DEA de Linguistique Théorique, Descriptive et automatique

Option linguistique informatique
Université Paris VII

Organisation du lexique pour assister l'accès lexical

Gaëlle Lortal

sous la direction de

Mme Laurence Danlos (Professeur, Paris VII)
et
Mme Brigitte Grau (Maître de conférences, LIMSI-CNRS, IIE)
M. Michaël Zock (Directeur de recherches, LIMSI-CNRS)
groupe LIR

septembre 2003

SOMMAIRE

II. Analyse du corpus

II.1. Structuration de l'analyse

II.1.1. Objectifs

Les objectifs de l'analyse sont de débroussailler la problématique de l'accès automatique au lexique ; Dans une petite mesure bien sûr, puisqu'il s'agit d'explorer un corpus minimal (quatre textes AFP) et d'étudier les accès au lexique possibles grâce aux outils automatiques mis à notre disposition au LIMSI (ROSA et SVETLAN') ou grâce à EWN. Le but d'un outil d'accès lexical étant d'être aussi puissant, ou le plus proche possible, de celui d'un locuteur humain, un de nos objectifs sera de quantifier le succès de ces outils aux tests. Ainsi des tests de validation seront à effectuer auprès de locuteurs natifs du français. On pourrait avoir alors une idée de l'efficacité d'un outil totalement automatique dans la recherche lexicale par rapport à des outils aux liens typés quasi-manuellement.

II.1.2. Ressources

Les tests mis en place sont censés répondre aux hypothèses posées ci-après grâce à diverses ressources, principalement ROSA, SVETLAN', EWN.
Nous espérons, grâce à ces différentes bases de connaissances mimer les connaissances qu'un locuteur peut mettre en jeu dans l'analyse ou la production d'un message.
Aucun test ne perd de vue que ROSA, SVETLAN', EWN ou un locuteur, n'ont pas des capacités identiques quant aux relations à mettre en jeu dans l'accès au lexique.
En effet, ROSA n'a aucune relation typée, cependant la construction des domaines implique des relations syntagmatiques abstraites ; SVETLAN' aurait alors à disposition des relations syntagmatiques abstraites ainsi que des relations syntaxiques typées ; EWN entretient entre les termes du thésaurus des relations paradigmatiques, et un locuteur les possède toutes, avec en plus une autonomie et une vitesse de calcul hors du commun.

II.1.2.1. ROSA

Les données de ROSA représentent une base lexicale organisée en thèmes. Ces informations dans un domaine thématique se présentent comme suit :

commercialJaponais

Nombre de lemmes : 667

américain, commercial, japonais, négociation, multilatéral, négociation_commercial, commerce_international, trade, droit_de_douane, round, commerce, libre-échange, négociation, commercial, discussion, déficit_commercial, relation_commercial, tarif_douanier, billet_vert, américain, bilatéral, pourparler, bilatéral, négociateur, ministre, discussion, responsable, accord, étranger, affaire, semi-conducteur, excédent_commercial, tenter, marché, mémorandum, protectionniste, japonais, représentant, protectionnisme, réduire, oléagineux, adjoindre, jeudi, énorme, pays, déficit, lundi, déclarer, commerce_mondial, excédent, vendredi, délégation, secteur, négociateur, relancer, objectif, impasse, taux_d'escompte, délégation, deux_partie, syrien, international, annoncer, reprise, ouvrir, février, reprendre, trouver, israélien, pièce_détaché, palestinien, commerce, milliard, économique, dernier, industrie, gazer, paix, contentieux, service_financier, acte_final, indiquer, politique, télécommunication, viser, président, mettre, juillet, moyen, jordanien, conférence_de_paix, commerce_extérieur, exportation, accord, préciser, équipement, journée, cutter, médical, ouverture, base, rencontre, astronaute, lanceur, manille, proche-orient, assurance, dollar, pièce, partie, mois, entamer, conseiller, détaché, haut, anonymat, automobile, responsable, jour_férié, chef_de_cabinet, bien_social, conseiller_technique, sous-directeur, ministre, importation, parapher, jeudi, marché_public, mardi, administration, efforcer, matin, ministère, produit, équipe, parvenir, mardi, tenter, métallurgie, affaire, devise_américain, administrateur_civil, adjoindre, impasse, étranger, lundi, samedi, tenir, échec, tomber, local, position, mondial, poursuivre, nippon, puissance, redémarrage, mort, estimer, clé, point, presse, représenter, trésor, référence, couvert, retrouver, question, compromis, sous-secrétaire, ouvrir, accord_de_principe, annoncer, marché_européen, produit_agricole, vendredi, représentant, directeur_de_cabinet, partenaire_social, investissement_étranger, yen, douanier, désarmement, charger, probable, porte-parole, finir, prévoir, conduire, émissaire, attendre, gouvernement, interrompre, concentrer, porter, ajouter, confirmer, secrétaire, entretien, grand, réaliste, principe, conclure, dire, prochain, marquer, cadre, agir, chiffrer, échange, sortir, renouer, constat, optimiste, soir, nouvelle, terminer, assouplissement, matière, côté, progresser, niveau, institut_d'études, matin, interrompre, tripartite, start, pourparler_de_paix, propriété_intellectuel, février, accord_commercial, transplant, convertible, rééchelonnement, relancer, mois, angolais, marché, politique_contractuel, administration, session, libeller, pays, grand_surface, hypermarché, matrimonial, baleinier, immatriculation, rencontrer, critère, bureau, consécutif, après-midi, lieu, conduite, former, méthode, discuter, situation, composer, séparer, massacre, geler, réaliser, suspendre, progrès, avancer, confiant, tarif, témoigner, jour, évaluation, aller, fixer, date, finance, devenir, général, venir, précaire, arriver, intervenir, proposition, travail, pessimiste, bill, redémarrer, poursuite, souligner, pouce, matinée, représentation, membre, borner, cinquième, excédent, heure, interroger, prendre, limiter, session, douanier, personne, couvrir, extérieur, issue, ancien, requérir, réunion, boucler, demande, contact, escompte, après-midi, bush, tomber, gouvernement_américain, adjudication, dimanche, secteur, tenir, déficit, international, recel, administration_central, sogo, samedi, dernier, plurilatéral, secrétaire, journée, sous-préfet, ancien_élève, marché_obligataire, préfet, puissance, juillet, saint-siège, balance_commercial, redémarrer, dumping, centre_commercial, matinée, fed, rétention, indiquer, accord_salarial, cadre, charger_de_mission, investissement_direct, reprendre, exportateur, importateur, retrouver, pentagone, gréviste, agroalimentaire, boulin, automobile, place, département, oublier, quitter, négocier, rapport, journal, préliminaire, réclamer, suffisant, unir, effort, englober, source, parler, demi, affirmer, congrès, prévu, pékin, avenir, rappeler, coup, voir, durer, différend, compte, engager, refuser, milieu, chaîne, télévision, fin, chose, minute, chercher, espérer, administrer, officiel, exigence, contraire, user, renoncer, forme, semaine, bloquer, séance, interruption, public, expliquer, numéro, condition, règles, maison, organisation, refus, parlement, résoudre, réduction, idée, comprendre, année, britannique, midi, successeur, bon, compter, absence, confier, indissociable, contenu, débloquer, aune, nuit, formuler, modalité, interview, offrir, signifier, seul, joindre, dialogue, revoir, définir, commission, avril, marathon, retrait, pourparler, débattre, économie, sembler, époque, significatif, lancer, sérieux, publier, période, expert, rédiger, comporter, donner, solder, demander, but, report, cas, rupture, pas, constituer, reporter, rester, capitale, signature, optimisme, moment, nécessité, fédéral, possible, signe, prolonger, retenir, conférence, part, cours, apprêter, divergence, téléphonique, mener, difficulté, futur, mesurer, ignorer, revenir, firme, paraître, concrétiser, tractation, attention, accès, discret, réussir, élément, programme, convaincre, service, passer, mauvais, conclusion, round, conversation, avancement, reconnaître, instaurer, retirer, détailler, figurer, début, examiner, entrer, possibilité, concerner, signal, réunir, essentiel, sortie, préjudice, syrien, ministère_de+le_finance, consécutif, préliminaire, pers, aide_social, amman, finir, conduire, place, accord_de_coopération, convention_collectif, bien_d'équipement, émissaire, pékin, critère, déficit_budgétaire, séparer, massacre, local, excédentaire, dette_publique, opérateur, confiant, chercher, espérer, traité_de_paix, clause, tarif, administrer, exigence, jour, contraire, concentrer, porter, fixer, date, règles, président, mondial, objectif, assurance, affaire_criminel, habillement, détroit, département, devenir, général, négoce, confirmer, longuet, membre_permanent, échec, blair, joint-venture, midi, probable, mettre, grand, fusée, bureau, nuit, cutter, bill, achopper, conclure, panel, gonflement, cessez-le-feu, autolimitation, tacite, khmer_rouge, food, butoir, segment, fin_de_journée, télécopieur, entamer, industrie_automobile, marché_automobile, tvhd, pénétration, cinquième, libéralisation, industrie, prochain, conseiller_régional, rupture, grève_général, heure, subventionné, moyen, ouverture, marquer, solde, affaire_étranger, consul, taxe, nécessité, échange, loyer_de_l'argent, signe, marché_mondial, reprise, pts, constat, trésor, textile, soir, concrétiser, riz, haut, nouvelle, référence, quitter, assouplissement, personne, tarifaire, issue, inspecteur_général, porte-parole, matière, parvenir, politique, trouver, championnat, chambre_d'accusation, déclaration_d'indépendance, instaurer, côté, contact, surveillant, fabiusien, niveau, haut_cour, compromis, affaire_européen, classement, résidentiel, japon, proposition_de_loi

Plus le nombre d'agrégations dans un domaine est important, plus le domaine va contenir de lemmes.
Ce type d'informations est similaire aux connaissances lexicales d'un locuteur organisées conceptuellement.

II.1.2.2. SVETLAN'

Les informations contenues dans les fichiers de SVETLAN' sont aussi des données lexicales organisées conceptuellement ; Elles possèdent de surcroît des informations grammaticales, puisque les noms sont liés aux verbes par des liens Sujet, COD ou préposition (à, sur, de, etc.).
Les fichiers d'UTSA sont donc plus ou moins similaires aux informations conceptuelles et syntaxiques que possède un locuteur sur un lemme. Un domaine présente les données sous la forme d'un verbe(et ses noms reliés) c'est-à-dire sous cette forme dans INCA :

prendreEngager

Nombre de verbes : 102

- prendre (fils, retard, temps, service, retard, fils, cassure, fédération)
- engager (pilote, nom, championnat, hôpital, hôpital)
- améliorer (sécurité, sécurité)
- effectuer (retour, année, retour, circuit)
- apprendre (vendredi, soir, service, organisateur)
- expérimenter (coureur)
- présenter (présidence, présidence, zone, vice-président)
- obtenir (service)
- entraîner (diminution, risque, mesure)
- lancer (défi)
- vivre (heure, drame, drame)
- assurer (sécurité, sécurité)
- aller (monoplace, élection)
- sembler (autorité, autorité, conflit)
- donner (conférence, veille)
- désigner (successeur, vice-président, vice-président)
- apparaître (problème, visage, sécurité, signe)
- intervenir (décision, confirmation)
- devenir (vainqueur, menace)
- rester (lit, stand)
- préparer (mile, pilote)
- dérouler (épreuve, réveil)
- appartenir (pilote)
- participer (écurie)
- comprendre (délégué, groupe)
- céder (départ)
- limiter (vitesse, voie)
- amortir (sortie)
- constituer (danger)
- signer (contrat)
- aboutir (négociation)
- révolter (pilote)
- redouter (moment, pilote)
- reporter (mars)
- considérer (développement)
- fuir (responsabilité)
- fier (sécurité)
- désamorcer (fronde)
- opérer (miracle)
- fournir (moteur, constructeur)
- briser (disparition)
- écrire (journal)
- décimer (année)
- apporter (circuit)
- représenter (danger)
- menacer (ministre)
- remplacer (saison)
- terminer (rallye)
- réunir (soir)
- succéder (coup)
- imposer (fédération)
- survenir (accident)
- planer (menace, déroulement)
- contrarier (carrière, blessure)
- ralentir (voiture, gravier)
- multiplier (accident)
- évoquer (problème)
- inscrire (message, capot)
- estimer (journal)
- couvrir (tour)
- recevoir (million, mort)
- annuler (prix)
- battre (année)
- élire (représentant)
- provoquer (cassure, incident)
- armer (revers)
- contraindre (étape)
- affirmer (journal)
- entendre (modification)
- respecter (lutte)
- posséder (chance)
- offrir (adversaire)
- tendre (bras)
- monter (podium)
- promettre (succès)
- oublier (différend)
- relancer (association, pilote)
- oeuvrer (sens)
- attendre (ravitaillement, stand)
- perdre (poignée)
- connaître (spirale)
- afficher (détermination)
- voir (image)
- réduire (performance)
- imiter (patron)
- occuper (fin)
- affronter (risque, circuit)
- dire (chevalier)
- trouver (moyen)
- éliminer (lutte, occasion)
- annoncer (ligne, président)
- partir (tête-à-queue)
- exiger (inspection)
- commencer (pilote)
- garantir (sécurité)
- revenir (saison)
- tourner (an, circuit, joueur)
- épargner (malchance)
- disparaître (rival)
- pencher (problème)
- préserver (pneu)
- déboucher (heure)

mais on peut aussi afficher, et donc utiliser, les informations sur des verbes et leur liens vers des lemmes.
En effet, les domaines de SVETLAN' sont constitués de paquets verbaux, c'est-à-dire un verbe avec ses liens et ses noms reliés. De la forme XML suivante :

```
<Structure>
        <Verbe>traduire</Verbe>
        <Poids>2</Poids>
        <Lien type="cod">
            <Lemme>soleil</Lemme>
            <PoidsLemme>1</PoidsLemme>
        </Lien>
        <Lien type="par">
            <Lemme>soleil</Lemme>
            <PoidsLemme>2</PoidsLemme>
        </Lien>
    </Structure>
```

Les UTLAs et UTSAs étant générées automatiquement, il existe une grande quantité de données. Afin d'avoir un corpus homogène et de taille non-excessive, il a été décidé de ne prendre que les domaines et domaines structurés issus d'un corpus textuel français (pas la partie anglaise issue du Los Angeles Times), et qu'un seul fichier de ROSA et SVETLAN' ayant pour origine des dépêches AFP (Agence France-Presse).

II.1.2.3. EWN

EWN serait le pendant électronique d'informations paradigmatiques sur un lemme dans l'esprit d'un locuteur.

Ces différents types d'informations représentent des données lexicales durablement en mémoire chez un locuteur. Cependant, lorsque l'on produit du langage, le choix d'un mot dans ces données possibles est guidé par le contexte dans lequel doit intervenir la production.

Ce contexte est simulé par l'introduction de vocabulaire issu de textes.

II.1.2.4. Texte à trou(s)

Le principe du texte à trou(s) (TàT) sera utile pour chacun des tests. En effet, il permet de guider plusieurs types d'observations.

Il s'agit d'observer l'accès à un mot absent dans un (con)texte. Le texte est un moyen intéressant d'observer l'accès lexical puisque dans ce cadre, le mot possède un contexte (voir Annexes 6) représenté par l'ensemble du texte conservé précédant et suivant le mot manquant. D'une certaine façon, on fait donc une recherche sur le sens même du mot (dans un ensemble de sens- le contexte-) ainsi que sur les associations de mots.

Quatre textes, nommés par la suite Irak, Libéria, Lockerbie, Nicolas, ont été choisis au sein de dépêches AFP pour constituer le corpus-test (en effet, la majorité des outils a des représentations du lexique de style journalistique).

Ces textes sont débarrassés manuellement de 5 verbes et 5 noms chacun, afin encore une fois que tous les systèmes soient sur un pied d'égalité (en effet, SVETLAN' ne répertorie que les verbes et leurs relations aux noms).

Ces mots sont conservés dans une liste de mots manquants spécifique par fichier de contexte étudié.

Le texte se présente de la sorte :

« Attentat de Lockerbie: Tripoli _____l'accord avec les familles (14/08/2003)
NEW YORK (AFP) L'ambassadeur de Libye à Londres, Mohammad Al-Zouai, a confirmé jeudi l'accord de son _____à dédommager les familles des victimes de l'attentat de Lockerbie, tout en accusant la France de vouloir le bloquer en menaçant d'empêcher la _____à l'Onu des sanctions contre Tripoli. »

(Texte Lockerbie voir Annexes 4)

mais sera considéré pour les traitements par les systèmes comme (en noir):

« Attentat de Lockerbie: Tripoli _____l'accord avec les famille (14/08/2003)
NEW YORK (AFP) L'ambassadeur de Libye à Londres, Mohammad Al-Zouai, confirmer jeudi l'accord de son _____à dédommager les famille des victime de l'attentat de Lockerbie, tout en accusant la France de vouloir le bloquer en menacer d'empêcher la _____à l'Onu des sanction contre Tripoli. »

Les mots manquants, soient : « confirme, pays, levée », seront considérés comme : « confirmer, pays, levée ».

Les mots manquants symbolisent les mots que l'on cherche à produire dans un contexte de production. Lors des résultats, une liste de mots est présentée comprenant ou non ces mots manquants. Du moment que ces mots seront compris dans la liste, ils seront considérés comme répondant aux critères de choix et donc comme choisis par un système de génération, malgré la multitude de mots proposée.

Pour limiter les propositions et se cantonner à des thèmes, des domaines, représentatifs du contexte donné, on utilisera les poids des mots dans les UTLAs ou UTSAs, mais aussi des

poids pour les domaines résultants, et les résultats en général. En effet, il faut quantifier les résultats afin de n'en prendre que les plus précis, les suffisamment représentatifs

II.1.2.5. Poids

Tous les fichiers sélectionnés pour le remplacement d'un mot manquant vont posséder un poids. De même le nombre de mots manquants trouvés sera représenté par des coefficients.
Ce poids sert à classer les outils par rapport à la possibilité d'accès au lexique plus ou moins adéquate qu'ils procurent.
Pour notre calcul, quatre paramètres sont pris en compte :
a°) le nombre de mots manquants : nombre de mots dans le domaine « mots manquants ».
b°) le nombre de mots trouvés : nombre de mots issu du système et appartenant au domaine « mots manquants ».
c°) le nombre total de mots sortis : nombre de mots issu de tous les domaines retenus, issus d'un système.
d°) le poids des mots dans les systèmes en disposant : poids des mots dans un domaine issu d'un système.

Trois coefficients sont à prendre en compte pour donner la valeur la plus objective possible quant au poids du domaine contenant les réponses (résultat final sur lequel on compare les systèmes).
Deux de ces coefficients sont valables dans tous les cas et se pondèrent l'un l'autre. Le troisième permet de pondérer un résultat plus global.

