Ainsi, les architectures GATE (Bontcheva *et al.*, 2004), UIMA (Ferrucci & Lally, 2004) ou de Textpresso (Müller *et al.*, 2004) visent généralement l'annotation linguistique et l'exploration de corpus de taille moyenne pour l'extraction d'information. LinguaStream (Widlöcher & Bilhaut, 2005), quant à elle, est conçue comme un outil de dépouillement de corpus et d'expérimentation, qui formalise des traitements complexes.

Ces plates-formes appuient leur analyse des documents sur des outils de Traitement Automatique des Langues existants. Ceux-ci sont réutilisés dans des modules qui les encapsulent et qui assurent la conformité des entrées/sorties. La définition d'un format d'échange et d'annotation suffisamment générique est également un point crucial pour les plates-formes d'annotation. Il s'agit d'assurer une communication correcte des informations entre les modules, mais aussi une réutilisation des annotations produites dans des applications externes. Ont ainsi été proposés différents formats d'échange et d'annotation qui reposaient généralement sur SGML puis XML. Le format d'échange et d'annotation de GATE, CPSL (Common Pattern Specific Language) et d'UIMA, CAS (Common Analysis Structure) sont inspirés du format d'annotation TIPSTER (Grishman, 1997). Afin de préserver une certaine flexibilité, les annotations y sont déportées.

Au regard de nos contraintes (généricité, performances et adaptabilité à un domaine de spécialité), les plates-formes d'annotation existantes ne paraissent pas adaptées à la recherche d'information spécialisée. Si les plates-formes GATE et UIMA sont plutôt conçues comme des solutions génériques, le système Textpresso (Müller *et al.*, 2004) poursuit un objectif similaire au nôtre : proposer une architecture générique capable de traiter des corpus de documents issus d'un domaine spécialisé. Cette plate-forme a été conçue pour la fouille des documents traitant de biologie, aussi bien des résumés que des articles complets. Son évaluation a porté sur un corpus relativement petit : 16 000 résumés et 3 000 articles en texte brut.

En règle générale, on dispose de très peu d'informations pour d'évaluer les performances de ces systèmes sur un corpus de documents. Un premier test nous a montré que GATE ne convient pas au traitement de gros corpus de documents : seuls de petits volumes de documents pouvaient être traités sans rencontrer des problèmes. Ceci s'explique par le fait que GATE ait été conçue comme un environnement puissant de développement et de conception d'applications de TAL dans le cadre de l'extraction d'information. Le passage à l'échelle n'était pas un objectif central. La *méta-plate-forme* KIM (Popov *et al.*, 2004), qui s'appuie sur GATE, tente cependant de satisfaire cette contrainte dans le cadre de projets d'annotation sémantique massive SWAN[3] et SEKT[4]. L'architecture est dédiée à l'enrichissement d'ontologies, l'indexation sémantique et la recherche d'information. Bien que les auteurs identifient le passage à l'échelle comme un paramètre critique, aucune performance en terme de temps de calcul et de volume de documents traités, n'est fournie. Le traitement de grande collections de documents est cependant, envisageable avec UIMA, les temps de calcul et l'adaptabilité des traitements restant à évaluer. Celle-ci offre en effet la possibilité de traiter les documents les uns après les autres ou sous forme d'une collection. Le Collection Processing Engine (CPE) gère alors la parallélisation et surveille les performances.

[3] http ://deri.ie/projects/swan
[4] http ://sekt.semanticweb.org

Les plates-formes existantes répondent donc partiellement aux contraintes de l'intégration d'informations linguistiques dans un moteur de recherche spécialisé. Il s'agit généralement plus d'environnements de dépouillement que d'architectures d'annotation de gros volumes de données pouvant être utilisées pour la recherche d'information spécialisée. Nous avons donc développé une plate-forme capable de gérer d'importants volumes de documents en mettant l'accent sur l'efficacité et la robustesse des traitements effectués.

3 Une plate-forme modulaire et adaptable

Nous avons choisi de développer une plate-forme d'annotation linguistique exploitant des outils de TAL existants plutôt que d'en développer de nouveaux[5]. Nous avons ainsi pu mettre l'accent sur la robustesse des traitements et la rapidité de l'annotation de grandes quantités de documents spécialisés, en proposant une architecture modulaire et distribuée. De plus, l'adaptation des traitements nécessaires à l'analyse de textes spécialisés est réalisée par l'intégration de ressources spécifiques au domaine ou l'utilisation d'outils spécialisés pour un domaine.

