Construire et Accéder à une Base de Données d'Expressions Figées à partir des Ressources de la Toile

Gaël Dias, Ludovina Carapinha, Rosa Trindade, Susana Mota, Marco Ribeiro et Jorge Dias

Université de la Beira Interior, Centre de Mathématique
Rue Marquês d'Ávila e Bolama, 6200-053, Covilhã, Portugal
ddg@di.ubi.pt
http://www.di.ubi.pt/~ddg

Mots clés: Constitution semi-automatique de terminologies, Toile, Accès libéralisé et standardisé, Analyse Probabiliste, *DicAssist*.

Résumé

La constitution de bases de données terminologiques s'adaptant dynamiquement aux constants changements de la langue est un axe de recherche prioritaire pour un nombre important d'applications du traitement automatique du langage naturel (TALN). Dans ce cadre, de nombreuses applications ont été proposées dont le but est d'extraire automatiquement les termes contenus dans les *corpora*. Cependant, la plupart de ces systèmes démontrent un statisme flagrant face au dynamisme toujours croissant du langage si bien illustré par le contenu de la toile. En effet, si l'on prétend attaquer le problème de la constitution de terminologies véritablement exploitables, la récolte des textes à traiter doit être systématique et réalisée à partir de sources réelles du langage et non pas à partir de *corpora* compilés dont le comportement n'évolue pas au fil du temps. Afin de répondre à ce problème, nous présentons une architecture qui s'appuie sur les ressources de la toile pour intégrer sur une seule chaîne de traitements toutes les étapes de la constitution d'une base de données terminologique, ceci de la récolte des textes à l'accès aux termes validés et enrichis linguistiquement. Ce système s'appelle *DicAssist*.

1. Introduction

La constitution de bases de données terminologiques s'adaptant dynamiquement aux constants changements de la langue est un axe de recherche prioritaire pour un nombre important d'applications du traitement automatique du langage naturel (TALN). En particulier, à travers les expressions figées qui représentent la plupart des réalisations terminologiques, il est possible de mieux appréhender le contenu des textes et ainsi de les indexer avec une plus grande précision afin de garantir le succès des moteurs de recherche (Evans, 1993).

Cependant, l'accroissement incessant des masses de textes ne permet pas leur traitement manuel. De plus, l'ensemble des termes est ouvert et à compléter (Habert, 1993). En effet, une partie essentielle de la néologie lexicale, en particulier dans les domaines techniques et

scientifiques, s'opère par le biais de séquences complexes. Afin de faire face à ces quantités gigantesques de données, les professionnels de l'information se sont dotés d'outils permettant d'extraire automatiquement les termes contenus dans les *corpora*. À titre d'exemple, nous retiendrons les systèmes XTRACT (Smadja, 1993), ACABIT (Daille, 1995), LEXTER (Bourigault, 1996) et SENTA (Dias, 1999).

Cependant, la plupart de ces systèmes démontrent un statisme flagrant face au dynamisme toujours croissant du langage si bien illustré par le contenu de la toile. En particulier, ils ne proposent pas une solution globale au problème de la constitution de bases de données terminologiques. D'une part, si l'on prétend attaquer le problème de la constitution de terminologies véritablement exploitables, la récolte des textes à traiter doit être systématique et réalisée à partir de sources réelles du langage et non pas à partir de *corpora* compilés dont le comportement n'évolue pas au fil du temps. D'autre part, la validation et l'enrichissement linguistique des termes extraits ne sont pas supportés par les systèmes qui se limitent à débiter des listes de termes candidats. Finalement, seule une utilisation intensive de ces ressources peut permettre de vérifier leur véritable utilité. Dans ce cadre, l'accès aux bases de données terminologiques doit être normalisé et libéralisé.

