Donner du sens à des documents semi-structurés : de la construction d'ontologies à l'annotation sémantique

Nathalie Aussenac-Gilles

Donner du sens à des documents semi-structurés
De la construction d'ontologies à l'annotation sémantique

Nathalie Aussenac-Gilles

Directeur de recherche au CNRS depuis 2010, habilitée à diriger des recherches depuis 2005, Nathalie Aussenac-Gilles est membre de l'Institut de recherche en informatique de Toulouse (IRIT) depuis son entrée au CNRS en 1991. Elle y anime depuis 2007 l'équipe IC3 devenue MELODI (Modèles et ingénierie de la langue, de l'ontologie et du discours) en 2012, ainsi que divers groupes de travail associés au GDR-I3 et à l'AFIA. Ses recherches portent sur les approches ascendantes pour l'acquisition et la modélisation de connaissances, abordées de manière interdisciplinaire par des collaborations avec des chercheurs en psychologie cognitive, ergonomie et linguistique. Depuis quinze ans, elle s'est orientée vers le traitement automatique des langues et les approches terminologiques pour la construction et l'utilisation d'ontologies en lien avec des textes, pour des applications comme la recherche sémantique d'informations ou le web sémantique. aussenac@irit.fr

Ce chapitre est consacré aux approches qui permettent de donner à des documents textuels un sens pertinent et « interprétable par un système informatique ». Cela revient à élaborer une représentation formelle et sémantique, plus ou moins proche du contenu du document, indépendante ou associée à celui-ci suivant les objectifs. En cela, ces questions rejoignent celles de la construction d'ontologies à partir de textes, qui fait l'objet de nombreuses recherches depuis dix ans. Du point de vue des techniques utilisées, ces approches s'appuient en particulier sur le traitement automatique du langage contenu dans les documents, sur de l'extraction d'information à partir de textes et sur les ontologies comme modèle résultat ou ressource du processus d'interprétation.

À partir des années 2000, plusieurs ouvrages ou articles de référence ont été publiés [32] [25] [12] puis [13] [26]. Plusieurs tutoriels nationaux ont également été proposés sur ce thème, d'abord à la communauté des chercheurs en ingénierie des connaissances (IC 2000 puis RFIA 2002), puis aux membres de la communauté du traitement automatique des langues (TALN 2003) [9]. Ces notions font maintenant partie d'enseignements pour les étudiants de master de ces domaines. En s'adressant, entre autres, aux professionnels de l'information et de la documentation, et presque dix ans après les premiers tutoriels, cette contribution se veut à la fois une présentation des travaux les plus opérationnels, un bilan sur les succès et les points difficiles qui demeurent, et surtout une actualisation des enjeux « à l'heure du web des données ».

Un bilan rapide et caricatural des travaux sur le traitement automatique des langues, la modélisation de connaissances et les ontologies fait ressortir les faits marquants suivants :
- des résultats performants et industrialisés en extraction d'information, en particulier pour des applications économiques (relations entreprises/personnes/pays/production) ou la datation et la localisation d'événements, ou plus récemment l'analyse d'opinions ou la recherche de compétences ou d'expertises ;
- des résultats capitalisés par domaine, en particulier en médecine et bio-médecine, domaine « favorable » car disposant à la fois de moyens, de ressources textuelles considérables (PubMed et ses giga-octets d'articles scientifiques) et des besoins forts (recouper des résultats éparpillés pour découvrir des connaissances, en particulier en génomique) ;
- pas de produit « clé en main » couvrant totalement chacun des processus (qu'il s'agisse de construire des ressources sémantiques ou de décrire sémantiquement des contenus textuels), mais des réponses partielles (soit tout le processus est couvert pour un type de corpus et de connaissances particuliers, soit seulement une partie du processus est traitée et s'applique à davantage de domaines). Dans tous les cas, il est souhaitable de savoir définir une chaîne de traitement adaptée à l'objectif que l'on vise et au jeu de données dont on dispose ;
- de perspectives de recherche toujours à explorer pour parvenir à des logiciels modulaires, intégrables, simples et capables de produire ou de traiter plusieurs caractérisations des textes, de manipuler des modèles plus riches ou des annotations à différents niveaux.

Ce chapitre alimente la question posée par le thème de cet ouvrage : que devient « le document numérique à l'heure du web des données » ? En effet, les chercheurs travaillant depuis plusieurs années sur « ontologies et analyse du langage » s'interrogent de la même manière sur l'impact du web des données : s'agit-il d'une évolution « naturelle » qui va substituer l'utilisation de données liées à celle des ontologies ? d'un recul des ambitions du web sémantique ou d'ambitions plus réalistes ? qu'est-ce qui change dans l'analyse de textes si l'on doit produire

des données liées (des triplets) plutôt que des ontologies ? On peut voir pour preuve de ces interrogations l'appel à communication d'un séminaire qui se tiendra en octobre 2012[1], dont les sous-titres provocateurs sont « *Did the current data-driven world kill ontologies?* » ou « *Are we navigating towards a shallow Web of Data?* »

Or, qu'il s'agisse d'une substitution ou d'une cohabitation entre données liées et ontologies, certaines questions demeurent les mêmes (construire des données liées qui ont du sens d'une part, utiliser ces données pour accéder aux contenus en ligne, donc aux documents numériques d'autre part). D'ailleurs, les chercheurs motivés par la production de données liées invitent déjà, à travers un autre séminaire comme Web of Linked Entities (WoLE) 2012, les communautés du traitement automatique des langues (TAL) et de l'extraction d'information à collaborer avec les chercheurs du web sémantique et à s'intéresser au web des données.

Dans la première partie de ce chapitre, nous allons donc revenir sur la notion d'ontologie pour les situer par rapport aux données liées. Nous reviendrons également sur leur production plus ou moins automatisée à partir de textes comme moyen de rendre compte de connaissances d'un domaine, et sur les résultats les plus intéressants et les plus opérationnels produits en la matière par des recherches croisées en ingénierie des ontologies, TAL, apprentissage automatique et extraction d'information.

Dans une deuxième partie, nous nous intéresserons au processus inverse qui consiste à caractériser, à l'aide de concepts d'un domaine ou de classes sémantiques, et de manière formelle, un document numérique (nous traiterons essentiellement de documents numériques textuels).

Pour conclure, nous évoquerons le problème difficile de l'évaluation de ces travaux ainsi que quelques systèmes qui se focalisent désormais sur la production de données liées.

1 Construction et peuplement d'ontologies à partir de textes

1.1 Rappel : pourquoi des ontologies ?

Avec le projet du web sémantique pour mieux accéder aux documents numériques du web et à leurs contenus, les ontologies comme structures de données et modèles de connaissances informatiques sont devenues incontournables et suscitent de nombreux espoirs. L'hypothèse est la suivante : pour produire, diffuser, rechercher, exploiter et traduire ces documents, les systèmes de gestion de l'information ont besoin de ressources termino-ontologiques qui articulent langue et connaissances et qui décrivent les termes et les concepts du domaine, selon un mode propre au type de traitement effectué par le système. Ces ressources doivent être consensuelles et définies précisément pour être associées à des fragments de documents, et en caractériser le contenu selon un point de vue et un niveau de granularité choisi *a priori*.

Si on les définit par ce qu'elles sont, les ontologies du web sémantique dérivent des classes conceptuelles et leur donnent du sens par les relations qu'elles entretiennent (et qui permettent de décrire leurs propriétés) ou par les règles et axiomes qui les définissent (et permettent de raisonner). Si on les définit par leur utilisation, les objectifs prioritaires auxquels doivent répondre les ontologies s'imposent d'emblée : le partage de connaissances, la communication facilitée entre applications et enfin l'annotation et la description des contenus des documents numériques. Dans ce dernier objectif, qui nous intéresse particulièrement, les ontologies jouent un rôle similaire à celui des langages documentaires dans une indexation manuelle : elles fournissent un vocabulaire accepté par une communauté d'utilisateurs et grâce auquel vont être décrits des contenus. Mais les ontologies se distinguent de ces langages par leur formalisation, qui permet à un programme de « donner du sens » aux éléments de ce vocabulaire.

Prenons un exemple[2] sur les conséquences liées à différentes manières de formaliser une même phrase :

> *Florence Aubenas (née le 6 février 1961 à Bruxelles) est une journaliste française*. (1)

La génération automatique et rapide de triplets pourrait conduire à la représentation des prédicats suivants :

> « Forence Aubenas » comme entité sur laquelle je connais deux assertions / deux triplets
> Française(Florence Aubenas) soit le triplet :FlorenceAubenas rdf:type :Français
> Journaliste(Florence Aubenas) soit le triplet :FlorenceAubenas rdf:type :Journaliste

[1] 1st International Workshop on Ontology Engineering in a Data-Driven World (OEDW 2012) : http://granvia.dia.fi.upm.es/oedw2012

[2] Repris à J. Corman, étudiant en thèse dans MELODI.

Je peux répondre ainsi à des questions du type :

Qui est journaliste ? Florence Aubenas
Florence Aubenas est-elle journaliste ? Oui
Florence Aubenas est-elle française ? Oui

mais pas aux questions suivantes :

Quelle est la profession de Florence Aubenas ?
Quelle est la nationalité de Florence Aubenas ?

car elles supposent de changer de niveau d'abstraction, de savoir que journaliste EST-UNE profession et que Français EST-UNE nationalité.

De plus, ces formalisations ne sont pas très satisfaisantes : dire que « :FlorenceAubenas est de type :Journaliste » suppose qu'elle est caractérisée par cette profession et n'en changera pas.

Ce serait encore plus choquant si, à partir de la phrase (2) dans un texte daté de plusieurs années en arrière :

Florence Aubenas, journaliste française, est otage depuis deux ans. (2)

on générait aussi le triplet :

:FlorenceAubenas rdf:type :otage

Ce triplet ne tient pas compte de la dimension temporelle, et donne le type « :otage » à un personne dont on espère qu'il s'agit d'une situation provisoire. Le fait de disposer d'une ontologie doit donc permettre de générer de meilleurs triplets, de définir vraiment à l'aide de classes, de changer de niveau d'abstraction, de produire des raisonnements plus précis.