Le premier coefficient se présente sous la forme :

p1.
$$\frac{\sum(mot(s)\_trouvé(s))}{\sum(mot(s)\_recherché(s))}$$

Il représente fondamentalement la réussite de la recherche de mots. Cependant, il est important de prendre en compte la quantité d'information sélectionnée. En effet, plus la quantité de données est importante, plus on a de possibilités d'accès aux lexèmes recherchés ; cependant, plus la quantité d'information est grande, plus on a aussi de bruit. C'est pourquoi il a été décidé de calculer le nombre total de lemmes sélectionnés dans un système, en tant que second coefficient. Le premier coefficient est donc pondéré par celui-ci :

p2.
$$\frac{\sum(mot(s)\_trouvé(s))}{\sum(mot(s)\_total)}$$

Un dernier coefficient est mis en place pour prendre en compte les multiples occurrences d'un lemme dans un domaine ou dans le système en général. En effet, si par exemple dans ROSA un lemme est présent plusieurs fois, c'est que l'on peut aussi bien le trouver textuellement (origine = T), que l'inférer par calcul (origine = I). On a donc au moins deux possibilités d'accéder à ce lemme. Cela n'est pas forcément négligeable puisque le double accès indique une certitude plus forte de trouver le mot manquant. De plus, ce coefficient permet de préférer

un domaine par rapport à un autre si les poids stricts des deux domaines sont identiques. Le troisième coefficient est tel que :

p3.

$$\frac{\sum(poids\_des\_mot(s)\_trouvé(s))}{\sum(poids\_de\_tous\_les\_mots)}$$

Le poids final du domaine des mots trouvés, qui va représenter la robustesse d'un système, est la moyenne de ces trois coefficients (le troisième ayant un poids moindre que les deux premiers).

p4.

$$\overline{C} = \frac{\sum(c1, c2, c3)}{3}$$

Tous les coefficients étant compris entre 0 et 1, on considère que plus le poids approche 1 plus le système est valable, et plus il approche de 0, moins il permet d'accès intéressants.
Ce calcul vaut pour les applications ROSA et SVETLAN' et permet de les comparer entre elles.
Pour EWN, les mots ne possédant pas de poids, et les liens nous menant directement aux lexèmes, il a été décidé que le coefficient 3 de pondération ne serait pas calculé ; il serait cependant possible de le fonder sur le type de lien suivi (d'une façon arbitraire toutefois !).
Bien que comparé aux résultats du calcul p4., le calcul effectué pour cette application sera donc :

p5.

$$\overline{C} = \frac{\sum(c1, c2)}{2}$$

Le calcul pour le test locuteur sera bien évidemment différent, mais comme dans le cas d'EWN, on tentera par un maximum d'objectivité dans les calculs de rendre comparables les résultats des calculs p4., p5., et p6..

Toutes ces ressources sont censées permettre la réponse aux hypothèses suivantes.

II.1.3. Hypothèses

On souhaite pouvoir définir quelles informations sont pertinentes dans l'accès lexical au niveau des relations principalement.
Les ressources précédemment présentées nous permettent de simuler les différents composants reconnus être mis en jeu dans le choix lexical. Il faut maintenant observer l'apport de ces différents types de ressources selon les liens qu'il procure vis-à-vis de l'organisation du lexique.
Trois hypothèses principales structurent l'analyse du corpus ;

II.1.3.1. Liens non typés

1. Les liens syntagmatiques non typés dans ROSA (liens thématiques) et SVETLAN' (liens thématiques et syntaxiques) sont suffisants pour l'accès au lexique.

II.1.3.2. Liens non typés conjointement avec des liens typés

2. Cependant, un typage conjoint avec EWN donnerait de meilleurs résultats, et la fusion entre les résultats des traitements de ROSA ou SVETLAN' et les liens paradigmatiques lexicaux d'EWN pourrait être automatisée et donc permettre un apport syntagmatique à EWN, et vice-versa.

II.1.3.3. Liens typés

3. Une fusion entre ressources de liens paradigmatiques et liens syntagmatiques explicites pourrait devenir une ressource complète et applicable dans de nombreuses implémentations. Il serait bien d'étudier la faisabilité (automatique, non manuelle) d'une telle ressource entre EWN et Dico par exemple.

II.2. Le traitement

Ceci est une présentation de la démarche suivie pour le traitement des tests, de la mise en place à l'analyse.

II.2.1. Traitement hypothèse 1

II.2.1.1. ROSA

1a

1. récupération de la liste des mots manquants dans TàT ;
2. récupération de la liste des mots pleins du contexte (Domaine du TàT) ;
3. matching du domaineTàT avec les domaines de ROSA (quantification des domaines de ROSA par rapport à domaineTàT) ;
4. sélection des domaines ROSA (voir Annexes 7) ;
5. renvoi de tous les lemmes constituant les domaines sélectionnés ;
6. matching des lemmes avec lemmes de la liste des mots manquants (avec quantification) ;
7. sélection du meilleur résultat objectif/automatique (le plus proche de 1).
8. *possibilité de tri manuel sur le genre et le nombre afin d'affiner (requantification après tri sur le déterminant).*

II.2.1.2. SVETLAN'

Dans le cas de SVETLAN', il est intéressant évaluer deux traitements :
1b.
Le premier traitement est un traitement quasi uniquement syntaxique. Il ne met en jeu que les occurrences lexicales structurées syntaxiquement. C'est-à-dire que seul le domaine de base précédant l'analyse syntaxique de SVETLAN' et issu de ROSA est pris en compte de façon sous-entendue. On n'utilise alors pas le maximum des capacités de SVETLAN'.

1. récupération de la liste des mots pleins du contexte (Domaine du TàT) de catégorie Verbe et Nom ;
2. récupération dans SVETLAN' de toutes les occurrences des mots du Domaine TàT avec leur(s) relation(s) syntaxique(s) et leur(s) lemme(s) lié(s) ;

3. pour chaque lemme repéré (matching des lemmes), renvoyer une liste de mot(s) proposé(s) grâce aux liens syntaxiques pour le TàT (quantifier les résultats) ;
4. sélection du meilleur résultat objectif/automatique (le plus proche de 1).
5. *possibilité de tri manuel sur le genre et le nombre afin d'affiner (requantification après tri sur le déterminant).*

1c.
Le second traitement possible est un traitement syntaxique-thématique, utilisant donc toutes les ressources possibles de SVETLAN'. On opère en quelque sorte le traitement 1a et 1b en même temps. Les étapes sont plus ou moins identiques, sauf un passage sur la sélection de domaine(s). On peut alors arriver à des différences très importantes entre les deux techniques, puisque le choix par domaine va se faire ici sur les occurrences de verbes uniquement (en effet, il n'est pas possible de retrouver les domaines de ROSA utilisés par SVETLAN' à la première constitution de domaines structurés, le nom de domaine ayant été modifié).
Les domaines de SVETLAN' sont donc beaucoup plus réduits, les accès au reste du lexique sont donc en théorie moins nombreux (à moins de la présence de Noms recherchés dans les relations des verbes), mais plus précis pour les verbes.

1. récupération de la liste des mots pleins du contexte (Domaine du TàT) de catégorie Verbe et Nom ;
2. matching des verbes du domaineTàT avec les verbes par domaine de SVETLAN' (quantification des domaines de SVETLAN' par rapport à domaineTàT) ;
3. sélection des domaines SVETLAN' ;
4. renvoi de tous les lemmes constituant les domaines sélectionnés avec leur(s) relation(s) syntaxique(s) (quantifier les résultats) ;
5. pour chaque lemme repéré (matching des lemmes), renvoyer la liste de mot(s) proposé(s) grâce aux liens syntaxiques pour le TàT (quantifier les résultats) ;
6. sélection du meilleur résultat objectif/automatique (le plus proche de 1).
7. *possibilité de tri manuel sur le genre et le nombre afin d'affiner (requantification après tri sur le déterminant).*

II.2.1.3. EWN

1d.

1. récupération de la liste des mots pleins du contexte (Domaine du TàT) ;
2. matching de chaque mot de la liste avec le variant d'EWN ;
3. extraire tous les mots possibles à un lien du mot source (lien direct sinon trop de possibilités et impossible à diriger) ;
4. renvoi d'une liste de mots à un lien d'éloignement du mot source ;
5. quantifier la liste obtenue par le calcul du poids en fonction de la liste de mots manquants.
6. *possibilité de tri manuel sur le genre et le nombre afin d'affiner d'autant plus que le genre est disponible dans la base (requantification après tri sur le déterminant).*

II.2.1.4. LOCUTEUR

1e.
Dans l'absolu, on considère que le résultat optimal est représenté par le coefficient 1. Cependant, afin de pondérer ce chiffre, on pourrait considérer que le taux de réussite moyen des locuteurs est le résultat optimal. C'est pourquoi il est important d'avoir un échantillon témoin des résultats qu'un locuteur natif de la langue peut obtenir.

Pour se faire des tests sur les mêmes bases sont présentés. Le déroulement de l'analyse, dans un lieu neutre et calme est celui-ci :

1. explication des conditions du test ;
2. explication du type de texte, de la date (journalistique, moderne) ;
3. distribution du TàT n°1 ;
4. lecture du TàT une fois en entier ;
5. relecture et remplissage par cinq mots maximum (au choix libre) de chacun des trous. Les mots doivent être classés du plus probable (1) selon le locuteur au moins probable (5).
6. Calcul du poids par rapport à la liste de mots manquants.

II.2.2. Traitement hypothèse 2

II.2.2.1. ROSA/SVETLAN'_EWN

2a.
Cette hypothèse pousse en fait à effectuer des traitements affinant les résultats précédents.
On peut a priori aussi bien affiner les résultats de ROSA que ceux de SVETLAN', mais selon les résultats au test précédent, nous allons considérer qu'il vaut mieux optimiser celui obtenant les valeurs les plus importantes (autant optimiser une base solide).
Affiner des accès lexicaux par EWN revient à diversifier les liens sur un mot, puisque le principal intérêt de cette base de données est le nombre important de liens.
Ainsi, après avoir obtenu un champ lexical (un thème) grâce à ROSA ou SVETLAN', où les propositions ne sont pas satisfaisantes dans le cas d'une recherche d'un lexème précis (les résultats obtiennent tout de même les coefficients les plus élevés), on pourrait faire embrayer la recherche sur des mots voisins, liés par des relations de synonymie, d'hyperonymie, etc.
Un ordre de traitement est donc nécessaire.

1. effectuer le traitement de type 1a/1b ou 1c ;
2. effectuer 1d. sur les résultats de 1a/b/c. ;

II.2.2.2. EWN_ROSA/SVETLAN'

2b.
On peut toujours expérimenter le traitement inverse, consistant alors à apporter à EWN des liens thématiques et syntaxiques, mais au vu des résultats d'EWN, beaucoup trop vagues, il semble plus adéquate de ne pas perdre de temps dans le traitement des données et de réduire le champ d'investigation le plus tôt possible. Cette possibilité ne sera donc pas testée ici.

II.2.2.3. LOCUTEUR

2c.
Afin de pondérer ce traitement, il est intéressant de moyenner les résultats par les scores d'un locuteur.

1. passer le TàT accompagné des résultats issus du test 1a/b/c.
2. faire effectuer le choix de cinq lexèmes à substituer aux vides, inspirés des résultats au test. Ces mots choisis librement peuvent ou non être identiques à la liste passée. Les mots doivent être classés du plus probable (1) selon le locuteur au moins probable (5).
3. Calcul du poids par rapport à la liste de mots manquants.

II.2.3. Traitement hypothèse 3

II.2.3.1. FL_EWN_ROSA/SVETLAN'

3a.
Cette hypothèse est assez lointaine dans la mesure ou de nombreux paramètres sont à prendre en compte.
La fusion de ressources de liens paradigmatiques et liens syntagmatiques pourrait devenir, à l'image de WordNet, une ressource pratique mise en place manuellement (comme DiCo). Cette optique semble faisable. Si une telle ressource intéresse autant que WN, et que d'aussi nombreuses implémentations sont permises permettant une réelle désambiguïsation, c'est un outil à mettre en place absolument.
Mais pour tester l'utilité et la faisabilité d'un tel outil, il faudrait au moins en partie une implémentation manuelle fort longue et qui nécessiterait un autre stage au moins.
Pour le moment, on peut toujours observer les vides laissés par la fusion SVETLAN'/ROSA_EWN. Dans le cas où ces vides sont dus à l'absence de liens syntagmatiques, on peut alors penser que ce type de ressources est très utile.

Afin de connaître les causes de ces impossibilités d'accès, on peut effectuer des tests sur les locuteurs. Il ne faut tout de même pas perdre de vue que SVETLAN' et ROSA finalement possèdent un certain nombre de cooccurrences et collocations même si celles-ci sont non typées, et que donc rien ne prouve que l'apport de cet outil vaut les déploiement de moyens nécessaires pour créer une telle base, à moins d'une implémentation facilement automatisable.

En effet, on peut considérer que si les données sont proches, et qu'il n'y a pas d'inconsistances entre les systèmes, une automatisation de cette fusion de ressources est possible ; en fait, d'une certaine manière, les fonctions lexicales sont directement en concurrence avec les systèmes statistiques et automatisés étudiés lors de ce stage.

L'analyse des données grâce à cette méthodologie peut nous apporter un certain nombre de réponses.

Une fois toute la méthodologie sur pied, on peut entamer les analyses, même si au fur et à mesure des changement mineurs sont à effectuer pour s'adapter à des imprévus.
Les résultats présentés sont simplifiés et ne comportent pas toutes les étapes décrites ci-dessus.

III. Prétraitement du corpus

III.1. Origine du corpus

L'étude des relations du lexique nécessite un corpus textuel. Plus il est conséquent, plus on a de chances d'y observer des lois générales ; c'est le principe de la linguistique de corpus. On peut penser que la linguistique de corpus dans sa démarche empirique s'oppose à la linguistique-informatique, cependant, celle-ci peut permettre aussi bien de formaliser des règles implémentables par la suite, que de valider des outils automatiques.

Le corpus utilisé dans notre sujet est complexe, dans le sens qu'il est construit de plusieurs résultats d'application, mais aussi parce qu'il est constitué de fichiers de différents formats.
En effet, on met à contribution la mémoire sémantique de ROSA (ensemble de tous les domaines construits automatiquement, les UTLAs) ainsi que toutes les Unités Thématiques Structurées de SVETLAN (UTSAs). Ces deux fichiers sont en format BQRDOC. Par contre, le fichier EWN est dans un format différent.
Il va donc falloir unifier ces formats « maison » afin de pouvoir les analyser par un même script facilement et les rendre compréhensibles en vue de réutilisation par exemple.
Il a donc été décidé de les traduire automatiquement (par des scripts en langage Perl) en format XML, puisque c'est un format courant et très utilisé.

Dans une première partie sont donc détaillés les différents formats et langage utilisés (BQRDOC, XML, Perl) ainsi que les principes de la conversion des fichiers.
Ensuite est présentée INCA (l'INterface du Corpus à Analyser, construite en Perl CGI et nécessaire pour une pré-analyse des fichiers d'UTLAs et UTSAs.

III.2. Traitement préalable du corpus

III.2.1. Le format Bqrdoc

[BQRDOC]

III.2.1.1. Historique du format

Le format BQRDoc trouve son origine dans un projet *Bonus Qualité Recherche* (projets financés par l'université Paris XI pour encourager les recherches mettant en jeu plusieurs équipes et/ou laboratoires au sein de l'Université) effectué avec des équipes du LRI (Laboratoire de Recherche en Informatique, Université Paris XI), l'action *Corval* et l'équipe L&C du LIMSI/CNRS. Ce projet, dénommé « Apprentissage de concepts », avait pour objectif de « concevoir et d'intégrer des méthodes d'apprentissage automatique et de

traitement automatique de la langue pour l'acquisition automatique de connaissances à partir de corpus de documents électroniques de types scientifiques et techniques ».
Il a donc semblé opportun de développer un nouveau format unique permettant aux diverses applications des divers laboratoires de communiquer.
Le format BQRDoc n'a finalement pas été utilisé dans le projet BQR lui-même puisqu'aucun prototype fonctionnel n'a été produit. Le temps disponible était trop court pour cela, mais la faisabilité du but annoncé a été démontrée et les difficultés pour l'atteindre identifiées. En revanche, des développements dans le cadre du système SVETLAN' ont profité de ce format.

III.2.1.2. Description du format

L'en-tête est composé des éléments suivants, chacun sur une ligne :

1. un identificateur du format de fichier (`#bqrdoc`) ;
2. un commentaire décrivant le type de données contenues dans le fichier (`Fichier contenant des UTLs`) ;
3. le nom du fichier source ;
4. le nom du fichier de l'en-tête ;
5. un numéro de version du format BQRDocBQRDoc (actuellement 0.1) ;
6. la ligne de description des colonnes ;
7. une ligne par type d'élément contenant la concaténation des noms de colonnes qu'il contient.
8. La ligne de l'en-tête de description des colonnes a le même format qu'une ligne : colonnes et sous-colonnes.
   Chaque sous-colonne de l'en-tête s'écrit ainsi :
   - []Nom :<nom>#Creat:<nom ou code du créateur de cette information>
   - #Dom:<regexp>#Code:<code Perl permettant d'extraire le contenu
   - de cette colonne>#Comment:un commentaire libre
   - avec :
     - `Nom` : le nom de la colonne qui devra être le plus expressif possible, tout en restant court ;
     - `Creat` : les initiales, le nom ou l'adresse E-mail de la personne ayant créé la colonne. Cette information n'est pas destinée à avoir une portée universelle, mais à être utile aux divers participants d'un projet de taille modeste ;
     - `Dom` : le domaine des valeurs que peut prendre l'élément, exprimé par une expression régulière. Ce domaine sera très souvent `.*` pour permettre d'insérer n'importe quelle chaîne de caractères ;
     - `Code` : un programme Perl permettant d'extraire le contenu de la colonne. Ceci est utile par exemple dans le cas ou la colonne contient un nombre variable de sous-colonnes dénotant des objets complexes ;
     - `Comment` : le commentaire devra être obligatoire. Les outils de gestion devront donc refuser la présence d'un commentaire vide !

Extrait de l'en-tête et deux lignes pour le fichier source exemple:

```
#bqrdoc
Ce fichier contient un exemple du format BQRDoc
monFichier
monEntete.hdr
0.1
Nom : Pos#Créat : gdc#Dom : nom|adj|...#Code :
  #Comment : ma colonne de description d'un élément du discours
  <TAB> Nom : Mode#Créat : cj#Cont : sing|plur...
Mot :Pos ;Mode ;...<TAB> SGMLTag :.....
<head type=PRINCIPAL id=afp_mai1994_1.1 n=1>|50 |100|SGMLTag| ...
Formule                                     |101|107|Mot    |nom|sing|...
```

Prés.2 fichier.bqrdoc

La première chose à faire est de choisir un caractère de séparation des colonnes : la tabulation. Nous devons aussi permettre une séparation des colonnes en sous-colonnes : #. Ces deux caractères devront être précédés d'une barre oblique inverse dans les données source pour éviter de les confondre avec les séparateurs, cette modification étant effectuée après avoir calculé les index des éléments, comme nous le verrons ci-dessous. Les colonnes ont les caractéristiques suivantes :

- La première colonne contient l'élément textuel : mot, ponctuation, balise, etc. ;
- La seconde contient l'indice de début de l'élément dans le fichier source et la troisième, l'indice de fin.
- Une quatrième colonne décrit un type d'élément. Pour chaque type d'élément, un certain nombre de colonnes sont valides, ceci étant spécifié dans l'en-tête. Cette quatrième colonne ne constitue qu'une recommandation. Elle peut être laissée vide par les utilisateurs ne désirant pas figer leurs catégories d'éléments. Les outils de gestion du format doivent donc prendre cette caractéristique en compte.