Afin de répondre à ces problèmes, nous présentons une architecture qui s'appuie sur les ressources de la toile pour intégrer sur une seule chaîne de traitements toutes les étapes de la constitution d'une base de données terminologique, ceci de la récolte des textes à l'accès aux termes validés et enrichis linguistiquement. Ce système s'appelle *DicAssist*. En résumé, cette architecture recueille quotidiennement un ensemble de textes directement de la toile nourrissant en amont l'extracteur d'associations lexicales SENTA (Dias, 1999). En aval, les termes extraits sont enregistrés dans une base de données pour être manuellement validés et enrichis linguistiquement grâce à un ensemble d'outils intégrés dans l'application. Parmi ceux-ci, on trouve un concordancier, une plate-forme statistique et le dictionnaire électronique POLLUX (Alves, 2002) de langue portugaise. Finalement, l'accès aux données terminologiques se fait par le biais d'un système de marquage qui repère et borne au moyen de balises XHTML tous les termes reconnus dans les textes soumis au *DicAssist*.

Afin de rendre cet exposé le plus clair possible, nous détaillerons dans une première étape l'architecture complète du *DicAssist*. Nous rappellerons ensuite les principes fondamentaux sous-jacents à l'extracteur d'associations lexicales SENTA afin de mieux comprendre les interfaces de validation et d'enrichissement linguistique. Finalement, nous expliquerons les règles du balisage XHTML des termes.

2. Architecture

Afin de répondre à tous ses objectifs, le *DicAssist* présente une architecture modulaire intégrant différents serveurs, bases de données et programmes temporisés dont il faut comprendre le fonctionnement. C'est ce que nous proposons d'expliquer dans cette partie.

Comme le montre la Figure 1, toute l'application est basée en entrée comme en sortie sur les ressources de la toile. Quotidiennement, le programme temporisé *WebGet* récupère tous les nouveaux documents édités par un portail donné de la toile (cf. 1). Dans notre cas, nous avons calibré le *WebGet* de manière à extraire tous les nouveaux articles du quotidien portugais *Correio da Manhã* (http://www.correiodamanha.pt). Les textes extraits sont ensuite enregistrés sur disque dur (cf. 2) et répertoriés dans une base de données (cf. 3) de manière à retrouver facilement leur localisation et leur état de traitement.