3.1 Contraintes spécifiques

Le fait de réutiliser des outils existants et d'autoriser le remplacement de certains outils par d'autres imposent des contraintes spécifiques. Il faut notamment gérer l'hétérogénéité des formats d'entrées/sorties des outils utilisés dans la plate-forme. Chaque outil ayant généralement ses formats propres, il est donc crucial de définir un format d'échange permettant d'interconnecter librement des outils ensemble et de distribuer correctement les traitements.

Le développement d'une plate-forme d'annotation des textes spécialisés intègre également des contraintes spécifiques au TAL, comme la disponibilité de ressources lexicales, terminologiques et ontologiques, ou la nécessité d'adapter des outils au domaine afin d'améliorer certains traitements, comme l'étiquetage morpho-syntaxique ou l'analyse syntaxique, sur des sous-langages particuliers. De plus, toutes les étapes de traitement n'étant pas également pertinentes pour toutes les applications, nous avons préservé au maximum l'approche modulaire.

3.2 Architecture générale

Les différentes étapes de traitement sont traditionnellement prises en charge par un ensemble de modules (Bontcheva *et al.*, 2004). Chaque module est dédié à un type de traitement : reconnaissance d'entités nommées, segmentation en mots, étiquetage morpho-syntaxique, analyse syntaxique, etc. Un module encapsule l'outil effectuant une analyse linguistique donnée et assure la conformité du format des entrées/sorties avec la définition de type de documents (dorénavant DTD). Les annotations sont enregistrées dans un format XML déporté afin de pouvoir mieux gérer l'hétérogénéité des entrées/sorties des outils de TAL. La DTD est décrite dans (Taylor, 2006; Nazarenko *et al.*, 2006). La modularité de l'architecture facilite la substitution d'un outil par un autre, car le remplacement d'un outil n'a aucun impact sur l'ensemble de l'architecture.

[5]Nous avons toutefois développé des outils lorsqu'aucun outil répondant à nos besoin n'était disponible ou nous convenait. Nous avons, de plus, choisi de préférence des logiciels sous licence GPL ou libre/gratuit pour un usage non commercial.

La spécialisation de la plate-forme pour un domaine spécifique est assurée par les ressources de chacun des modules. Par exemple, une liste d'espèces ou de gènes peut être ajoutée au module de repérage d'entités nommées spécifiques à la biologie, afin de traiter des résumés de Medline. L'adaptabilité des traitements peut aussi se faire par l'intégration d'outils spécialisés.

La figure 1 présente l'architecture de la plate-forme dans son état actuel. D'autres modules tels que l'étiquetage sémantique et la résolution d'anaphores seront prochainement intégrés. Les boîtes représentent les différents modules composant la chaîne de traitements linguistiques. Ces modules sont décrits dans la section 3.3. Les flèches en traits pleins représentent le flux de données lors du traitement tandis que les flèches en pointillés représentent les ressources qui peuvent être utilisées dans la plate-forme.

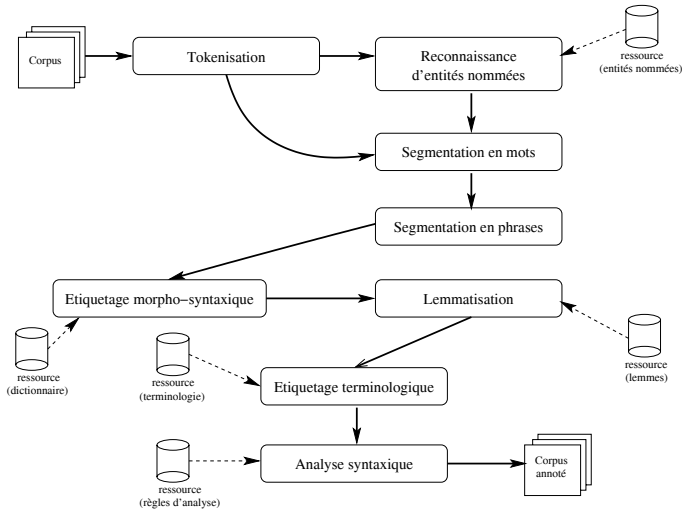

FIG. 1 – Architecture de la chaîne de traitement

Nous partons du principe que les documents Web donnés en entrée ont déjà été téléchargés, nettoyés, codés en UTF-8 et convertis au format XML (Taylor, 2006). Les documents sont d'abord tokenisés, ce qui permet de définir des offsets (indices délimitant une séquence, en nombre de caractères par rapport au début du document) pour garantir l'homogénéité des différentes annotations. Les tokens seront utilisés par les modules suivants. Les documents sont ensuite traités par divers modules : repérage d'entités nommées, segmentation en mots et en phrases, lemmatisation, étiquetage morpho-syntaxique, étiquetage terminologique et analyse syntaxique.