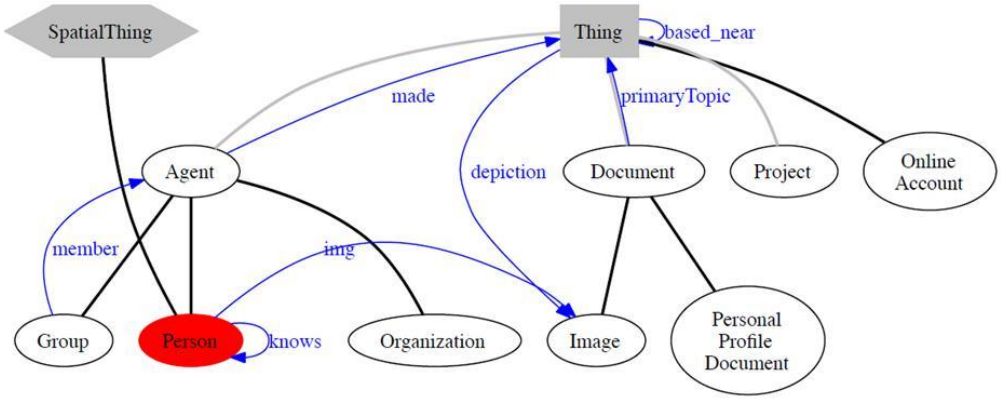
A-Box (faits)
aPourNationalité(Florence Aubenas, français) :FlorenceAubenas :apourNationalité :français
aPourProfession(Florence Aubenas, journaliste) :FlorenceAubenas :apourProfession :journaliste
Nationalité(français) :français rdf:type :Nationalité
Profession(journaliste) :journaliste rdf:type :Profession
Personne(Florence Aubenas) :FlorenceAubenas rdf:type :Personne

T-Box(ontologie)
?xy (aPourNationalité (x , y) -> Nationalité (y) ET Personne(x)) apourNationalité rdfs:domain :Personne ;
:apourNationalité rdfs:range :Nationalité .
?xy (aPourProfession(x , y) -> Profession(y) ET Personne(x)) :apourProfession rdfs:domain :Personne ;
:apourProfession rdfs:range :Nationalité

On construit ainsi une ontologie « personnelle » avec trois classes : « :Personne », « :Profession » et « :Nationalité », qui permet d'être plus précis, d'anticiper qu'une autre personne puisse exercer une autre profession ou avoir une autre nationalité.

Pour mieux partager cette ontologie, il est préférable de recourir à FOAF[3], une petite ontologie très utilisée sur le web, qui définit la notion de personne et permet de préciser les propriétés « FirstName » et « LastName », par exemple [figure 1].

[3] http://xmlns.com/foaf/spec

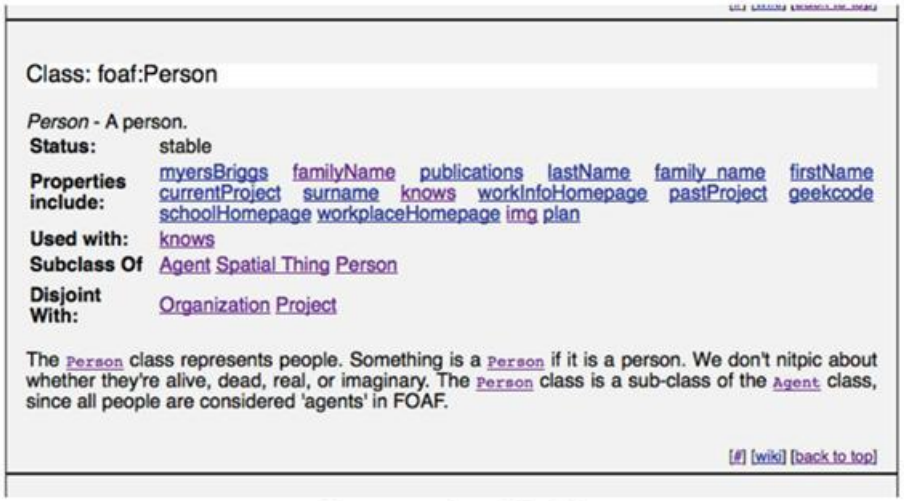


FOAF

- Thing : classe-racine de OWL
- SpatialThing : classe de W3C Basic Geo
- Arcs noirs : relation IS-A (gris : implicites)
- Arc bleus : autres relations entre catégories, décrites parmi les "properties"

Figure 1 – Les principaux concepts de l'ontologie FOAF

Mais FOAF ne répond pas exactement aux besoins et devra de toute façon être enrichie en ajoutant la notions de nationalité et de profession et en les reliant à « :foaf:person ». De plus, si l'ontologie doit être partagée sur le web, le W3C recommande que les identifiants soient des URIs. Si deux personnes définissent « #FlorenceAubenas », avec deux URIs différentes, elles pourront indiquer dans leur ontologie qu'il s'agit de la même entité [figure 2].



Class: foaf:Person

Person - A person.

Status: stable

Properties include: myersBriggs familyName publications lastName family_name firstName currentProject surname knows workInfoHomepage pastProject geekcode schoolHomepage workplaceHomepage img plan

Used with: knows

Subclass Of Agent Spatial Thing Person

Disjoint With: Organization Project

The Person class represents people. Something is a Person if it is a person. We don't nitpic about whether they're alive, dead, real, or imaginary. The Person class is a sub-class of the Agent class, since all people are considered 'agents' in FOAF.

[#] [wiki] [back to top]

Personne dans FOAF

Figure 2 – Le concept de personne dans FOAF

Considérons maintenant les deux phrases :

> *Florence Aubenas (née le 6 février 1961 à Bruxelles) est une journaliste française. Elle parle français, italien et anglais.* (3)

Pour rendre compte de ces connaissances, on peut compléter l'ontologie avec la notion de langue (« :langue »), dont trois instances sont :

:français rdf:type :langue
:italien rdf:type :langue
:anglais rdf:type :langue

et la relation (« ObjectProperty » en OWL) :

:sait_parler rdfs:domain :personne rdfs:range :langue

On peut alors déclarer les faits suivants :

:FlorenceAubenas :sait_parler :français
:FlorenceAubenas :sait_parler :italien
:FlorenceAubenas :sait_parler :anglais

Mais alors « :français » appartient à deux classes « :langue » et « :nationalité ».

Ceci pourrait ne pas poser problème mais, de fait, cela correspond à deux sens du mot « français » : dans la première phrase, l'adjectif « française » définit une nationalité particulière, la « nationalité française », alors que dans le second cas, le nom « français » correspond à la « langue française ».

Le fait de définir une ontologie oblige ainsi à être plus précis dans la définition de connaissances et dans une analyse (ici manuelle) de contenus, et à définir donc :

:français rdf:type :langue
:nationalité_française rdf:type :nationalité

Enfin, l'ontologie permet de raisonner sur les connaissances et d'en synthétiser. Par exemple, si on définit deux sous-classes de « :personne » : « :homme » et « :femme », alors les propriétés vraies pour « :personne » le seront pour « :homme » et « :femme » :

:sait_parler rdfs:domain :homme rdfs:range :langue
:sait_parler rdfs:domain :femme rdfs:range :langue
:apourProfession rdfs:domain :homme rdfs:range :profession
etc.

De plus, on peut construire le concept de « :JournalisteFrançais » non pas comme sous-classe de « journaliste » mais comme un concept défini par une formule axiomatique :

pour toute :Personne(p) telle que (p : apourProfession :journaliste) Et (p :apourNationalité :NationalitéFrançaise),
alors :JournalisteFrançais(p)

1.2 Démarche générale

1.2.1 Pourquoi s'appuyer sur les textes : limites et avantages

Dans une précédente publication [4], nous citions plusieurs avantages attendus de la construction d'ontologie à partir de textes : définir des ontologies de domaine très spécifiques et en adéquation avec le point de vue présent dans des corpus spécialisés ou avec des usages précis, des ontologies « idiosyncrasiques » ; réduire le coût de construction des ontologies, et les associer à des schémas ou des ontologies existantes.

Une des hypothèses qui sous-tendent cette démarche est que ce genre d'ontologie est dans certains cas plus pertinent que des ontologies génériques, en particulier dans des applications associant ontologies et textes. Une autre hypothèse est que les techniques actuelles d'analyse du langage et d'apprentissage supervisé à partir de textes permettent de rendre compte de manière rapide des vocabulaires utilisés et de leur sémantique, ou de faire des propositions à affiner.

Plus récemment, le web lui-même a été pris comme corpus. Il peut être utilisé dans son ensemble pour constituer des ressources générales (soit des ontologies, soit des connaissances linguistiques utilisées par le TAL, comme des patrons linguistiques). Le web sert également à former des corpus spécialisés ou thématiques de grande taille, sur lesquels des approches statistiques seront efficaces pour constituer des ontologies spécialisées. La sélection de documents se fait à l'aide de mots-clés et de moteurs de recherche classiques, ou d'indices plus fins (présence de mots-clés et de définitions, par exemple). Actuellement, beaucoup d'approches exploitent Wikipedia comme corpus d'analyse en complément ou en lieu de corpus d'étude.

1.2.2 Les difficultés rencontrées sont de trois ordres

● *Difficultés liées à la complexité de la langue*. Nous choisissons trois phénomènes classiques à titre d'illustration : les termes polysémiques (cf. « français » dans l'exemple (3) sur F. Aubenas) ; le calcul de références (toujours dans l'exemple (3), savoir que « elle » renvoie à « Florence Aubenas » dans la phrase « Elle parle français, italien et anglais ») ; les calculs complexes de rattachement d'adjectifs à des groupes nominaux ou verbaux (par exemple, dans un corpus de jardinage, « croissance : lente, au printemps » suppose de rattacher l'adjectif « lente » à « croissance », ce qui échoue dans la plupart des systèmes d'analyse à cause du signe de ponctuation « : »).

● *Difficultés liées aux méthodes utilisées en traitement automatique des langues*. La plupart des logiciels actuels s'appuient sur des mesures statistiques au sein de corpus volumineux, sur des connaissances linguistiques modélisées ou apprises, ou sur une combinaison des deux. De manière générale, les approches linguistiques supposent des analyses fines et complexes. Sinon, elles restent approximatives, peu productives ou bruitées. Par exemple, l'approche par patrons pour l'extraction de relations est souvent mise en avant pour la construction d'ontologies depuis les travaux de Marti Hearst [23]. Or un patron trop simple comme :

X de Y, avec Y nom propre -> membreDe(X,Y)

est pertinent pour reconnaître que « MembreDe(:pays, :Otan) » dans les pays de l'Otan mais génère beaucoup de bruit avec des exemples erronés comme la ville de Rome ou la chèvre de Monsieur Seguin. L'analyse de corpus spécifiques (par opposition à la langue générale) se heurte à deux restrictions supplémentaires : les corpus sont de taille moyenne, souvent trop petite pour produire des analyses statistiques pertinentes, et les connaissances linguistiques doivent être adaptées aux particularités du corpus. C'est principalement cette difficulté qui rend nécessaire le développement de chaînes de traitement dédiées.

● *Difficultés liées à la modélisation*. Étant donné un texte, plusieurs interprétations en sont possibles par les lecteurs, suivant leur objectif. Un même article de journal, par exemple ne sera pas lu de la même manière pour s'informer, faire un dossier de presse sur un thème particulier ou sur l'auteur, une analyse de société, etc. De même, suivant le modèle de connaissances que l'on veut construire, on va retenir différents types et niveaux d'information du texte, et les restituer différemment au sein d'une ontologie ou d'annotations.

Or l'analyse du langage ne véhicule qu'en partie l'intention de modélisation. Dans les systèmes d'extraction d'information, le filtre de lecture est donné par les classes sémantiques recherchées et les lexiques qui leur sont associés. Si l'on a prévu, par exemple, de rechercher des noms de personnes, de villes et d'entreprises, un processus d'apprentissage peut permettre d'apprendre, à partir d'exemples de textes annotés, de nouveaux noms pour enrichir les lexiques correspondants. Mais le système ne pourra pas reconnaître, par exemple, des lieux géographiques autres que les villes, etc. Dans les systèmes plus généraux comme les extracteurs de termes, le point de vue de l'analyste est absent.