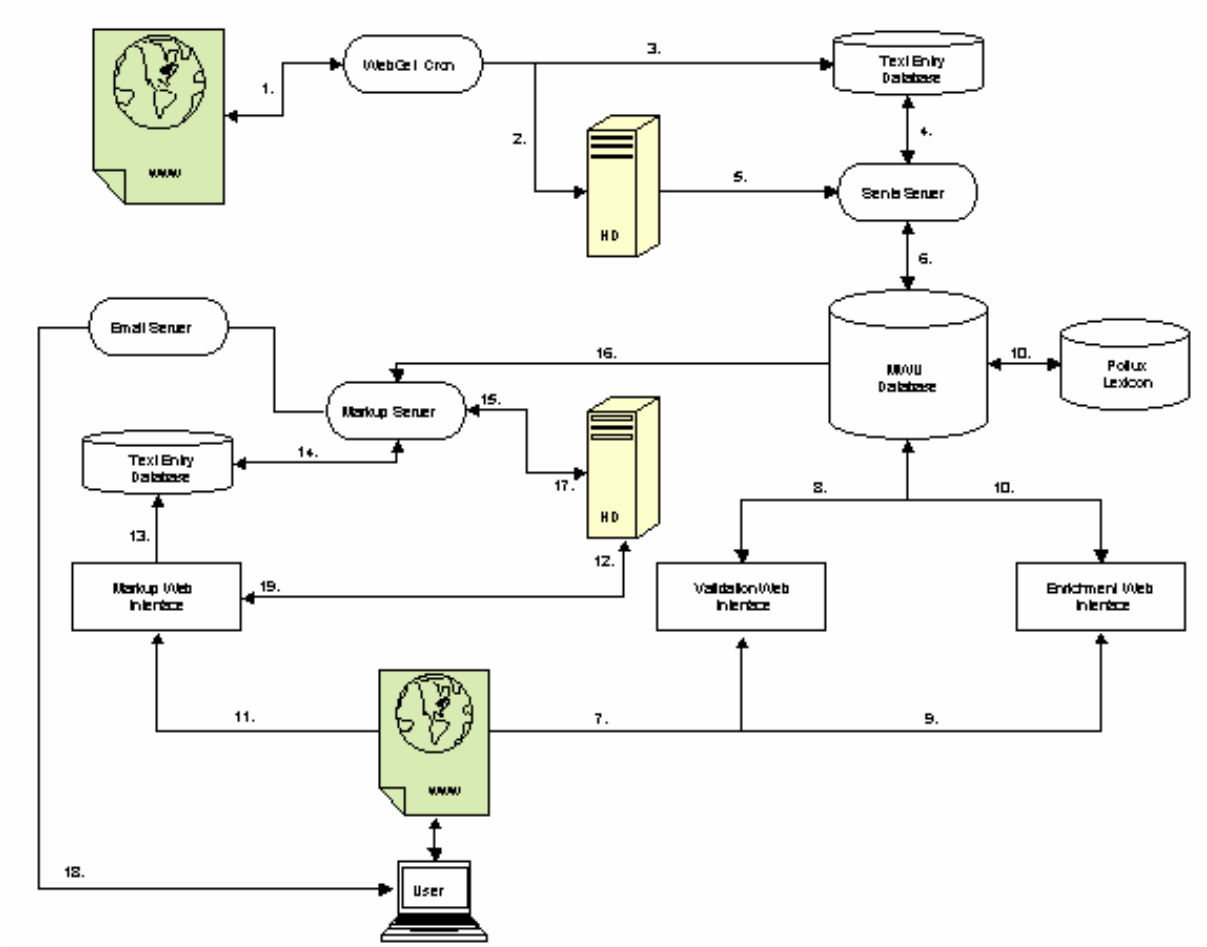

Figure 1: Architecture du *DicAssist*

Chacun de ces textes doit alors être traité automatiquement de manière à extraire un nouvel ensemble de termes candidats. C'est l'objectif du serveur SENTA. Celui-ci fonctionne en arrière plan et interroge en continu la base de données d'entrée de textes afin de vérifier l'existence d'énoncés en attente de traitement (cf. 4). Tant qu'il existe de nouveaux textes à traiter, SENTA extrait les nouvelles expressions figées (cf. 5) et les enregistre dans la base de données d'expressions en attente d'être validées et enrichies linguistiquement (cf. 6).

À partir d'un simple navigateur, un utilisateur autorisé du système peut à tout moment accéder aux interfaces de validation (cf. 7 et 8) et d'enrichissement (cf. 9 et 10). Comme son nom l'indique, la première option sert à valider les expressions de la base de données. Pour se faire, le *DicAssist* tient à la disposition de l'utilisateur trois sources d'information: une plate-forme statistique, un concordancier et le texte intégral dans lequel l'expression à valider est coloriée i.e. marquée grâce à des étiquettes HTML. Chaque expression validée doit ensuite être enrichie linguistiquement. L'objectif de cette phase est de définir la catégorie de chaque expression selon la nomenclature de Gaston Gross (1996) et de lui associer toutes ses informations morpho-syntaxiques grâce au dictionnaire POLLUX (cf. 10) et aux interfaces réalisées pour cet effet. Le travail des linguistes permet ainsi d'enrichir la base de données des expressions figées au fur et à mesure que les expressions apparaissent dans le système.

Finalement, l'accès à la base de données terminologique est rendu libre grâce à l'interface de marquage de textes (cf. 11). Cette option permet à toute personne reliée à la toile de soumettre un texte dans le but de repérer les termes qu'il contient. Chaque fois qu'un texte est

introduit dans le système, celui-ci est sauvegardé sur disque dur (cf. 12) et toutes ses caractéristiques sont enregistrées dans une base de données d'entrée de textes (cf. 13). Chacun de ces textes est alors localisé sur disque dur (cf. 15) grâce à l'information contenue dans la base de données d'entrée de textes (cf. 14) et marqué par le serveur *Markup*. Le résultat du balisage XHTML est alors gardé sur disque dur (cf. 17) de manière à permettre leur accès via navigateur (cf. 19). Pour les utilisateurs détenteurs de courrier électronique, les résultats sont automatiquement envoyés à leur adresse définie lors de l'enregistrement (cf. 18).