Le contenu de chaque colonne ou sous-colonne est ensuite libre, pourvu que le créateur les ait documentées dans l'en-tête. Malgré tout, il est recommandé que les sous-colonnes aient un contenu homogène. Une colonne pour laquelle l'information est valide mais inconnue sera laissée vide, c'est-à-dire que deux tabulations se suivront immédiatement.
Toute ligne commençant par le caractère « % » sera considérée comme une ligne de commentaire.

Le format est donc assez clair et permet entre autres une transformation simple en un autre format.

Les fichiers UTLAs et UTSAs utilisés sont respectivement, afp_0594.sgml0.all.utla.bqrdoc et afp_0594.sgml0.all.utsa.bqrdoc, respectivement d'environ 11,4 Mo et 1 Mo. Ces fichiers sont au format BQRDOC décrit ci-dessus. Ils se présentent sous cette forme :

```
#bqrdoc
Fichier contenant des UTL A
/home/gael/Smalltalk/Corpus/These/AFP/afp_0594.sgm
l0.1/afp_0594.sgml0.1.nodbq.sel.bqrdoc
/win/Smalltalk/Corpus/These/AFP/Encore/afp_0594.sg
ml0.all.utla.bqrdoc
0.1
Nom:Lemme#Creat:of#Dom:[\l|\+\é\à\è....]+#Code:#Co
mment:Un lemme correspondant à un ou des mots de
corpus
Nom:NbOcc#Creat:gdc#Dom:\d+#Code:#Comment:Le
nombre d'occurrences d'un lemme
Nom:Orig#Creat:gdc#Dom:T|I#Code:#Comment:
Nom:Balise#Creat:gdc#Dom:UTLA|finUTLA#Code:#Commen
t:indique le début ou la fin d'une UTLA
Nom:NomUTLA#Creat:gdc#Dom:[a-zA-Z_][a-zA-Z0-
9_\.]*#Code:#Comment:Le nom de l'UTLA, souvent
calculé par l'appli
Nom:NbAgr#Creat:gdc#Dom:\d+#Code:#Comment:Le
nombre d'aggrégations de l'UTLA
Nom:UTLs#Creat:gdc#Dom:#Code:#Comment:Les noms des
UTLs contenant le Lemme
Mot : Lemme;NbOcc;Orig;UTLs
Balise : Balise;NomUTLA;NbAgr

      0     0     Balise      UTLA
      blockhausDébarquement   1
      0     0     Mot   cas   1.0   T
      afp_.sel525
      0     0     Mot   blockhaus   1.0   T
      afp_.sel525
      0     0     Mot   débarquer   1.0   I
      afp_.sel525
      0     0     Mot   débarquement      1.0   I
      afp_.sel525
      0     0     Balise      finUTLA
      blockhausDébarquement   1
```

```
#bqrdoc
Fichier contenant des UTS A
/home/gael/Smalltalk/Corpus/These/AFP/afp_0594.sgm
l0.1/afp_0594.sgml0.1.nodbq.sel.bqrdoc
/win/Smalltalk/Corpus/These/AFP/Encore/afp_0594.sg
ml0.all.utsa.bqrdoc.2
0.1
Nom:Balise#Creat:gdc#Dom:UTSA|finUTSA#Code:#Commen
t:indique le début ou la fin d'une UTLA
Nom:NbAgr#Creat:gdc#Dom:\d+#Code:#Comment:Le
nombre d'aggrégations de l'UTSA
Nom:NomUTSA#Creat:gdc#Dom:[a-zA-Z_][a-zA-Z0-
9_\.]*#Code:#Comment:Le nom de l'UTSA, souvent
calculé par l'appli
Nom:UTSs#Creat:gdc#Dom:#Code:#Comment:Les noms des
UTSs agregees dans l'UTSA
Nom:Lien#Creat:gdc#Dom:#Code:#Comment:
Balise : Balise;NbAgr;NomUTSA
Lien : Lien UTSs : UTSs

      0     0     Balise      UTSA  5
      comblerSouligner
      0     0     Lien  exister#2#entre#niveau#2
      0     0     Lien  commencer#1#sujet#vie#1
      0     0     Lien
      souligner#1#cod#condition#1#sujet#rapport#1
      0     0     Lien
      réaliser#1#sujet#économiste#1
      0     0     Lien  estimer#1#cod#économiste#1
      0     0     UTSs  UTS  March  13,  20002:10:02
am#UTS  March  13,  20002:12:38  am#UTS  March  13,
20002:16:36  am#UTS  March  13,  20003:01:09  am#UTS
March 13, 20003:01:09 am
      0     0     Balise      finUTSA     5
      comblerSouligner
```

Le problème d'unification de formats s'est posé, puisque BQRDOC n'est unifié qu'au niveau du LIMSI et de ces partenaires pour ce projet ; il n'est pas réutilisable. Il valait donc mieux transformer ce format en un autre format plus conventionnel et reconnu, tel que XML, recommandation du W3C ([W3C]).

III.2.2. Le format XML

III.2.2.1. Historique du XML

Le langage de balisage extensible (Extensible Markup Language, XML) est un sous-ensemble de SGML. XML 1.0 est passé recommandation en 2000. Son but est de permettre au SGML générique d'être transmis, reçu et traité sur le Web de la même manière que l'est HTML aujourd'hui. XML a été conçu pour être facile à mettre en œuvre et interopérable avec SGML et HTML.
Ainsi, contrairement au HTML, le XML est un langage ouvert. En effet, il est possible de créer ses propres variables, ses propres balises (plus besoin de vérifier si tel ou tel attribut est homologué par le W3C).

III.2.2.2. Description du format XML

Un document XMl est dit bien formé lorsqu'il respecte les règles de la grammaire XML. C'est-à-dire :

1. que les informations sont entourées de balises ouvrantes et fermantes comme : <LIVRE> information </LIVRE>.
   On parle alors d'éléments. Les éléments doivent s'imbriquer proprement les uns dans les autres sans chevauchement.
2. ou que les informations sont incluses à l'intérieur d'une balise comme : <LIVRE SUJET="XML"> </LIVRE>
   On parle alors d'attribut
3. ou encore, les informations sont définies sous forme d'entités.
   Les entités sont des abréviations. Par exemple, "Extensible Markup Language" peut être déclaré comme une entité associée à la notation "xml"; cette chaîne de caractères pourra être abrégée en "&xml;" dans tout le fichier XML.

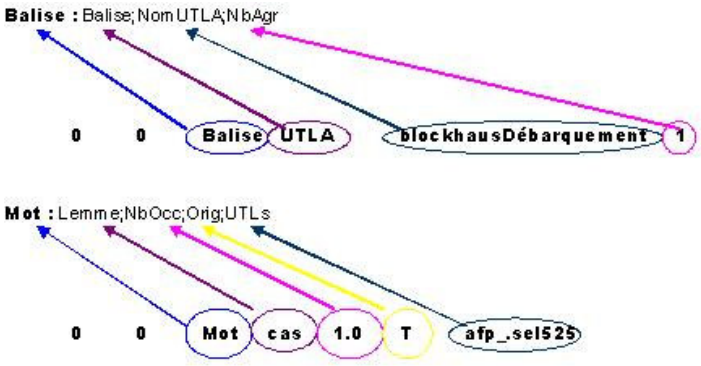
Un document XML est dit valide lorsqu'il possède une DTD (Définition de Type de Document) associée et la respecte. Cette déclaration est facultative. On n'écrira donc une DTD que lorsqu'il y aura vraiment intérêt à le faire (par exemple pour contraindre la saisie/mise à jour du document XML).
La DTD contient la structure arborescente du document XML (intitulé des balises, imbrications des balises, caractère obligatoire ou facultatif des balises et de leur ordre de succession…).

III.2.3. La « BtoX » (bqrdoc2XML)

Afin d'éviter tout problème au niveau du traitement du corpus, un choix s'est opéré sur la conversion des fichiers au format .bqrdoc en fichier .xml.

Cette conversion a un double objectif :
Tout d'abord, homogénéiser le corpus en vue d'obtenir des fichiers uniquement en XML, que ce soit les corpus « maison » (du LIMSI), ou la base de données EWN. Ils sont ainsi plus facile à traiter puisque répondent à un seul outil.
Ensuite, installer le projet dans une optique de réutilisabilité, avec des fichiers dans un format standardisé largement exploité. Par exemple, il existe plusieurs API XML dans la plupart des langages. De plus, c'est un format de fichier simple à traiter pour la génération d'applications Web.

Une première application simple (scripts Perl) a donc été implémentée, à l'aide de l'éditeur Xemacs (fichiers BtoX, voir Annexes 3), permettant de transformer les fichiers issus de ROSA ou SVETLAN en fichier XML, mais aussi de me familiariser avec un langage nouveau pour moi, Perl.
Le principe est simple vu la forme des données en BQRDOC. Grosso modo, il s'agit de reconnaître par des expressions régulières les éléments importants et les encadrer avec des balises correspondantes en respectant les niveaux d'enchâssement.



Prés.3 forme_données_UTLAs/entête.

Deux programmes sont donc disponibles, l'un pour les fichiers de ROSA (annexe myUTLAbqrdoc2XML.pl), l'autre pour les fichiers de SVETLAN' (myUTSAbqrdoc2XML.pl).
Les fichiers sous format XML sont de la forme suivante. Certaines balises seront expliquées plus loin lors de leur traitement.

```
<?xml version="1.0" encoding="iso-8859-1"?>
<ROSA>
<UTLA>
        <NomUTLA>blockhausDébarquement</NomUTLA>
        <NbrAggr>1</NbrAggr>
        <Lemme>
                <NomLemme>cas</NomLemme>
                <Poids>1.0</Poids>
                <Orig>T</Orig>
                <UTLs>afp_.sel525</UTLs>
        </Lemme>
        <Lemme>
                <NomLemme>protéger</NomLemme>
                <Poids>1.0</Poids>
                <Orig>T</Orig>
                <UTLs>afp_.sel525</UTLs>
        </Lemme>
        <Lemme>
                <NomLemme>plage</NomLemme>
                <Poids>1.0</Poids>
                <Orig>T</Orig>
                <UTLs>afp_.sel525</UTLs>
        </Lemme>
        <Lemme>
                <NomLemme>abriter</NomLemme>
                <Poids>1.0</Poids>
                <Orig>T</Orig>
                <UTLs>afp_.sel525</UTLs>
        </Lemme>
</UTLA>
</ROSA>
```

```
<?xml version="1.0" encoding="iso-8859-1"?>
<SVETLAN>
<UTSA>
<NomUTSA>comblerSouligner</NomUTSA>
        <NbrAggr>5</NbrAggr>
        <Structure>
                <Verbe>exister</Verbe>
                <Poids>2</Poids>
                <Lien type="entre">
                        <Lemme>niveau</Lemme>
                        <PoidsLemme>2</PoidsLemme>
                </Lien>
        </Structure>
        <Structure>
                <Verbe>commencer</Verbe>
                <Poids>1</Poids>
                <Lien type="sujet">
                        <Lemme>vie</Lemme>
                        <PoidsLemme>1</PoidsLemme>
                </Lien>
        </Structure>
        <Structure>
                <Verbe>souligner</Verbe>
                <Poids>1</Poids>
                <Lien type="cod">
                        <Lemme>condition</Lemme>
                        <PoidsLemme>1</PoidsLemme>
                </Lien>
                <Lien type="sujet">
                        <Lemme>rapport</Lemme>
                        <PoidsLemme>1</PoidsLemme>
                </Lien>
        </Structure>
</UTSA>
</SVETLAN>
```

Prés.4 ROSA/SVETLAN'.xml

III.3. INCA

III.3.1. Fondements

Pour analyser d'une façon économique le corpus (du point de vue du temps), il est beaucoup plus simple d'avoir une interface permettant un accès rapide aux données. En effet, il n'est pas possible de traiter manuellement des milliers de mots au cours d'un stage de quelques mois.
C'est pourquoi l'INterface du Corpus à Analyser a été développée.
Elle devait tout d'abord l'être en XSL, format recommandé par le W3C pour l'affichage de données XML, cependant, pour effectuer les analyses désirées, il fallait une interface dynamique.
Ainsi, on a opté pour Perl CGI. (Common Gateway Interface) qui est un module disponible dans la distribution standard de Perl et qui permet de générer des pages HTML dynamiques.
Le langage Perl (Langage pratique d'extraction et de rapport, Practical Extraction and Report Language, créé par Larry Wall en 1987) est un langage optimisé pour extraire des informations de fichiers texte et imprimer des rapports basés sur ces informations [Perl].

Après avoir effectuer le cahier des charges de l'interface, il a fallu voir à l'implémenter, et donc se remettre dans le HTML et approfondir le Perl pour utiliser le CGI.
Pour les spécifications du programme, il s'agissait en fait d'avoir un outil pouvant afficher certaines informations de plusieurs fichiers, selon des critères désignés.
Une interface Perl CGI a déjà été développée au LIMSI, pour le projet [MorTAL], avec ce type de spécifications.
L'étude est donc tout d'abord passée par l'observation du code de cette interface (visible à [MorTaLInt]) composé d'un dictionnaire, de trois bibliothèques et d'un moteur d'analyse.
Ce programme m'a permis d'avoir le code HTML nécessaire à l'affichage de mes données (boîtes, menus déroulants…).
Cependant cette interface était un galop d'essai pour apprendre le langage PERL. N'étant que débutante en programmation, de l'aide m'a été fournie par des informaticiens chevronnés.
J'ai donc effectué une partie de l'interface, mais de nombreux remodelages et corrections ont été apportés par une aide extérieure.

III.3.2. Fonctionnement d'INCA

Cette partie décrit le fonctionnement de l'interface au niveau utilisateur aussi bien qu'au niveau algorithmique. Le code de l'interface est disponible dans l'annexe.

III.3.2.1. Programmation

L'interface est constituée de deux modules ; un moteur qui traite les données (fichier lex.cgi) et une bibliothèque (package Inca.pm) gérant l'affichage de ses données. L'interface utilise le module Perl CGI pour construire les pages HTML.
Les données brutes (fichiers XML) sont tout d'abord traitées par lex.cgi qui effectue les sélections nécessaires selon les paramètres entrés par l'utilisateur. Ce module transmet ensuite les données nécessaires au module d'affichage de l'interface.
Le principe de navigation entre les pages est fondé sur les formulaires HTML traités par le serveur Web qui garde en mémoire les paramètres d'une page à une autre. Lors de l'entrée de

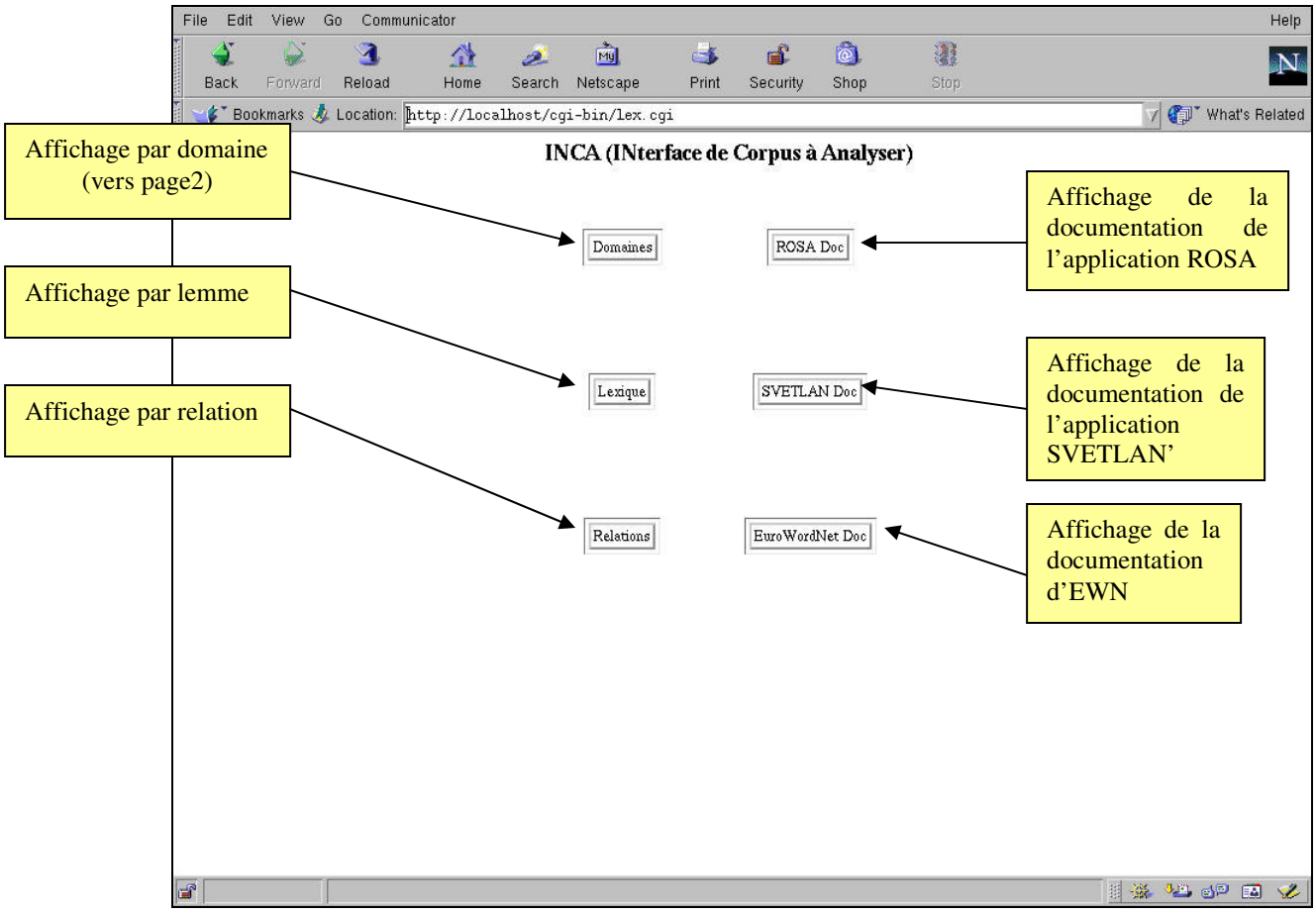
nouveaux paramètres, le serveur Web appelle le fichier lex.cgi qui traite à nouveau les fichiers bruts selon les paramètres envoyés. Une fois traitées avec les nouveaux paramètres, les données obtenues sont renvoyées au module d'affichage qui créé une page HTML et la renvoie au navigateur par l'intermédiaire du serveur Web.
Les paramètres renvoyés sont les valeurs nécessaires à l'affichage d'une autre page selon les choix de l'utilisateur. Ils peuvent donc être « page » de valeur : Défaut, Accueil, Domaines 111, etc. (nécessaire à la détermination du modèle de page contenu dans l'interface), ou principalement « lemme », « domaines », « verbe », « nbraggr » pour faire les choix nécessaires dans les données des fichiers XML.