Cette architecture est assez traditionnelle mais certains points méritent d'être soulignés :

– La tokenisation constitue la première étape de la chaîne. Elle procède à une première segmentation, non linguistique, utilisée par la suite par les autres outils. Le token est donc l'unité textuelle de base dans la chaîne de traitements, et n'est qu'un point de départ pour les autres annotations. Ce niveau d'annotation suit les recommandations du groupe TC37SC4/TEI, même si nous employons le terme d'*offset de caractère* plutôt que celui de *pointeur d'élément* pour désigner les frontières de chaque token. Pour simplifier les traitements suivants, nous distinguons quatre types de tokens : alphabétiques, numériques, séparateurs et symboliques.

– L'étiquetage des entités nommées se produit très tôt dans la chaîne de traitement car l'identification des entités nommées facilite la désambiguïsation d'un certain nombre de marques de ponctuation lors de la segmentation en mots ou en phrases.
– L'étiquetage terminologique est utilisé tel quel mais peut également être considéré comme un préalable à l'analyse syntaxique. Cette dernière demandant beaucoup de temps de calcul, nous exploitons le fait qu'une analyse terminologique réduit le nombre d'analyses syntaxiques possibles (Aubin *et al.*, 2005).

3.3 Description des modules disponibles

Les modules sont appelés de manière séquentielle pour chaque document. Les sorties (annotations) sont stockées en mémoire jusqu'à la fin du traitement du document en cours, puis enregistrées dans un format XML.

Cette section décrit les différents modules intégrés à l'heure actuelle au sein de la chaîne de traitement. Il s'agit d'une description des modules par défaut de la plate-forme pour le traitement de l'anglais. Des outils similaires sont également intégrés pour le français (à l'exception de l'analyse syntaxique). De plus, la conception et l'implémentation de la plate-forme permet aisément une substitution d'un outil par un autre.

Etiquetage d'entités nommées. Le module assurant la reconnaissance des entités nommées identifie les séquences textuelles qui renvoient à une entité, leur associe un type sémantique (dépendant du domaine – pour la biologie, les étiquettes `gene` et `species`, par exemple) et, le cas échéant, normalise cette séquence. Dans la suite des traitements, une entité nommée est considérée comme une seule unité et assimilée à un mot. En les reconnaissant à un stade très préliminaire dans l'analyse, on évite des ambiguïtés ultérieures. Le module encapsule TagEN (Berroyer, 2004), qui repose essentiellement sur des dictionnaires et l'application de règles décrites sous formes de transducteurs.

Segmentation en mots et en phrases. Ce module identifie les phrases et les mots. Il exploite un ensemble d'expressions régulières reprenant l'algorithme proposé dans (Grefenstette & Tapanainen, 1994). Une partie de la segmentation est effectuée par le module de reconnaissance des entités nommées dans la mesure où celui-ci résout un grand nombre des problèmes liés à la ponctuation. C'est par exemple le module traitant les entités nommées qui permet de reconnaître la séquence « B. subtilis » , et qui met en rapport l'abréviation « B. » avec la forme étendue « Bacillus » . Le point présent dans la séquence « B. subtilis » n'a plus à être pris en compte au niveau de la segmentation en phrases.

Étiquetage morpho-syntaxique. Ce module associe une étiquette morpho-syntaxique à chaque mot du texte. Il repose sur la segmentation effectuée à l'étape précédente. Nous utilisons à l'heure actuelle le TreeTagger (Schmid, 1997). Nous avons aussi testé l'intégration de l'étiqueteur GeniaTagger (Tsuruoka *et al.*, 2005) qui est spécialisé pour le biologie, même si on observe que le gain en qualité de l'étiquetage se fait au détriment des performances.