Les systèmes d'analyse du langage font donc des propositions, ils présentent des extraits linguistiques, mais nullement des fragments de modèle. Le processus de modélisation consiste justement à faire des choix et à décider du ou des fragments de modèle à produire.

Un premier choix est de savoir si l'on produit des faits, des informations impliquant des « instances » d'une ontologie : il s'agit alors de *peupler* une ontologie ; ou bien, si l'on produit de nouvelles « classes » et « relations », des connaissances à ajouter dans l'ontologie : il s'agit de *construire* ou *enrichir* l'ontologie. Peupler une ontologie suppose en général de disposer de l'ontologie à instancier. Peupler une ontologie à partir de texte se rapproche donc d'une extraction d'information avec de multiples classes. Construire l'ontologie est en général une démarche ascendante, qui peut être combinée à la réutilisation d'ontologies existantes ou de noyaux génériques qu'il s'agit de spécialiser à des domaines ou des corpus spécifiques.

Toujours à partir de l'exemple (3), nous avons fait (manuellement) une modélisation possible, à la fois au niveau des classes (relatives à « :personne », « :profession », « :nationalité », « :langue », etc.) et au niveau de l'instance (« :FlorenceAubenas rdf:type :Personne » ; : »FlorenceAubenas :apourProfession :Journaliste », etc.).

D'autres choix portent sur les informations à représenter ou non, et sur la manière de représenter une connaissance. Dans l'exemple (3), nous n'avons pas exploité l'information sur l'âge : peut-être l'application visée ne nécessite-t-elle pas d'indiquer l'âge d'une personne. Nous avons associé la nationalité à une personne, alors que nous aurions pu l'associer à la profession journaliste, en ajoutant une sous-classe à « :Journaliste » qui serait « :JournalisteFrançais », par opposition à « :JounalisteAnglais », par exemple. Nous aurions aussi pu décider que « :Journaliste » est une sous-classe de « :personne » (ce qui est fait par exemple dans DBpedia) au lieu de le représenter sous forme de relation, ce qui revient à produire la même « incohérence » que si l'on ne crée pas le

concept de personne : une personne ne peut pas exercer deux professions à un instant donné ou même au cours de sa vie.

1.3 Critères de bonne structuration d'une ontologie

Quelle que soit la manière de trouver des connaissances à intégrer dans le modèle que constitue une ontologie, plusieurs méthodes proposent de s'interroger sur le caractère définitoire et ontologique de ces connaissances. Buitelaar, Cimiano et Völker [12] proposent de fixer explicitement le paradigme retenu pour déterminer le contenu de l'ontologie. Ce paradigme peut être cognitif (un concept est défini pour représenter une entité mentale, utilisé dans un raisonnement, par exemple), linguistique (un concept rend compte des différences véhiculées par la langue), philosophique ou pragmatique si les concepts sont ceux utiles pour une application.

La méthode Archonte de Bruno Bachimont [5] reprend des principes différentiels adaptés des définitions d'un concept par différence par rapport à un concept générique pour suggérer de ne définir un concept que si l'on sait formuler au moins une différence et au moins un point commun avec ses frères et son père (critères de différenciation). Ce principe conduit à élaborer des spécialisations de concepts homogènes et à expliciter les critères de structuration à chaque niveau de l'ontologie.

Enfin, pour corriger une ontologie, la méthode OntoClean [21] applique les principes de l'ontologie formelle à l'aide de métapropriétés des concepts et des relations entre les concepts de cette ontologie. OntoClean suppose que les concepts de l'ontologie de domaine sont situés comme sous-classes de ceux de l'ontologie de haut niveau DOLCE. OntoClean peut également servir à corriger une ontologie au fur et à mesure de sa construction. Il s'agit de vérifier des propriétés liées à la durée, à l'évolution ou la stabilité dans le temps (rigidité) d'un concept ou d'une propriété, de bien distinguer les objets (classes) des individus (instances), les objets des rôles et des qualités, de vérifier leur unicité, leur identité, etc.

1.4 Typologies des logiciels de TAL pour faciliter la construction d'ontologie

Si les logiciels de TAL pour la construction d'ontologie sont bien une réalité, il est difficile aujourd'hui de fournir une liste de programmes directement opérationnels et utilisables. En effet, il s'agit pour le moment de prototypes de recherche et non de produits industriels. De plus, la diversité des langues, des domaines, des plates-formes informatiques et des formats d'ontologies pour lesquels ces logiciels ont été définis rend leur maintenance difficile. Ces logiciels servent plus de preuve de l'efficacité de nouveaux algorithmes que d'aides opérationnelles à des utilisateurs. Nous mentionnerons ici plusieurs logiciels et plates-formes disponibles en ligne et qui sont soit téléchargeables, soit exécutables sur un site web à partir duquel on récupère des fichiers de résultats, et qui sont parfois disponibles sous forme de services web à intégrer dans une suite de traitements.

Les logiciels réalisés s'appuient sur une ou plusieurs manières d'appréhender des textes : des approches linguistiques (faisant référence à des connaissances linguistiques sur le fonctionnement de la langue) et/ou des approches statistiques (faisant appel à des modèles statistiques du comportement des mots dans les corpus étudiés), les deux étant souvent combinés. On oppose également les approches ascendantes, qui tirent des fragments lexicaux de corpus et essaient de les structurer et de les organiser pour faire des éléments de modèles, aux approches descendantes, de type « extraction d'information », où un cadre ontologique ou sémantique déjà défini permet de cibler les connaissances recherchées et d'orienter la recherche d'indices linguistiques de connaissances.

Enfin, on peut distinguer les logiciels en fonction des tâches qu'ils réalisent dans le processus de construction d'éléments d'ontologie. La liste de ces tâches est aujourd'hui clairement identifiée, bien que, selon les techniques, certaines ne soient pas systématiquement mises en œuvre ou qu'elles soient intégrées à d'autres. La première est la constitution d'un corpus. Avec la croissance des documents disponibles sur le web, beaucoup de chercheurs tendent à former les corpus de domaine en cherchant des pages analogues sur le web. Suivant la nature de l'ontologie à construire et celle du domaine, un corpus tiré du web peut éloigner des textes initiaux et de leur point de vue. En revanche, il permet en général de conduire des analyses statistiques de meilleure qualité. Au sein du corpus, les genres, les auteurs, les niveaux de langue et la qualité syntaxique, etc., jouent un rôle important sur la possibilité d'appliquer certaines techniques ou sur la nature des résultats.

L'identification de fragments ou de modèles préliminaires d'ontologie à partir de texte s'organisent en général selon les tâches suivantes :
- préparation du corpus par des traitements préliminaires (segmentation, étiquetage syntaxique, analyse des dépendances syntaxiques ;
- extraction de termes, éventuellement d'entités nommées et de la structure des termes ;
- regroupement des termes : regroupement des variantes de forme, des synonymes ou de termes formant des classes conceptuelles (*clustering*) ;
- organisation des classes en hiérarchies (en général, par la relation générique / spécifique mais aussi hiérarchies de parties à tout) ;
- recherche de relations sémantiques autres qui ont un intérêt pour le domaine et l'ontologie à concevoir ;
- organisation de ces classes et relations au sein d'un modèle (appelé modèle termino-conceptuel dans DAFOE ou TERMINAE) ;
- vérification de la qualité de ces données.

Nous distinguons par la suite les logiciels qui réalisent certaines de ces tâches indépendamment, des ateliers qui les mettent en oeuvre dans une perspective de modélisation terminologique ou ontologique, et enfin des ateliers plus généraux de TAL utilisables pour la construction d'ontologies.

Pour une dimension historique de l'évolution de ces logiciels, et pour des états de l'art plus complets sur l'anglais et le français, on consultera [12] [9] [4], [7] [26]. Nous présentons ici quelques logiciels de référence ainsi que des logiciels disponibles et opérationnels.

1.5 Les extracteurs de termes

1.5.1 Caractéristiques

Les extracteurs de termes ont pour but de fournir une liste de candidats à devenir des termes, liste accompagnée de critères pour sélectionner les plus pertinents. Les termes sont ici vus comme des traces linguistiques de concepts, et le fait d'étudier leur usage en corpus contribue à appréhender leurs différents sens dans le domaine étudié ainsi que les relations qu'ils entretiennent entre eux.

Trois éléments vont donc déterminer l'intérêt d'un extracteur de termes pour la construction d'ontologie ou pour l'identification des termes d'un domaine présents dans un document :
- l'algorithme selon lequel les termes sont identifiés, les langues auxquelles il s'applique ;
- les critères selon lesquels les termes peuvent être filtrés et les termes les plus pertinents étudiés en priorité ou sélectionnés pour former des concepts. Deux notions sont utilisées pour indiquer les critères indicateurs de la « force » d'un candidat terme : *termhood*, qui renvoie au fait que ce mot est à la fois important et représentatif du domaine, et *unithood*, qui évalue à quel point les mots formant le terme ont une bonne cohésion, plus forte que si l'on enlève ou ajoute un mot ;
- la possibilité d'exploiter les relations syntaxiques entre termes et leurs contextes d'usage pour les regrouper ou les mettre en relation (automatiquement ou manuellement).

1.5.2 Travaux de référence

Parmi les travaux de référence pour la langue française, ACABIT [17], FASTR [24] et SYNTEX [10] sont des systèmes encore maintenus et qui peuvent être obtenus auprès de leurs concepteurs.

● **ACABIT** extrait des candidats termes nominaux composés de deux mots à partir d'un corpus préalablement étiqueté et désambiguïsé. Ce programme réalise une analyse en deux étapes : 1) l'analyse linguistique et le regroupement de variantes consiste à appliquer un ensemble de transducteurs sur le corpus étiqueté pour extraire des séquences nominales et les ramener à des candidats termes binaires ; 2) le filtrage statistique consiste à trier les candidats termes binaires produits à l'étape précédente au moyen de mesures statistiques pour ne retenir que les plus pertinents. Différentes versions existent pour le français et l'anglais, faisant appel à un étiqueteur et à des règles propres à chaque langue.

● **FASTR** est un analyseur syntaxique robuste dédié à la reconnaissance en corpus de termes appartenant à une liste contrôlée fournie au système. Les termes n'ayant pas toujours, en corpus, la même forme linguistique, le principal enjeu est de pouvoir identifier leurs variantes. Ce système est donc souvent utilisé pour apprendre des variantes de termes, une fois établie une première liste. FASTR est doté d'un ensemble élaboré de métarègles qui lui permettent de repérer différents types de variation : les variantes syntaxiques, morpho-syntaxiques et sémantico-syntaxiques.

