Désambiguïsation d'un Dictionnaire de Synonymes

Jean-Pascal Pfister
supervisé par
J.-C. Chappelier et M. Rajman

Octobre 2000

Résumé

Cet ouvrage traite de la désambiguïsation automatique d'un dictionnaire de synonymes. Le but est de regrouper les différents synonymes par classes de même signification. Dans un premier temps, la préoccupation est de formater les données de façon à ce qu'elles soient exploitables facilement. Dans un deuxième temps, l'étude se porte sur la recherche d'une heuristique de regroupement des synonymes en évitant les glissements de sens.

Table des matières

Chapitre 1

Introduction

1.1 But du projet

Dans le cadre des études de physique, le département demande à ses étudiants d'effectuer un stage d'une durée minimale d'un mois dans un domaine autre que la physique. Ce stage a été effectué du 2 août au 6 octobre dans le Laboratoire d'Intelligence Artificielle (LIA) à l'EPFL.

Le but de ce projet est de désambiguïser un dictionnaire de synonymes, c'est-à-dire de ne plus considérer les mots qui sont pour la plupart ambigus, mais chacun des identificateurs de sens que comporte le mot donné. Nous pourrons par la suite regrouper ces mots associés à leur numéro de sens et obtenir des classes qui possèdent la même signification.

Ce but précis s'inscrit dans un cadre beaucoup plus général qu'est par exemple celui de la recherche documentaire. En effet, qu'il s'agisse de retrouver un document dans énorme bibliothèque ou de trouver une page web très précise, le problème est le même, car celui qui opère la recherche ne connaît pas a priori, les mots précis du texte qu'il recherche, mais bien plutôt l'idée ou le concept qui se cachent derrière le texte. C'est pourquoi il peut être utile d'élargir les requêtes en utilisant un dictionnaire de synonyme.

Pour pouvoir étudier la similarité entre deux textes, il est nécessaire de recourir à des outils de représentation de texte et donc de représentation de mots. Pour l'instant, il existe une définition du sens d'un mot qui est vecteur dont chacune des composantes est un coefficient de co-occurence avec un autre mot dans un contexte donné. Il serait certainement intéressant de raffiner la représentation par la prise en compte des sens et non des mots. Le dictionnaire des synonymes désambiguïsé est certainement une aide pour atteindre ce but.

Sous un autre angle, ce travail pourrait être utile pour les lexicographes du dictionnaires des synonymes. En effet, puisque nous allons effectuer énormément de traitements sur les données de bases, il sera nécessaire d'utiliser des données propres. Ce travail met en évidence les incohérences du dictionnaire de base.

1.2 Définitions et modélisation

Pour ne pas tomber dans le piège de la confusion au niveau des termes employés dans cet ouvrage, il nous semble judicieux de commencer par clarifier certaines expressions, quitte à leur donner un sens un peu plus précis que leur sens usuel.

- La *graphie* est la représentation écrite d'un mot donné.
- *c.m.s.* est l'abréviation pour Catégorie Morpho-Synthaxique. Celle-ci informe sur le type grammatical (nom, adj., adv.,..), le genre et nombre du mot considéré. S'il

s'agit d'un verbe, la c.m.s. donnent des informations sur le temps, le mode du verbe en question...

- La *polysémie* est la propriété d'un mot qui possède plusieurs sens.

Arrêtons-nous un instant sur la notion de *synonymie* puisque cette notion est centrale à cette étude.

Dans cet ouvrage, nous utiliserons plusieurs types de relations synonymiques; il est donc nécessaire de définir chacune de ces relations.

Les auteurs de «la grammaire méthodique du français» [2] donnent une définition de la synonymie que nous utiliserons comme définition de la *synonymie stricte*:

> «Phénomène inverse de l'homonymie, la synonymie est la relation entre deux formes lexicales formellement différentes (elles se distinguent par leurs signifiants) mais de même sens (elles ont le même signifié). Au sens strict du terme, deux unités synonymes seraient donc sémentiquement équivalentes, c'est-à-dire librement substituables sans modifier le sens de l'énoncé où elles figurent.»

Cette définition est problématique pour un grand nombre de cas. «Si *briser, rompre* et *casser* sont commutables dans de nombreux contextes, on *brise la glace* mais on ne la *rompt* pas; on *rompt un traité*, mais non *la glace*,... De tels termes entretiennent alors une relation de *synonymie partielle* (ou *contextuelle*).»

Considérons un petit exemple:

```
<mot>rapport
<cms>n. m.
<sens>1. (Evaluer le rapport d'un placement)  <liste>BÉNÉFICE, FRUIT,...
<sens>2. (Donner lecture du rapport des gendarmes) <liste>COMPTE RENDU, EXPOSÉ,...
```

La première constatation est que le mot *rapport* est polysème. En fait, la majorité des mots sont polysèmes. Il est donc absurde de parler de *synonymie stricte* pour la relation entre *rapport* et *fruit* puisque justement le mot *rapport* possède d'autres sens pour lesquels *fruit* ne peut être substitué.

Par contre, si nous considérons uniquement les situations dans lesquelles le mot *rapport* est utilisé dans son sens premier, nous pouvons le remplacer par le mot *fruit*. Cette constatation nous amène à définir deux termes:

- Le *triplet* (Gr,CMS,N) est un ensemble composé d'une graphie **Gr** , d'une catégorie morpho-synthaxique **CMS** et d'un numéro de sens **N**, définissant ainsi de manière unique le sens précis.

- La *graphie-synonymie* est la relation qui lie un triplet (Gr1,CMS,N) à une graphie (Gr2) dans le cas où la graphie Gr2 peut remplacer la graphie Gr1 lorsque celle-ci est utilisée au sens N.

Ainsi, dans notre exemple, *fruit* est *graphie-synonyme* du triplet (rapport, n. m., 1). Nous noterons cette relation de la façon suivante:

$$Tr \stackrel{gs}{\Longrightarrow} Gs \tag{1.1}$$

où Tr est un triplet et Gs est une graphie-synonyme correspondante.

Intuitivement, la relation de synonymie est symétrique. Or, la définition que nous venons de poser n'est pas symétrique puisque la relation se situe entre deux objets différents: le triplet et la graphie.

Par la suite (cf. section 3.1 et 3.2), nous définirons une nouvelle relation de synonymie qui lie deux triplets entre eux que nous appellerons *triplet-synonymie*. Cette relation sera du type

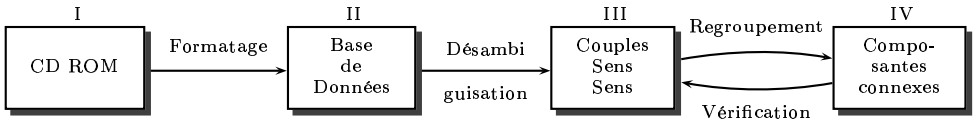


FIG. 1.1 – *Organigramme des étapes du projet*

$$Tr_1 \overset{s}{\Longrightarrow} Tr_2 \tag{1.2}$$

Ainsi, il sera possible d'obtenir la relation de symétrie puisque les deux objets comparés sont de même type. Il sera également possible d'imposer la transitivité de la relation, c'est-à-dire:

$$(Tr_1 \overset{s}{\Longleftrightarrow} Tr_2) \wedge (Tr_2 \overset{s}{\Longleftrightarrow} Tr_3) \Longrightarrow Tr_1 \overset{s}{\Longleftrightarrow} Tr_3 \tag{1.3}$$

où

- les Tr_i, $i = 1..3$ sont des triplets distincts;
- le symbole $\overset{s}{\Longleftrightarrow}$ représente la relation de triplet-synonymie.

Avec la une relation synonymique symétrique et transitive, il est possible de créer des classes d'équivalence qui représenteront chacune d'elles une *signification* différente.

Maintenant que les termes utilisés sont clairement définis, il est nécessaire d'accorder à chaque élément un symbole qui le caractérise.