Comme nous venons de le montrer, en s'appuyant sur les ressources de la toile, l'architecture du *DicAssist* intègre sur une seule chaîne de traitements toutes les étapes de la constitution d'une base de données terminologique. Dans la prochaine partie, nous détaillons le cœur du système *DicAssist*, l'extracteur probabiliste SENTA.

3. L'Extracteur SENTA

Le logiciel SENTA (Dias, 1999) est un extracteur probabiliste qui se base sur trois concepts principaux: les modèles *N*-gram positionnels, la mesure d'association Expectative Mutuelle et l'algorithme d'extraction GenLocalMaxs. Nous rappelons dans cette partie les fondements qui accompagnent chacun de ces trois points.

3.1. Modèles N-gram Positionnels

L'objectif principal du logiciel SENTA (*Software for the Extraction of N-ary Textual Associations*) est d'identifier et d'extraire un ensemble, le plus vaste possible, d'associations lexicales à partir des seules contraintes présentes dans les énoncés. Ainsi, les textes ne sont ni lemmatisés, ni épurés à partir de listes de mots vides simplifiant ainsi l'ensemble de l'architecture du *DicAssist* et élargissant par la même occasion son champ d'application.

La première étape du processus d'extraction commence donc par la construction des modèles *N*-gram positionnels. Un *N*-gram positionnel est en fait une séquence ordonnée de *N* unités lexicales correspondant à une séquence continue ou non d'un énoncé (séquence interrompue ou ininterrompue) délimitée par la taille d'un environnement. Dans le cadre des expressions figées, nous avons limité cet environnement à sept unités lexicales. Ainsi, seuls les *N*-grams positionnels tels que *N*=1..7 sont construits. Nous présentons dans le Tableau 1 deux *N*-grams positionnels calculés à partir de l'énoncé suivant: "*Après moultes négociations, le Traité de Maastricht a été ratifié par tous les Etats Membres*".

***N*-gram = séquence ordonnée**	**Séquence associée de l'enoncé**
[0 *Traité* +1 *de* +2 *Maastricht*]	*Traité de Maastricht*
[0 *le* +1 *Traité* +2 *de* +3 *Maastricht* +5 *été*]	*le Traité de Maastricht* _____ *été*[1]

Tableau 1: Exemples de *N*-grams positionnels

Afin de généraliser la structure des *N*-grams positionnels, la notation suivante a été proposée où u_i correspond à une unité lexicale et p_{1i} à la position de l'unité lexicale u_i par rapport à l'unité u_1: $[p_{11}u_1\ p_{12}u_2\ ...\ p_{1i}u_i\ ...\ p_{1n}u_n]$. Par convention, la position p_{ii} vaut 0.

Une fois que tous les *N*-grams positionnels ont été construits à partir du texte à traiter, il faut mesurer l'aptitude de chaque séquence à représenter une expression figée. C'est le rôle

[1] L'espace (i.e. "_____") correspond au saut d'un mot graphique dans le texte. Dans ce cas, l'occurrence "*a*".

des mesures d'associations. Dans le cas du logiciel SENTA, une nouvelle mesure a été introduite: l'Expectative Mutuelle.

3.2. Mesure d'Association Expectative Mutuelle

Les modèles statistiques proposés dans la littérature (Salem, 1987; Church et Hanks, 1990; Gale, 1991; Smadja, 1993; Dunning, 1993; Smadja, 1996; Shimohata, 1997) ne sont définis que pour les *2*-grams et ne permettent ainsi que l'acquisition d'associations lexicales binaires. Pour les associations de plus de deux unités lexicales, l'acquisition requiert un travail complémentaire où les paires d'association acquises initialement jouent le rôle d'amorce. Parallèlement, la plupart des modèles mathématiques sont sensibles à l'occurrence d'unités textuelles élémentaires fréquentes[2] et leur normalisation aboutit à la définition de valeurs de cohésion incohérentes. Afin de résoudre ces problèmes, Dias *et al.* (1999) ont défini un nouveau modèle probabiliste appelé Expectative Mutuelle qui mesure le degré de cohésion qui lie entre eux tous les éléments d'un *N*-gram positionnel (i.e. $\forall N, N \geq 2$) et permet ainsi d'acquérir des associations *N*-aires sans recourir aux techniques d'amorçage. L'Expectative Mutuelle (EM) est basée sur la notion d'Expectative Normalisée (EN).