● **SYNTEX** est un analyseur syntaxique de corpus qui existe en deux versions, l'une pour le français et l'autre pour l'anglais. Ce système réalise une analyse syntaxique en dépendance de chacune des phrases du corpus : chaque mot est relié au mot dont il dépend (et le verbe principal à la phrase) par une relation étiquetée (SUJ, OBJ, DET, PREP, etc.). À partir de cette analyse, Syntex construit un réseau de mots et de syntagmes (verbaux, nominaux, adjectivaux), dit « réseau terminologique », dans lequel chaque syntagme est relié d'une part à sa tête et d'autre part à ses expansions. Les éléments du réseau (mots et syntagmes) sont appelés « candidats termes » (CT). À chaque CT sont associées des données quantitatives sur sa présence en corpus (fréquence mais aussi productivité, à savoir le nombre de CT dans lesquels il entre en composition). Un module complémentaire, UPERY [8], réalise une analyse des voisins distributionnels en corpus des candidats termes (et non des mots seuls). Les voisins d'un CT sont les CT qui dépendent au moins une fois des mêmes CT que lui. Les CT qui appartiennent aux mêmes voisinages sont regroupés en classes. La position des termes dans le « réseau terminologique » ainsi que leurs éventuels voisins distributionnels constituent des informations très utiles pour décider ou non de retenir un candidat terme et pour amorcer une structuration d'un réseau conceptuel. Une interface spécifique, TermOnto, permet d'ailleurs de sélectionner et valider des termes, de leur associer des concepts et d'élaborer un tel réseau.

1.5.3 Extracteurs de termes disponibles en ligne

Parmi les extracteurs de termes aujourd'hui disponibles en ligne, nous présentons Yatea [2], TermExtractor [30] et Termostat [18]. Nous mentionnons ici une plate-forme dédiée à la construction de terminologie, TERMINUS[4],

[4] http://terminus.upf.edu

proposée par l'université Pompeu Fabra, car elle comporte également un extracteur de termes, basé sur des critères statistiques, et un éditeur de fiches terminologiques. Les termes sont d'autant plus pertinents que leur fréquence est différente de leur fréquence dans un corpus de référence. L'accent est mis plus sur le choix des termes que sur leur structuration.

- **YaTeA**[5] est un extracteur de termes qui identifie des groupes nominaux candidats. Ce programme s'appuie sur une analyse syntaxique du corpus par TreeTagger[6]. Il repère des candidats termes selon des critères syntaxiques (à l'aide de patrons syntaxiques). Ces candidats sont ensuite désambiguïsés selon une stratégie hybride qui exploite à la fois des connaissances propres au corpus d'étude et des connaissances extérieures à ce corpus. Chaque candidat terme est analysé syntaxiquement pour faire apparaître sa structure sous la forme de têtes et de modifieurs. Les ressources linguistiques nécessaires à l'identification et à l'analyse des candidats termes sont fournies pour le français et l'anglais, et peuvent être modifiées par l'utilisateur. À chaque candidat est associé un poids révélateur de son importance en tant que terme.

- **TermExtractor**[7] est disponible en ligne pour extraire les candidats termes de fichiers en langue anglaise. Son originalité ne réside pas tant dans la manière de proposer des candidats termes (selon la combinaison de plusieurs techniques linguistiques) que dans celle de filtrer la liste ainsi obtenue. Plusieurs critères sont pris en compte pour mettre en avant certains termes et des heuristiques permettent d'en rejeter d'autres : pertinence par rapport au domaine (le terme est plus fréquent dans le corpus que dans un corpus de langue générale), consensus dans le domaine (le terme a une distribution équiprobable dans les différents documents du corpus), cohésion des termes (qui reflète le critère *unithood* mentionné plus haut : les mots formant un terme sont essentiellement présents dans ce terme et ne sont pas utilisés isolés) et fréquence. De plus, on peut indiquer des types de mise en forme (niveau de titre, casse) qui vont augmenter le poids des mots qui auront cette mise en forme. Le poids global d'un terme est une combinaison pondérée de ces critères selon des poids ajustables au corpus.

- **Termostat**[8] acquiert automatiquement des termes à partir de corpus en exploitant une méthode qui met en opposition des corpus spécialisés et non spécialisés en vue d'identifier les termes les plus représentatifs du corpus et du domaine (critère de pertinence dans TermExtractor). La version disponible en ligne de ce programme prend en charge le français, l'anglais, l'espagnol, l'italien et le portugais. TermoStat reçoit un texte en entrée et retourne comme résultat principal une liste de candidats termes (CT) tirés du texte. Un terme peut être simple (un mot) ou complexe (une suite de mots). L'utilisateur peut fixer la nature grammaticale des termes recherchés. Les termes sont extraits à l'aide de patrons syntaxiques qui caractérisent les structures de surface attendues pour les termes (de type « Nom, Nom Nom », « Adj Nom » ou « Nom Adj », etc.). La liste des patrons est adaptée à la nature grammaticale des termes recherchés. Pour faciliter la sélection parmi tous les candidats retournés, chaque terme reçoit un score basé sur sa fréquence dans le corpus analysé, le corpus d'analyse (CA), et sa fréquence dans un autre corpus prétraité, un corpus de référence (CR).

Les travaux de Patrick Drouin et Philippe Langlais [19] sur la notion de *termhood* les ont conduits à définir plusieurs scores statistiques qui sont combinés ou non dans Termostat. Le plus simple est la fréquence, qui est associée à d'autres critères (spécificité, X^2, log-likelyhood, etc.) exploitant les différences de fréquence des termes étudiés et des autres termes dans CA et CR. Après évaluation, la fréquence et la *log-likelyhood* semblent les meilleurs critères. Comme YaTeA et Syntex, Termostat gère les relations syntaxiques entre les termes (relations en tête et expansion des syntagmes complexes) qui permettent de « visualiser » les modifieurs d'un terme, ses composants, etc. Une interface web permet d'appliquer Termostat sur des corpus, d'en récupérer des listes de termes avec leurs scores et d'en valider une partie en fonction du score mais aussi des relations syntaxiques.

1.5.4 Extraire les entités nommées

L'extraction d'entités nommées, à savoir d'unités textuelles particulières comme les noms propres (personnes, lieux, organisations) mais aussi les expressions temporelles (dates, durées, horaires) et les noms de quantités (monétaires, unités de mesure, pourcentages), consiste à les repérer en corpus mais aussi à leur associer un type sémantique. De nombreux systèmes d'extraction d'information incluent cette tâche dans la mesure où l'extraction cherche à mettre en relation des entités dont certaines sont souvent des entités nommées (par exemple des personnes travaillant ensemble ou des noms de maladies, de gênes, etc.).

Les systèmes classiques recherchent souvent des types « classiques » comme des noms de personnes ou de lieux, ou des dates. Des applications récentes ont rendu la tâche plus complexe en traitant cent à deux cents types. Dans le contexte de construction d'ontologies et d'annotation sémantique, la recherche d'entités nommées relève du peuplement d'ontologies ainsi que de l'annotation, puisqu'il s'agit d'associer un type à des entités spécifiques, ou de « remplir » l'ontologie par ces entités spécifiques comme les instances des classes représentant les types. Mais cette recherche peut aussi permettre d'identifier des concepts spécifiques

---

[5] http://www-limbio.smbh.univ-paris13.fr/membres/hamon/YaTeA

[6] http://www.ims.uni-stuttgart.de/projekte/corplex/TreeTagger

[7] http://lcl.di.uniroma1.it

[8] http://olst.ling.umontreal.ca/~drouinp/termostat_web/ ;
http://olst.ling.umontreal.ca/~drouinp/termostat_web/doc_termostat/doc_termostat.htm

représentés comme des sous-classes de concepts existants. Dans ce cas, l'extraction d'entités nommées contribue à la construction d'ontologie.

Parmi les logiciels existants, nous retenons les modules d'extraction d'entités nommées des plates-formes de TAL GATE[9] et d'extraction d'information KIM. Au sein de GATE, c'est le module d'extraction d'information ANNIE qui réalise l'extraction d'entités nommées. Ce logiciel s'appuie sur des transducteurs qui appliquent des heuristiques linguistiques propres à la recherche d'entités nommées. La tâche délicate est alors de résoudre des co-références, c'est-à-dire d'être capable, par exemple, de considérer comme synonymes un sigle et sa forme développée, ou de décider si deux mentions d'un même texte renvoient ou non à la même entité. ANNIE gère des listes de termes par type sémantique (des *gazeteers*) qui sont enrichies au fur et à mesure des activités.

1.6 Organiser les termes en classes ou en hiérarchies

L'organisation des termes en classes, et de ces classes en hiérarchies, vise différents objectifs. La facette la plus linguistique est d'associer des variantes de forme, comme avec FASTR, ou des sigles et leurs formes développées. Une étape plus élaborée est d'associer des termes supposés synonymes, et de choisir le terme le plus représentatif de cette classe pour la nommer. Les divers algorithmes de *clustering* peuvent s'adapter à l'étude des termes. La difficulté est d'identifier des indices en corpus qui vont permettre de supposer que des termes peuvent faire partie de la même classe.

L'hypothèse distributionnelle de Harris [22] suppose que les mots ayant une même distribution d'utilisation en corpus, qui donc partagent les mêmes contextes, ont des chances de faire partie d'une même famille, ou d'être en relation hiérarchique, l'un subsumant les autres. La qualité de la caractérisation du contexte est alors déterminante pour former de bonnes classes. Une méthode élémentaire est de rechercher des séquences de *k* mots consécutifs communs dans les voisins de termes à comparer (*k* plus proches voisins). Cette méthode, qui s'appuie sur des critères de surface et ne nécessite donc que des traitements préalables légers, comme une lemmatisation, produit cependant de bons résultats pour des termes peu fréquents et permet de gérer des termes composés (et pas seulement des mots). L'étude des co-occurrences sur une fenêtre de cinq à dix mots est également un bon critère si le corpus est assez volumineux. Cependant, la nature de la relation entre les termes en collocation est très difficile à identifier, et ne peut pas systématiquement être interprétée comme une relation de hiérarchie entre classes [14].

Si une analyse de relations syntaxiques a été réalisée (cas de Syntex), les contextes peuvent être beaucoup plus précis : mêmes termes voisins et mêmes relations de dépendance. Mais, dans ce cas, sur des corpus de petite taille, le nombre de classes trouvées ainsi est marginal (de l'ordre de la dizaine pour près de dix mille termes retrouvés et quelques centaines de concepts retenus [9]. L'approche est intéressante sur des corpus de grande taille tirés du web ou de collections éditoriales de journaux ou revues. Des travaux pionniers ont utilisé comme contexte les frames associés aux verbes et ont regroupé les termes jouant les mêmes rôles par rapport aux mêmes verbes.

Ainsi, ASIUM [20] réalise une classification ascendante itérative des noms qui apparaissent dans les mêmes contextes de verbes. À chaque étape, ce système propose de regrouper les extensions les plus similaires des arguments de deux verbes. L'utilisateur intervient pour valider chacune des propositions et éviter des erreurs. ASIUM forme ainsi non seulement une hiérarchie de concepts, mais aussi une généralisation des catégories sémantiques acceptées comme arguments par ces verbes. D'autres voisinages utilisés peuvent être les appositions de noms ou les relations entre noms. L'étiquetage des classes peut se faire en recherchant en corpus des marqueurs de relations de classe / sous-classe, comme le patron « X est un Y qui ... » ou les patrons de Hearst [23] entre les termes d'une même classe.