Soient:

- $GR = \{Gr_1, \ldots, Gr_g, \ldots, Gr_G\}$, l'ensemble des G graphies figurants en entrée du dictionnaire;
- $Tr_{g,t} = (Gr_g, CMS_g, t)$ est le triplet composé d'une graphie Gr_g, d'une c.m.s. CMS_g et d'un numéro de sens t;
- $\Gamma = \{\Gamma_1, \ldots, \Gamma_g, \ldots, \Gamma_G\}$, l'ensemble des G ensembles de triplets pour une graphie G_g donnée tel que $\Gamma_g = \{Tr_{g,1}, \ldots, Tr_{g,t}, \ldots, Tr_{g,T_g}\}$;
- $GS_{g_t} = \{Gs_{g,t,1}, \ldots, Gs_{g,t,s}, \ldots, Gs_{g,t,S_{g,t}}\}$, l'ensemble des *graphies-synonymes* correspondants au triplet $Tr_{g,t}$.

1.3 Étapes du projet

La tâche à effectuer se divise en plusieurs étapes. La première consiste en la préparation d'une base de donnée dans un format facilement exploitable. Par la suite, il s'agira de la désambiguïser. Cette étape est clairement la plus importante du travail. Pour valider l'heuristique utilisée nous utiliserons plusieurs outils différents. Finalement, nous établirons les classes de signification à partir du de la base de données désambiguïsée.

Chapitre 2

Formatage des données

2.1 Aperçu du formatage

La donnée brute pour cette étude de désambiguïsation est un dictionnaire de synonymes Hachette. Il s'agit de formater les données de base de façon à ce qu'elles soient exploitables pour la désambiguïsation.

Cette tâche serait triviale si les données brutes étaient consistentes. Puisque tel n'est pas le cas, il a fallu opérer plusieurs corrections dont certaines à la main.

La première étape n'est qu'une transformation de format; il n'est donc pas nécessaire de s'y attarder. Cette transformation donne lieu à un dictionnaire balisé sous la forme suivante:

```
<mot>rigolo, ote
<cms>adj.
<sens>1. Fam.  <liste>AMUSANT, COMIQUE, DÉSOPILANT, DRÔLE, MARRANT (fam.),
         PLAISANT, TORDANT (fam.).<forme particuliere>n.
<sens>2. Fam. <forme particuliere>rigolo, ote (L'oncle Jean-Pierre,
         c'est un vrai rigolo) <liste>BOUTE-EN-TRAIN, COMIQUE,
         FARCEUR, PLAISANTIN.
<mot>abîme
        <cms>n. m.
        <sens>1. Litt. (Descendre dans un abîme)  <liste>AVEN, GOUFFRE, PRÉCIPICE.
        <sens>2. Spécialement sous la mer <liste>ABYSSE, FOSSE.
```

Dans la deuxième étape quelques petites corrections ont été effectuées à la main sur ce fichiers. Il s'agit de corrections qui ne peuvent pas être détectées automatiquement puisqu'il s'agit d'erreurs ponctuelles dans le dictionnaire de base.

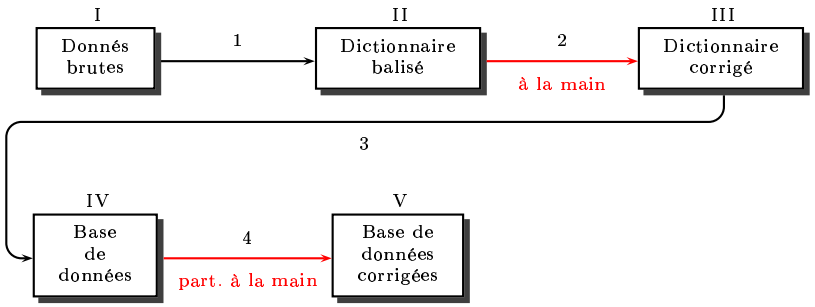


FIG. 2.1 – *Organigramme du formatage*

A partir du fichier corrigé, nous pouvons créer une base de donnée plus facilement exploitable. Celle-ci est constituée des dix champs suivants:

1. la **graphie** figurant en entrée du dictionnaire;
2. le suffixe donnant la marque du **féminin** s'il s'agit d'un adjectif ou d'un nom qui possède une forme au féminin;
3. catégorie morpho-synthaxique (**C.m.s.**);
4. le **numéro de sens** est simplement le numéro qui figure après la balise `<sens>` (pour plus de détails, se référer à la section 3.2.1);
5. le **niveau de langue du sens** indique s'il faut considérer le sens dans un contexte littéral, familier, figuré etc;
6. une petite **phrase illustrative** qui donne un exemple d'utilisation de la graphie dans un sens donné;
7. le champ **spécialement** contient une petite description de l'usage spécifique de la graphie dans le sens donné;
8. la **graphie-synonyme** (la définition se trouve dans la section 1.2);
9. parfois, le lexicographe indique le **niveau de langue de la graphie-synonyme** (par exemple: familier, figuré,...); celui-ci figure entre parenthèses apposé à la graphie-synonyme;
10. lorsqu'une graphie est dérivée d'une autre (par exemple, *s'abaisser* dérive de *abaisser*), le dictionnaire possède une balise `<forme particuliere>`. Nous avons donc créé une nouvelle entrée de la base de donnée tout en gardant l'information de la graphie de laquelle la nouvelle dérive dans un champ que nous appellerons **forme particulière**.

Pour le $5^{\text{ième}}$ champ, se pose le problème du pluriel. Lorsque certains mots ont un sens différent au pluriel, le lexicographe a décidé de le mentionner par *Plur.* à la place du niveau de langue du sens, comme le montre l'exemple suivant:

```
<mot>abord
<cms>n. m.
<sens>1. (Un lieu d'un abord facile)  <liste>ACCÈS.
<sens>2. Plur. (Visiter les abords d'une ville)
<liste>ALENTOURS, ENVIRONS, VOISINAGE.
```

Nous avons pour cela créé une nouvelle entrée contenant le mot au pluriel. Comme le montre l'exemple, le lexicographe ne donne pas la forme lexical au pluriel, c'est pour cela qu'une règle pour accorder au pluriel a été nécessaire: à moins que la graphie se termine par «s, z» ou «x», ajouter le suffixe «s»; pour les graphies se terminant par «eau» ou «eu», ajouter un «x»; les terminaisons en «al» sont remplacées par «aux»; le cas de «aileul» est traité séparément.

A partir du dictionnaire balisé, il est possible de créer une base de donnée. Le tableau 2.1 illustre le format de cette base de données.

La dernière étape du formatage est aussi une étape qui a nécessité quelques corrections à la main. Les corrections sont de deux types différents: premièrement au niveau des *formes particulières* et deuxièmement au niveau des majuscules.

1	2	3	4	5	6	7	8	9	10
rigolo		adj.	1	Fam.			amusant		
rigolo		adj.	1	Fam.			comique		
rigolo		adj.	1	Fam.			désopilant		
rigolo		adj.	1	Fam.			drôle		
rigolo		adj.	1	Fam.			marrant	fam.	
rigolo		adj.	1	Fam.			plaisant		
rigolo		adj.	1	Fam.			tordant	fam.	
rigolo	ote	n.	2	Fam.	L'oncle ... rigolo		boute-en-train		rigolo
rigolo	ote	n.	2	Fam.	L'oncle ... rigolo		comique		rigolo
rigolo	ote	n.	2	Fam.	L'oncle ... rigolo		farceur		rigolo
rigolo	ote	n.	2	Fam.	L'oncle ... rigolo		plaisantin		rigolo
abîme		n. m.	1	Litt.	D. ... un abîme		aven		
abîme		n. m.	1	Litt.	D. ... un abîme		gouffre		
abîme		n. m.	1	Litt.	D. ... un abîme		précipice		
abîme		n. m.	2			sous la mer	abysse		
abîme		n. m.	2			sous la mer	fosse		

TAB. 2.1 – *Exemple de base de données de graphies-synonymes construite à partir du dictionnaire.*

2.2 Problème de la forme particulière

Comme nous le voyons dans l'exemple précédent, les nom *rigolo, rigolote* sont une forme particulière de l'adjectif *rigolo*. Jusque là, il n'y a pas de problèmes, car la c.m.s. «n.» a été explicitement donnée après la balise `<forme particuliere>`. Dans beaucoup de cas, la c.m.s. des formes particulières n'est pas donnée explicitement. Dans la plupart de ces situations, la c.m.s. est la même que celle donnée dans l'entrée principale; dans les autres situations, il a fallu établir les c.m.s. unes à unes. La liste détaillée des c.m.s. rajoutées se trouve en annexe C.