Problèmes rencontrés

De nombreux problèmes se sont posés lors de l'implémentation de cette interface, principalement des problèmes de vitesse d'exécution, dus à la lourdeur des fichiers XML. En général, lors de l'affichage de gros fichiers, on ne choisit pas ce format pourtant très répandu.
On a préféré à l'usage de parsers spécifiques à l'XML, l'utilisation d'expression régulières courantes en PERL pour des raisons pédagogiques, mais celles-ci se sont avérées de surcroît plus efficaces lors de traitements de fichiers importants.

III.3.2.2. Utilisateur

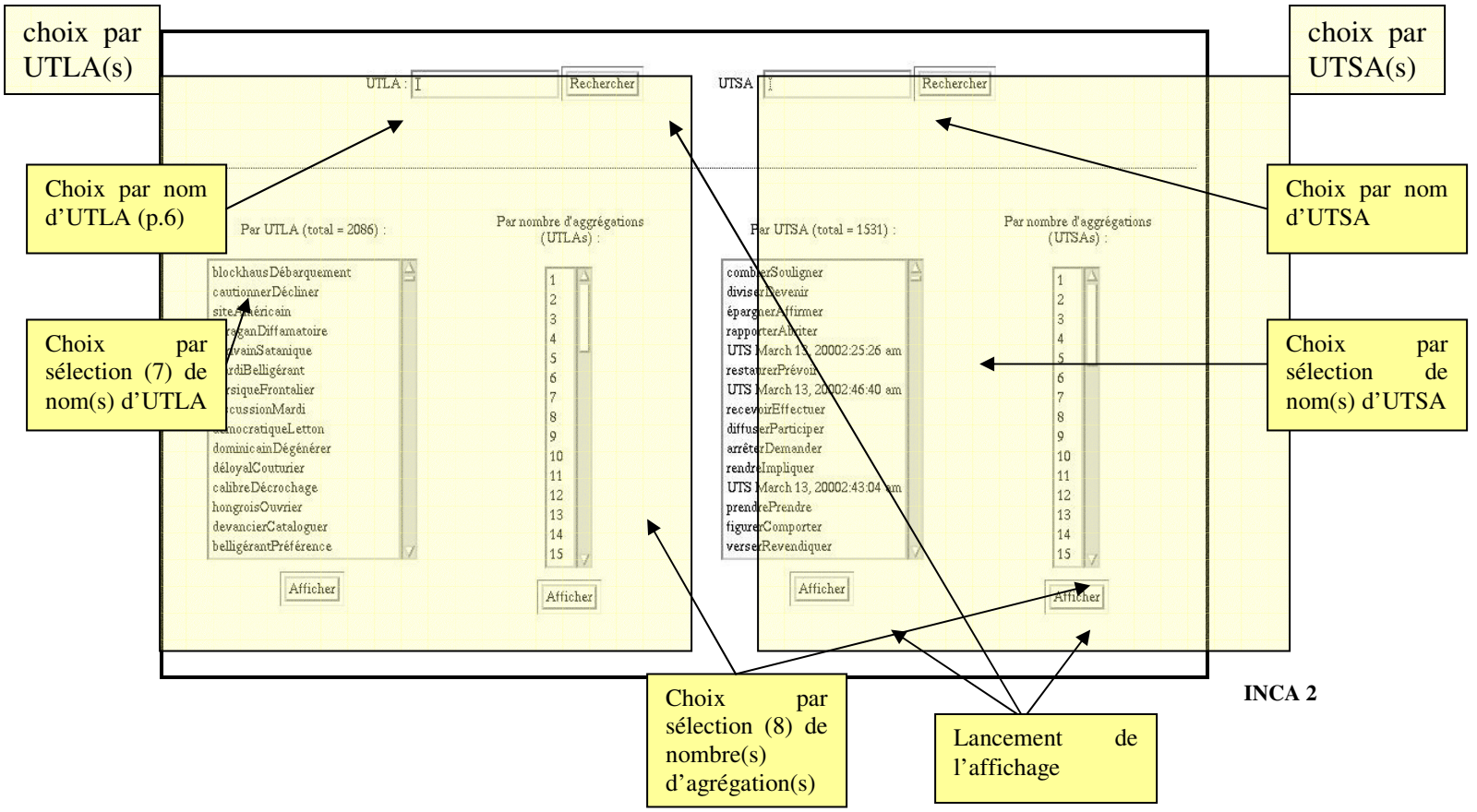
La page d'accueil :



INCA 1

Elle comprend tous les boutons nécessaires aux différents types d'affichage.
Ces différentes formes d'affichage permettent de parcourir les fichiers reliés en mettant en exergue un élément et ses propriétés.
« Domaines » représente un ensemble de lemmes ; « lexique » constitue la base de données de lemmes individuels, quant à « relations », ce bouton permet de mettre en valeur les liens entre les lemmes. Chacun de ces éléments sont appréhendables au travers des autres interfaçage, mais d'une façon moins adéquate.
Une documentation est affichable pour rappeler les fondements de ROSA, SVETLAN' ou EWN qui sont les applications et bases de données comprenant les données affichables grâce à cette interface.

Page de choix par domaine :

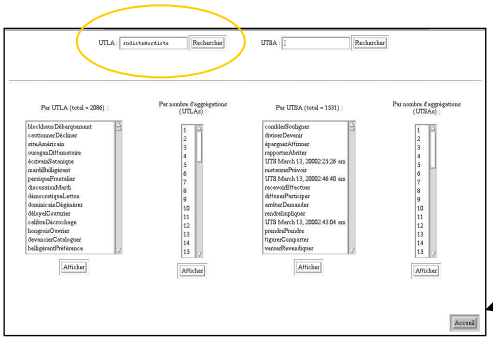


INCA 2

C'est la page de choix du domaine à afficher. Le programme permet d'en afficher plusieurs à la fois, selon différents critères ; le nom de domaine, ou le nombre d'agrégations faites pour constituer ce domaine. Cependant, on ne peut ni combiner les recherches entre UTLAs et UTSAs, ni les recherches entre domaines « nommés » et nombre d'agrégations de domaine. Malgré ces limitations, on peut obtenir de nombreux renseignements utiles.
Par exemple, tous les nombres d'agrégations sont disponibles (le programme ne fait qu'afficher), mais au niveau des résultats, ils ne deviennent intéressants qu'à partir d'une dizaine d'agrégations.

Domaines INCA

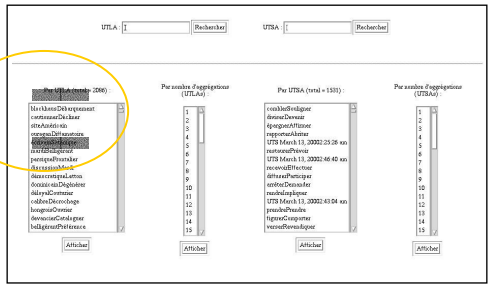
Les captures d'écran suivantes montrent l'affichage de résultats selon les différents types de choix dans « Domaines »



INCA 6

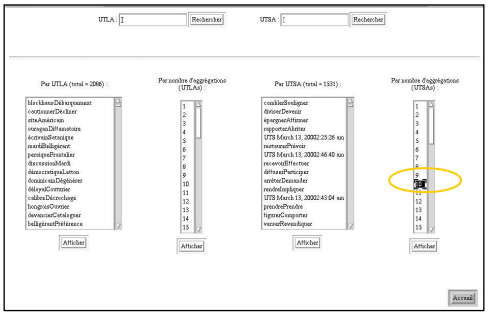
Choix par nom d'UTLA/UTSA
Toutes les pages possèdent ce bouton de retour vers la page d'accueil. Sinon, le retour se fait par le bouton de l'explorateur.

bouton de retour à la page d'accueil



Choix par sélection d'UTLA(s) – idem pour UTSA-

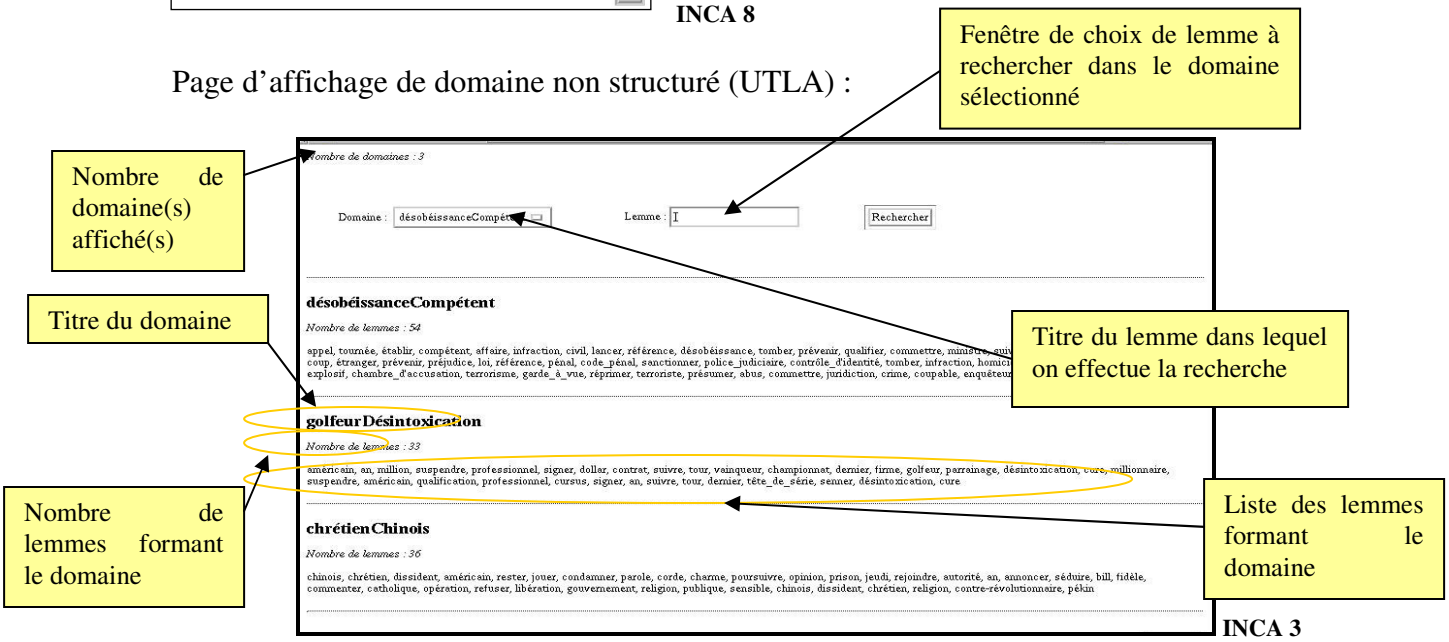
INCA 7



Choix par sélection(s) d'agrégation(s)

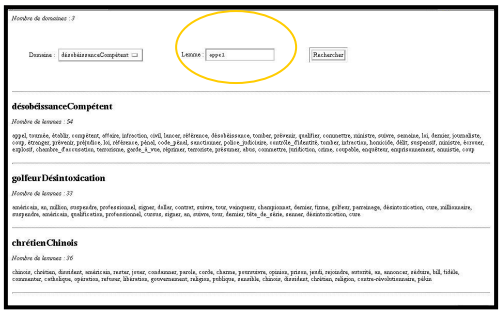
INCA 8

Page d'affichage de domaine non structuré (UTLA) :



INCA 3

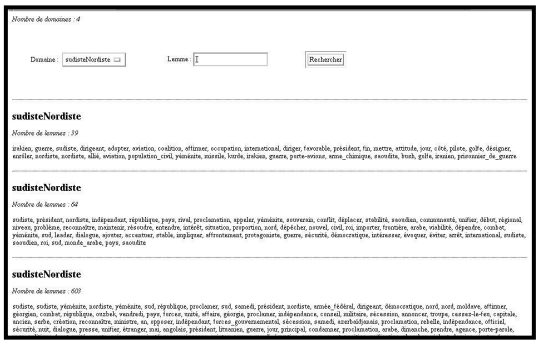
Chaque domaine sélectionné, ou répondant au(x) critère(s) de choix, est affiché sur cette page.
La recherche de lemme est utile lorsque l'on a beaucoup de lemmes dans le domaine sélectionné (parfois plus de mille). Cette recherche affiche les informations sur un lemme.



Page d'affichage de domaine avec choix pour visualisation des informations d'un lemme

INCA 4

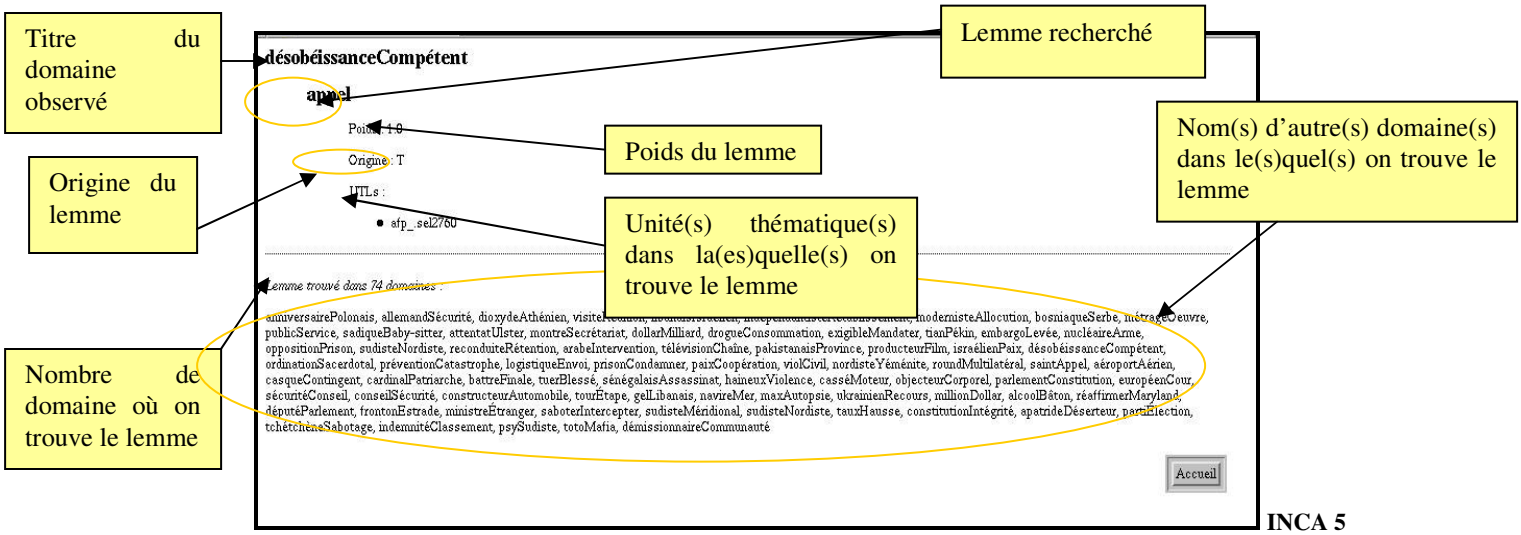
Visualisation de plusieurs domaines de même nom (recherche par nom de domaine) :



Lorsque l'on a plusieurs domaines de même nom, tous sont affichés. Le nombre de lemmes de chaque domaine diffère alors.

INCA 9

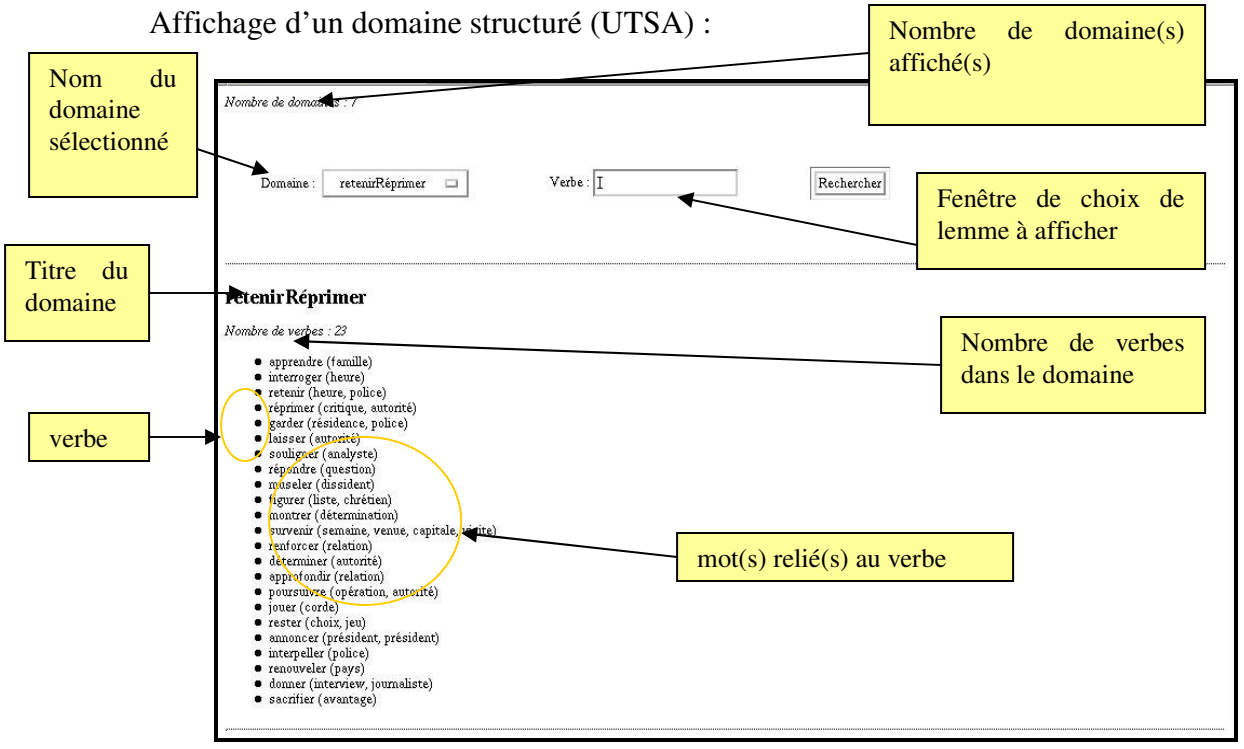
Visualisation des informations d'un lemme dans une UTLA :



INCA 5

Le détail des informations d'un lemme présent dans un domaine se présente de la sorte.
L'origine du domaine ne peut-être que T ou I, c'est-à-dire que l'origine est Textuelle ou Inférée (par un calcul). On eut ainsi avoir dans un domaine deux occurrences du même lemme, mais avec deux origines différentes.
Lorsque aucune occurrence du lemme recherché n'est trouvée, on a alors un message « aucune occurrence trouvée », mais les domaines autres dans lesquels on peut trouver le mot sont listés.