Enfin, des approches s'appuient sur l'apprentissage ou des statistiques d'utilisation des termes dans des corpus volumineux. Dans ce cas, l'analyse statistique de matrices de co-occurrences de termes produit des résultats très pertinents pour retrouver des termes synonymes. Buitelaar, Volker et Cimiano [12] citent, par exemple, la méthode de réduction de matrices et de calcul de vecteurs propres *Latent Semantic Indexing* et ses variantes (LSI, LSA, PLSI) pour faire ressortir des relations implicites entre mots qui permettent de former des groupes. Bien sûr, ces méthodes sont beaucoup plus compliquées à mettre en œuvre si l'on veut regrouper des termes composés. Mais, avec l'essor des documents numériques sur le web, elles connaissent un fort succès.

Cimiano, Hotho et Staab [14] ont testé et comparé plusieurs techniques, telles que l'Analyse formelle de concepts ou FCA (*Formal Concept Analysis*) en exploitant les relations sujets / verbes ou objets / verbes, Bi-section-Kmeans, ou des approches faisant intervenir des mesures de similarité. L'analyse formelle de concepts est basée sur l'identification d'attributs et d'objets, et le regroupement des objets en classes en fonction des attributs communs, pour former un treillis. Les objets classés sont ici des termes et leurs attributs sont les verbes dont ils sont arguments. Une fois le treillis appris à partir de corpus, il est restructuré en ontologie. Les résultats sont d'autant meilleurs que des regroupements de termes basés sur leur forme sont faits avant le calcul du treillis. Les attributs regroupés qui caractérisent une classe permettent de définir des concepts intermédiaires dans la hiérarchie, comme l'ensemble des objets « mangeables et cuisinables » qui définirait la classe des aliments. Cette méthode s'avère beaucoup plus productive que les mesures de similarités entre contextes, par exemple.

[9] http ://gate.ac.uk

Un dernier type de travaux utilise des ressources externes (WordNet, DBpedia) et des modules de désambiguïsation de sens adaptés à ces ressources pour reprendre les relations hiérarchiques présentes dans ces ressources entre des termes extraits des corpus. Avec la mise à disposition d'ontologies plus nombreuses sur le web, et la disposition de moteurs de recherche d'ontologies, l'approche retenue pour définir SCARLET consiste à rechercher une ontologie mettant en relation (directe ou non) deux concepts donnés. Alors les relations trouvées sont ajoutées à l'ontologie à construire. Cette méthode, séduisante sur le principe, fonctionne assez mal pour plusieurs raisons : la plupart des ontologies étant en anglais, les termes fournis doivent être en anglais. De plus, si on n'introduit pas un module de désambiguïsation, le système peut proposer différentes ontologies de domaines différents. Enfin, il existe rarement une ontologie contenant à la fois deux concepts fixés.

De l'ensemble de ces techniques et travaux, il est difficile d'extraire des logiciels disponibles et faciles à utiliser. Cependant, nous retenons le système TaxoLearn[10] [28], accessible en ligne depuis 2011. Ce logiciel utilise l'algorithme Word Class Lattices qui, pour un terme donné, recherche des contextes définitoires et un hyperonyme dans ce contexte. Il enrichit si besoin le corpus avec des pages trouvées à l'aide de Google:define. Chaque définition extraite produit une partie de graphe, et l'analyse de l'ensemble du corpus revient à extraire des graphes. Cette extraction réalise simultanément l'extraction de termes, de définitions et d'hyperonymes. TaxoLearn amorce le processus par le résultat d'un extracteur de termes (TermExtractor) sur le corpus. Puis il extrait des graphes qui mettent ces termes en relation avec des hyperonymes. Il utilise l'algorithme SSI pour la désambiguïsation des termes et Wordnet++ comme ressource sémantique. Le premier résultat produit sont des graphes déconnectés qui, une fois nettoyés du bruit qu'ils contiennent et associés via Wordnet++, forment une taxonomie lexicale. L'élimination du bruit dans les graphes se fait sur la base de propriétés topologiques du graphe et de principes généraux de structuration d'une taxonomie. Elle permet de produire une hiérarchie. L'évaluation de TaxoLearn sur deux corpus différents a montré que la qualité du résultat dépend avant tout du volume et de la qualité des données analysées. C'est un des systèmes les plus performants en la matière.

1.7 Extraction de relations

L'extraction des relations entre concepts est une généralisation de l'identification de taxonomies ou de hiérarchies de concepts, qui revient à chercher des relations « est-un » ou d'hyperonymie entre termes, et de considérer que ces termes sont les labels de concepts. À partir des travaux de Hearst en 1992, de nombreux systèmes ont mis en œuvre une approche à base de patrons lexico-syntaxiques pour la recherche de relations [26]. Un patron caractérise le contexte linguistique dans lequel on peut interpréter qu'il existe une relation sémantique entre deux termes. Les patrons de Hearst permettent d'identifier des relations d'hyperonymie et d'organiser ensuite des concepts dans une hiérarchie de classes. Ils sont toutefois peu productifs et ne couvrent qu'une partie des manières d'exprimer l'hypéronymie. La mise au point de patrons propres à chaque corpus étudié, ou à chaque type de relation, est souvent indispensable pour produire des résultats significatifs. C'est ce que proposent les systèmes Prométhée et Caméléon pour le français[11], l'un par apprentissage à partir d'exemples, l'autre par observation manuelle des contextes dans lesquels des termes en relation sont identifiés.

La mise au point de patrons étant laborieuse, l'idée est d'une part de les capitaliser et de les réutiliser. Or, à ce jour, il n'existe pas de format standard de représentation de patron, ni de lieu où ceux-ci sont systématiquement mis à disposition. Certains travaux se proposent donc d'apprendre les patrons à partir d'exemples sur corpus étiquetés. D'autres font appel à des algorithmes de classification pour extraire des relations entre concepts, avec les limites que nous avons énoncées plus haut.

Pour mettre en œuvre les approches par patrons, on peut utiliser aujourd'hui des plates-formes de traitement automatique des langues qui permettent de définir des expressions régulières ou des transducteurs, comme la plate-forme GATE[12], Open Calais, mais aussi LinguaStream[13], Nooj ou Unitex. On trouvera des listes de patrons réutilisables dans les travaux de Marshman [27], Séguéla [31] ou Auger et Barrière [3].

Nous aurions pu présenter TaxoLearn dans cette partie. Un autre logiciel disponible en ligne et particulièrement intéressant pour la recherche de relations est Terminoweb[14] [6]. C'est un support complet à la réalisation de terminologies, qui permet de construire des listes de termes validés à partir de corpus extraits du web. Reprenant la notion de *knowledge rich context* de I. Meyer, ce système propose de chercher des concepts et des relations sémantiques dans les textes qui ont le plus de chances d'en contenir, et qui sont pertinents pour le domaine à modéliser.

TerminoWeb se présente comme un logiciel disponible en ligne dans lequel on peut gérer des projets de construction de terminologies en français ou en anglais. La première étape est de fournir un ensemble de mots représentatifs du domaine pour amorcer la recherche de documents. Le système cherche alors des textes qui contiennent ces termes et des patrons définitoires. C'est un peu l'idée présente dans TaxoLearn, mais on n'extrait pas encore de connaissances au niveau de la phrase. Une fois le corpus constitué, le système applique un

[10] http://lcl.uniroma1.it/taxolearn

[11] Logiciels aujourd'hui non maintenus, dont on trouvera les références dans [9].

[12] http://gate.ac.uk

[13] http://www.linguastream.org

[14] http://terminoweb.iit.nrc.ca/TE.html ; http://terminoweb.iit.nrc.ca/terminoWeb-v2_f.html

algorithme d'extraction de termes qui peut aussi tenir compte de la mise en forme et de titres. Les termes sont consultables en contexte et ont un poids de pertinence. L'étape suivante est la recherche de relations à l'aide de patrons. Le système propose une liste de relations sémantiques et, pour chacune d'elles, des patrons prédéfinis. Mais il est possible d'ajouter des relations et des patrons propres au corpus. La projection des patrons en corpus retourne des phrases et permet de définir manuellement de nouveaux termes, et de les mettre en relation. On ne parle pas ici de concepts, ce serait une étape suivante dans le processus. Mais l'esprit est le même.

Enfin, nous mentionnerons Text2Onto [16], qui est un des systèmes les plus aboutis en la matière. Text2Onto est une évolution de Text-to-Onto, un des modules de la suite de logiciels KAON pour la gestion d'ontologies. Ce système facilite la construction d'ontologie à partir de textes à l'aide d'une analyse de surface du corpus combinée à de l'apprentissage. Les termes sont extraits selon des critères de fréquence et de distribution des mots simples. Les relations hiérarchiques entre concepts et les autres relations sémantiques sont extraites à l'aide de règles d'association ou de patrons syntaxiques. Les relations sont pondérées selon la confiance avec laquelle le patron a été reconnu en corpus. Le modèle ainsi obtenu est ensuite filtré pour éliminer les termes trop ou trop peu fréquents, en comparant les fréquences dans le corpus analysé et dans un corpus de langue générale. Les résultats obtenus forment un modèle d'ontologie probabiliste (POM), qui doit être ensuite validé et corrigé avant de constituer l'ontologie finale. Dans sa dernière version, Text2Onto examine également les évolutions du corpus d'étude pour y détecter des changements et suggérer ainsi de faire évoluer l'ontologie.

Text2Onto est disponible comme un module indépendant ou comme un *plug-in* de la plate-forme NEON Toolkit de gestion d'ontologie.

1.8 Ouvertures

Les perspectives sont nombreuses pour rendre ces travaux plus performants mais aussi plus accessibles.

Au niveau de la qualité des résultats, l'exploitation de grands corpus choisis pour leur qualité est sans doute une perspective prometteuse. À ce titre, exploiter le web dans l'esprit du web des données peut orienter les travaux de manière intéressante. Mais d'autres perspectives plus techniques consisteraient à exploiter différents niveaux d'étiquetage des corpus, de dépasser l'analyse de la phrase pour aller vers celle de plusieurs phrases consécutives, de prendre en compte des éléments liés au discours, de mieux prendre en compte la structure des documents, de recherche des relations n-aires, etc.

Cependant, dans une perspective d'appropriation du web par ses utilisateurs, on est encore loin de services facilitant la construction d'ontologies de domaines (tâche qui, de toute façon, demande du recul sur le domaine) ou de services qui facilitent la description de contenus à l'aide de ces ontologies. Il y a un besoin évident de logiciels opérationnels et éprouvés, facilement accessibles pour la structuration des connaissances et leur identification en corpus. Les logiciels disponibles en ligne montrent qu'une évolution est en cours et que de nouveaux systèmes seront sans doute bientôt disponibles.