Certaines c.m.s. corrigées à la main ont été classées dans des groupes assez larges bien qu'il existe un type de c.m.s. plus précis. Par exemple *en abondance* est un groupe prépositionnel, mais il a été classé comme un adv. car il n'y avait pas d'autres groupes prépositionnels dans le dicionnaire.

2.3 Problèmes des majuscules

Le dictionnaire de base a été construit comme suit: les entrées sont en minuscules et les graphies-synonymes sont en majuscules. Nous avons décidé de tout mettre en minuscule pour qu'il puisse y avoir correspondance entre les entrées et les graphies-synonymes. Le seul problème rencontré est qu'il existe tout de même quelques majuscules, par exemple dans des mots comme dans «l'Éternel». C'est pour cela qu'il a fallu les répertorier et effectuer les changements nécessaires dans les listes des graphies-synonymes.

Chapitre 3

Désambiguïsation

3.1 Première heuristique

Nous arrivons maintenant au point crucial de notre étude. Quels sont les critères qui nous permettent de relier deux triplets par une relation de synonymie?

La première heuristique considérée fut la suivante: deux triplets sont dits *triplet-synonymes* s'ils partagent au moins une graphie en commun dans leur liste respective de *graphies-synonymes* (sans considérer la graphiqe source elle-même). Cette situation est représentée graphiquement dans la figure 3.1

Nous noterons cette relation de la façon suivante:

$$Tr_1 \overset{s_1}{\Longleftrightarrow} Tr_2 \tag{3.1}$$

Cette heuristique a l'avantage d'être relativement simple. Par contre, il est clair qu'elle donne lieu à des déviations de sens. Par exemple, le triplet (main, n. f., 1) est lié à (pli, n. m.,2) car ils ont tout deux la graphie *pince* dans la liste de leur graphie-synonymes. Cet exemple montre bien la faiblesse de l'heuristique, car celle-ci ne tient pas compte du fait que les graphies-synonymes communs aux deux triplets sont ambiguës.

3.2 Deuxième heuristique

3.2.1 Principe

Les règles qui gouvernent la deuxième heuristique sont un peu peu plus élaborées que pour la première.

1. Un triplet $Tr_{g,t}$ est lié au triplet $Tr_{g',t'}$ si la graphie-synonyme $Gs_{g,t,s}$ est la même que la graphie $Gr_{g'}$ et si la cardinalité de l'intersection de leur graphies-synonymes est maximal, c'est-à-dire si

$$I_{g,t,g',t'} = |\{(Gr_g \cup GS_{g,t}) \cap GS_{g',t'}\}| \tag{3.2}$$

   est maximal pour $t' \in \{1..T_{g'}\}$.

2. S'il n'existe pas de graphies-synonymes communes, c'est-à-dire si $I_{g,t,g',t'} = 0, \forall t' \in \{1..T_{g'}\}$ alors un $T_{g'} + 1$ème sens est crée. Le caractère «-» sera placé devant le numéro du sens pour rappeler que ce triplet a été créé automatiquement. La liste des graphies associées au nouveau triplet $Tr_{g',T_{g'}+1}$ est $GS_{g',T_{g'}+1} = \{Gs_{g,t,s}\}$

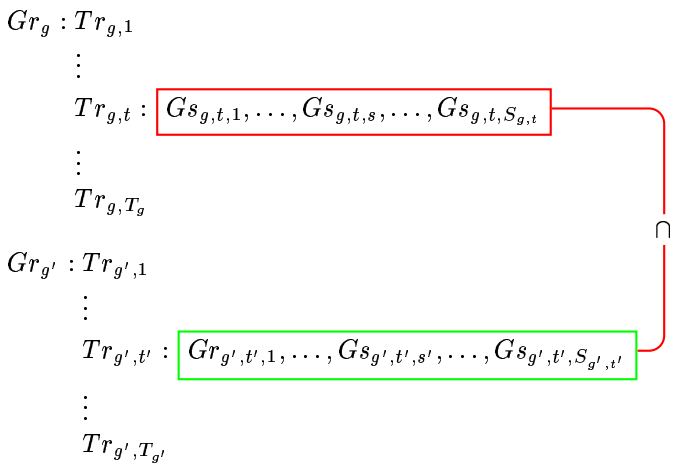


FIG. 3.1 – *Première heuristique*

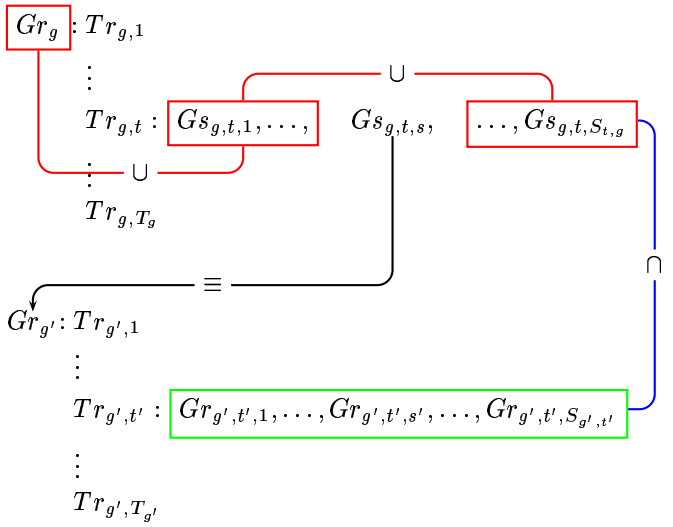


FIG. 3.2 – *Deuxième heuristique*

3. Si la graphie $Gr_{g,t,s}$ n'est pas une entrée, l'entrée est créée et nous associons le numéro de sens «0» pour garder à l'esprit que le nouveau triplet est créé automatiquement. Dans ce cas nous ne connaissons pas a priori la c.m.s. du nouveau triplet. Pourtant, nous savons que celui-ci a la même c.m.s. principale que celle du triplet $Tr_{g,t}$. Si par exemple, la c.m.s. du triplet $Tr_{g,t}$ est *n. f.*, celle du nouveau triplet sera *n. IC. IC* est juste une marque pour montrer que cette c.m.s. n'existait dans le dictionnaire original.

La figure 3.2 représente graphiquement la deuxième heuristique.
Nous noterons cette nouvelle relation de triplet-synoymie de la façon suivante:

$$Tr_1 \overset{s_2}{\Longrightarrow} Tr_2 \tag{3.3}$$

Cette relation n'est pas symétrique dans sa définition, mais nous la rendons symétrique pour répondre à nos exigences de synonymies:

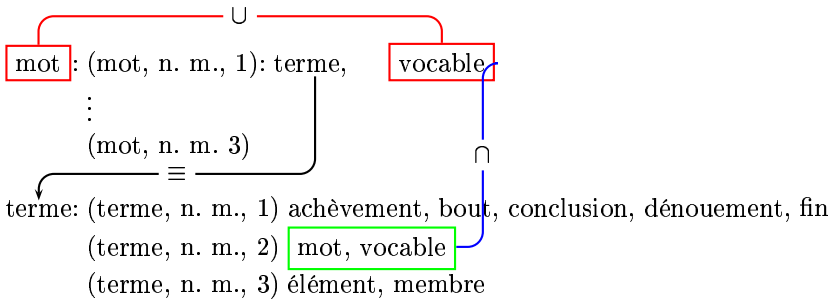


FIG. 3.3 – *Illustration de la deuxième heurisitque*

$$Tr_1 \overset{s_2}{\Longleftrightarrow} Tr_2 \tag{3.4}$$

La figure 3.3 illustre bien la deuxième heuristique.