3.2.1 Expectative Normalisée

L'idée de base de l'Expectative Normalisée est d'évaluer le coût de la perte d'un mot dans un *N*-gram positionnel. Ainsi, plus une séquence de l'énoncé est figée et témoigne d'une forte cohésion, moins elle accepte la perte de l'un de ses constituants et plus la valeur de l'EN doit être élevée. Dans ce cadre, l'EN d'un *N*-gram positionnel est l'expectative moyenne de voir apparaître une unité lexicale dans une position donnée sachant que les *N*-1 autres apparaissent dans son environnement immédiat contraintes par leur position.

Le concept sous-jacent à l'EN est celui de la probabilité conditionnelle. Ainsi, l'EN mesure l'expectative moyenne incarnée par les *N* probabilités conditionnelles qui résultent de la décomposition d'un *N*-gram en *N* (*N*-1)-grams positionnels, chacun représentant la perte d'une des *N* unités lexicales. Dans le cadre de la normalisation de la probabilité conditionnelle, Dias *et al.* ont introduit la notion d'Argument Moyen Conditionnel (AMC) défini comme étant la moyenne arithmétique de toutes les probabilités conjointes des *N* (*N*-1)-grams positionnels contenus dans un *N*-gram (Equation 1)[3] i.e. la moyenne arithmétique des dénominateurs des *N* probabilités conditionnelles[4].

$$AMC([p_{11}u_1\, p_{12}u_2 \ldots p_{1i}\ u_i \ldots p_{1N}\ u_N]) = \frac{1}{N}\left(p([p_{22}u_2 \ldots p_{2i}\,u_i \ldots p_{2N}\,u_N]) + \sum_{i=2}^{N} p\left(\left[p_{11}u_1 \ldots \hat{p}_{1i}\ \hat{u}_i \ldots p_{1N}\,u_N\right]\right)\right) \quad (1)$$

Ainsi, l'EN d'un *N*-gram positionnel est introduite comme étant une probabilité conditionnelle "juste" qui utilise le concept d'AMC et est définie par l'équation (2).

---

[2] En conséquence, (Daille, 1995) et (Enguehard, 1993) ne considèrent que les occurrences des mots pleins pour l'évaluation des forces de cohésion.

[3] L'accent circonflexe "^" correspond à une convention fréquemment utilisée en Algèbre qui consiste à écrire un "^" au-dessus du terme omis d'une suite indexée de 1 à n.

[4] En effet, les numérateurs restent inchangés d'une probabilité à l'autre. Ainsi, la normalisation peut être réalisée par le calcul du centre de gravité des dénominateurs.

$$EN([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N]) = \frac{p([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N])}{AMC([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N])} \quad (2)$$

3.2.2 Expectative Mutuelle

Cependant, l'un des critères les plus importants pour l'identification d'associations lexicales est la fréquence d'occurrence. Or, l'Expectative Normalisée mesure le degré de cohésion qui lie les constituants d'un *N*-gram mais ne rend pas compte de l'hypothèse formulée précédemment. Certains de cette supposition, Dias *et al.* déduisent qu'entre deux *N*-grams positionnels ayant la même EN, il est plus probable que le plus fréquent des deux corresponde à une association lexicale pertinente. Ainsi, l'Expectative Mutuelle (EM) est définie grâce à l'équation (3).

$$EM([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N]) = p([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N]) * EN([p_{11}u_1 ... p_{1i}u_i ... p_{1N}u_N]) \quad (3)$$

L'EM permet donc de mesurer le degré de cohésion de tout *N*-gram positionnel sans être limitée aux associations binaires. Ainsi, il est possible de classer tout *N*-gram (i.e. $\forall N, N \geq 2$) suivant son degré de pertinence. Il ne reste donc plus qu'à choisir parmi cet ensemble de *N*-grams positionnels valués, les plus pertinents. C'est le rôle de l'algorithme de sélection GenLocalMaxs.