2 « Donner du sens » à des contenus : l'annotation sémantique

Le processus qui nous intéresse ici est celui qui consiste à produire des représentations formelles à partir de données linguistiques (des textes au sein de documents) afin de permettre à un programme informatique de manipuler ces documents en fonction de leur contenu, voire de traiter ces contenus. Nous appelons ce processus « annoter sémantiquement les contenus ». Cette annotation répond à un paradoxe : on souhaite à la fois rester au plus près des contenus linguistiques et produire des représentations conformes à des standards et prenant du sens pour d'autres, donc si possible utilisant un langage partagé, une ontologie.

Dans cette partie, nous reviendrons d'abord sur les différentes réalités que couvre le mot « annotation sémantique », avant de présenter les principales étapes du processus tel que nous le définissons. Nous évoquerons ensuite les ressources ontologiques et terminologiques pertinentes pour ce processus, ainsi que les modèles d'annotation. Enfin, nous présenterons quelques réalisations significatives et opérationnelles. Nous terminerons par des perspectives d'actualité : des logiciels utilisant les données liées pour rendre compte de contenus linguistiques.

2.1 Associer des données et des modèles sémantiques

Parmi les modalités envisagées pour parvenir à un web sémantique opérationnel figure l'idée, mise en avant par Tim Berners-Lee, de ne pas modifier les pages existantes mais de leur ajouter une description sémantique, manipulable par des logiciels. Cette couche sémantique serait, *a minima*, un ensemble de métadonnées et, au mieux, une représentation sémantique en lien avec une ontologie. Si le continuum qui existe entre métadonnées et annotations sémantiques est bien réel, il nous faut examiner leurs différences pour mieux comprendre les processus liés à leur production.

Les métadonnées sont en général produites par l'auteur ou une personne autre, dans le but de retrouver facilement le document, de le situer par rapport à d'autres documents. En général celui qui choisit les mots-clés formant les métadonnées connaît les utilisateurs potentiels ou a en tête des utilisateurs particuliers ; il s'adapte à leur point de vue et à leurs mots-clés autant qu'au contenu du document lui-même pour étiqueter celui-ci.

Souvent les métadonnées prennent du recul par rapport au contenu, et il est rare de retrouver parmi les mots-clés des mots présents dans les documents.

L'annotation, elle aussi, est produite soit par l'auteur, soit par une autre personne à destination d'un type particulier de lecteur. Il peut s'agir de commentaires, de renvois, d'explications, d'évaluations, d'actions à réaliser, etc., destinés à faciliter la lecture ou la relecture, à mieux exploiter le document lu, etc. Dans le cas d'une annotation formelle, le vecteur visé est l'application informatique, et son utilisateur. L'annotation doit être pertinente par rapport aux objectifs du système à concevoir. De ce fait, mais surtout parce que la langue est sujette à de multiples interprétations, l'annotation revient à fixer une interprétation. La restitution des contenus n'est pas complètement neutre, et l'ontologie utilisée pour annoter est, que ce soit voulu ou non, le moyen de fixer ce point de vue.

2.1.1 Qui doit annoter et quand ?

Une des craintes de l'échec du web sémantique était la difficulté d'annoter des documents. Or, avec le web 2.0, les internautes ont montré que, pour leur intérêt personnel (être lu, connu, diffusé, commenté, etc.) ou pour l'intérêt collectif (capitaliser, comparer, récupérer des données ou des résultats), ils n'hésitaient pas à annoter leur production, autant que celles des autres. Tous ces tags ont une multitude de statuts et l'on trouve aussi bien des données descriptives proches du contenu, des commentaires, des caractérisations de haut niveau, que des métadonnées. Les tags sont autant produits par les auteurs que par les lecteurs. Cependant, dans le cas d'applications plus élaborées, les annotations sémantiques sont produites par celles qui utilisent des pages web, et exploitent des ressources soit génériques, soit reflétant le point de vue pertinent pour l'application visée.

2.1.2 Annotation sémantique en linguistique vs pour le web sémantique

Utilisée en linguistique et en traitement automatique des langues avant d'être reprise pour le web sémantique, l'expression « annotation sémantique » prend un sens différent dans chacun de ces domaines[15], ce qui donne lieu à des annotations de nature différente et à des processus différents. En linguistique, on peut entendre par là soit le fait de poser des relations sémantiques entre les éléments d'une phrase (relations agent / patient d'une action, par exemple), soit le fait d'annoter les mots de la phrase par leur sens dans ce contexte.

La première interprétation correspond à un étiquetage des rôles sémantiques, et s'appuie sur la structure prédicative des verbes. Ce processus se situe à l'articulation entre syntaxe et sémantique. La seconde interprétation correspond à l'identification d'unités de sens (un ou plusieurs mots) et à la désambiguïsation de leur sens, c'est-à-dire à l'identification de la bonne classe sémantique à associer à un terme. Il s'agit d'étiqueter par le sens (*sense tagging*) à partir d'un dictionnaire ou d'une base de données lexicales comme WordNet. Cette tâche est difficile non seulement parce que les mots sont polysémiques ou ambigus, mais aussi parce que plusieurs niveaux d'abstraction peuvent être choisis. Une variante de ce type d'annotation consiste à faire appel à une ressource propre à un domaine pour désambigüiser. Dans ce cas, on annote les mots par des concepts d'un domaine, ce qui correspond tout à fait à l'annotation sémantique telle qu'on l'entend en utilisant une ontologie de domaine. On parle parfois d'annotation par le domaine. Il y a en général moins de polysémie et d'ambiguïtés possibles pour trouver le type d'un terme donné, et un processus automatique est envisageable.

Un troisième point de vue est celui de l'extraction d'information, où l'on s'intéresse seulement à quelques classes sémantiques, qui ne forment pas un modèle à proprement parler et pour lesquelles on dispose de lexiques correspondant aux termes renvoyant à chaque classe. Souvent celles-ci (personnes, lieux, entreprises) renvoient à des noms propres, à des dates ou à des quantités, et l'on retrouve le problème de l'identification d'entités nommées.

Les enjeux applicatifs de ces différents types d'annotation sont nombreux. Pour celui qui nous intéresse, à savoir la description des contenus documentaires, c'est l'annotation par le sens, et même par le domaine, que nous appelons annotation sémantique. Cependant, l'identification de ces annotations peut encore renvoyer à deux significations : étant donné un mot dans son contexte, on peut considérer l'annotation au niveau de la classe conceptuelle qui lui correspond ou à celui d'une instance de cette classe. On trouve en effet des systèmes d'annotation sémantique qui créent une classe nommée lorsqu'ils ont identifié une instance de concept, ou des systèmes qui restent au niveau de la classe.

Finalement, on peut considérer divers niveaux d'annotation sémantique : annotation par des instances ou des classes, par un domaine sémantique, par un champ sémantique, par un rôle sémantique.

Nous considérons désormais que, lorsqu'une annotation sémantique est produite, elle fixe un point de vue sur des documents, tout comme une ontologie fixe un point de vue sur le monde. L'ontologie qui a servi à produire les annotations influence en grande partie ce point de vue. Le résultat de l'annotation sémantique rend compte du contenu textuel de manière formelle, partielle et partiale, et c'est elle qui sera désormais manipulée, en lieu ou en complément du texte d'origine, par les applications du web sémantique.

[15] Voir la thèse d'Antonio Pareja Lora [29].

2.2 Démarche générale

L'annotation sémantique peut être réalisée manuellement, par exemple en vue de produire des données pour entraîner un système d'apprentissage qui produira des règles pour annoter ensuite automatiquement d'autres textes, ou automatiquement.

Il est classique de distinguer les étapes suivantes :
- prétraitement des textes par des analyseurs ;
- reconnaissance de termes ou de syntagmes ayant une unité ;
- projection / appariement des textes et de la ressource servant à annoter (il s'agit de reconnaître les concepts de la ressource à partir des termes / entités du texte) ;
- choix du concept le plus pertinent (en général le plus spécifique) parmi ceux possibles pour annoter, et éventuellement reconnaissance de relation si l'annotation rend compte aussi des relations ;
- enregistrement de l'annotation au format prévu par le système (sous forme de concept ou d'instance, par exemple, mais aussi de graphe d'instance plus complexe dans le cas où l'on veut rendre compte non seulement des concepts mais aussi des relations qu'ils entretiennent dans la phrase, etc.) ;
- visualisation des annotations : présentation à l'utilisateur pour validation (éventuellement). Lorsqu'elle se fait dans le document, cette visualisation utilise souvent des jeux de couleurs associés aux classes sémantiques. La visualisation peut être à côté du document ou présentée à la demande, ou même implicite et cachée.

Les deux étapes du processus qui nous intéressent sont la reconnaissance des termes puis l'identification des bons concepts.

Pour la reconnaissance des termes, tous les algorithmes et logiciels d'extraction peuvent s'appliquer ici. On peut même avoir besoin de critères de sélection de *termhood* si l'on ne veut annoter que les termes les plus importants d'un document, et non tout son contenu ; on a besoin des critères d'unité des termes pour savoir où sont les frontières des syntagmes à annoter, s'il vaut mieux annoter un ou plusieurs mots, etc. Donc tous les travaux sur l'extraction de termes, présentés en première partie, se retrouvent dans le processus d'annotation. Si on annote par les relations, les travaux sur l'identification de relations sont également concernés. Dans le contexte local d'une phrase, seule des approches par patron peuvent s'appliquer ici.

La reconnaissance des bons concepts revient à typer les termes et, là encore, renvoie à des questions soulevées par l'identification de classes et l'organisation des termes en classes dans la première partie, lorsque nous avons présenté les systèmes de structuration des termes et de définition de concepts. Les étapes classiques de ce processus sont : désambiguïser, et donc identifier les classes possibles et, parmi celles-ci, la plus pertinente ; puis reconnaître s'il s'agit plutôt d'une classe ou d'une instance.

Toutefois, pour réaliser ces étapes, les méthodes et algorithmes peuvent varier et correspondre à des procédés très différents : reconnaissance de chaînes de caractères via les labels, calcul de distances sémantiques, application de règles apprises à partir de corpus d'exemples déjà étiquetés, etc. Ces algorithmes peuvent partir des termes du corpus pour rechercher les classes présentes dans l'ontologie ou, inversement, partir de l'ontologie et la projeter sur le corpus. Le premier choix revient à balayer l'ensemble des termes du corpus, donc un index du corpus, avec chacun des labels (ou identifiants ?) de l'ontologie ; le deuxième à parcourir l'ontologie pour chacun des termes du corpus. Selon le volume du corpus et la nature de l'ontologie (ontologie de domaine *vs* WordNet, par exemple), l'une ou l'autre des solutions peut être plus pertinente. Mais, de manière générale, on projette plutôt l'ontologie sur le corpus.