3.2.2 Cas particulier

La deuxième heuristique a également des faiblesses. Prenons l'exemple de *rejeter* et *éliminer*.

```
<mot>éliminer
<cms>v.
<sens>1. (Eliminer une équipe de football)  <liste>DISQUALIFIER,
         EXCLURE.
<sens>2. (Eliminer un collaborateur incompétent)  <liste>BALANCER
         (fam.), CHASSER, SE DÉBARRASSER DE, ÉCARTER, ÉVINCER,
         EXPULSER, RENVOYER, VIDER (fam.), VIRER (fam.).
<sens>3. (Eliminer un témoin gênant)  <liste>ABATTRE, ASSASSINER,
         SE DÉBARRASSER DE, DESCENDRE (fam.), LIQUIDER (fam.),
         SUPPRIMER, TUER.
<sens>4. (Eliminer qqch de sa mémoire)  <liste>CHASSER, DISSIPER, EFFACER.
<sens>5. (Eliminer un calcul rénal)  <liste>ÉVACUER, EXPULSER, REJETER.

<mot>rejeter
<cms>v.
<sens>1. (Rejeter une balle)  <liste>RELANCER, RENVOYER.
<sens>2. (Rejeter un paragraphe à la fin d'un chapitre)
         <liste>REPORTER, REPOUSSER.
<sens>3. (Rejeter ce qui avait été absorbé)  <liste>ÉVACUER, EXPULSER,
         RENDRE, VOMIR.
<sens>4. (Des malheureux que tout le monde rejette)  <liste>BANNIR,
         CHASSER, ÉCARTER, EXCLURE, PROSCRIRE, REFOULER.
<sens>5. (Rejeter une offre)  <liste>BALAYER, DÉCLINER, ÉCARTER,
         ÉLIMINER, RÉCUSER, REFUSER, REPOUSSER.
<sens>6. (Rejeter une faute sur son associé)  <liste>ATTRIBUER À,
         IMPUTER À.
```

En appliquant l'heuristique, nous obtenons la situation décrite par la figure 3.4.

La première constatation est que notre algorithme ne donne pas des relations symétriques, contrairement à celui de la première heuristique.

En fait nous devrions avoir la situation suivante:

Malheureusement, nous n'avons pas trouvé de critères objectifs pour arriver automatiquement à cette situation idéale.

Il peut aussi arriver qu'une fusion de sens soit désirable, comme dans l'exemple suivant:

```
<mot>vomir
<cms>v.
<sens>1. (La voiture lui donne envie de vomir)  <liste>DÉGOBILLER
```

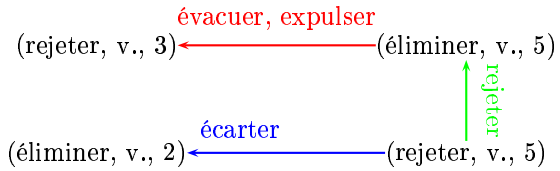


FIG. 3.4 – *Représentation des relations entre* rejeter *et* éliminer. *Les graphies apposées aux flèches sont les graphies-synonymes partagées par les deux triplets*

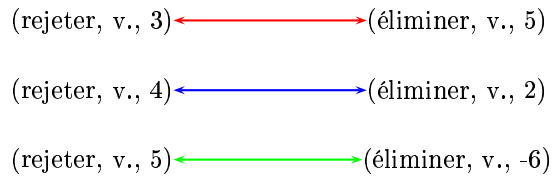


FIG. 3.5 – *Situation idéale pour rejeter et éliminer*

```
        (fam.), DÉGUEULER (fam.), GERBER (fam.), RENDRE.
<sens>2. (Vomir tout son déjeuner)  <liste>DÉGURGITER, RÉGURGITER,
        REJETER, RENDRE.
<sens>3. Fig. (Vomir des injures)  <liste>BALANCER (fam.), JETER,
        LANCER, PROFÉRER, SORTIR (fam.).
<sens>4. Fig. et litt. (Vomir les tièdes)  <liste>ABHORRER (litt.),
        ABOMINER (litt.), DÉTESTER, EXÉCRER, HAÏR, HONNIR (litt.).
```

Dans cette situation, il serait judicieux de pouvoir fusionner le sens 1 et le sens 2 car, ils sont identiques.

3.3 Paramètres de modification de l'heuristique

Pour changer le résultat obtenu par la deuxième heuristique, il est possible de modifier quelques paramètres:

- Il est possible d'introduire un paramètre α qui représente le nombre minimal de graphies-synonymes communs entre deux triplets $Tr_{g,t}$ et $Tr_{g',t'}$ pour attribuer la relation de *graphie-synonymie* entre ces deux triplets. En d'autres termes, si $I_{g,t,g',t'} \geq \alpha$, la relation est créée, sinon, un $T_{g'} + 1$ème sens est crée.
- Il est aussi possible de définir la fonction $I_{g,t,g',t'}$ de manière un peu différente. Soit $I^{\beta}_{g,t,g',t'} = \beta * |\,(Gr_g \cap GS_{g',t'})\,| + |\,(GS_{g,t} \cap GS_{g',t'})\,|$ notre nouvelle fonction. Celle-ci met une importance différente à la présence de la graphie Gr_g parmi l'ensemble des graphies-synonymes $GS_{g',t'}$ par rapport à la présence d'une graphie-synonyme de GS_{g_t} parmi $GS_{g',t'}$. Si $\beta = 1$, nous avons $I^{\beta}_{g,t,g',t'} = I_{g,t,g',t'}$

Chapitre 4

Outils de vérification

4.1 Composantes connexes

Maintenant que des couples de triplets sont formés, nous pouvons utiliser la relation de transitivité (relation 1.3) pour créer des composantes connexes. Le programme qui construit ces composantes créé en même temps un fichier dans lequel se trouve toute l'information nécessaire pour faire les statistiques voulues. Ce fichier est en fait une base de données constituée des huit champs suivants:

1. **numéro** de la composante
2. liste de tous les **triplets** figurant dans cette composante. Un numéro est attribué, de façon relative à la composante, à chaque triplet.
3. le **nombre de triplets**
4. la liste des **arcs orientés**. Chacun des éléments de cette liste est un couple de deux nombre correspondants respectivement à deux triplets mentionnées dans la liste de triplets.
5. le **nombre d'arcs orientés**
6. la liste des **arcs non-orientés**.
7. le **nombre d'arcs non orientés**.
8. le coefficient de **connectivité** κ défini par l'équation 4.2

4.2 Représentation graphique

Pour déterminer rapidement les liaisons gênantes d'un graphe, il est utile de recourir à un programme de représentation graphique. Mais avant de continuer, il est peut-être bon d'éclairer quelques notions de base de la théorie des graphes.

- Un *graphe* consiste en une collection finie d'objets V et une relation binaire et non-réflexive sur V [1]. Dans ce rapport, les objets ou les nœuds représentent les triplets et la relation binaire est donnée par la relation de triplet-synonymie.
- Un *graphe orienté* est un graphe pour lequel la relation binaire peut être représentée par une collection de paires ordrées.
- Un *graphe complètement connecté* est un graphe qui possède $\frac{N(N-1)}{2}$ arcs si N est le nombre de nœuds.

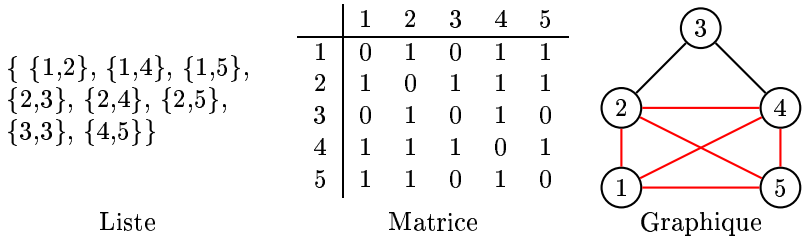


FIG. 4.1 – *3 représentations de graphe*

- Une *composante connexe* est un graphe connecté, c'est-à-dire. qu'il existe au moins un chemin pour relier deux nœuds quelconques.
- Une *clique* est un sous-graphe complètement connecté

Un graphe peut se représenter de plusieurs façons différentes:

Liste: La représentation sous forme de liste est judicieuse lorsque la connectivité du graphe est faible, c'est-à-dire lorsque le nombre d'arcs par rapport au nombre de nœuds est faible.

Matrice: La représentation matricielle permet quant à elle, de vérifier aisément certaines propriétés. Par exemple, si le graphe est orienté, la matrice devient symétrique.

Graphique: Le graphique est la représentation qui donne le plus rapidement un aperçu global du graphe.

Remarquons que l'ensemble des nœuds $\{1, 2, 4, 5\}$ forment une clique puisque chacun des nœuds de cet ensemble est relié avec 3 autres nœuds de ce même ensemble. Il s'agit en fait de la clique maximale puisqu'il n'y a pas de cliques plus grandes que celle-ci

Nous avons utilisé premièrement Mathematica, mais bien vite, nous avons réalisé que cet outil souffrait de deux grandes faiblesses: premièrement, il n'y a pas de fonction qui dispose les nœuds de façon automatique autrement que sur un cercle; deuxièmement, il n'y a pas la possibilité de bouger à la main les nœuds pour obtenir une configuration différente. Par contre Mathematica a l'avantage de déterminer la clique maximale. Quelques lignes de programmation sous Mathematica permettent de déterminer les cliques maximales. Néanmoins nous avons opté pour la solution *vgj*. Il s'agit d'une petite application Java qui permet de remplir les fonctionnalités mentionnées.