Lemmatisation. Ce module associe un lemme à chaque mot du texte. Si le mot ne peut pas être lemmatisé (nombres, mots étrangers, mots inconnus), aucune information n'est associée à la forme. Ce module suppose que l'analyse morpho-syntaxique a préalablement été effectuée. Dans notre implémentation, la lemmatisation est effectuée en même temps que l'analyse morpho-syntaxique par le TreeTagger mais quand on utilise un étiqueteur qui ne fournit pas de lemmes, comme l'analyseur de Brill (Brill, 1995), il faut faire appel à un module spécifique pour la lemmatisation.

Étiquetage terminologique. Ce module vise à repérer les expressions du domaine qui ne sont pas des entités nommées, comme *gene expression* ou *spore coat cell* dans le domaine de la biologie. L'analyse peut être réalisée en projetant les termes fournis en entrée. Ceux-ci peuvent être issus de ressources terminologiques comme Gene Ontology (GO Consortium, 2001), le MeSH (MeSH, 1998) ou UMLS (UMLS, 2003) ou d'une ressource construite à l'aide d'un extracteur de termes. L'analyse morpho-syntaxique et la lemmatisation du texte sont nécessaires pour procéder à l'analyse terminologique.

Analyse syntaxique. L'analyse syntaxique vise à produire, pour chaque phrase du texte, un graphe reflétant les dépendances entre mots au sein de la phrase. L'analyse repose sur les sorties de l'analyse morpho-syntaxique. La plupart des analyseurs n'exigent pas une analyse terminologique préalable mais celle-ci permet de faire décroître largement l'ambiguïté et donc la complexité de l'analyse (Aubin *et al.*, 2005).

L'analyse syntaxique demande encore aujourd'hui des temps de calcul beaucoup plus importants que les autres étapes d'analyse, dans la mesure où elle opère sur un espace de recherche très vaste (tous les mots de la phrase sont potentiellement liés deux à deux). Nous avons choisi d'intégrer le Link Grammar Parser (Sleator & Temperley, 1993), qui repose sur des grammaires de dépendance, comme traitement par défaut. Pour le traitement de textes biomédicaux, l'adaptation de cet outil au domaine de la biologie BIOLG (Pyysalo *et al.*, 2006) est utilisée.

3.4 Implémentation

La plate-forme est implémentée en Perl et est disponible sous forme de modules CPAN (`http://search.cpan.org/~thhamon/Alvis-NLPPlatform-0.3/`). Nous avons utilisé un modèle client/serveur, mais la plateforme peut également traiter séquentiellement et de manière autonome une collection de documents. Dans le contexte d'utilisation client/serveur, chaque client récupère auprès du serveur les documents à traiter les uns après les autres et les analyse. Les documents annotés sont ensuite renvoyés au serveur qui, dans l'ensemble de la chaîne de traitement de recherche d'information d'ALVIS, les envoie au moteur d'indexation.

4 Analyse des performances

La plate-forme que nous avons développée vise à analyser des textes provenant du web pour des moteurs spécialisés dans des domaines techniques. Bien qu'il ne s'agisse pas d'analyse en temps réel, les performances doivent être acceptables. On vise ainsi l'analyse de plusieurs giga-octets de données par jour. Ce type de performances implique une architecture distribuée, qui est par

définition robuste dans la mesure où l'on peut ajouter de nouvelles machines en fonction de la charge. Au-delà des performances, le système doit également être robuste face aux documents fournis en entrée, qui peuvent être très variables quant à leur taille ou leur contenu, notamment quand il s'agit de documents issus du web.

Nous avons mené une expérience d'annotation de deux collections de documents issus du Web. La première collection regroupe 55 329 documents biomédicaux (désormais BIO). La plupart des documents XML ont une taille comprise entre 1 kilo-octet et 100 kilo-octets. La taille du plus grand document est 5,7 méga-octets. La seconde collection comporte 48 422 dépêches relatives aux moteurs de recherche (désormais SEN). La taille des documents varie entre 1 et 150 kilo-octets.

Nous nous sommes placés dans le contexte d'annotation d'un flux de documents venant du Web. Ainsi, nous avons réalisé l'ensemble des traitements jusqu'à l'étiquetage terminologique. Pour l'annotation de la collection BIO, nous avons exploité une liste de 375 000 termes issus du MeSH et de Gene Ontology, Sur la collection SEN, la liste comportait 17 341 termes extraits automatiquement. Nous avons utilisé une liste d'environ 400 000 entités nommées, incluant des noms d'espèce et de gènes sur le corpus BIO, ou des noms de personne, de logiciel et de société sur le corpus SEN.