3.3. Algorithme de Sélection GenLocalMaxs

La plupart des méthodes de sélection proposent des valeurs limites globales (ou seuils) qui permettent de définir si une séquence est d'intérêt ou non. Ces seuils font l'objet d'un ajustement qui est crucial pour la réussite des expérimentations statistiques (Church et Hanks, 1990; Smadja, 1993; Dunning, 1993; Daille, 1995; Smadja, 1996; Shimohata, 1997). Il s'agit d'un compromis entre des valeurs (de fréquence ou de mesure d'association) assez permissives pour que la collecte soit importante (taux de rappel) et des valeurs pas trop généreuses pour que le résultat soit précis (taux de précision). Malheureusement, cette approche se révèle peu fiable et peu flexible. En effet, les résultats dépendent de l'expérimentation, et, suivant la longueur, le type, le domaine et la langue du *corpus*, il est nécessaire de réajuster les valeurs des seuils. Ainsi, Dias *et al.* ont introduit un nouvel algorithme, le GenLocalMaxs qui ne dépend d'aucun seuil (pré-établi ou mesuré par expérimentation) et qui élit tout *N*-gram positionnel dont le degré d'association est un maximum local.

Soient, une mesure d'association, *assoc*, un *N*-gram positionnel, *W*, l'ensemble de tous les (*N*-1)-grams contenus dans *W*, Ω_{N-1}, l'ensemble de tous les (*N*+1)-grams contenant *W*, Ω_{N+1} et une fonction *taille* qui calcule la longueur d'un *N*-gram *W* donné en argument, alors:

$\forall x \in \Omega_{N-1}$ et $\forall y \in \Omega_{N+1}$, *W* est une association pertinente si

$$(taille(W)=2 \ \wedge \ assoc(W) > assoc(y))$$

$$\vee$$

$$(taille(W)\neq 2 \ \wedge assoc(W) \geq assoc(x) \ \wedge \ assoc(W) > assoc(y))$$

Dans le cadre du logiciel SENTA, la mesure d'association utilisée est bien entendue l'Expectative Mutuelle. L'avantage principal de l'architecture proposée par Dias *et al.* dans le

cadre du *DicAssist* est de ne nécessiter d'aucun calibrage particulier pour fonctionner. SENTA s'alimente d'un texte en entrée et retourne un ensemble de termes candidats sans se préoccuper de l'ajustement de quelconque paramètre. Le travail du linguiste i.e. de l'utilisateur du *DicAssist* n'est donc pas une tâche exploratrice mais uniquement un travail de validation et d'enrichissement des termes proposés. Afin de mieux comprendre cette dernière phase, nous nous proposons d'expliquer en détail les fonctionnalités du logiciel *DicAssist* dans la prochaine partie.

4. Le Logiciel *DicAssist*

Le logiciel *DicAssist* se décompose en trois grandes phases: la validation, l'enrichissement linguistique et le marquage XHTML de textes.

4.1. Validation

Comme son nom l'indique, l'interface de validation permet au linguiste de déterminer si une expression extraite par le processus probabiliste est effectivement un terme.

Afin d'accéder aux termes à valider, trois options lui sont proposées: le mode séquentiel, le mode liste ou le mode recherche. Dans le premier cas, le système propose automatiquement le terme candidat le plus ancien de la base de données qui n'a pas encore été traité. Dans le mode liste, le valideur peut choisir n'importe quel terme à valider à partir de la liste de tous les termes en attente de traitement. Finalement, le *DicAssist* propose une option recherche qui permet à l'utilisateur de déterminer un patron de terme donné dans le genre moteur de recherche.