Sur le graphique 4.3, nous voyons bien que le lien entre le triplet (récapitulation, n. f.,1) et (revue, n. f., 1) devrait être détruit pour obtenir deux graphes à plus grand connectivité. Il est évident, qu'il y aurait d'autres liens à détruire ou à rajouter. La figure 4.2 montre un autre exemple de composante connexe. Pour plus de lisibilité, l'étiquetage a été supprimé. Cet outil de représentation nous permet d'imaginer les changements dans l'heuristique pour obtenir des composantes plus compactes.

4.3 Statistique

Une autre façon d'approcher le problème est de le regarder de façon globale et de ne pas s'arrêter sur une composante précise qui peut être un cas particulier.

Avant de poursuivre cette approche, définissons un coefficient de connectivité κ. Celui-ci vaut 0 si la composante a un nombre minimal d'arcs ($N-1$ arcs) et 1 si la composante est une clique ($\frac{N(N-1)}{2}$ arcs). Nous obtenons:

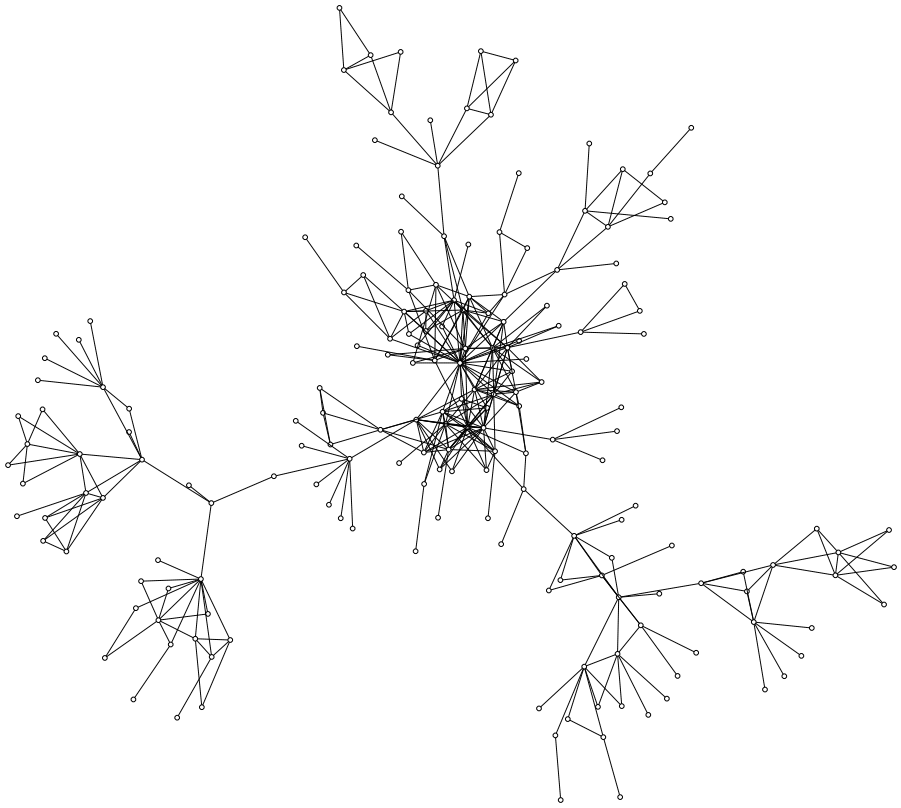


FIG. 4.2 – *Composante à 185 éléments*

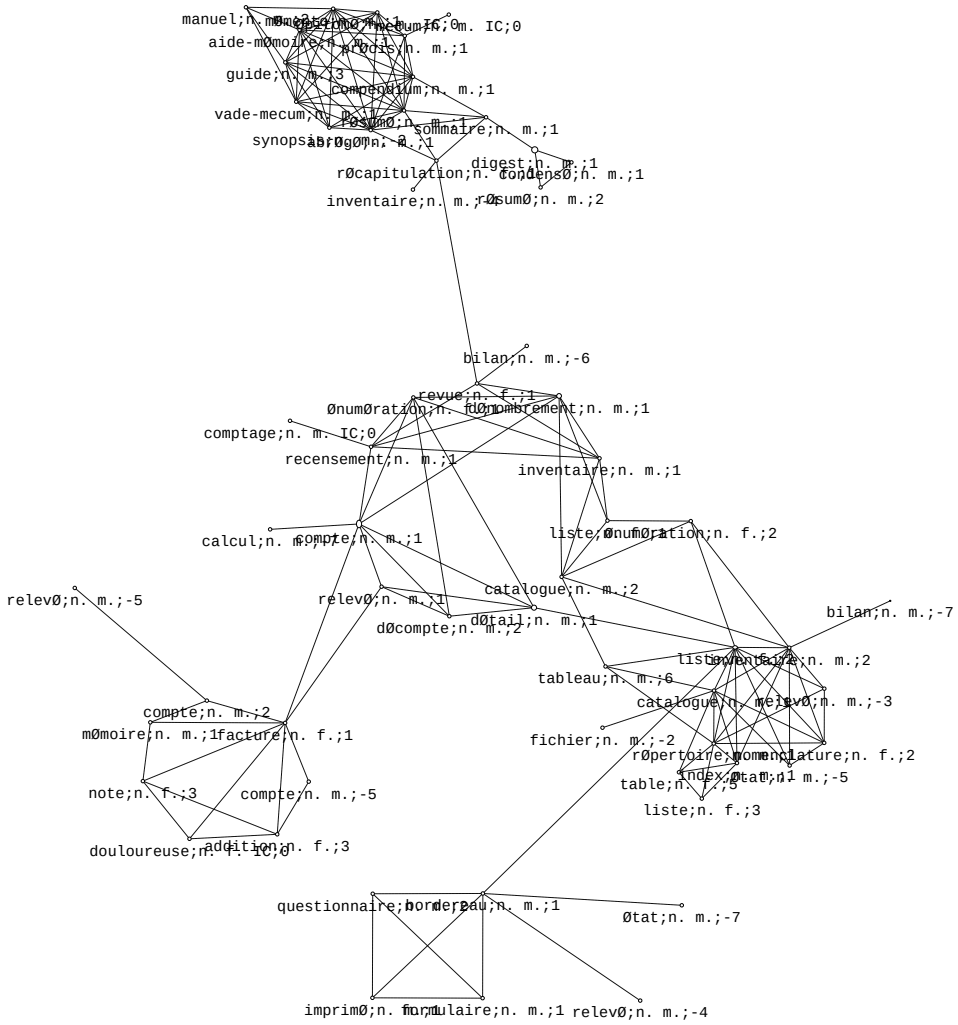


FIG. 4.3 – *Composante à 60 éléments*

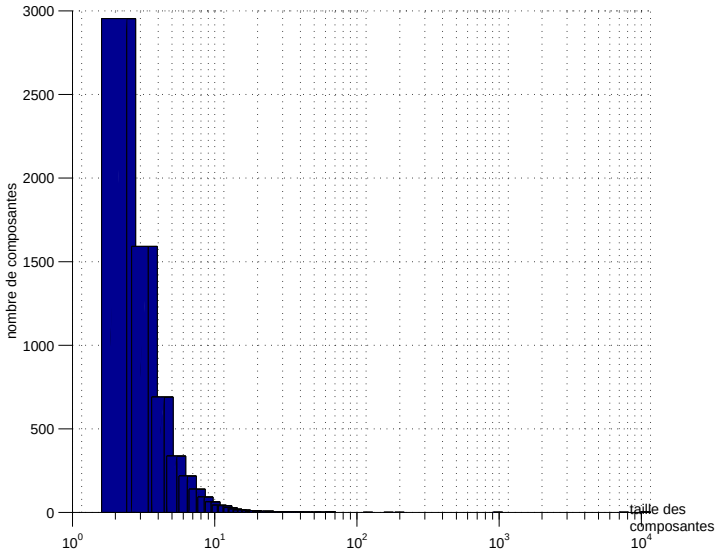


FIG. 4.4 – *Distribution des nœuds*

$$\begin{aligned} \kappa(A,N) &= 2\frac{A-(N-1)}{(N-1)(N-2)} \\ \kappa(1,2) &= 1 \end{aligned} \tag{4.1}$$

où A est le nombre d'arcs et N est le nombre de nœuds

Il s'agit donc d'observer la distribution des nœuds, des arcs et du coefficient κ sur l'ensemble des composantes.