Les algorithmes d'annotation sémantique doivent faire des propositions, des choix pour réaliser les trois tâches suivantes :
- identifier les termes du corpus : c'est un problème d'extraction de terme ;
- étant donné un terme du corpus et un label d'un concept de l'ontologie, décider si ces deux chaînes de caractères sont proches, égales, complètement différentes, et donc si elles renvoient ou non au même concept et dans quelle mesure, avec quel degré de confiance : c'est un problème de calcul de distance entre chaînes de caractères (distance d'édition) ;
- une fois un ou plusieurs concepts identifiés comme pouvant annoter un terme, identifier le plus pertinent : c'est à la fois un problème de désambiguïsation de sens (quel est le sens du terme dans la phrase à annoter) et un problème de choix du bon concept d'annotation (une fois le sens valide identifié, et étant donné le modèle d'annotation, voire l'algorithme de recherche d'information qui va exploiter les annotations, vaut-il mieux annoter avec une instance ou avec le concept ? avec ce concept ou un concept plus précis ou un concept plus générique ?).

Le principe de l'annotation sémantique étant de rendre compte de manière fine des contenus, de nombreux systèmes prévoient une annotation par les instances ou par les classes les plus précises.

2.3 Quelques logiciels d'annotation sémantique disponibles

● *Annotation sémantique par des classes sémantiques.* Nous avons déjà évoqué le module ANNIE[16] qui procède par extraction d'information et recherche d'entités nommées [figure 3]. Il peut être associé comme un *plug-in* à NEONToolKit. Ce système permet d'annoter avec des classes sémantiques qui ne forment pas une ontologie,

[16] http://gate.ac.uk/sale/tao/splitch6.html#chap:annie

mais aussi avec certains concepts d'une ontologie de domaine. Pour ces concepts, il faut fournir une liste de termes qui forment le lexique. Le module d'annotation par les entités nommées de la plate-forme KIM joue un rôle similaire, mais il n'est pas possible d'utiliser une ontologie.

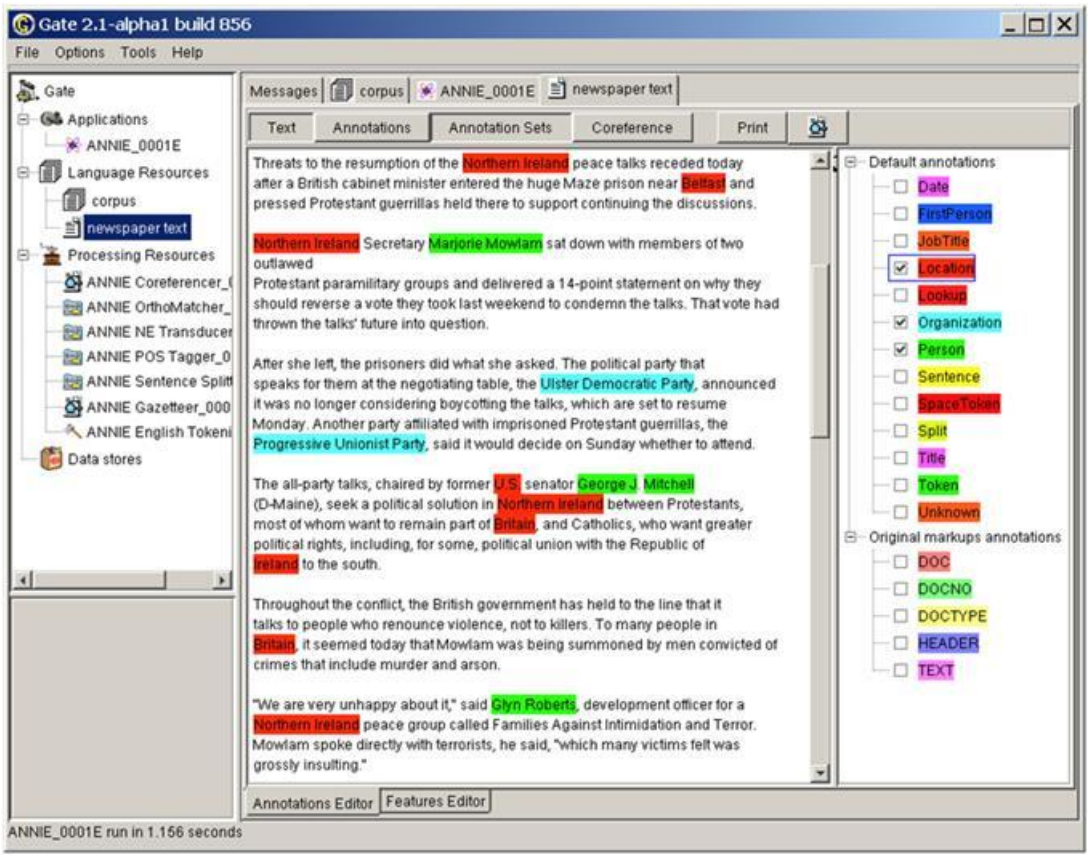


Figure 3 – L'interface d'ANNIE : résultat de l'annotation d'un texte, avec les concepts sélectionnés sur la droite

● *Annotation sémantique spécialisée en médecine ou en génomique.* Nous citerons un logiciel du site du NCBO et celui de l'INRA.

● *Annotation sémantique avec des ontologies spécialisées.* De nombreux systèmes ont été développés, parmi lesquels TextAnnot, Magpie et PANKOW.

PANKOW (Pattern-based ANnotation through Knowledge On the Web) réalise une annotation sémantique non pas sur la base d'un calcul de distance entre chaînes de caractères, mais à l'aide de patrons d'annotation [15]. Ce système a été conçu pour faciliter le travail d'un annotateur humain en lui suggérant les bons concepts d'une ontologie donnée, à associer aux instances présentes sur une page web. Pour cela, PANKOW génère des instances de patrons lexico-syntaxiques adaptées à la recherche de certaines relations sémantiques. Il compte les occurrences de ces patrons sur le web à l'aide de Google API afin de décider de la pertinences de ces patrons. La distribution des occurrences de ceux-ci permet ensuite à l'annotateur de prendre une décision sur le choix du concept le plus pertinent pour annoter le terme qu'il est en train de traiter. Les données du web tiennent lieu de connaissances collectives sur la manière dont un terme a été interprété dans un contexte proche de celui dans lequel il apparaît en corpus. Cette aide est précieuse dans le cas où l'annotateur rencontre des termes ou des instances inconnues.

TextAnnot[17] est le fruit de deux projets ANR[18] nationaux. Il permet une annotation sémantique par un graphe d'instances de concepts d'une ontologie donnée et proche du corpus. L'ontologie est représentée selon un format particulier : c'est une ressource termino-ontologique, à savoir une ontologie avec une composante terminologique riche qui permet de stocker, pour chaque concept, ses différentes réalisations linguistiques en corpus. Le résultat de l'annotation est un graphe formé des instances des concepts reconnus en corpus et mis en relation selon les connaissances présentes dans l'ontologie. TextAnnot utilise le logiciel de recherche d'information Lucène pour construire un index du corpus à annoter, et se base sur la reconnaissance de termes associés aux concepts de

[17] http://www.irit.fr/TextViz

[18] Agence nationale de la recherche

l'ontologie pour projeter celle-ci sur le corpus et générer les annotations. Les textes du corpus sont annotés chacun par un ou plusieurs graphes. TextAnnot est utilisable en ligne pour un corpus et une ontologie fixés. Il sera bientôt paramétrable de manière à charger le corpus et l'ontologie souhaités. Un service web assurant le processus d'annotation est également disponible.

2.4 Conclusion

Les travaux sur l'annotation sémantique progressent et des logiciels pour prendre en charge tout ou partie du processus sont disponibles. Comme pour l'extraction de connaissances pour la construction d'ontologies, peu de ces logiciels sont de niveau professionnel (la plate-forme KIM ou ANNIE).

3 Conclusion. TAL et données liées

Apprendre, réutiliser, construire, enrichir, faire évoluer... Si l'on observe les contextes dans lesquels les textes sont utilisés comme sources de connaissances pour construire des ontologies, on constate que les techniques d'analyse du langage se prêtent bien à un processus de fouille, dans lequel on complète, on enrichit ou on instancie une ontologie avec de nouvelles connaissances, à conditions que celles-ci ne viennent pas remettre en question les connaissances existantes (très peu de travaux prévoient d'automatiser ce cas). Dans le cas de corpus spécialisés et de taille volumineuse, l'apprentissage automatique permet de développer des outils *ad hoc* pour apprendre de nouvelles connaissances. Mais, pratiquement, la mise au point de logiciels performants reste complexe et, en dehors de travaux universitaires, il existe peu d'offres disponibles et publiques. De plus, l'utilisation de ces logiciels et l'interprétation ou la modélisation de leurs résultats requiert des compétences en ingénierie des connaissances, des ontologies ou des documents numériques.

3.1 Finalement, a-t-on vraiment besoin d'ontologies à l'heure du web des données ?

Il semblerait que oui ! Prenons pour exemple deux logiciels annoncés début 2012. Ils extraient « directement » des triplets à partir de textes et sont capables d'associer des termes à des catégories sémantiques; Il n'est pas demandé de construire une ontologie pour cela. Cependant, si l'on regarde de près leur description ci-dessous, on constate que ces classes sémantiques sont empruntées à des ontologies génériques validées et reconnues. Les types associés aux triplets seront ceux de ces ontologies. Ce choix favorise la réutilisation, l'interopérabilité, mais ne permet pas de rendre compte de points de vue spécifiques ou précis sur des contenus. On pourrait imaginer une approche analogue s'appuyant sur des ontologies de domaine. Mais, dans ce cas, les triplets générés ne seront sans doute pertinents que pour une ou des applications ciblées.

Ces deux logiciels ont été mis récemment à disposition publique par A. Gangemi et ses collègues V. Presutti, F. Draicchio, A. Musetti, A. Nuzzolese. Ils témoignent des nouvelles tendances utilisant le TAL pour l'identification de données liées et de ressources sémantiques. Ils offrent une vision intéressante de ce que pourrait être un web sémantique intégrant le web des données, ou un web des données rendu plus pertinent par le web sémantique. Ces travaux illustrent également le fossé qui se creuse entre l'anglais, langue pour laquelle il existe de nombreux outils et ressources réutilisables, et les autres langues, pour lesquelles des développements importants restent à réaliser. Les deux logiciels proposés sont complémentaires : FRED est dédié à l'extraction de triplets à partir de textes, et Tipalo assure l'organisation sémantique de ces triplets en les associant à des classes sémantiques et en les connectant au sein de graphes.

FRED[19] est supposé analyser des phrases en langage naturel pour en produire des fragments d'ontologies et de données liées en RDF/OWL. Pratiquement, ce logiciel traite des phrases bien formées en anglais, dans lesquelles le verbe doit être correctement reconnu. En effet, le système base la reconnaissance de relations sur ce verbe. Pour cela, il fait appel à des analyseurs C&C[20] et Boxer[21], qui produisent à partir du texte analysé une représentation logique compatible avec la DRT (*discourse representation theory*). Cette représentation est ensuite transformée en triplets RDF, selon un ensemble d'heuristiques. Ces heuristiques cherchent à reconnaître, autour des verbes, les syntagmes qui jouent les rôles attendus pour ceux-ci dans FrameNet et VerbNet. D'autres heuristiques visent à produire une ontologie de qualité en appliquant des patrons de conception d'ontologie (*ontology design patterns*).