Figure 2: Interface de Validation

Une fois choisi le terme à traiter, le *DicAssist* met à disposition trois sources d'information pour la validation proprement dite: une plate-forme statistique, un concordancier et le texte intégral dans lequel l'expression à valider est coloriée. La figure 2 montre l'interface de validation. Les avantages du concordancier et du texte intégral sont évidents et nous ne nous attarderons pas sur leurs spécificités. En revanche, il convient d'expliquer plus en détail la plate-forme statistique. En particulier, celle-ci permet au valideur de comparer les données numériques du terme en traitement (i.e. fréquence d'occurrence et Expectative Mutuelle) aux

données moyennes de validation (i.e. la fréquence moyenne d'occurrence et l'Expectative Mutuelle moyenne des termes déjà validés).

Parallèlement à la validation, l'enrichissement linguistique a pour but d'enregistrer toutes les données morpho-syntaxiques des termes validés afin de compléter la base de données terminologique.

4.2. Enrichissement Linguistique

L'enrichissement linguistique suit trois étapes. Premièrement, chaque terme préalablement validé doit être classé suivant la nomenclature des expressions figées proposée par Gaston Gross (1996) i.e. en noms et déterminants composés, en locutions verbales, adjectivales, adverbiales, prépositives et conjonctives. Pour accéder à cette première option, l'utilisateur dispose des trois modes d'accès déjà énoncés: mode séquentiel, mode liste et mode recherche. Dans un deuxième temps, toutes les unités lexicales qui composent le terme en traitement sont associées aux entrées respectives du dictionnaire POLLUX (Alves, 2002). Finalement, l'utilisateur associe au terme toutes ses informations morpho-syntaxiques comme par exemple son genre, son nombre ou son temps. Nous ne nous attarderons pas plus sur cette phase qui ne pose aucun problème de compréhension et qui n'introduit pas de nouveaux concepts. La Figure 3 montre l'interface d'enrichissement.

Figure 3: Interface d'Enrichissement

Finalement, l'accès à la base de données terminologique est rendu libre grâce à l'interface de marquage de textes que nous développons dans la partie suivante.

4.3. Marquage XHTML de textes

Comme nous l'avons déjà mentionné, cette option du *DicAssist* permet à toute personne reliée à la toile de soumettre un texte dans le but de repérer les termes qu'il contient. Dans ce cadre, nous avons voulu que cet accès soit standardisé de manière à ce que les données recueillies puissent être facilement réutilisées.

Nous avons donc décidé d'utiliser le langage XHTML (*eXtensible HyperText Markup Language*) pour baliser les expressions figées présentes dans les textes. Dans ce cadre, nous avons dû définir un nouvel élément du DTD (*Definition Type Document*), l'élément `mwu`[5]:

```
<!ELEMENT mwu --(#PCDATA|mwu)+>
```

Ce formalisme veut simplement dire qu'une expression figée est une combinaison de chaînes de caractères et/ou d'autres expressions figées. Suivant cette définition, les deux expressions figées *Traité de Maastricht* et *Journal Officiel des Communautés Européennes*, seraient balisées comme suit:

```
<mwu>Traité de Maastricht</mwu>
<mwu><mwu>Journal Officiel</mwu> des <mwu>Communautés Européennes</mwu></mwu>
```

Cependant, les expressions figées extraites par le logiciel SENTA peuvent représenter des séquences d'unités lexicales interrompues. Pour rendre compte de cette spécificité, nous avons dû introduire certains attributs à l'élément `mwu` formalisés de la manière suivante dans le DTD:

```
<!ATTLIST mwu
        sequence (cont|non-cont) cont
        length (2|3|4|5|6)
        id ID #IMPLIED
        next ID #IMPLIED
        prev ID #IMPLIED>
```

Ainsi, toute expression ou partie d'une expression a un identifiant représenté par l'attribut `id`. Chaque expression est également cataloguée suivant son type de séquence (continue ou non). Dans le cas d'une expression non continue, les attributs `next` et `prev` identifient les liaisons entre les différentes parties de l'expression. Ainsi, l'expression « *tomber* _____ *amoureux* » serait balisée de la manière suivante:

```
<mwu sequence=non-cont length=2 id=1 prev=0 next=2>tomber</mwu> MOT-QUELCONQUE
        <mwu sequence=non-cont length=2 id=2 prev=1 next=0>amoureux</mwu>
```

Cette standardisation a pour objectif de permettre l'accès normalisé aux données repérées et ainsi de répondre à la première des deux préoccupations exprimées précédemment. L'autre objectif du *DicAssist* est de libéraliser cet accès. Dans ce cadre, n'importe quelle personne reliée à la toile par un simple navigateur peut remplir un formulaire d'accès au système. Si l'accès est autorisé, l'utilisateur peut dès lors recourir au service de marquage et recevoir les résultats du processus de balisage par courrier électronique ou bien les consulter directement à partir de l'interface de marquage. Nous finissons ainsi l'exposé sur la définition fonctionnelle de l'architecture du *DicAssist*.