L'annotation des documents a été distribuée sur vingt ordinateurs. La plupart sont des ordinateurs classiques (de type PC) avec 1 giga-octet de mémoire vive (RAM) et un processeur cadencé à 2,9 ou 3,1 GHz. Nous avons également utilisé un ordinateur avec 8 giga-octets de RAM et deux processeurs Xeon cadencés à 2,8 GHz (processeur Xeon dual core). Le système d'exploitation utilisé est Linux (Debian ou Mandrake). Le serveur et trois clients étaient hébergés sur la machine bi-processeur Xeon. Chaque ordinateur personnel abritait un seul client réalisant l'ensemble de la chaîne de traitement.

Les performances obtenues donnent une bonne idée des performances globales de la plate-forme (une évaluation complète aurait demandé des séries plus importantes de test). Le temps d'exécution de chaque module a été enregistré à l'aide du module Perl `Time::Hires`. Les temps d'analyse sont inscrits dans le fichier XML produit en sortie.

L'annotation de la collection, à l'exception de deux documents, a été effectuée en 35 heures. Le corpus est composé de 106 millions de mots et 4,72 millions de phrases. 147 documents ne contenaient aucun mot, ils n'ont donc pas été analysés au-delà de l'étape de tokenisation. Un des clients a analysé un document composé de 414 995 mots.

Les documents du corpus BIO sont analysés en moyenne en 35 secondes. La génération du fichier XML prend en moyenne 2 secondes supplémentaires. Les étapes les plus coûteuses en temps de traitement sont celles qui demandent le plus de ressources, à savoir la reconnaissance des termes (56 % du temps de traitement global) et la reconnaissance des entités nommées (16 % du total).

Lors ces deux expériences, l'ensemble des documents a été traité sans rencontrer de problème. Les performances obtenues montrent que la plate-forme développée est robuste, et qu'elle peut traiter des grandes masses de textes dans des temps raisonnables. Celles-ci pourraient être encore améliorées par une optimisation du code, et un travail approfondi sur le module d'étiquetage terminologique. Le processus permet une indexation précise de documents spécialisés.

5 Conclusion

Nous avons présenté une plate-forme, OGMIOS, destinée à enrichir des documents issus de domaines spécialisés avec des annotations linguistiques. Les expériences présentées ont porté sur des collections de documents issus du web. Nous avons montré que l'architecture et les modules intégrés à la plate-forme sont adaptés au traitement de textes de langue de spécialité. L'architecture est en outre suffisamment générique pour permettre l'adaptation à d'autres domaines. La plate-forme est actuellement utilisée par d'autres partenaires du projet ALVIS et notamment pour l'annotation de documents issus de bibliothèques numériques en biomédecine.

La stratégie adoptée consiste à réutiliser des modules existants et à les adapter au domaine visé. Ceux-ci peuvent bien évidemment être remplacés par d'autres et les traitements peuvent être enchaînés de différentes façons en fonction du résultat visé. Les modules intégrés sont pour l'instant : la reconnaissance des entités nommées, la segmentation en phrases et en mots, l'analyse morpho-syntaxique et la lemmatisation, la reconnaissance des termes et l'analyse syntaxique. Un module de résolution d'anaphore ainsi que d'autres outils terminologiques seront prochainement intégrés.

Les performances sont le point clé de ce type d'application. Nous avons décrit une implémentation distribuée de la plate-forme permettant le traitement de la collection de documents sur plusieurs machines. Les temps de calcul obtenus sont acceptables pour une tâche de RI.

Remerciements

Ce travail a été réalisé, pour l'essentiel, dans le cadre du projet ALVIS (projet européen IST du 6ème programme cadre – Partenaires : HIIT (Helsinki, Finlande), MIG-INRA (Jouy en Josas, France), LSIR-EPFL (Lausanne, Suisse), ULUND (Lund, Suède), DTU (Copenhague, Danemark), LIPN (Paris, France), JSI (Liubliana, Slovénie), DCSTH (Tsinghua, Chine), IndexData (Copenhague, Danemark), Exalead (France), ALMA Bioinformatica (Madrid, Espagne)). Les données et les exemples fournis ont été obtenus en interaction avec les partenaires du projet. La conception de cette plateforme a bénéficié d'une collaboration de plusieurs années avec le groupe MIG de l'INRA qui a notamment défini le cadre des expériences en biologie.