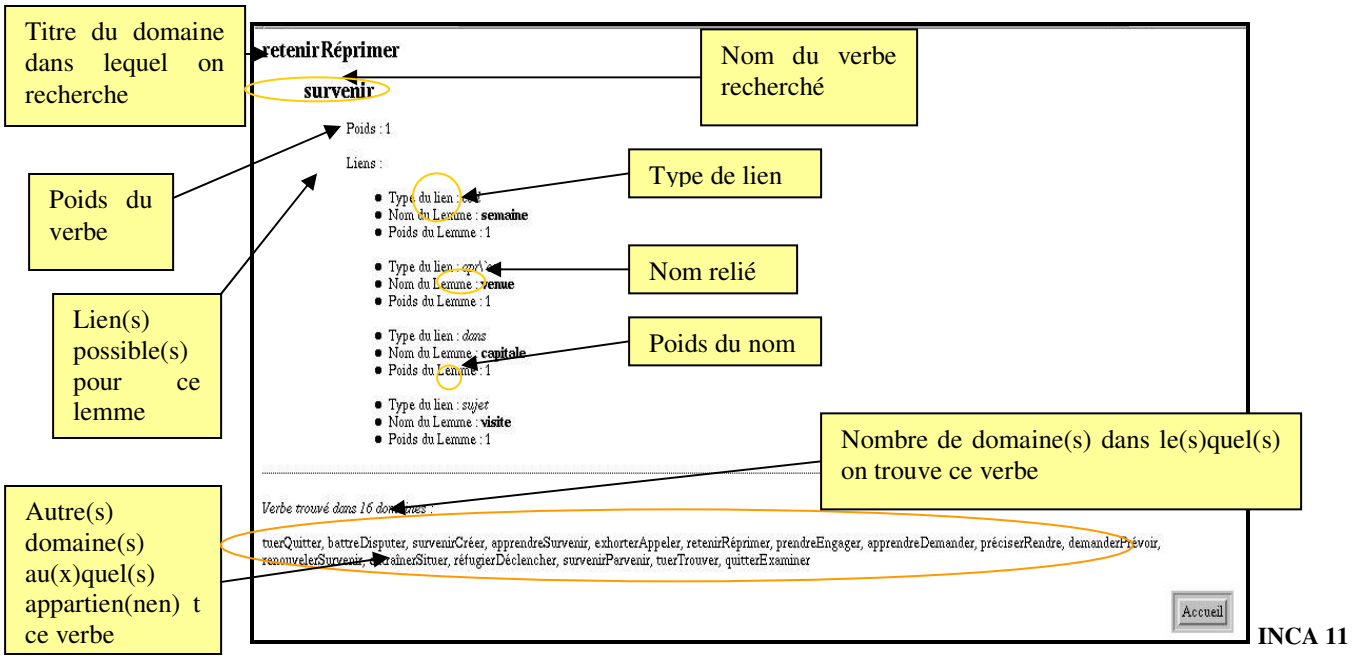
Affichage d'un domaine structuré (UTSA) :



INCA 10

Les lemmes des UTSAs ne sont que des verbes auxquels sont reliés syntaxiquement des noms. Les verbes sont donc affichés avec leurs noms reliés entre parenthèses. Une recherche par lemme est possible ici aussi.

Visualisation des informations d'un lemme dans un domaine structuré :



INCA 11

L'affichage du lemme dans une unité structurée est proche de celui du lemme dans une UTLA, à part que les liens sont aussi affichés. Pour voir le lemme avec tous ses liens possibles, il faut utiliser un autre type de recherche que « domaines » (« lexique »).
Les liens possibles affichés avec un lemme sont sujet, COD ou prép. (avec le nom de la préposition) puisque se sont les liens syntaxiques définis dans SVETLAN'.

III.4. Autres bases de corpus

III.4.1. EWN_fr.xml

Un stagiaire de l'IIE (Vincent Barbier) a convertit en XML, pour ses propres besoins, la base de données EWN. Vu que j'ai intérêt aux relations, il a accepté de la partager. Grâce à son format, elle pourra rapidement être ajoutée à INCA et permettra de diversifier et relier les relations déjà présentées dans le corpus.
C'est un fichier d'environ 6Mo, puisque seul le fichier ayant trait au français est nécessaire pour notre étude. Il contient environ 244700 lignes, soit environ 32800 « mots » ou variants et leurs liens.
Certaines données d'EWN ont été simplifiées, par exemple les liens « reversed ». Ces liens ont été générés automatiquement par l'inversion d'une relation entre deux noms (une relation « X holonyme Y » devient « Y holonyme X » automatiquement), et sont notés par exemple : « holo_reversed » dans EWN. Dans notre fichier XML, ils ont été simplifiés en une relation classique (ex : holo_reversed ➔holonymie).
Exemple du fichier :