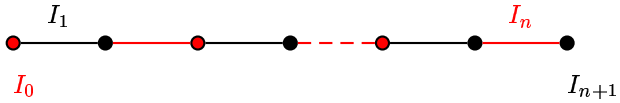
Pour le détail des valeurs numériques se référer à l'annexe A.

Les distribution des nœuds et des arcs sont représentées respectivement par les figures 4.4 et 4.4. Nous observons tout de suite qu'il existe d'énormes composante à plus de 7000 éléments. Ce graphique montre bien la limite de la deuxième heuristique.

Pour savoir si le taux de connectivité est élevé, il peut être intéressant de considérer la distribution du coefficient κ. Pour pouvoir représenter convenablement cette distribution, il faut déterminer une partition de l'intervalle $I = [0, 1]$. Comme le point 0 et le point 1 sont des points intéressants, l'intervalle a été partitionné en $n + 1$ intervalles disjoints comme suit:

- $I_0 = \{0\}$
- $I_k =]\frac{k-1}{n}, \frac{k}{n}], k \in 1, \ldots, n-1$
- $I_n =]\frac{n-1}{n}, 1[$
- $I_{n+1} = \{1\}$

Ce partitionnement peut être représenté de la façon suivante:



Nous voyons rapidement sur le graphe 4.6 qu'il y a beaucoup de composantes à connectivité maximale. La question se pose de savoir s'il s'agit de composantes à deux nœuds

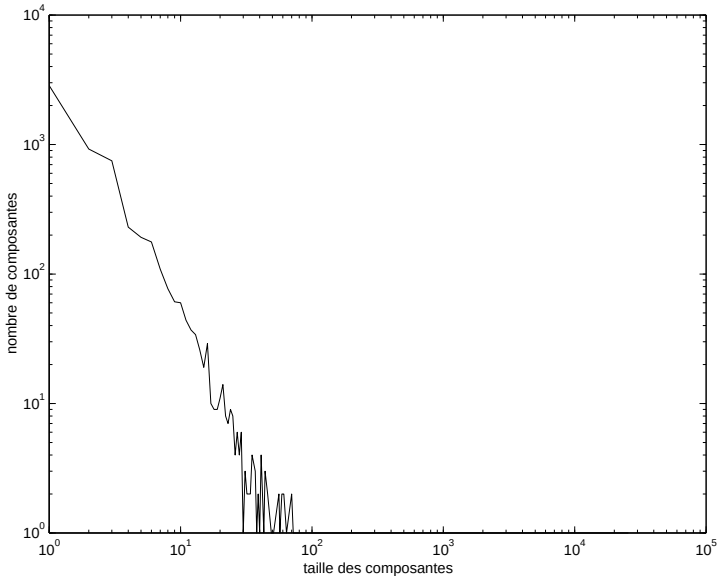


FIG. 4.5 – *Distribution des arcs*

uniquement, ou est-ce qu'il y a effectivement une majorité de cliques. Pour répondre à cette question, il faut établir une distribution en fonction de deux paramètres: le nombre de nœuds dans la composante, le coefficient de celle-ci. Nous obtenons le graphique 4.7

Remarquons premièrement que pour des questions de lisibilité, le graphe a été volontairement coupé dans l'axe représentant la taille des composantes.

Nous constatons aisément que la majorité des cliques sont des composantes à deux éléments. Si nous considérons les composantes à trois éléments, nous voyons qu'il n'y a que la valeur 1 ou 0, ce qui est normal puisque le graphe peut être soit totalement connecté (3 arcs) ou connecté au minimum (2 arcs). Le graphique devient donc intéressant à partir de composante contenant 4 triplets (cf. figure 4.8)

Un autre élément que nous pouvons observer globalement est le taux de doublons. Nous appellerons *doublons*, les triplets qui possèdent la même graphie et la même c.m.s. dans une même composante. Soient:

- $\Xi(N)$, le nombre de composantes à N nœuds.
- $\Delta(N)$, le nombre de composantes à N nœuds qui possèdent un ou plusieurs doublons.

Avec ces deux définitions nous pouvons définir une nouvelle entité que nous appellerons le *coefficient de doublage*

$$\delta_1(N) = \frac{\Delta(N)}{\Xi(N)}$$

Graphiquement, nous obtenons la figure 4.9

Nous voyons que le coefficient converge vite vers 1. Cela est dû au fait que le nombre de grosses composantes est très petit et bien souvent égal à 1. Il serait plus judicieux de définir un taux de doublons pour chaque composante et effectuer la moyenne pour toutes les composantes de même taille. Il est donc intéressant de définir un coefficient plus raffiné. Soient

- $u(C)$, le nombre de triplets uniques à un numéro de sens près, dans une composante C donnée.