Références

AUBIN S., NAZARENKO A. & NÉDELLEC C. (2005). Adapting a general parser to a sublanguage. In *Proceedings of the International Conference on Recent Advances in Natural Language Processing (RANLP'05)*, p. 89–93, Borovets, Bulgaria.

BERROYER J.-F. (2004). Tagen, un analyseur d"entités nommées : conception, développement et évaluation. Mémoire de D.E.A. d'intelligence artificielle, Université Paris-Nord.

BONTCHEVA K., TABLAN V., MAYNARD D. & CUNNINGHAM H. (2004). Evolving GATE to meet new challenges in language engineering. *Natural Language Engineering*, **10**(3-4), 349–374.

BRILL E. (1995). Transformation-based error-driven learning and natural language professing : A case study in part-of-speech tagging. *Computational Linguistics*, **21**(4), 543–565.

CUNNINGHAM H., BONTCHEVA K., TABLAN V. & WILKS Y. (2000). Software infrastructure for language resources : a taxonomy of previous work and a requirements analysis. In *Proceedings of the 2nd International Conference on Language Resources and Evaluation (LREC-2)*, Athens.

FERRUCCI D. & LALLY A. (2004). UIMA : an architecture approach to unstructured information processing in a corporate research environment. *Natural Language Engineering*, **10**(3-4), 327–348.

GO CONSORTIUM (2001). Creating the Gene Ontology Resource : Design and Implementation. *Genome Res.*, **11**(8), 1425–1433.

GREFENSTETTE G. & TAPANAINEN P. (1994). What is a word, what is a sentence ? problems of tokenization. In *The 3rd International Conference on Computational Lexicography*, p. 79–87, Budapest.

GRISHMAN R. (1997). *Tipster architecture design document version 2.3*. Rapport interne, DARPA.

MESH (1998). Medical subject headings. Library of Medicine, Bethesda, Maryland, WWW page `http ://www.nlm.nih.gov/mesh/meshhome.html`,.

MÜLLER H.-M., KENNY E. E. & STERNBERG P. W. (2004). Textpresso : an ontology-based information retrieval and extraction system for biological literature. *PLoS Biology*, **2**(11), 1984–1998.

NAZARENKO A., ALPHONSE E., DERIVIÈRE J., HAMON T., VAUVERT G. & WEISSENBACHER D. (2006). The ALVIS format for linguistically annotated documents. In *Proceedings of LREC 2006*.

POPOV B., KIRYAKOV A., OGNYANOFF D., MANOV D. & KIRILOV A. (2004). Kim – a semantic platform for information extraction and retrieval. *Natural Language Engineering*, **10**(3-4), 375–392.

PYYSALO S., SALAKOSKI T., AUBIN S. & NAZARENKO A. (2006). Lexical adaptation of link grammar to the biomedical sublanguage : a comparative evaluation of three approaches. In J. F. SOPHIA ANANIADOU, Ed., *Proceedings of the Second International Symposium on Semantic Mining in Biomedicine (SMBM 2006)*, p. 60–67, Jena, Germany.

SCHMID H. (1997). Probabilistic part-of-speech tagging using decision trees. In D. JONES & H. SOMERS, Eds., *New Methods in Language Processing Studies in Computational Linguistics*.

SLEATOR D. D. & TEMPERLEY D. (1993). Parsing English with a link grammar. In *Third International Workshop on Parsing Technologies*.

TAYLOR M. (2006). Report on metadata frameworks, including concrete representations, for network nodes and semantic document analyses. ALVIS Deliverable 3.1.

TSURUOKA Y., TATEISHI Y., KIM J.-D., OHTA T., MCNAUGHT J., ANANIADOU S. & TSUJII J. (2005). Developing a robust part-of-speech tagger for biomedical text. In *Proceedings of Advances in Informatics - 10th Panhellenic Conference on Informatics*, LNCS 3746, p. 382–392.

UMLS (2003). UMLS knowledge source. National Library of Medicine.

WIDLÖCHER A. & BILHAUT F. (2005). La plate-forme linguastream : un outil d'exploration linguistique sur corpus. In *Actes de la conférence TALN 2005*, p. 517–522, Dourdan, France.