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<dictionnaire>
<word offset="4604" cat="n" ILI="2619896-n">
<synset>
<variant>"casque 1"</variant>
</synset>
<relations>
<has_hyperonym>"n capuchon 2"</has_hyperonym>
<has_hyponym>"n casque 2"</has_hyponym>
<has_hyponym>"n casque de motocycliste
1"</has_hyponym>
<eq_synonym>"n 2619896"</eq_synonym>
</relations>
</word>
<word offset="4605" cat="n" ILI="2620140-n">
<synset>
<variant>"casque de motocycliste 1"</variant>
</synset>
<relations>
<has_hyperonym>"n casque 1"</has_hyperonym>
<eq_synonym>"n 2620140"</eq_synonym>
</relations>
</word>
<word offset="4606" cat="n" ILI="2621207-n">
<synset>
<variant>"héroïne 1"</variant>
<variant>"came 1"</variant>
</synset>
<relations>
<has_hyperonym>"n drogue dure 1"</has_hyperonym>
<eq_synonym>"n 2621207"</eq_synonym>
</relations>
</word>
<word offset="3999" cat="n" ILI="2393768-n">
<synset>
<variant>"bicyclette 1"</variant>
</synset>
<relations>
<has_hyperonym>"n véhicule à roue
1"</has_hyperonym>
<has_hyponym>"n bicyclette 2"</has_hyponym>
<has_hyponym>"n moto 1"</has_hyponym>
<has_hyponym>"n scooter 1"</has_hyponym>
<has_hyponym>"n tricycle 1"</has_hyponym>
<eq_synonym>"n 2393768"</eq_synonym>
</relations>
</word>
<word offset="772" cat="n" ILI="166345-n">
<synset>
<variant>"déplacement 9"</variant>
<variant>"voyage 4"</variant>
</synset>
<relations>
<has_hyperonym>"n déplacement 1"</has_hyperonym>
<has_hyponym>"n marche 3"</has_hyponym>
<has_hyponym>"n conduite 2"</has_hyponym>
<has_hyponym>"n conduite en moto 1"</has_hyponym>
<has_hyponym>"n voyage 2"</has_hyponym>
<eq_synonym>"n 166345"</eq_synonym>
<eq_generalization>"n 1911"</eq_generalization>
</relations>
</word>
<word offset="23107" cat="v">
<synset>
<variant>"utiliser un ordinateur 1"</variant>
</synset>
<relations>
<has_hyperonym>"v recourir à 1"</has_hyperonym>
<has_hyponym>"v annuler 8"</has_hyponym>
<has_hyponym>"v deziper 1"</has_hyponym>
<has_hyponym>"v ziper 1"</has_hyponym>
</relations>
</word>
</dictionnaire>
```

Prés.5 EWN.xml

III.4.2. Corpus de test

Le corpus de texte est composé de quatre articles [AFP] (voir Annexes 4). Ils sont lemmatisés à la main pour le traitement du corpus par les outils. C'est-à-dire que seuls les mots pleins ont un intérêt dans le test, afin de garder un équilibre entre les outils (ROSA, SVETLAN', EWN) qui ne comportent que des entrées lemmatisées et constituées par des mots pleins ; Si les mots outils ou les mots différents des lemmes des systèmes étaient considérés, ils ne feraient que créer du bruit. On conserve ainsi les mots sous forme de vedettes normalisées (verbes à l'infinitif), et seules (suppression des déterminants). Les chiffres ne sont pas conservés, que ce soit des déterminants ou des adjectifs, qu'ils soient sous forme de chiffre ou de lettre.

III.5. Problèmes rencontrés

III.5.1. Corpus

A la base, la fenêtre de choix de domaine mettait côte à côte les domaines UTLAs et UTSAs afin que des recherches combinées soient faites sur le même nom de domaine. Cependant, à l'affichage, nous nous sommes rendus compte que les noms de domaines ne correspondaient pas, ayant été recalculés par l'application SVETLAN'. On a donc des noms de domaines Verbe/Nom, Nom/Verbe ou Nom/Nom Verbe/Verbe dans ROSA, alors que dans SVETLAN', ils sont tous Verbe/Verbe.
Une autre solution était d'apparier les domaines sur leurs UTLs d'origine (en effet, les fichiers de base étaient équivalents et choisis en conséquence) mais celle-ci n'était pas récupérable dans les fichiers résultats disponibles. Il a donc fallu changer le type de recherche et le traitement, certaines hypothèses n'étant plus directement observables.

Certains problèmes ont été constatés au niveau des fichiers de résultats, comme des problèmes au niveau de la lemmatisation (verbe gazer ➔ Gaza) ou un balisage problématique (balisage d'UTL en UTSA). Vu que les analyses ne sont pas complètement automatisées, les erreurs de ce type dans les fichiers étudiés seront contournées. Cependant, il est clair qu'avec une automatisation de ce système, il faudra supprimer ce genre d'inconsistances au niveau du corpus.

III.5.2. Occurrences

De même, avec une première visualisation des fichiers, on se rend compte (alors que la logique de ROSA ou SVETLAN' laisse à penser que ce n'est pas possible) que l'on trouve plusieurs occurrences d'un même mot dans un même domaine.
Il a donc fallu adapter le traitement et principalement le calcul du poids pour que soit prise en compte cette multiplicité, sans pour autant la rendre primordiale.
La présence de doublons à tous les niveaux des analyses rend parfois les coefficients difficiles à appréhender.

III.5.3. Mise en place des tests

Informatiquement :

Au fur et à mesure de la mise en place des tests, il a fallu ajouter des scripts, ou modifier des affichages afin d'avoir en main toutes les données nécessaires à l'analyse.
Par exemple un appariement automatique des domaines, la récupération de certains poids, le focus sur les différents éléments à afficher entraîne des modifications de script parfois profondes.

Locuteur

L'analyse grâce à des locuteurs est beaucoup plus contraignante à mettre en place qu'une analyse automatique ou sur un outil. Elle prend plus de temps, demande de trouver des cobayes volontaires avec une unicité d'âge, une répartition équivalente des sexes, etc.

Une fois toutes les parties du corpus observées et traitées, elles sont analysées par des scripts Perl permettant de valider les hypothèses précédentes. Les résultats sont présentés ci-après.

IV. Résultats de l'analyse

Afin de calculer les résultats suivants, différents scripts ont été écrits. Chacun s'applique à un type de fichier de données XML.
Tout d'abord, seul un script de choix de domaines était prévu (voir Annexes 7). Mais vu le nombre de domaines sélectionnés et surtout le nombre de mots dans chaque domaine, il a été nécessaire d'effectuer tous les appariements et les calculs automatiquement.

IV.1. Sélection des contraintes et des valeurs

Les analyses définies précédemment se sont avérées encore trop imprécises à la suite des premiers résultats obtenus. C'est pourquoi il a été ajouté de nouveaux facteurs pour renforcer les contraintes et donc la finesse sur la sélection des domaines et des mots nécessaires à l'accès à nos mots manquants.

Trois contraintes sont sélectionnables, ayant chacune deux valeurs. Les scripts d'analyse permettent de combiner ces contraintes entre elles et génèrent ainsi autant de fichiers de résultats que de combinaisons possibles (2^3 solutions possibles pour ROSA et autant pour SVETLAN').

Les deux premières contraintes viennent directement des résultats de ROSA et SVETLAN' et étaient le plus souvent utilisées dans l'analyse de leurs résultats.

Le nombre d'Agrégations d'un domaine permet d'augmenter l'importance d'un domaine par rapport à un autre puisqu'il indique que celui-ci possède plus de segments de discours pris en compte pour ROSA, ou le plus d'unités thématiques similaires pour SVETLAN'.
Dans notre corpus, le nombre de domaines total est de 2086 pour ROSA et 1531 pour SVETLAN', avec une majorité de domaines à une seule agrégation. En effet, on a environ 75% de domaines d'une agrégation, et ainsi seuls 507 domaines possèdent de 2 à 480 agrégations dans ROSA pour 357 domaines de 2 à 312 agrégations dans SVETLAN'.
En général, plus le nombre d'agrégations est important, plus le nombre de mots constituant le domaine est important. Par exemple, dans ROSA, au sein du domaine « battreFinale » qui possède le maximum d'agrégations, on compte 3837 lemmes.

Le Poids d'un lemme représente son importance au sein d'un domaine, c'est pourquoi il semble utile de poser une contrainte sur ce poids afin de réduire le bruit dans nos résultats dû à des lemmes trop courants. Ce poids ne correspond pas aux poids des mots calculés dans les applications ROSA ou SVETLAN', c'est le nombre d'agrégations d'un lemme ou d'un verbe dans un domaine, ou le nombre d'agrégations d'un nom par rapport à un verbe. Dans notre cas, un poids se rapporte à un lemme tandis qu'agrégation référera à un domaine.
Le poids minimum, et le plus représenté, est de : 1.0. Le poids maximum trouvé, dans le domaine le plus important de ROSA « battreFinale » est de : 472. Cependant, dans ce

domaine, le nombre de mots de poids 1 est de : 1923, tandis que seuls 1914 lemmes possèdent un poids supérieur à 1. Environ 50% des lemmes peuvent créer du bruit dans ce domaine.

La dernière contrainte est liée au choix de domaine à étudier. Il spécifie un quota minimum de mots du Contexte à retrouver dans le domaine qui sera analysé. Il permet un premier choix qui restreint le nombre de domaines et donc le temps d'analyse en réduisant le bruit créé par des domaines de moindre importance. Cette contrainte a été modifiée par rapport à l'analyse initialement prévue, puisque après analyse des fichiers de ROSA, on a pu se rendre compte que malgré la contrainte sur la moitié des mots du contexte à apparier avec des domaines il en résultait encore un grand nombre de domaines à étudier.
Les résultats suivants présenteront donc des analyses avec deux valeurs de la contrainte « Contexte », l'une à ½ (la moitié des mots du contexte présents dans un domaine sélectionne ce dernier pour l'analyse), l'autre à ¾. La contrainte à 0 n'est de toute façon pas envisageable et n'est présentée qu'à titre indicatif et de témoin.

Certaines de ces combinaisons engendrent des résultats intéressants, d'autres sont trop strictes ou trop larges. L'observation des résultats nous permet de conserver certains facteurs valables pour une analyse et un choix lexical avec une granularité adéquate.
Les résultats présentés regroupent selon l'origine du fichier de données observé (fichier issu de ROSA, puis SVETLAN', puis EWN), le nombre de mots manquants trouvés par rapport au nombre total de mots dans un domaine. Le fichier affiché et ses résultats est en fait un de ceux qui présentent globalement les résultats les plus significatifs sur les analyses puisque d'une façon générale les fichiers traités conduisent au même type de conclusions.

IV.1.1. ROSA

Analyse sur les fichiers Irak (TàT, Liste de mots manquants)

			nombre de domaines affichés	nombre de domaines total	coefficient domaine	moyenne mots trouvés
Agrégation = 0	**Poids = 0**	**Contexte = 0**	2036	2086	0,9760	0,4401
Agrégation = 0	**Poids = 0**	**Contexte = 1/2**	28	2086	0,0134	5,6071
Agrégation = 0	**Poids = 0**	**Contexte = 3/4**	10	2086	0,0048	6,9
Agrégation = 0	**Poids > 1**	**Contexte = 1/2**	10	2086	0,0048	5,3
Agrégation = 0	**Poids > 1**	**Contexte = 3/4**	5	2086	0,0024	6,2
Agrégation >1	**Poids = 0**	**Contexte = 1/2**	28	507	0,0552	5,6071
Agrégation >1	**Poids = 0**	**Contexte = 3/4**	10	507	0,0197	6,9
Agrégation >1	**Poids > 1**	**Contexte = 1/2**	10	507	0,0197	5,3
Agrégation >1	**Poids > 1**	**Contexte = 3/4**	5	507	0,0099	6,2
Agrégation >10	**Poids = 0**	**Contexte = 1/2**	28	82	0,3415	5,6071
Agrégation >20	**Poids = 0**	**Contexte = 3/4**	10	42	0,2381	6,9
Agrégation >80	**Poids = 0**	**Contexte = 3/4**	10	17	0,5882	6,9
Agrégation >90	**Poids = 0**	**Contexte = 3/4**	9	15	0,6000	7,11
Agrégation >100	**Poids = 0**	**Contexte = 3/4**	9	14	0,6429	7,11

Tab.1 ROSA Résultats contraintes_valeurs_Irak

Facteurs à tendance positive

Facteur à tendance négative

IV.1.2. SVETLAN'

Analyse sur les fichiers Irak (TàT, Liste de mots manquants)

			nombre de domaines affichés	nombre de domaines total	coefficient domaine	moyenne mots trouvés
Agrégation = 0	**Poids = 0**	**Contexte = 0**	973	1531	0,6355	0,2765
Agrégation = 0	**Poids = 0**	**Contexte = 1/2**	10	1531	0,0065	4,6
Agrégation = 0	**Poids = 0**	**Contexte = 3/4**	4	1531	0,0026	4
Agrégation = 0	**Poids > 1**	**Contexte = 1/2**	0	0	0,0000	0
Agrégation = 0	**Poids > 1**	**Contexte = 3/4**	0	0	0,0000	0
Agrégation >1	**Poids = 0**	**Contexte = 1/2**	10	357	0,0280	4,6
Agrégation >1	**Poids = 0**	**Contexte = 3/4**	4	357	0,0112	4
Agrégation >1	**Poids > 1**	**Contexte = 1/2**	0	0	0,0000	0
Agrégation >1	**Poids > 1**	**Contexte = 3/4**	0	0	0,0000	0
Agrégation >10	**Poids = 0**	**Contexte = 1/2**	10	58	0,1724	4,6
Agrégation >20	**Poids = 0**	**Contexte = 3/4**	4	30	0,1333	4
Agrégation >80	**Poids = 0**	**Contexte = 3/4**	4	13	0,3077	4
Agrégation >100	**Poids = 0**	**Contexte = 3/4**	4	11	0,3636	4
Agrégation >200	**Poids = 0**	**Contexte = 3/4**	3	5	0,6000	4,33
Agrégation >250	**Poids = 0**	**Contexte = 3/4**	1	2	0,5000	4
Agrégation >300	**Poids = 0**	**Contexte = 3/4**	0	0	0,0000	0

Tab.2 SVETLAN' Résultats contraintes_valeurs_Irak

Facteur à tendance positive

Facteurs à tendance négative

Les résultats ci-dessus présentent le nombre de domaines au total contenus dans ROSA ou SVETLAN', le nombre de domaines affichés, c'est-à-dire le nombre de domaines retenus par un appariement à un contexte, ainsi que deux moyennes.
Le « coefficient domaine » est obtenu par le rapport :

$$\frac{\text{Nombre de domaines affichés}}{\text{Nombre de domaine total}}$$

et représente la finesse du choix de domaine (il doit être le plus proche possible de 0).

La « moyenne mots trouvés » ou

$$\frac{\text{Nombre de mots manquants trouvés au total}}{\text{Nombre de domaines affichés}}$$

représente le nombre moyen de mots que l'on retrouve dans tout le fichier, grâce aux réglages sélectionnés (à droite dans le tableau) la moyenne doit être la plus proche possible de 10 ; nombre de mots manquants).

En observant ces deux tableaux de résultats, on peut constater que les contraintes entraînent d'importantes différences selon le type de données analysé.
En effet, on peut considérer que pour une analyse fondée sur des fichiers issus de ROSA deux facteurs ont un rôle positif et permettent de raffiner le nombre de domaine proprement. Il s'agit du facteur Contexte et du facteur Poids.
La contrainte sur la quantité de Contexte à apparier sur le fichier de ROSA permet de diminuer de façon conséquente le bruit créé par l'utilisation de domaines non adaptés dans l'analyse. De plus, jouer sur ce facteur peut permettre de réduire au moins de moitié les fichiers provocant du bruit dans le système. En effet le facteur contraignant à 50% d'appariement peut être resserré en contraignant l'appariement du contexte sur la totalité du fichier à 75% ; grâce à ce facteur, on peut améliorer de 50% à 64% les résultats. Pour SVETLAN', on constate qu'une réduction de 60% du bruit est possible grâce à ce facteur.
On considèrera donc ce facteur validé pour la suite des expériences. Pour le moment, on peut décider du réglage à 75% de l'appariement correct, puisqu'il semble que les résultats soient intéressants et ne pâtissent pas de la réduction des domaines de sélection. Ceci sera à confirmer par une analyse des mots trouvés en détail et non sur des statistiques générales de leur appariement.
Le second facteur intéressant semble être le Poids, puisqu'il pondère l'ensemble des domaines exactement dans les mêmes proportions que le facteur Contexte. On peut donc penser que le facteur est utilisable seul.
Un bémol est à apporter puisque, lors d'une utilisation seul, il obtient un score inférieur à celui du facteur Contexte seul. Lorsqu'il est combiné avec Contexte, le score remonte (de 5,3 à 6,2 au lieu de 6,9 lorsque Contexte est seul).
Avec une analyse sur les mots appariés, on pourra décider de l'utilisation ou non de ce facteur.

Un autre point négatif à cette contrainte est les résultats qu'il apporte dans SVETLAN'. En effet, il serait plus objectif d'utiliser les mêmes contraintes pour les différentes analyses, et dans ce cas ce facteur est catastrophique puisque tous les résultats sont alors nuls. Quels que soient les résultats des analyses avec un fichier SVETLAN' et les contraintes sur le Contexte et l'Agrégation, les résultats sont rendus nuls par l'utilisation d'un facteur appariant les mots sur un poids supérieur à 1 (qui est le poids minimum d'un mot dans les fichiers SVETLAN' et ROSA). Les analyses sur le facteur poids n'ont donc pas été plus poussées, ce facteur étant déjà sélectionné implicitement dans SVETLAN' qui n'a conservé que les mots de plus fort poids dans un domaine.
Le facteur Agrégation est lui tout à l'inverse dans la mesure où il n'a absolument aucune incidence sur les résultats à moins de lui affecter une valeur supérieure à 80 dans les fichiers ROSA et pour SVETLAN' il faut dépasser les 200, sachant qu'à 250 les résultats baissent et deviennent nuls à 300.
Ce facteur permet cependant d'élaguer assez largement le nombre de domaines à analyser ce qui influe sur le temps d'exécution. Dans la mesure où nos fichiers d'exemples à traiter ne comportent pas des millions de mots, on considèrera que ce facteur à une utilité limitée. On le conservera donc avec une valeur >1 afin de réduire le bruit le plus important créé par des fichiers de peu d'importance dans les systèmes ROSA et SVETLAN' et de conserver une certaine qualité de résultats.
Cette attitude est raisonnable dans notre cas vu la taille du corpus, mais ce facteur doit être pris en compte dans le cas d'un corpus important. De plus, le temps d'exécution est dans notre cas ralenti par un autre facteur, l'optimisation des scripts ; ne sachant quel facteur prévaut ici, autant ne pas stigmatiser la contrainte Agrégation.

Au niveau des résultats en détail, il faut observer les listes de mots manquants trouvés de plus près et dans les différents cas d'analyse. En effet, le nombre de mots trouvés peut être équivalent, mais ne pas représenter les mêmes lemmes dans les différents cas.

IV.1.3. Lemmes manquants

D'après l'observation des données suivantes, on peut confirmer que l'utilisation du facteur Contexte à 75% est correcte, puisque dans le cas de ROSA, les mêmes lemmes et en nombre d'occurrences identiques sont trouvés avec cette valeur, mais un nombre inférieur de domaines est à analyser. Dans le cas de SVETLAN', le nombre de lemmes trouvés reste suffisant.

L'Agrégation semblait avoir le même type de conséquences sur le nombre de domaines à analyser en observant les tableaux de résultats précédents. Cet effet générique sur les domaines semble confirmé ici, et malgré une réduction importante du nombre d'occurrences de lemmes, ce facteur semble trancher dans les résultats d'une façon homogène, conservant plus ou moins le nombre de mots trouvés initialement.

Le facteur Poids fait apparaître encore une fois dans ces résultats sa puissance, puisque c'est encore le facteur qui fait disparaître un mot trouvé, le passant de 2 occurrences à 0. C'est donc peut être une contrainte trop forte dans nos fichiers où des mots très divers sont recherchés.
Cependant, dans le cas d'une recherche précise, il est certainement intéressant d'en disposer.

Analyse sur les fichiers Irak (TàT, Liste de mots manquants et ROSA)

	Agrég. = 0 Poids = 0 Cont. = ½	Agrég. = 0 Poids = 0 Cont. = 3/4	Agrég. = 0 Poids > 1 Cont. = 1/2	Agrég. = 0 Poids > 1 Cont. = 3/4	Agrég.>1 Poids = 0 Cont. = ½	Agrég.>1 Poids = 0 Cont. = ¾	Agrég.>1 Poids > 1 Cont. = 1/2	Agrég.>1 Poids > 1 Cont. = 3/4	Agrég.>100 Poids = 0 Cont. = 3/4
Partisan	11	5	3	2	11	5	3	2	5
Lundi	28	10	10	5	28	10	10	5	9
Militaire	25	10	10	5	25	10	10	5	9
Force	16	7	5	4	16	7	5	4	7
Amende	7	5	3	1	7	5	3	1	5
Lancer	22	10	10	5	22	10	10	5	9
Confirmer	19	8	6	5	19	8	6	5	7
Étudier	9	4	1	1	9	4	1	1	4
Représenter	18	8	5	3	18	8	5	3	8
Enfreindre	2	2	0	0	2	2	0	0	1

Tab.3 ROSA Résultats Lemmes_contraintes_valeurs_Irak

Contexte
Poids
Agrégation

Analyse sur les fichiers Irak (TàT, Liste de mots manquants et SVETLAN')

	Agrég. = 0 Poids = 0 Cont. = ½	Agrég. = 0 Poids = 0 Cont. = 3/4	Agrég. = 0 Poids > 1 Cont. = 1/2	Agrég. = 0 Poids > 1 Cont. = 3/4	Agrég.>1 Poids = 0 Cont. = 1/2	Agrég.>1 Poids = 0 Cont. = ¾	Agrég.>1 Poids > 1 Cont. = 1/2	Agrég.>1 Poids > 1 Cont. = 3/4	Agrég.>200 Poids = 0 Cont. = 3/4
Partisan	2	0	0	0	2	0	0	0	0
Lundi	9	4	0	0	9	4	0	0	3
Militaire	0	0	0	0	0	0	0	0	0
Force	3	2	0	0	3	2	0	0	2
Amende	3	0	0	0	3	0	0	0	0
Lancer	9	4	0	0	9	4	0	0	3
Confirmer	8	3	0	0	8	3	0	0	3
Étudier	4	1	0	0	4	1	0	0	1
Représenter	8	2	0	0	8	2	0	0	1
Enfreindre	0	0	0	0	0	0	0	0	0

Tab.4 SVETLAN' Résultats Lemmes_contraintes_valeurs_Irak

Contexte
Agrégation
Poids

IV.1.4. EuroWordNet

Analyse sur les quatre fichiers d'exemple (TàT, Liste de mots manquants et fichier xml d'EWN)

	1 Lien		2 Liens	
Irak	Nombre _mots trouvés	**7**	Nombre _mots trouvés	**8**
	Nombre mots total	**32809**	Nombre mots total	**32809**
	moyenne	**0,0002**	moyenne	**0,0002**
Liberia	Nombre _mots trouvés	**7**	Nombre _mots trouvés	**10**
	Nombre mots total	**32809**	Nombre mots total	**32809**
	moyenne	**0,0002**	moyenne	**0,0003**
Lockerbie	Nombre _mots trouvés	**8**	Nombre _mots trouvés	**9**
	Nombre mots total	**32809**	Nombre mots total	**32809**
	moyenne	**0,0002**	moyenne	**0,0003**
Nicolas	Nombre _mots trouvés	**6**	Nombre _mots trouvés	**7**
	Nombre mots total	**32809**	Nombre mots total	**32809**
	moyenne	**0,0002**	moyenne	**0,0002**

Tab.5 EWN Résultats contraintes_valeurs_Ir.Lib.Loc.Ni

Au vu de ces résultats, il est décidé de conserver les contraintes suivantes :
Contexte : valeur →3/4
Puisqu'il permet une bonne réduction du nombre de domaines.

Agrégation : >1
Pour permettre une réduction des domaines qui compensera l'absence du paramètre Poids, inutilisable dans SVETLAN'.

Dans le cas d'EWN, les paramètres sont différents. On ne peut en fait qu'influer sur le nombre de liens mis en jeu pour accéder aux mots manquants. Concrètement cela implique que l'on prenne soit les liens directs, soit les liens vers d'autres mots d'une façon récurrente.