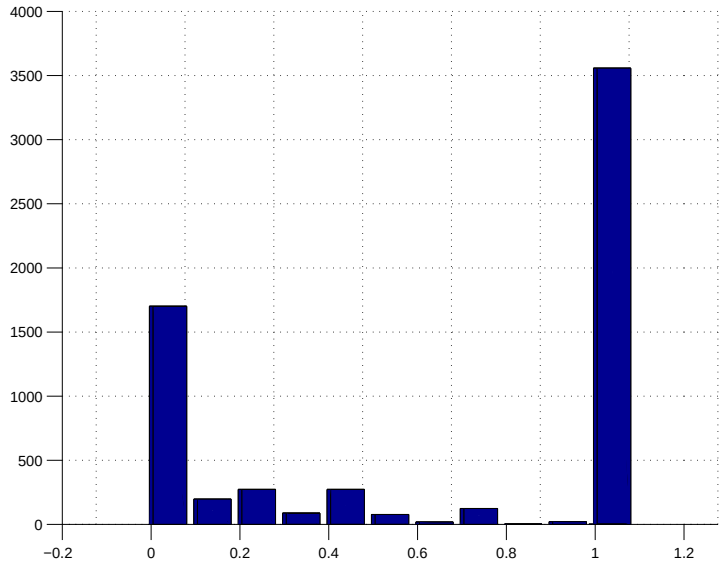


FIG. 4.6 – *Distribution de κ*

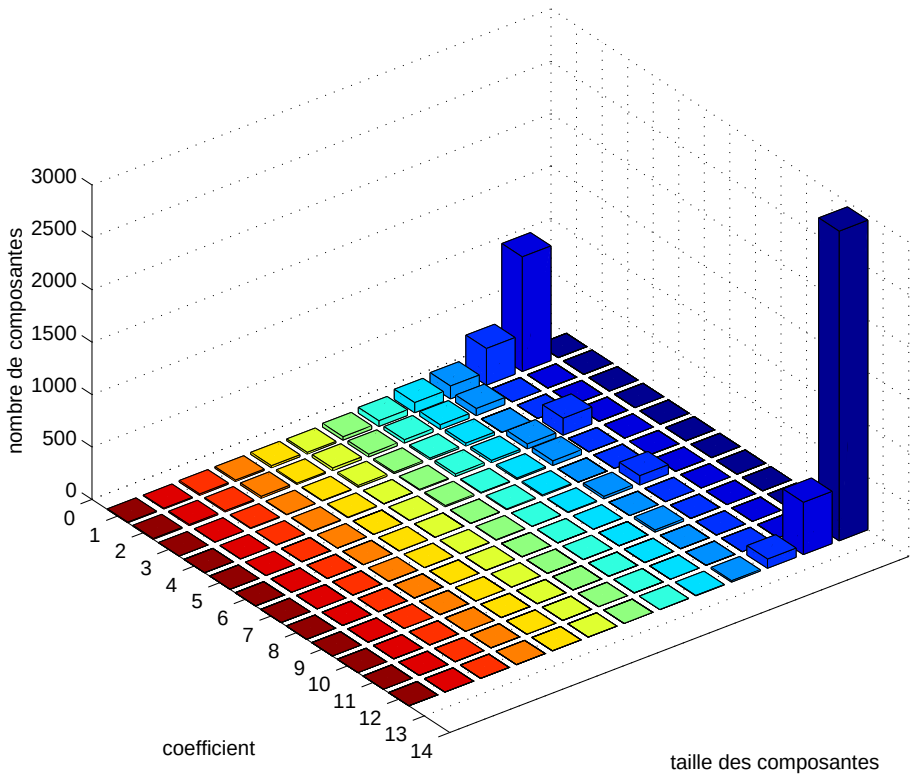


FIG. 4.7 – *Double distribution*

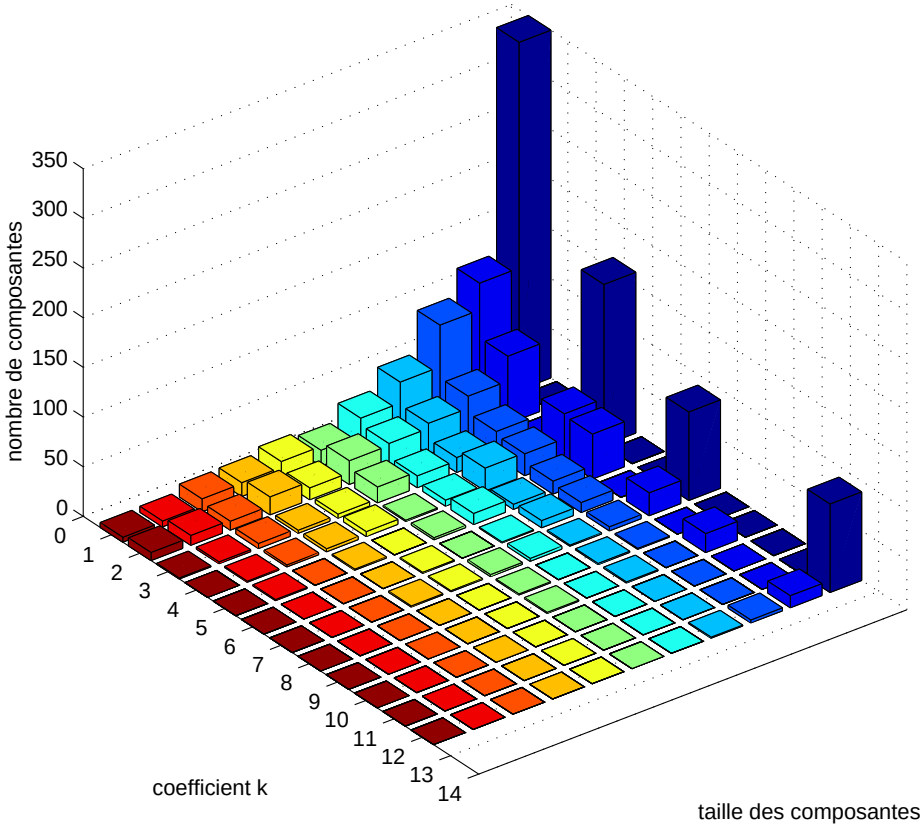


FIG. 4.8 – *Double distribution coupée*

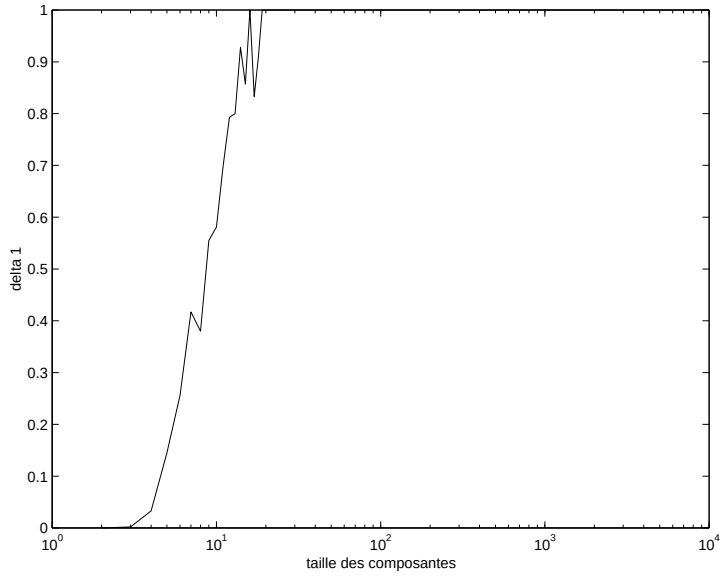


FIG. 4.9 – *Coefficient de doublage*

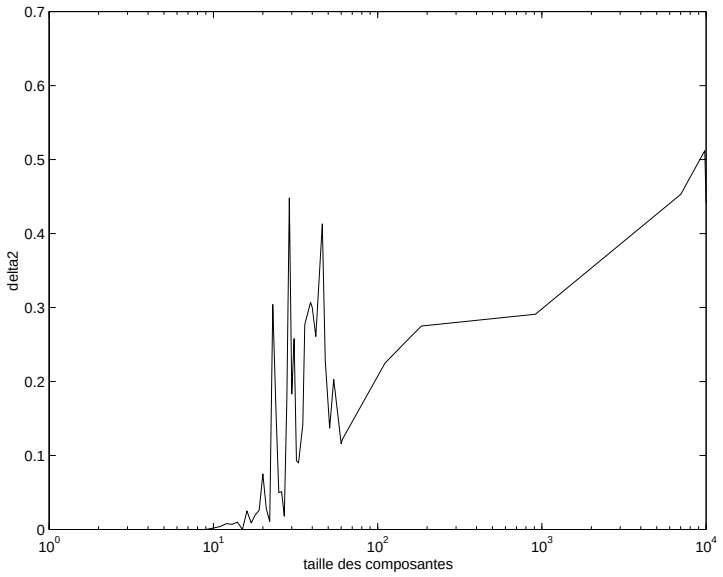


FIG. 4.10 – *Coefficient raffiné de doublage*

- $\nu(C)$, le nombre de triplets dans une composante C donnée.
- $d(C) = 1 - \frac{u(C)}{\nu(C)}$, le taux de doublons pour une composante C donnée.
- $\Omega(N) = \{C | \nu(C) = N\}$, l'ensemble des composantes à N nœuds.

Définissons notre deuxième coefficient que nous appellerons *coefficient raffiné de doublage*

$$\delta_2(N) = \frac{1}{\Xi(N)} \sum_{C \in \Omega(N)} d(C) \tag{4.2}$$

Ce nouveau coefficient est représenté par la figure 4.10 et la liste des valeurs numériques pour les deux coefficients sont en annexe B.

Ce dernier coefficient indique soit la faiblesse de l'heuristique parce qu'elle fusionne des sens différents, soit mauvaise qualité des lexicographes qui divisent en plusieurs sens une seule et même signification. La première hypothèse paraît plus raisonnable.

Chapitre 5

Traitement final des données

5.1 Division de composantes

Comme nous l'avons vu dans la section 3.3, il est possible de jouer sur certains paramètres pour changer la désambiguïsation et ainsi changer la distribution du nombre de nœuds. Dans cette section, nous allons voir s'il est judicieux d'augmenter le paramètre α pour casser les grosses composantes.

Comme nous le voyons sur le tableau 5.1, dès que le paramètre α arrive à 2, le nombre total de nœuds a augmenté de plus de 66%, ce qui est énorme. Nous voyons immédiatement que cette solution n'est pas bonne pour diviser les grosses composantes.

5.2 Fusion de nœuds

Jusqu'à un certain point, l'hypothèse que tous les éléments d'une même composante ont la même signification est vérifiée. Au delà, les composantes sont de tailles déraisonnables ; elles ne peuvent donc pas satisfaire notre hypothèse. Nous avons déterminer empiriquement cette limite aux composantes dont la taille ne dépasse pas 50 nœuds. En dessous de cette limite, nous pouvons fusionner les triplets qui ont la même graphie et même c.m.s., car la distinction de sens devient inutile.