5. Conclusion

L'architecture actuelle du *DicAssist* permet, sur la base d'une application informatique totalement intégrée, de construire une base de données terminologique à partir des ressources linguistiques de la toile. Dans ce cadre, nous avons voulu démontrer que seuls un traitement systématique des textes et une libéralisation de l'accès aux données terminologiques peuvent permettre de mesurer l'utilité réelle des terminologies électroniques. Cette application devra

[5] De l'anglais, *Multiword Unit*.

néanmoins évoluer dans le sens d'un plus grand automatisme et intégrer une palette plus vaste de services. En effet, il est aujourd'hui techniquement possible d'automatiser les tâches de l'enrichissement linguistique et on espère pouvoir travailler sur la validation automatique des termes. Le *DicAssit* devra aussi être capable de franchir le pas entre base de données terminologique et base de connaissance terminologique. En effet, tous les fondements sont construits pour relier conceptuellement entre eux tous les termes de la base. L'application *DicAssist* est disponible à partir de http://dicassist.di.ubi.pt.

Références

ALVES M. (2002). Engenharia do Léxico Computacional : Princípios, Tecnologia e o caso das Palavras Compostas. *Master Thesis.* New University of Lisbon. Portugal.

BOURIGAULT D. (1996). Lexter, a Natural Language Processing Tool for Terminology Extraction. In *Proceedings of 7th EURALEX International Congress*.

CHURCH K.W. & HANKS P. (1990). Word Association Norms Mutual Information and Lexicography. In *Computational Linguistics*, 16 (1), pp 23-29.

DAILLE B. (1995). Study and Implementation of Combined Techniques for Automatic Extraction of Terminology. In *The balancing act combining symbolic and statistical approaches to language*. MIT Press.

DIAS G. & GUILLORE S. & LOPES J.G.P. (1999). Language Independent Automatic Acquisition of Rigid Multiword Units from Unrestricted Text corpora. In *Proceedings of Traitement Automatique des Langues Naturelles*. Institut d'Etudes Scientifiques, Cargèse, France.

DUNNING T. (1993). Accurate Methods for the Statistics of Surprise and Coincidence. *Computational Linguistics*, vol. 19-1.

ENGUEHARD C. (1993). Acquisition de Terminologie à partir de Gros Corpus. In Proceedings of *Informatique & Langue Naturelle*, pp 373-384.

EVANS D & LEFFERTS R. (1993). Design and Evaluation of the CLARIT-TREC-2 System. In *TREC'93*, pp. 137-150.

GALE W. (1991). Concordances for Parallel Texts. In *Proc. of Seventh Annual Conference of the UW Center for the New OED and Text Research, Using Corpora*. Oxford.

GROSS G. (1996). *Les expressions figées en français*. Paris, Ophrys.

HABERT B. & JACQUEMIN C. (1993). Noms composés, termes, dénominations complexes: problématiques linguistiques et traitements automatiques. In *Traitement Automatique des Langues* 34(2), Association pour le Traitement Automatique des langues, France.

SALEM A. (1987). *La pratique des segments répétés*. Klincksieck. Paris.

SHIMOHATA S. (1997). Retrieving Collocations by Co-occurrences and Word Order Constraints. In *Proc. of ACL-EACL'97*.

SMADJA F. (1993). Retrieving Collocations From Text: XTRACT. In *Computational Linguistics*, 19 (1), pp 143-177.

SMADJA F. (1996). Translating Collocations for Bilingual Lexicons: A Statistical Approach. In *Association for Computational Linguistics*, 22 (1).