Lorsque l'on parle d' « 1 lien », il s'agit en fait des mots que l'on peut atteindre directement grâce aux mots du contexte. Les mots trouvés font partie des hyponymes, hyperonymes ou synonymes directs du variant (vedette EWN) appartenant au contexte.
« 2 liens », c'est en fait le lancement dans l'analyse de tous ces mots récupérés directement. Ils mènent alors à d'autres mots qui sont le cas échéant appariés à la liste de mots manquants. Cette récurrence a pour conséquence un accroissement très important des données résultantes.

Par exemple :

Mot du contexte	Liens	Mots liés		Liens	Mots liés
Soldat	Synonyme	soldat	**ouvrier spécialisé**	Synonyme	ouvrier spécialisé
	Synonyme	militaire		Synonyme	O.S.
	has_hyperonym	**ouvrier spécialisé**		has_hyperonym	travailleur
	has_hyponym	artilleur		has_hyponym	aviateur
	has_hyponym	marin		has_hyponym	boulanger
	has_hyponym	appelé		has_hyponym	aéronaute
	has_hyponym	simple soldat		has_hyponym	boucher
	has_hyponym	fusilier marin		has_hyponym	calligraphe
	has_hyponym	officier militaire		has_hyponym	cuisinier
	has_hyponym	militaire		has_hyponym	artisan
	has_hyponym	ancien combattant		has_hyponym	coupeur
	antonym	civil		has_hyponym	dessinateur
				has_hyponym	éditeur
				has_hyponym	électricien
				has_hyponym	pêcheur
				has_hyponym	fondeur
				has_hyponym	chasseur
				has_hyponym	ordonnateur des pompes funèbres
				has_hyponym	lunetier
				has_hyponym	peintre
				has_hyponym	plâtrier
				has_hyponym	imprimeur
				has_hyponym	rénovateur
				has_hyponym	réparateur
				has_hyponym	marin
				has_hyponym	militaire
				has_hyponym	forgeron
				has_hyponym	technicien

Tab.6 EWN Liens

Ces traitements sont très longs, principalement lors d'un appel récurrent du fichier EWN, ce qui n'est pas négligeable lors de l'analyse d'un grand corpus.

Dans le tableau 5, le « nombre mots trouvés » est le nombre de mots appariés dans la liste manquants grâce au fichier EWN. Il est donc de 10 au maximum.
Le « nombre mots total » est la somme des 32809 variants existant dans EWN ; il sert de diviseur à la « moyenne » qui permet de représenter la quantité de mots trouvés par rapport au nombre de mots proposés.

Dans les résultats présentés lors de traitement de fichier à un ou deux liens, on constate que la récurrence n'apporte que peu de mots à la liste recherchée, par rapport au nombre de variants traités dans EWN. De plus, la moyenne reste sensiblement la même, et dans aucun cas le double appariement n'optimise les résultats. On se contentera donc d'un seul passage et de l'accès aux mots manquants par des liens directs.

IV.2. Sélection des applications par coefficient

D'après les calculs exposés dans la partie précédente « structuration de l'analyse », on obtient les résultats suivants (résultats les plus importants surlignés en orange) :

« Nom Domaine » correspond au nom d'un domaine affiché par appariement suffisant avec le contexte. Le Poids final est calculé par rapport aux coefficients définis précédemment.
La ligne en gras de « Total » permet de voir le résultat global par application, et non selon un domaine.
Dans la colonne de droite sont affichées les applications ayant servi pour l'analyse.
Les résultats EWN sont un peu différents puisqu'EWN n'est qu'un seul et grand domaine. De plus, les coefficients diffèrent un peu vu que les mots ne sont pas pondérés dans ce thésaurus. Les coefficients 1 et 2 sont identiques à ceux de ROSA et SVETLAN', mais le coefficient 3 représente le nombre de mots appartenant à la liste manquante qui sont possibles à apparier à EWN dans l'absolu. En effet, EWN ne possède pas forcément tous les lemmes proposés, puisque le lexique de cette base de données ne représente pas forcément le lexique AFP de ROSA ou SVETLAN'.
Ainsi, les coefficients indiqués ne sont pas tous comparables directement dans la mesure où les résultats d'EWN ne représentent pas strictement la même chose. On peut toujours observer les coeff1 et coeff2 qui sont, eux, identiques par leur calcul aux coefficients de ROSA et SVETLAN'. On constate d'ailleurs rapidement qu'ils sont nettement inférieurs pour EWN.

Analyse sur les fichiers Irak

	Nom Domaine	Poids_final	Coeff1	Coeff2	Coeff3
ROSA	palestinienIsraélien	0.1692	0.5	0.002	0.0057
	bosniaqueSerbe	0.2355	0.7	0.0023	0.0043
	dollarMilliard	0.2011	0.6	0.0018	0.0014
	israélienPaix	0.1701	0.5	0.0039	0.0063
	prisonCondamner	0.3031	0.9	0.0023	0.0071
	aéroportAérien	0.1693	0.5	0.0024	0.0055
	battreFinale	0.2344	0.7	0.0018	0.0015
	tuerBlessé	0.2697	0.8	0.0023	0.0068
	sudisteNordiste	0.3059	0.9	0.0025	0.0151
	partiElection	0.2687	0.8	0.0029	0.0031
Total	**10 domaines**	**0.2327**	**10**		
SVETLAN'	prévoirReprendre	0.136	0.4	0.0032	0.0049
	tuerPrendre	0.1023	0.3	0.0032	0.0037
	commanderPoursuivre	0.1362	0.4	0.0023	0.0064
	tuerTrouver	0.171	0.5	0.0038	0.0091
Total	**4 domaines**	**0.1364**	**6**		
		Poids_final	**Coeff1**	**Coeff2**	**Coeff3**
EWN		0.4927	0.7	0.0002	0.7778

Tab.7 Résultats Poids_Irak

Analyse sur les fichiers Liberia

	Nom Domaine	Poids_final	Coeff1	Coeff2	Coeff3
ROSA	palestinienIsraélien	0.2683	0.8	0.0031	0.0017
	bosniaqueSerbe	0.2354	0.7	0.0023	0.004
	dollarMilliard	0.201	0.6	0.0018	0.0013
	prisonCondamner	0.268	0.8	0.0021	0.0019
	aéroportAérien	0.1376	0.4	0.0019	0.0108
	battreFinale	0.1673	0.5	0.0013	0.0007
	tuerBlessé	0.3015	0.9	0.0025	0.0021
	réfugierCamp	0.1686	0.5	0.0037	0.002
	sudisteNordiste	0.2690	0.8	0.0022	0.0047
	partiElection	0.1688	0.5	0.0018	0.0045
Total	**10 domaines**	**0.2186**	**10**		
SVETLAN'	prévoirReprendre	0.2056	0.6	0.0048	0.0119
	commanderPoursuivre	0.2372	0.7	0.0041	0.0076
	tuerTrouver	0.2728	0.8	0.0061	0.0123
	donnerObtenir	0.1055	0.3	0.0031	0.0133
Total	**4 domaines**	**0.2055**	**8**		
		Poids_final	**Coeff1**	**Coeff2**	**Coeff3**
EWN		0.4667	0.7	0.0002	0.7

Tab.8 Résultats Poids_Liberia

Analyse sur les fichiers Lockerbie

	Nom Domaine	Poids_final	Coeff1	Coeff2	Coeff3
ROSA	palestinienIsraélien	0.2347	0.7	0.0027	0.0013
	bosniaqueSerbe	0.2365	0.7	0.0023	0.0071
	dollarMilliard	0.2681	0.8	0.0024	0.0019
	israélienPaix	0.1708	0.5	0.0039	0.0085
	prisonCondamner	0.268	0.8	0.0021	0.0018
	AéroportAérien	0.1343	0.4	0.0019	0.0009
	BattreFinale	0.2012	0.6	0.0016	0.002
	TuerBlessé	0.2346	0.7	0.002	0.0017
	DéputéParlement	0.1685	0.5	0.0027	0.0029
	SudisteNordiste	0.2348	0.7	0.0019	0.0025
	TauxHausse	0.1007	0.3	0.0012	0.0009
Total	**11 domaines**	**0.2047**	**10**		
SVETLAN'	prévoirReprendre	0.2747	0.8	0.0064	0.0178
	prévoirAtteindre	0.1378	0.4	0.0039	0.0096
	commanderPoursuivre	0.2384	0.7	0.0041	0.0110
	tuerTrouver	0.1713	0.5	0.0038	0.0101
Total	**4 domaines**	**0.2056**	**8**		
		Poids_final	**Coeff1**	**Coeff2**	**Coeff3**
EWN		0.5630	0.8	0.0002	0.8889

Tab.9 Résultats Poids_Lockerbie

Analyse sur les fichiers Nicolas

	Nom Domaine	Poids_final	Coeff1	Coeff2	Coeff3
ROSA	palestinienIsraélien	0.134	0.4	0.0016	0.0004
	bosniaqueSerbe	0.1338	0.4	0.0013	0.0002
	prisonCondamner	0.2012	0.6	0.0015	0.002
	battreFinale	0.2006	0.6	0.0016	0.0002
	tuerBlessé	0.2011	0.6	0.0017	0.0015
	sudisteNordiste	0.1338	0.4	0.0011	0.0004
Total	**6 domaines**	**0.1674**	**9**		
SVETLAN'	commanderPoursuivre	0.1351	0.4	0.0023	0.003
	apprendreDemander	0.102	0.3	0.003	0.0029
	tuerTrouver	0.136	0.4	0.0031	0.0048
Total	**3 domaines**	**0.1244**	**7**		
		Poids_final	**Coeff1**	**Coeff2**	**Coeff3**
EWN		0.4223	0.6	0.0002	0.6667

Tab.10 Résultats Poids_Nicolas

Les résultats sur l'ensemble des propositions des applications indiquent qu'en général ROSA permet de trouver plus de mots, avec un poids final supérieur à SVETLAN' (sur quatre analyses, trois posent ROSA supérieur) :

	IRAK		LIBERIA		LOCKERBIE		NICOLAS	
ROSA	**0.2327**	**10**	**0.2186**	**10**	**0.2047**	**10**	**0.1674**	**9**
SVETLAN'	**0.1364**	**6**	**0.2055**	**8**	**0.2056**	**8**	**0.1244**	**7**

Tab.11 Résultats Applis_Poids_max

Cependant, on s'aperçoit que SVETLAN' propose des résultats beaucoup plus homogènes. En effet, les éléments surlignés représentant les meilleurs coefficients appartiennent en général à un seul et même domaine.
Seul le Coeff2 est le plus souvent supérieur dans SVETLAN'. En effet, SVETLAN' propose beaucoup moins de domaines à analyser, ce qui d'ailleurs optimise les recherches en terme de vitesse d'exécution.

Au niveau d'une comparaison par domaine, on constate qu'aucun domaine ne renvoie de scores équivalents ou meilleurs que le score global où tous les domaines sont pris en compte. Pourtant, le fait de n'avoir qu'un seul domaine de pris en compte pourrait optimiser l'analyse.

En comparant le meilleur domaine de chaque application, on obtient :

	IRAK		LIBERIA		LOCKERBIE		NICOLAS	
ROSA	0.3059	0.9	0.3015	0.9	0.2681	0.8	0.2012	0.6
SVETLAN'	0.171	0.5	0.2728	0.8	0.2747	0.8	0.136	0.4

Tab.12 Résultats Domaine_Poids_max

Une possibilité envisageable serait alors de trouver les facteurs qui contraignent un seul domaine de ROSA (le meilleur) à être sélectionné. Le problème est que l'on ne peut prévoir ce domaine comme le montre le tableau d'exemple suivant, les scores trouvés à partir des éléments de calcul connus (du fichier Liberia) ne menant pas à ce domaine, mais à un domaine moindre et en général de beaucoup :

	nombre de mots trouves	nombre de mots du domaine	nombre d'agrégations	nombre de mots du contexte matchés	nombre de mots matchés du contexte / nombre total de mots	nombre de mots matchés du contexte/ nombre d'agrégation
palestinienIsraélien	8	2561	213	273	**0,1066**	**1,2817**
bosniaqueSerbe	7	3035	288	309	**0,1018**	**1,0729**
dollarMilliard	6	3366	205	231	**0,0686**	**1,1268**
prisonCondamner	8	3884	299	293	**0,0754**	**0,9799**
aéroportAérien	4	2103	119	240	**0,1141**	**2,0168**
battreFinale	5	3837	480	263	**0,0685**	**0,5479**
tuerBlessé	9	3541	337	316	**0,0892**	**0,9377**
réfugierCamp	5	1361	87	224	**0,1646**	**2,5747**
sudisteNordiste	8	3671	380	356	**0,0970**	**0,9368**
partieElection	5	2773	208	261	**0,0941**	**1,2548**

Tab.13 Résultats ROSA_domaine_unique_Liberia

En définitive, on constate pour l'évaluation des applications, que SVETLAN' donne moins d'information, qui est donc plus précise en comparaison avec la quantité d'information délivrée. On prendra dorénavant les résultats de cette application comme base puisqu'elle répond en partie à l'hypothèse 1, effectuant le meilleur accès au lexique pour le moment.
Il faut tout de même noter que l'application qui donne les meilleurs résultats est celle qui contient des liens, puisque SVETLAN' fonctionne grâce à des liens syntaxiques.
De même pour ces résultats, qu'il s'agisse de ROSA ou de SVETLAN', il ne faut pas oublier que ces tests n'ont été effectués que sur un seul fichier de résultats ; il est possible d'augmenter la quantité d'informations dans ces applications.

SVETLAN' permet un accès plus clair au mot, mais cette application est plus faible au niveau de la récupération de mots et en laisse souvent de côté (en moyenne 1/10).
Il faudrait donc optimiser SVETLAN' au niveau de la quantité de mots récupérée sans pour autant démultiplier le nombre de mots total.
Il serait possible par exemple d'étendre le lien d'un nom vers d'autres noms ou verbes afin de récupérer d'autres verbes ou d'étendre les liens d'un verbe apparié au contexte pour récupérer d'autres noms. C'est un genre de double appariement comme effectué sur EWN et qui risque donc de faire exploser le nombre de domaines renvoyés et de mots donc. Si l'on effectue ce traitement au sein du domaine présélectionné, on risque de ne pas étendre beaucoup la couverture, puisque tous les mots du domaine sont déjà appariés.

Parallèlement, malgré la limitation du nombre de domaines, le nombre de mots proposés au sein de ces domaines est très important. On pourrait donc prolonger les liens de ces mots en utilisant ceux d'EWN. En intégrant EWN on peut par ailleurs retrouver le type de liens utilisés.
Cependant, est-ce que l'information apportée par des liens typés est vraiment valable ? Qu'apporte un système qui comporte conjointement des liens typés et des liens non typés.

IV.3. Accès au lexique par intégration d'EWN à SVETLAN'

IV.3.1. La compatibilité

Avant de répondre à ses questions, il faut savoir quel type de données prendre en compte en tant que fondement de la base de données pour fusion de données.
Au vu des résultats précédents et du type d'information délivrées par les différentes bases, il semble que SVETLAN' et EWN puissent être complémentaires. Avec une analyse des fichiers Irak, on trouve par exemple :

	partisan	**lundi**	**militaire**	**force**	**amende**	**lancer**	**confirmer**	**étudier**	**représenter**	**enfreindre**
Svetlan'	0	4	0	2	0	4	3	1	2	0
EWN	5	0	4	18	5	1	1	0	14	0

Tab.14 Résultats Mots_trouvés_Occurences_SVETLAN'_EWN

Des mots tels que « partisan » ou « militaire » peuvent donc être récupérés par une association d'EWN à SVETLAN', quant à « enfreindre », pourtant disponible dans les

deux fichiers, aucun lien ne semble l'atteindre. « lundi », lui n'est pas disponible dans EWN, donc irrécupérable a posteriori.

On peut donc considérer qu'il y a des chances de complémentarité entre les deux bases, d'autant plus que d'une façon générale, les liens que ce soit leur type ou le mots servant de lien, sont très différents selon SVETLAN' ou EWN.

	SVETLAN'		**EWN**	
	Lien	Lemme	Lien	Lemme
Partisan	\`a	Déclarer	synonymie	Partisan
	\`a	Proposer		
	cod	Accuser		
	cod	Inciter		
	entre	Provoquer		
	pr\`es	Éliminer		
	sujet	Affirmer		
	sujet	Commencer		
	sujet	Insister		
Lundi	\`a	Assister		
	\`a	Atteindre		
	\`a	Fonctionner		
	\`a	Repousser		
	aux	Rendre		
	avant	Dégager		
	avant	Reprendre		
	cc	Traiter		
	cod	Apprendre		
	cod	Avancer		
	cod	Destiner		
	cod	Fêter		
	cod	Obtenir		
	cod	Parvenir		
	cod	Placer		
	cod	Proposer		
	cod	Publier		
	cod	Repartir		
	cod	Tomber		
	d\`es	Opposer		
	d\`es	Prévoir		
	depuis	Détenir		
	depuis	Tuer		
	du	Apprendre		
	entre	Prévoir		
	jusqu'\`a	Donner		
	sujet	Constituer		
	sujet	Marquer		
	sujet	Prévoir		
	sujet	Rendre		
	sujet	Stabiliser		
	sujet	Stipuler		
Militaire			synonymie	Soldat

			has hyponym	Soldat
			has hyperonym	Soldat
Force	cod	Devenir	synonymie	Force
	cod	Employer	has hyponym	Force
	cod	Employer	has hyperonym	Force
	cod	Employer	has hyperonym	Autorité
	cod	Mettre	synonymie	Pouvoir
	cod	Redouter	has hyperonym	Intérêt
	par	Disperser	has hyperonym	Pouvoir
	par	Imposer		
	par	Intervenir		
	sujet	Redouter		
	sujet	Réprimer		
	sujet	Résider		
Amende	cod	Infliger	synonyme	Amende
	cod	Payer	has hyponym	Amende
	cod	Payer	has hyperonym	Amende
	cod	Risquer		
	cod	Verser		
Lancer	119 liens		has hyperonym	Ouvrir
Confirmer	62 liens		synonyme	Affirmer
Étudier	33 liens			
Représenter	86 liens		synonyme	Représenter
			has hyponym	Représenter
			has hyponym	mal représenter
			has hyperonym	Représenter
			synonyme	Être
			has hyperonym	Être
			has hyponym	Être
Enfreindre	0 lien			

Tab.14 Résultats Liens_ Mots_trouvés_SVETLAN'_EWN

(Tous les liens pour les verbes dans SVETLAN' ne sont pas cités puisqu'en trop grand nombre).

On constate que les verbes amènent en général aux noms dans SVETLAN', alors que dans EWN les noms ont tendance à ne mener qu'aux noms et les verbes aux verbes.
Ainsi, non seulement les liens sont de type très différent, mais en plus ils ne relient pas le même type d'information. On a donc des chances de beaucoup diversifier l'accès.

Dans EWN le lien principal d'accès est synonyme, puisque dans la plupart des cas, il représente un appariement direct d'un mot du contexte avec un mot de la liste manquant. (sauf dans le cas de « militaire » repris pas « soldat »). On peut penser ce lien comme valable, d'autant plus que dans la langue française, on aime à utiliser plutôt des synonymes d'une phrase à l'autre que plusieurs occurrences du même mot. Les liens d'hyperonymie sont eux aussi très représentés. On les prendrait donc en compte même dans le cas d'une réduction d'EWN sur les liens (réduction de liens inutiles afin de gagner du temps à l'exécution).
Dans SVETLAN', les liens sont mixtes. Comme ils sont syntaxiques, on peut apparier le nom directement sur les lemmes du domaine, ou par l'intermédiaire d'un verbe. Le verbe lui ne trouve qu'un type d'accès, direct dans le domaine, puisque les liens des noms ne

sont pas étendus aux autres domaines. Les doublons contenus dans le ocntexte et dans la liste de mots manquants sont alors très utiles. Ces liens sont très nombreux, surtout dans le cas de verbes très représentés qui possèdent alors beaucoup de noms liés (119 pour le verbe lancer).

On peut conclure de cette observation qu'une association de ces différents liens peut être positive à deux niveaux : tout d'abord, réduire la quantité de réponses apportées, et deuxièmement affiner l'accès au lexique par un typage exprimé.

IV.3.2. SVETLAN' < EWN

Il s'agit ici d'apparier le contexte avec SVETLAN' puis d'ajouter à chaque mot du domaine sélectionné les liens EWN par une mise en correspondance des mots de SVETLAN' avec les variants d'EWN.
A partir de ce test, le script prendra directement en compte le choix de recherche sur un nom ou un verbe, ce qui évite d'une part de trouver « lancer » en tant que nom par exemple, alors que seul le verbe est recherché et d'autre part cela limite le temps d'exécution, la liste à analyser étant réduite de moitié.

Un premier calcul est effectué pour quantifier entre autres l'ajout de résultats.
On peut comparer les résultats précédents avec une analyse avec SVETLAN' seul ou EWN seul, avec les résultats d'un double appariement, du contexte dans SVETLAN' tout d'abord puis des vocables du domaine avec EWN.

	IRAK	**LIBERIA**	**LOCKERBIE**	**NICOLAS**
	Trouvés	Trouvés	Trouvés	Trouvés
EWN	7	7	8	6
SVETLAN'	6	8	8	7
SVETLAN'<EWN	10 / 8.5[1]	9 / 8.25	10 / 7.5	8 / 7.6667

Tab.15 Résultats Occurrences_Mots_trouvés__Poids_Fichier_SVETLAN'_EWN_SVETLAN'<EWN

(attention le calcul du poids n'est pas effectué de la même manière entre EWN et SVETLAN')

A priori, on retrouve au moins autant voir plus de mots manquants grâce à ce type d'analyse.
Un problème subsiste, le nombre de mots à passer au test est encore plus important. Pour le texte Irak, l'ensemble de mots à analyser à la base est de : 3881 pour SVETLAN' et 32809 pour EWN. Il faut prendre en compte que de nombreux doublons sont présents et que tous passent dans EWN un par un et sont analysés pour chaque variant identique parfois dans de multiples synsets.
Ces résultats ne sont donc satisfaisants que dans la mesure où sans perdre trop de nos liens utiles on réussit à réduire le nombre de mots en input dans SVETLAN' et/ou EWN.

[1] premier chiffre = Nombre de mots trouvés ;
deuxième chiffre = Nombre de mots trouvés en moyenne.

IV.4. Réduction des informations non pertinentes

Plusieurs facteurs sont à l'origine du nombre important de réponses non pertinentes, c'est-à-dire du nombre de mots non pertinents proposés à l'analyse.
Il est possible de réduire ce nombre de mots à différents niveaux, principalement au niveau du contexte lui-même, ou au niveau de la liste de résultats à la sortie de SVETLAN'.

IV.4.1. Réduction contextuelle

IV.4.1.1. Appariement du contexte seul

C'est un cas où seul les mots du contexte de départ ne seront appariés, que ce soit dans SVETLAN' ou dans EWN.

Environ 200 mots appartenant au contexte du texte initial sont tout d'abord passés à SVETLAN' qui sélectionne les domaines contenant 75% de ce contexte. Ensuite ne seront passés à EWN pour extension des liens que les mots de ce contexte initial qui ont été retrouvés dans SVETLAN'. Peu de mots sont donc conservés et l'exécution est beaucoup plus rapide. Les résultats obtenus sont probants, mais on peut voir une perte des mots récupérés précédemment.

	IRAK		LIBERIA		LOCKERBIE		NICOLAS	
	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés
SVETLAN'<EWN	10	8.5	9	8.25	10	7.5	8	7.6667
SVETLAN'<EWN Restriction match contexte	7	6.25	8	8.25	9	9.5	8	5.6667

Tab.16 Résultats Occurrences_Mots_trouvés_Poids_Fichier_SVETLAN'<EWN_SVETLAN'<EWNRestricContexte

Il faut bien prendre en compte que l'on fait passer le nombre de mots analysés d'environ 4000 mots (nombre de mots dans les domaines sélectionnés pour un fichier) à environ 900. Avec un passage dans EWN, le nombre de mots du contexte augmente de 5 fois en moyenne (doublons).
Le résultat avec une restriction sur le contexte est donc très intéressant puisque l'accès aux mots manquants se fait au milieu de beaucoup moins de mots et que le temps d'exécution est largement amélioré.

Si la restriction au contexte était trop forte, et que trop peu de mots restent, il y a possibilité d'étendre tout d'abord les liens des mots du contexte avec EWN puis de réeffectuer la réduction avec le passage normal dans SVETLAN' puis EWN.
On peut aussi relâcher ce contexte en effectuant un autre type de restriction qui tranche moins vivement dans le contexte et dans la quantité de mots disponible pour les mises en correspondance suivantes.

De plus, cette restriction agissait en fait sur deux niveaux d'analyse en même temps, le passage dans SVETLAN' et celui dans EWN ; on peut tenter de restreindre plus proprement le contexte avant analyse.

IV.4.1.2. Division en Segments Thématiques (ST)

Cette restriction contextuelle consiste en la division du texte d'entrée (le contexte même) en segments thématiques. Un segment thématique est une partie du texte qui possède une unité thématique. Les précédents segments thématiques utilisés lors de la constitution de ROSA ou de SVETLAN' étaient faits automatiquement ; ce n'est pas le cas ici.

Les textes sont divisés en 2 à 5 segments thématiques (ST Irak/Liberia/Lockerbie/Nicolas, voir Annexes 4). Le texte Nicolas étant plus court, il ne possède que 2 segments et la division n'est donc pas sur le même modèle.