α	nbre de composantes	taille de la comp. max.	nbre de nœuds
1	1	912	912
2	126	419	1520
3	287	179	2134
4	412	202	2663
5	483	170	2973
6	547	73	3222

(a)

α	nbre de composantes	taille de la comp. max.	nbre de nœuds
1	1	9997	9997
2	1715	6039	16887
3	3629	540	24093
⋮	⋮	⋮	⋮

(b)

TAB. 5.1 – *Évolution de deux composante en fonction de* α

Chapitre 6

Conclusion

6.1 Buts atteints et suite du travail

Nous sommes parvenus à formater correctement les données, à tester deux heuristiques pour la désambiguïsation et à regrouper les triplets en composantes connexes. Nous sommes également parvenus à représenter graphiquement les composantes, ce qui est un des buts que nous espérions atteindre.

Il reste néanmoins un certain travail à fournir pour que la grande partie des données soient exploitables. En effet, les grosses composantes connexes (composantes contenant plus de 100 triplets) contiennent en tout 28179 triplets sur les 50913, ce qui représente plus de 55%. Il s'agira de trouver une heuristique plus efficace ou raffiner celle qui existe. Une autre possibilité serait de trouver une heuristique de cassure de composantes. Ceci n'a malheureusement pas été possible durant ce stage puisque le programme *vgj*, outil indispensable pour une heuristique de cassure, ne fut découvert qu'à la toute fin du stage.

6.2 Rétrospective du stage

Ce stage était très intéressant et m'a permis de découvrir énormément dans le domaine de la programmation avec Perl, Mathematica, gawk, commandes unix,...

J'ai beaucoup apprécié la bonne volonté et la patience des assistants pour expliquer tel ou tel astuce.

S'il est vrai qu'il est enrichissant d'avoir deux superviseurs qui n'ont pas toujours le même point de vue, il n'est pas moins vrai que cela peut mener à passer du temps dans une direction que l'autre superviseur jugera inutile. Ceci était particulièrement frappant lors du retour de vacances de J.-C. Chappelier. Mais dans l'ensemble la communication a bien fonctionné.

Bibliographie

[1] Golumbic M. Ch., *Algorithmic Graph Theory and Perfect Graphs*, Academic Press, p. 3.

[2] Riegel M., Pellat J.-C., Rioul R. *Grammaire méthodique du français*, PUF, 1994, 5^e éd. 1999, p. 560.

Annexe A

Distributions

N	$\Xi(N)$
2	2954
3	1591
4	690
5	337
6	218
7	139
8	92
9	63
10	43
11	40
12	29
13	20
14	14
15	14
16	10
17	6
18	11
19	4
20	4
21	5
22	8
23	1
24	1
25	4
26	3
27	4
28	1
29	1
30	2
31	2
32	1
33	2
35	1
36	1
39	1
40	1
42	1
46	1
48	1
51	1
54	1
60	2
61	2
111	1
155	1
185	1
912	1
7007	1
9812	1
9997	1

A	nbre de comp.
1	2857
2	924
3	748
4	230
5	192
6	177
7	109
8	77
9	61
10	60
11	44
12	37
13	34
14	26
15	19
16	29
17	10
18	9
19	9
20	11
21	14
22	8
23	7
24	9
25	8
26	4
27	6
28	4
29	6
30	1
31	3
32	2
33	2

A	nbre de comp.
34	2
35	4
37	3
38	1
39	2
40	1
41	4
42	2
43	1
44	3
46	2
49	1
50	1
51	1
56	2
57	1
59	2
61	2
64	1
70	2
72	1
82	1
89	1
92	1
94	1
98	1
101	1
110	1
149	1
2228	1
19978	1
25392	1
25605	1

intervalle de κ	nbre comp
I_0	1701
I_1	197
I_2	273
I_3	88
I_4	273
I_5	78
I_6	19
I_7	124
I_8	3
I_9	21
I_{10}	0
$I11$	3557

TAB. A.1 – *Distribution du nombre de nœuds* N, *du nombre d'arcs* A *et du coefficient de connectivité* κ

par composante

Annexe B

Doublons

N	$\Xi(N)$	$\Delta(N)$	δ_1	δ_2
2	2954	0	0	0
3	1591	4	0.002	0
4	690	23	0.033	0
5	337	49	0.145	0
6	218	56	0.256	0
7	139	58	0.417	0
8	92	35	0.380	0
9	63	35	0.555	0
10	43	25	0.581	0.002
11	40	28	0.7	0.004
12	29	23	0.793	0.008
13	20	16	0.8	0.007
14	14	13	0.928	0.010
15	14	12	0.857	0
16	10	10	1	0.025
17	6	5	0.833	0.009
18	11	10	0.909	0.020
19	4	4	1	0.026
20	4	4	1	0.075
21	5	5	1	0.028
22	8	8	1	0.011
23	1	1	1	0.304
24	1	1	1	0.166
25	4	4	1	0.05
26	3	3	1	0.051
27	4	4	1	0.018
28	1	1	1	0.178
29	1	1	1	0.448
30	2	2	1	0.183
31	2	2	1	0.258
32	1	1	1	0.093
33	2	2	1	0.090
35	1	1	1	0.142
36	1	1	1	0.277
39	1	1	1	0.307
40	1	1	1	0.3
42	1	1	1	0.261
46	1	1	1	0.413
48	1	1	1	0.229
51	1	1	1	0.137
54	1	1	1	0.203
60	2	2	1	0.116
61	2	2	1	0.122
111	1	1	1	0.225
155	1	1	1	0.258
185	1	1	1	0.275
912	1	1	1	0.291
7007	1	1	1	0.453
9812	1	1	1	0.512
9997	1	1	1	0.441

TAB. B.1 – *Coefficient de doublage (δ_1) et coefficient de doublage raffiné (δ_2)*

Annexe C

Liste des c.m.s. corrigées

graphie	c.m.s. corrigée	graphie	c.m.s. corrigée
à compter de	prép.	en fonction de	prép.
à contresens	adv.	en forme	adv.
à côté	adv.	en pagaille	adv.
à défaut de	prép.	en panne	adv.
à faux	adv.	en pleurs	adv.
à l'écoute	adv.	en présence de	prép.
à l'étuvée	adv.	en trompette	adv.
à part	adv.	envoyer par le fond	v.
à plat	adv.	état d'esprit	n. m.
à proximité de	prép.	état de choses	n. m.
à proximité	avd.	être au fait	v.
aimer mieux	v.	être digne de	v.
au compte-gouttes	adv.	face à face	adv.
au cours de	prép.	faible d'esprit	adj.
au diable	adv.	faire défaut	v.
au fait	adv.	faire face	v.
aux crochets de	prép.	faire fonction de	v.
avoir foi en	v.	faire l'aveu de	v.
avoir froid	v.	faire son deuil de	v.
bien élevé	adj.	fête foraine	n. f.
bien foutu	adj.	frais et dispos	adj.
bilan de santé	n. m.	gens de maison	n. m. pl.
bonnes manières	n. f. Pl.	homme à	prép.
bras droit	n. m.	il convient de	prép.
brebis galeuse	n. f.	la boucler	v.
brouillard givrant	n. m.	maîtrise de soi	n. f.
comme il faut	adv.	mal choisi	adj.
commissaire de la République	n. m.	mal élevé	adj.
copie conforme	n. f.	mal foutu	adj.
crier sur	v.	marcher sur les brisées	v.
d'abord	adv.	mettre en branle	v.
d'élite	adj.	mise en demeure	v.
d'enfer	adj.	monter à la tête	v.
de côté	adv.	par accès	adv.
de fait	adv.	par définition	adv.
de fer	adj.	perdre contenance	v.
de service	adj.	porte de sortie	n. f.
de travers	adj.	porter atteinte à	v.
débit de boissons	n. m.	prendre corps	v.
débit de tabac	n. m.	prendre fin	v.
détective privé	n. m.	reconduire à la frontière	v.
devoir sur table	n. m.	sans délai	adv.
donner faim	v.	sans façon	adv.
emboîter le pas	v.	sans précédent	adv.
en abondance	adv.	sans trêve	adv.
en bref	adv.	sous le manteau	adv.
en ce qui concerne	adv.	sur le papier	adv.
en comparaison de	prép.	tout bêtement	adv.
en connaissance de cause	adv.	tout bien pesé	adv.
en conscience	adv.	trêve de	prép.
en ébullition	adv.	vieux garçon	adj.
en face	adv.	vieux jeu	adj.
en fait;adv.	y voir clair	v.	