Tipalo[22] est présenté comme un logiciel d'extraction et de typage d'entités nommées tirées de pages web de Wikipedia. Il organise toutes les entités qu'il trouve dans une page Wikipedia au sein d'un graphe RDF en utilisant les relations suivantes : « rdf:type », « rdfs:subClassOf », « owl:sameAs », « owl:equivalentTo ». Les types des entités sont tirés des textes puis traités par un système de reconnaissance d'entités nommées (basé sur Apache Stanbol[23]), et désambiguïsés en utilisant UKB[24]. Les triplets sont alors alignés avec des ressources partagées,

[19] http://wit.istc.cnr.it/stlab-tools/fred

[20] http://svn.ask.it.usyd.edu.au/trac/candc

[21] http://svn.ask.it.usyd.edu.au/trac/candc/wiki/boxer

[22] http://wit.istc.cnr.it/stlab-tools/tipalo

[23] http://incubator.apache.org/stanbol

qui sont actuellement l'ontologie DBpedia[25], WordNet3.0[26] en RDF, DUL[27] et DolceZero[28]. Les résultats sont disponibles en RDF, HTML (avec LODE[29]), et sous forme de graphes. Tipalo utilise FRED pour l'extraction des triplets, et peut traiter tout type d'ontologie de tout domaine.

3.2 TAL pour les entités liées ?

Pour terminer, nous ferons mention d'un atelier qui déplace les perspectives du « TAL pour ontologies et annotation sémantique » vers le « TAL pour générer automatiquement des entités liées et annoter à l'aide de ces entités » : l'atelier WoLE2012 : The Web as a Web of Linked Entities. « T*he WoLE2012 workshop envisions the Web as a Web of Linked Entities (WoLE), which transparently connects the World Wide Web (WWW) and the Giant Global Graph (GGG) using methods from Information Retrieval (IR) and Natural Language Processing (NLP). The focus of this workshop is to bring together the Information Retrieval, Semantic Web and NLP communities. The primary goal is to strengthen research techniques that provide access to textual information published on the Web to further improve the adoption of Semantic Web technology.* »

On trouve parmi les sujets de cet atelier presque toutes les problématiques du TAL appliquées au web : *text and web mining*, *pattern and semantic analysis of natural language*, *reading the web*, *learning by reading*, *large-scale information extraction*, *entity resolution and automatic entities discovery*, *frequent pattern analysis of entities*, *ontology representation of natural language text*, *analysis of ontology models for natural language text*, *learning and refinement of ontologies*, etc.

Références

[1] Florence AMARDEILH. *Web sémantique et informatique linguistique : propositions méthodologiques et réalisation d'une plateforme logicielle*. Thèse de doctorat en informatique, Université Paris 10, 2007

[2] Sophie AUBIN, Thierry HAMON. « Improving term extraction with terminological resources ». *In* : S. Pyysalo, T. Salakoski, F. Ginter, T. Phikkala (eds.). *Advances in natural language processing, 5th International Conference on NLP (FinTAL 2006*), Turku, Finland, August 23-25, 2006. P. 380-387. Berlin : Heidelberg : Springer-Verlag, 2006

[3] Alain AUGER, Caroline BARRIERE. « Pattern based approaches to semantic relation extraction: a state-of-the-art ». *Terminology*, 2008, vol. 14, n° 1, p. 1-19

[4] Nathalie AUSSENAC-GILLES, DAGOBERT SOERGEL. « Text analysis for ontology and terminology engineering ». *Applied Ontology*, 2005, vol. 1, n° 1, p. 35-46

[5] Bruno BACHIMONT. *Arts et Sciences du numérique : ingénierie des connaissances et critique de la raison computationnelle*. Mémoire d'HDR, Université de technologie de Compiègne, 2004

[6] Caroline BARRIERE, Akakpo AGBADO. « TerminoWeb: A software environment for term study in rich contexts ». *In* : *3rd International Conference on Terminology, Standardization and Technology Transfer (TSTT 2006)*, Beijing, China, August 25-26, 2006. P. 103-113

[7] Chris BIEMANN. « Ontology learning from text: A survey of methods ». *LDV-Forum*, 2005, vol. 20, n° 2, p. 75-93

[8] Didier BOURIGAULT. « Upery : un outil d'analyse distributionnelle étendue pour la construction d'ontologies à partir de corpus ». *In* : *Actes de la 9e conférence annuelle sur le Traitement Automatique des Langues Naturelles (TALN 2002)*, Nancy, 24-27 juin 2002. P. 75-84

[9] Didier BOURIGAULT, Nathalie AUSSENAC-GILLES. « Construction d'ontologies à partir de textes ». *In* : *10e conférence sur le Traitement Automatique des Langages Naturelles (TALN 2003), Batz-sur-Mer, 11-14 juin 2003*. P. 27-47

[10] Didier BOURIGAULT, Cécile FABRE. « Approche linguistique pour l'analyse syntaxique de corpus ». *Cahiers de grammaires* [Université Toulouse - Le Mirail], 2000, n° 5, p. 131-151

[11] Didier BOURIGAULT, Christian JACQUEMIN. « Construction de ressources terminologiques ». *In* : J.-M. Pierrel (éd.). *Industrie des langues*. Paris : Hermès, 2000. P. 215-233

[12] Philipp CIMIANO, Johanna VOLKER, Paul BUITELAAR. « Ontology construction ». *In* : N. Indurkhya, F. J. Damerau (eds.). *Handbook of natural language processing*. 2nd ed. Boca Raton, FL : CRC Press, 2010. 48 p.

[13] Philipp CIMIANO. *Ontology learning and population from text: Algorithms, evaluation and applications*. Berlin : Heidelberg : Springer-Verlag, 2006

[14] Philipp CIMIANO, Andreas HOTTO, Steffen STAAB. « Learning concept hierarchies from text corpora using formal concept analysis ». *Journal of artificial intelligence research*, 2005, n° 24, p. 305-339

---

[24] http://ixa2.si.ehu.es/ukb

[25] http://dbpedia.org/Ontology

[26] http://semanticweb.cs.vu.nl/lod/wn30

[27] http://www.ontologydesignpatterns.org/ont/dul/DUL.owl

[28] http://www.ontologydesignpatterns.org/ont/d0.owl

[29] http://www.essepuntato.it/lode

[15] Philipp CIMIANO, Günter LADWIG, Steffen STAAB. « Gimme' the context: context-driven automatic semantic annotation with c-pankow ». *In* : *Proceedings of the 14th International Conference on WorldWideWeb (WWW'05*), New York, NY, USA, December 12-15, 2005. P. 332-341. New York : ACM, 2005

[16] Philipp CIMIANO, Johanna VOLKER. « Text2onto: A framework for ontology learning and data-driven change discovery ». *In* : A. Montoyo, R. Munoz, E. Metais (eds.). *Proceedings of the 10th International Conference on Applications of Natural Language to Information Systems (NLDB 2005)*, Alicante, Spain, June 15-17, 2005. P. 227-238. Berlin : Heidelberg : Springer-Verlag, 2005

[17] Béatrice DAILLE. « Study and implementation of combined techniques for automatic extraction of terminology ». *In* : *Proceedings of the 32nd Annual Meeting of the Association for Computational Linguistics (ACL 1994)*, Las Cruces, New Mexico, USA, June 27-30, 1994

[18] Patrick DROUIN. « Term extraction using non-technical corpora as a point of leverage ». *Terminology*, 2003, vol. 9, n° 1, p. 99-117

[19] Patrick DROUIN, Philippe LANGLAIS. « Évaluation du potentiel terminologique de candidats termes ». *In* : *8es Journées internationales d'Analyse Statistique des Données Textuelles (JADT 2006*), Besançon, 19-21 avril 2006. P. 379-388

[20] David FAURE, Claire NEDELLEC. « A corpus-based conceptual clustering method for verb frames and ontology ». *In* : P. Velardi (ed.). *Proceedings of the LREC Workshop on Adapting lexical and corpus resources to sublanguages and applications*, 1998. P. 5-12

[21] Nicola GUARINO, Chris WELTY. « An overview of OntoClean ». *In* : S. Staab, R. Studer (eds.). *Handbook on ontologies*. P. 151-159. Berlin : Heidelberg : Springer-Verlag, 2004

[22] Zellig HARRIS. *Mathematical structures of language*. Wiley, 1968

[23] Marti A. HEARST. « Automatic acquisition of hyponyms from large text corpora ». *In* : *Proceedings of the 14th International Conference on Computational Linguistics (COLING*), Nantes, 23-28 août 1992, Nantes. P. 539-545

[24] Christian JACQUEMIN. *Variation terminologique : reconnaissance et acquisition automatiques de termes et de leurs variantes en corpus*. Mémoire d'HDR en informatique fondamentale, Université de Nantes, 1997

[25] Alexander MAEDCHE. *Ontology learning for the Semantic Web*. Kluwer Academic Publisher, 2002

[26] Alexander MAEDCHE, Steffen STAAB. « Ontology learning ». *In* : S. Staab, R. Studer (eds.). *Handbook on ontologies*. 2nd ed. P. 245-268. Berlin : Heidelberg : Springer-Verlag, 2009

[27] Elizabeth MARSHMAN. *Lexical knowledge patterns for the semi-automatic extraction of cause-effect and association relations from medical texts: A comparative analysis of English and French*. PhD Dissertation, Département de linguistique et de traduction, Université de Montréal, 2007

[28] Roberto NAVIGLI, Paola VELARDI, Stefano FARALLI. « A graph-based algorithm for inducing lexical taxonomies from scratch ». *In* : *Proceedings of the International Joint Conferences on Artificial Intelligence (IJCAI 2011*), Barcelona, Catalonia, Spain, July 16-22, 2011

[29] Antonio PAREJA-LORA. *Ontotag: A linguistic and ontological annotation model suitable for the Semantic Web*. Ph.D. thesis, Universidad Poltecnica de Madrid, 2012

[30] Francesco SCLANO, Paola VELARDI. « TermExtractor: A Web application to learn the common terminology of interest groups and research communities ». *In* : *9th Conference on Terminology and Artificial Intelligence (TIA 2007)*, Sophia Antipolis, October 8-9, 2007

[31] Patrick SEGUELA. *Construction de modèles de connaissances par analyse linguistique de relations lexicales dans les documents techniques*. Mémoire de thèse en informatique, Université Toulouse 3, 2001

[32] Steffen STAAB, Alexander MAEDCHE. « Ontology learning for the Semantic Web ». *IEEE Intelligent Systems*, 2001, vol. 16, n° 2, p. 72-79 [Special issue on the Semantic Web]

[33] Paola VELARDI, Michele MISSIKOFF, Roberto BASILI. « Identification of relevant terms to support the construction of domain ontologies ». *In* : *ACL WS on Human Language Technologies and Knowledge Management,* Toulouse, 6-7 juillet 2001. P. 18-28

[34] Paola VELARDI, Roberto NAVIGLI, Alessandro CUCHIARELLI, Francesca NERI. « Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies ». *In* : P. Buitelaar, P. Cimiano, B. Magnini (eds.). *Ontology learning from text: Methods, evaluation and applications*. P. 92-106. Amsterdam : IOS Press, 2005