En effet, le ST1 représente une phrase introductive présente dans tous les articles AFP, tandis que le ST5 relate souvent un événement un peu extérieur au sujet même, qui image le reportage.

En même temps que cette restriction est mise en place une restriction par anti-dictionnaire. Elle permet d'épurer le texte des mots trop courants qui perdent alors tout sens (auxiliaires, modaux principalement). Cette « stop-list » est minimale vu la taille de notre corpus.

Voici les résultats d'une analyse avec SVETLAN' et EWN combinés et une combinaison de la restriction sur les mots du contexte appariés entre SVETLAN et EWN et une restriction sur le contexte input (restriction maximum pour le moment).

		ST1	ST2	ST3	ST4	ST5
Moyenne trouvés		3.6667	5.1667	5.6	5.25	4.4286
Trouvés	**1**	Force	Force	Force	Force	Force
	2	Lundi	Lundi	Lundi	Lundi	Lundi
	3	Étudier	Amende	Partisan	Amende	Amende
	4	Représenter	Partisan	Militaire	Étudier	Partisan
	5	Lancer	Militaire	Étudier	Représenter	Étudier
	6	Confirmer	Étudier	Représenter	Lancer	Représenter
	7		Représenter	Lancer	Confirmer	Lancer
	8		Lancer	Confirmer		Confirmer
	9		Confirmer			
	10					
Domaines sélectionnés		fuirContrôler commanderPoursuivre tuerTrouver	prévoirReprendre tuerPrendre commanderPoursuivre élireEffectuer apprendreDemander tuerTrouver	prévoirReprendre tuerPrendre commanderPoursuivre tuerTrouver donnerObtenir	prévoirReprendre commanderPoursuivre apprendreDemander tuerTrouver	tuerPrendre prévoirAtteindre commanderPoursuivre maintenirDemander apprendreDemander tuerTrouver donnerObtenir

Tab.17 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestricContexte_ST_Irak

Dans cet exemple on voit que le segment thématique 3 donne les meilleurs résultats au sein d'un fichier, cependant, c'est le ST2 qui procure le plus des mots recherchés.

Il est à noter que non seulement cette technique donne en résultats moins de domaines, mais en plus il ne donne pas les plus importants en termes d'agrégation qui représentent en général des domaines avec moins de mots.
Malgré les multiples restrictions, on trouve sur un fichier en moyenne la moitié des mots. Il semblait pourtant prévisible que ces multiples restrictions allaient jouer négativement sur les résultats.

Bien que ces résultats soient intéressants, on peut tester les effets d'une réduction n'agissant pas sur le contexte de près ou de loin. En effet, il est possible d'opérer une réduction intermédiaire entre le domaine et le mot dans SVETLAN' (qui ne serait pas possible dans ROSA ou EWN par exemple), la réduction à la structure.

IV.4.2. Réduction dans SVETLAN'

Une réduction des domaines est possible avec simplement la structure dans laquelle il y a un appariement avec le contexte. On ne récupère donc pas seulement un mot, mais un ensemble de mots possédant des liens syntaxiques entre eux. On peut ainsi prendre en compte des mots ayant en plus d'un simple lien syntaxique un véritable lien sémantique, voire de cooccurrence.
Cette réduction se fait grâce à l'architecture des fichiers d'UTSAs mettant en valeur la structure, c'est-à-dire un verbe et ses liens vers des noms.

Cela permet de ne pas conserver un lemme unique apparié du contexte et limiter l'accès à EWN à un trop petit nombre de mots, ni l'inverse en envoyant les 1000 ou 1500 mots d'un domaine.
Le résultat est très bon, bien qu'avec en moyenne environ 5 mots par structure, on ait en résultat un plus grand nombre de mots qu'avec les restrictions précédentes sur le contexte.

	IRAK		**LIBERIA**		**LOCKERBIE**		**NICOLAS**	
	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés	Trouvés	Moyenne Trouvés
SVETLAN'<EWN Restriction Structure	9	7.4	10	7.5	9	7	8	6.5

Tab.18 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestric_Structure

L'augmentation des résultats est importante par rapport à Tab.16, où il n'y a qu'une seule restriction, d'autant plus qu'il y a ici le filtre de l'anti-dictionnaire.

A l'intérieur d'un domaine apparié le nombre de mots est nettement moindre (exemple Irak), ce qui allège considérablement le fichier et ce malgré un passage sur les données EWN :

	Nom Domaine	Trouvés	Nbre mots du domaine
SVETLAN'	prévoirReprendre	4	1247
	tuerPrendre	3	925
	commanderPoursuivre	4	1709
	tuerTrouver	5	1309
SVETLAN'<EWN **Restriction match contexte + Structure**	prévoirReprendre	7	532
	tuerPrendre	7	372
	commanderPoursuivre	6	922
	tuerTrouver	8	563

Tab.19 Résultats Irak_SVETLAN_SVETLAN_restriction

Vu les résultats de ces différentes restrictions, il semble naturel de tester plus avant les combinaisons de restrictions de différents types.

IV.4.3. Combinaison de restriction contextuelle et d'application

Il est possible de combiner une restriction sur le contexte et une restriction sur SVETLAN'. Nous prendrons celles qui semblent les plus mesurées sur l'un et l'autre, c'est-à-dire les ST pour le contexte et la conservation des structures dans un domaine pour SVETLAN'.
Cela donne les résultats suivants :

		ST1	ST2	ST3	ST4	ST5
Moyenne trouvés		4.3333	6.3333	6.8	7	5
Trouvés	**1**	Force	Force	Force	Force	Force
	2	Lundi	Lundi	Lundi	Lundi	Lundi
	3	Militaire	Amende	Partisan	Amende	Amende
	4	Étudier	Partisan	Militaire	Militaire	Partisan
	5	Représenter	Militaire	Amende	Étudier	Militaire
	6	Lancer	Étudier	Étudier	Représenter	Étudier
	7	Confirmer	Représenter	Représenter	Lancer	Représenter
	8		Lancer	Lancer	Confirmer	Lancer
	9		Confirmer	Confirmer		Confirmer
	10					
Domaines sélectionnés		fuirContrôler commanderPoursuivre TuerTrouver	prévoirReprendre tuerPrendre commanderPoursuivre élireEffectuer apprendreDemander tuerTrouver	prévoirReprendre tuerPrendre commanderPoursuivre tuerTrouver donnerObtenir	prévoirReprendre commanderPoursuivre apprendreDemander tuerTrouver	tuerPrendre prévoirAtteindre commanderPoursuivre maintenirDemander apprendreDemander tuerTrouver donnerObtenir

Tab.20 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestricContexte_Structure_ST_Irak

		ST1	ST2	ST3	ST4	ST5
Moyenne trouvés		5	5.2	7.5	7.333	7
Trouvés	**1**	Espoir	Espoir	Mois	Mois	Mois
	2	Mois	Mois	Appareil	Appareil	Appareil
	3	Appareil	Appareil	Pouvoir	Pouvoir	Pouvoir
	4	Pouvoir	Pouvoir	Diriger	Diriger	Diriger
	5	Demeure	Demeure	Marquer	Marquer	Marquer
	6	Diriger	Diriger	Fournir	Fournir	Fournir
	7	Marquer	Marquer	Déployer	Déployer	Déployer
	8	Fournir	Fournir	Déclencher	Déclencher	Déclencher
	9	Déployer	Déployer			
	10	Déclencher	Déclencher			
Domaines sélectionnés		prévoirReprendre battreDisputer prévoirAtteindre commanderPoursuivre élireEffectuer poursuivreDemander apprendreDemander demanderPrévoir tuerTrouver donnerObtenir OuvrirApporter	prévoirReprendre tuerPrendre battreDisputer présenterJouer prévoirAtteindre commanderPoursuivre élireEffectuer poursuivreDemander maintenirDemander apprendreDemander prendreRemporter demanderPrévoir tuerTrouver donnerObtenir	commanderPoursuivre tuerTrouver	prévoirReprendre commanderPoursuivre tuerTrouver	commanderPoursuivre

		ouvrirApporter			

Tab.21 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestricContexte_Structure_ST_Liberia

		ST1	ST2	ST3	ST4	ST5
Moyenne trouvés		5.25	6.25	6.5	7.5	6.25
Trouvés	**1**	Lettre	Lettre	Lettre	Lettre	Lettre
	2	Levée	Levée	Levée	Levée	Levée
	3	Pays	Pays	Pays	Pays	Pays
	4	Envoyer	Envoyer	Courrier	Courrier	Courrier
	5	Rencontrer	Rencontrer	Envoyer	Envoyer	Envoyer
	6	Payer	Payer	Rencontrer	Rencontrer	Rencontrer
	7	Conclure	Conclure	Payer	Payer	Payer
	8	Confirmer	Confirmer	Conclure	Conclure	Conclure
	9			Confirmer	Confirmer	Confirmer
	10					
Domaines sélectionnés		tuerQuitter prévoirReprendre tuerPrendre présenterJoue prévoirAtteindre commanderPoursuivre poursuivreDemander tuerTrouver	prévoirReprendre prévoirAtteindre commanderPoursuivre apprendreDemander	prévoirReprendre prévoirAtteindre commanderPoursuivre tuerTrouver	prévoirReprendre tuerPrendre commanderPoursuivre poursuivreDemander apprendreDemander tuerTrouver	prévoirReprendre commanderPoursuivre tuerTrouver ouvrirApporter

Tab.22 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestricContexte_Structure_ST_Lockerbie

		ST1	ST2
Moyenne trouvés		4.6667	7
Trouvés	**1**	Enfant	Enfant
	2	Famille	Famille
	3	Examen	Parent
	4	Parent	Recueillir
	5	Recueillir	Placer
	6	Placer	Décéder
	7	Décéder	Maintenir
	8	Maintenir	
	9		
	10		
Domaines sélectionnés		prévoirReprendre tuerPrendre prévoirAtteindre commanderPoursuivre élireEffectuer poursuivreDemander apprendreDemander tuerTrouver donnerObtenir	commanderPoursuivre

Tab.23 Résultats Occurrences_Mots_trouvés_SVETLAN'<EWNRestricContexte_Structure_ST_Nicolas

En général, le découpage en segments thématiques permet une bonne économie, avec des pertes de trois mots au maximum selon le segment. Les ST3 semblent contenir le plus d'informations, en offrant un minimum de domaines. Ils ne contiennent pas forcément le nombre de mots le plus important :

ST1	25,67
ST2	52,67
ST3	55,00
ST4	55,67
ST5	21,67

Tab.24 Nombre_Mots_ST

Ce ne sont pas non plus forcément les segments qui comportent le plus de mots manquants. Mais c'est grâce au contexte qu'ils apportent que l'on accède à ces mots.

D'autres tests seraient à effectuer, comme un test sur l'intérêt des structures seules avec leurs éléments correspondant à ceux du contexte, indépendamment des domaines. Le temps nous arrête là.

IV.5. La vérification ou l'accès au contexte par les mots manquants

Afin de vérifier ou découvrir quels mots du contexte mènent aux mots manquants, on peut remonter le fil d'Ariane. Un script a donc fait le chemin de mise en correspondance inverse.
Tout d'abord, pour avoir les meilleurs résultats, il aurait été intéressant d'apparier les 10 mots manquants dans le contexte, or ce n'est pas possible pour tous les textes et encore moins avec les contraintes mises sur notre liste d'input (et que nous avons conservées pour garder les mêmes contraintes dans l'accès cad stop-liste, structure, essai avec ST).

On retrouve par contre de nombreuses fois ces domaines comprenant les 10 mots manquants pour Liberia et Lockerbie ce qui est concluant ; ce sont les suivants :

	Liberia	Lockerbie	
Noms de domaines	prévoirReprendre	prévoirReprendre	maintenirDemander
	prévoirAtteindre	battreDisputer	apprendreDemander
	commanderPoursuivre	commanderPoursuivre	tuerTrouver
	tuerTrouver	poursuivreDemander	donnerObtenir

Tab.25 Domaines_trouvés_liste_manquants

Les résultats ci-dessous sont ceux de plusieurs domaines pour Irak et sont représentatifs des résultats avec une restriction sur les STs des différents textes.

	Liste matchés manquants		Domaines trouvés	Liens + Trouvés contexte	Nbre trouvés	Nbre contexte
Fichier complet	1	amende	battreDisputer apprendreDemander donnerObtenir	SVETLAN' : amende commencer demande force intérêt lundi mardi nuit opération partisan payer plan président proposer représenter source souveraineté verser EWN : amende->has_hyperonym->amende amende->has_hyponym->amende confirmer -> causes -> affirmer confirmer -> causes -> appuyer force -> has_hyperonym -> force force -> has_hyponym -> force partisan -> antonym -> partisan partisan -> has_holo_member -> partisan partisan -> has_hyperonym -> partisan partisan -> has_hyponym -> partisan représenter -> has_hyperonym -> représenter représenter -> has_hyponym -> représenter	**20**	**214**
	2	confirmer				
	3	Étudier				
	4	Force				
	5	Lancer				
	6	Lundi				
	7	partisan				
	8	représenter				

Tab.26 Domaines_liens_trouvés_liste_manquants

En fait, cela montre que principalement les doublons nous mènent aux mots manquants. On peut y ajouter les synonymes d'EWN, c'est-à-dire les variants d'un même synset qui jouent ce rôle lorsque l'on part de la liste_contexte, ainsi que donc tous ces hyponymes/hyperonymes qui se répondent la même vedette.

Les résultats les plus intéressants sont ceux apportés dans EWN par le lien « causes » qui permet de diversifier les lemmes qui permettent l'accès. Ce ne sont malheureusement pas des liens très courants.
On peut aussi considérer que seuls une vingtaine des mots de notre contexte nous permettent un accès aux mots recherchés. Ces mots n'ont pas de liens spécifiques, et seules les collocations, qui se trouvent dans SVETLAN grâce à des étiquetages syntaxiques redondants sur un vaste corpus d'informations textuelles, mènent en fait aux mots manquants.

IV.6. Découvertes originales

Les différentes listes de mots manquants pouvaient chacune souligner des particularités.

Deux mots « athématiques » se sont glissés par exemple dans ces listes : « lundi et mois ». Ces deux mots qui ne devraient appartenir à aucun domaine précis appartiennent en fait à un maximum de domaines et se trouvent alors dans les plus retrouvés.
Contrairement à cela des mots qui pour presque tous les locuteurs du français (le test ne sera pas validé faute de temps) seraient retrouvés grâce à des associations, sont perdus dans la masse d'information de SVETLAN' ou d'EWN et aucun accès n'y est trouvé.

Par exemple pour « enfreindre » alors que le contexte contient « loi » ou « dédommagement » alors que le contexte contient « victime » et « payer ». D'une façon générale on retrouve pourtant des liens de cooccurrence entre verbe et nom.
De même alors que (en jouant sur les sens) la liste des mots manquants contient « courrier, levée, lettre et envoyer », seul « courrier et lettre » sont reliés dans EWN, alors que tous sont associés dans une relation de type script. On n'« envoie » pas de « lettre » ou de « courrier » au mieux des « messages » et en général des « troupes ». Tous ces mots sont retrouvés par des liens relativement éloignés qui ne se croisent pas. Dans l'esprit d'un locuteur, il est pourtant possible qu'ils ne soient pas très éloignés ; mais dans l'esprit d'un locuteur, la connaissance du monde ne s'arrête pas aux dépêches belligérantes d'AFP.
Il pourrait donc être intéressant de continuer à augmenter des bases de connaissances telles que SVETLAN' afin de couvrir d'autres types de textes (Gala ?), qui amèneraient non seulement un autre type de lexique mais aussi de nouveaux domaines en général. Le choix du domaine serait alors plus déterminant encore.

Une autre prise de position est possible ; elle est de limiter cette importance du domaine en soi (un domaine AFP et un domaine Gala), et de porter l'intérêt sur les structures comportant des mots appariés avec le contexte. Cela permettrait par exemple de récupérer « dédommagement », qui se trouve dans les UTSAs de SVETLAN' dans la même structure que « victimes « (appartenant au contexte) et que « verser » (appartenant au synset de « payer ») par exemple, mais jamais dans un domaine qui soit conservé pour la suite de l'analyse.

Finalement, toutes les relations qui mettent en jeu deux mots au même moment (collocations par exemple) sont récupérables dans un système construit par SVETLAN' bien qu'elles ne soient pas typées. Les relations analogiques le sont aussi puisque la relation peut tout à fait apparaître d'une façon « statistique ». Une association de ressemblance peut tout à fait être obtenue sans typage du moment que les deux mots apparaissent en même temps ou dans le même environnement.
Les relations fonctionnelles nominales ou thématiques verbales sont typiquement des relations récupérables par leurs structures dans SVETLAN', sans être expressément typées.

Des relations converses sont elles aussi récupérables dans la mesure ou SVETLAN' et EWN sont reliés. En effet, elles possèdent des structures très proches et une relation d'antonymie dans EWN directe ou sur certains mots de leur structure (il y a tout de même besoin d'une certaine quantification ; à combien de liens antonymes on peut considérer que l'association est converse ?) pourrait permettre de conclure à une relation d'antonymie.

Par contre les relations mettant en cause plus directement une définition (les relations structurante par exemple), ou une qualité grammaticale (les dérivés syntaxiques) sont difficilement récupérables par SVETLAN'. EWN peut alors jouer un rôle avec ses définitions en synsets. On peut aussi ajouter un type de lien à EWN reliant les mots d'une même « famille » afin de retrouver au moins les dérivés syntaxiques, EWN possédant déjà la nature du lemme.

De même, certaines associations sont plus ou moins présentes dans EWN, tels que les relations de type lexical « mots composés » ou « formules », mais la forme des expressions régulières dans mon script ne sont pas efficaces pour leur récupération.
D'autres relations lexicales ne sont pas disponibles avec la forme des données de SVETLAN', puisque aucune relation entre le nom et son expression de départ n'est conservée ; il n'y a pas de mémoire de l'expression par rapport au corpus AFP de départ.

Toutes ces associations reposent la question de l'ajout de certains liens dans EWN.
Des liens « grammaticaux » (dérivés syntaxiques par exemple) doivent pouvoir être ajoutés automatiquement vu le grand nombre de ressources électroniques disponibles sur ce thème.
On peut aussi ajouter au système un dictionnaire « classique » qui permettrait de faire le lien avec l'un des types de recherche (recherche par superposition du graphe-message et du graphe lexical) de l'application d'aide à l'accès au lexique de MM. Zock et Fournier.

Il est aussi intéressant de souligner certaines limites d'EWN. En effet, les labels présents dans EWN ne sont pas directement liés aux mots, mais dans un fichier joint ; du fait, dans la traduction d'EWN en XML on perd tout à fait ces informations.
De même les liens entre noms et verbes sont très limités dans EWN et finalement c'est SVETLAN', grâce à ses liens syntaxiques qui en relient beaucoup plus (certes ses liens sont très différents), qui permet de faire un pont entre les différentes catégories.
Une optique intéressante serait peut-être d'ajouter à ce fichier itinérant de labels d'EWN, les liens de SVETLAN mêmes afin d'enrichir le thésaurus.
Le vocabulaire d'EWN est très large et il est possible qu'une fusion automatisable soit possible.
Une extension où tous les liens seraient typés pourrait permettre de garder un type « dictionnaire » humain à EWN, mais ne semble pas forcément utile pour une application d'accès lexical.

Pour permettre un accès lexical de n'importe quel type, SVETLAN' a besoin d'être étendu, mais pas seulement par des liens typés comme ceux disponibles dans EWN. Il faut aussi une extension au niveau des adjectifs par exemple (présents dans ROSA), des noms dans des structures avec adjectifs (collocations) ou avec prépositions (noms composés par exemple) ou plus largement des domaines afin d'être représentatifs de différents types de textes.
Pour le moment, SVETLAN' aiderait principalement les rédacteurs d'articles de type AFP.

Conclusion

Réponses aux hypothèses

Les liens non typés de SVETLAN' (liens thématiques) et ses liens syntaxiques sont suffisants pour un accès au lexique massif. On a cependant avant cela pondéré par des contraintes le choix du domaine et pour plus de précision on a besoin de restrictions sur le contexte et d'extension de ces liens.
L'intégration des informations d'EWN est donc très utile pour permettre d'étendre les données de SVETLAN'.
Cependant, cette extension peut-être nécessaire à divers moments du processus d'analyse, c'est pourquoi il est peut-être mieux de laisser ces deux bases de données séparées si on opère sans faire de choix sur les liens eux-mêmes.
Si les liens sont directement pris en compte, il est alors possible de fusionner les deux types d'information en faisant appel selon le traitement voulu aux relations d'EWN ou aux liens syntaxico-thématiques de SVETLAN'.

Les liens offerts par les FLs n'ont pas pu être étudiés. Cependant, il semble possible que tout comme une fusion entre EWN et SVETLAN' soit utile si il y a appel à des liens typés dans l'analyse, l'ajout de liens typés des FLs dans SVETLAN' permette de mener plus précisément à un terme, malgré des restrictions plus fortes (moins de mots du contexte en input par exemple).

Avenir du travail

L'intérêt de cette étude était d'observer les relations mises en jeu dans la recherche lexicale au travers d'applications de création de domaines automatique, en vue d'intégrer un module sémantique à une application d'aide à l'accès lexical.
Il semble que l'application SVETLAN', qui structure en domaines (liens syntagmatiques implicites) des informations textuelles typées syntaxiquement, soit valable dans la recherche lexicale.
Cependant, une application telle qu'EWN (et DiCo peut-être) permet d'étendre les liens et donne lieu à un accès lexical plus important.
Des combinaisons de recherche lexicale grâce à ces deux applications restent à tester, principalement en s'appuyant plus fortement sur les structures constituant les domaines.

Cette étude est une observation des possibilités de mise en place d'un module sémantique dans un système d'aide à l'accès lexical, en combinaison principalement avec les modules graphique et phonologique de MM. Zock et Fournier.

Il prendrait place après une analyse graphique et phonologique de l'information entrée, et permettrait de valider l'homogénéité des informations au sein d'un domaine et de mesurer la validité des réponses (mots) proposées.
Il devrait être possible aujourd'hui de tenter un prototype de ce système à trois modules où SVETLAN' (et ses liens dans EWN) servirait à fournir un contexte à une amorce donnée par un locuteur qui recherche un mot cible.
Dans le cas d'une recherche d'un mot seul vers un autre, il peut être possible de suivre les liens d'EWN.

Ce type de recherche lexicale, développé et validé pourrait aussi être introduit sous forme de module de désambiguïsation sémantique (thématique) dans un système de question-réponse par exemple.