# Heuristique pour la résolution d'anaphores dans les textes d'accidents de la route

Farid Nouioua
LIPN, UMR 7030 du CNRS
Institut Galilée, Université Paris 13, F-93430, Villetaneuse
e-mail : nouiouaf@lipn.univ-paris13.fr

## 1. Motivation

Dans cette communication, nous ne prétendons pas proposer une méthode originale pour la résolution des anaphores dans le cas général, mais nous décrivons plutôt un algorithme qui fait partie d'une chaîne complète de traitement automatique de textes décrivant des accidents de la route[1]. Nous voulons montrer d'une part la pertinence du problème de résolution d'anaphores pour des applications de compréhension automatique de textes réels et d'autre part que la restriction du domaine d'étude permet de tirer profit des spécificités de celui-ci pour développer des heuristiques très efficaces en exploitant au mieux les connaissances sémantiques du domaine. Certes, l'inconvénient de cette approche est son manque de généralité, mais cet inconvénient peut être pallié si plusieurs équipes travaillent sur différents domaines d'applications et développent leurs propres heuristiques. Un algorithme centralisé et assez léger pourrait ensuite déterminer les techniques les plus adaptées pour le traitement d'un nouveau texte et en recueillant les réponses de celles-ci, choisirait le meilleur résultat.

La résolution d'anaphores s'appuie globalement sur deux outils principaux : des contraintes qui sont des conditions obligatoires reliant l'anaphore à son antécédent (accord en genre et en nombre, les contraintes de c-commande, ou la compatibilité sémantique) ; et les préférences qui permettent d'établir un ordre entre différents antécédents possibles (le parallélisme syntaxique, le parallélisme sémantique, le centrage) [Mitkov, 02].

De nombreux algorithmes ont été proposés dans la littérature pour résoudre ce problème. Les premiers travaux étaient fondés sur une analyse syntaxique fine : on trouve par exemple l'algorithme de Hobbs [Hobbs, 78] qui utilise des contraintes syntaxiques pour le filtrage des candidats. L'algorithme de [Lappin & Leass, 94] opère sur le résultat du *"Slot grammar Parser"* de McCord et propose une méthode de calcul des saillances pour les antécédents potentiels. La difficulté d'obtenir une analyse syntaxique correcte et complète dans le cas général a conduit au développement de méthodes qui ne requièrent pas forcément une telle analyse. On trouve par exemple, l'utilisation de structures syntaxiques incomplètes ou ambiguës [Stuckardt, 01] ou le raffinement de techniques de calcul de la saillance en se basant sur un étiqueteur morpho-syntaxiques [Kennedy & Boguraev, 96] [Mitkov, 02].

Nous présentons, dans la section 2, la place du module de résolution d'anaphores dans l'architecture globale de notre application. Dans la section 3 nous discutons les principes de notre méthode de résolution d'anaphores ainsi que l'algorithme que nous proposons. Les résultats obtenus en appliquant notre heuristique sur un corpus de textes décrivant des accidents de la route sont présentés et discutés dans la section 4. Enfin la section 5 conclut et trace les perspectives de travaux futurs.

## 2. Résolution d'anaphore pour une compréhension automatique

Nous nous intéressons à la résolution d'anaphores comme sous-tâche d'un système complet fondé sur une approche de raisonnement non monotone. Le système répond automatiquement à la question de savoir la cause d'un accident à partir d'un texte qui le décrit [Nouioua & Kayser, 06], en concevant la cause comme une violation d'une norme du domaine. D'abord, une analyse syntaxique de surface est effectuée sur le texte en entrée et produit un certain nombre de relations syntaxiques de surface reliant

[1] Nous remercions la MAIF de Cergy-Pontoise qui nous donné accès aux textes constituant notre corpus.

les mots du texte[2]. Ces relations appelées "littéraux linguistiques bruts" subissent des post-traitements permettant de les adapter afin de produire "les littéraux linguistiques finaux". Ces derniers représentent les prémisses d'un processus de raisonnement non monotone dont le but est d'inférer la cause de l'accident exprimée dans un langage logique adéquat (pour plus de détail sur le système, voir [Nouioua, 07]). La résolution des anaphores est l'un des post-traitements que nous utilisons pour adapter la sortie de l'analyse syntaxique de surface au processus de raisonnement.

## 3. Principes de la méthode

Les principes sur lesquels se base notre méthode sont résumés dans les points suivants :

- Nous considérons uniquement la résolution d'anaphores pouvant désigner des agents actifs dans l'accident décrit. Il y a dans notre corpus, une métonymie très répandue qui assimile souvent des véhicules à des personnes et inversement (ex. j'ai heurté…, le véhicule a perdu le contrôle, …). En exploitant cette remarque, nous considérons touts les référents pouvant désigner des personnes ou des véhicules. Nous déterminons ainsi, a priori, la liste des anaphores à résoudre. Les anaphores peuvent être des groupes nominaux contenant plusieurs mots, nous choisissons dans ce cas le nom "central" du groupe comme référent. (p.ex. le nom "*chauffeur"* dans *"le chauffeur du véhicule"*).

- Nous cherchons à affecter à chaque anaphore une référence dans une liste prédéfinie. Cette liste est constituée des constantes : auteur, veh_A, veh_B, …, nom_de_personne1, …, nom_de_vehicule, …[3]. Ces constantes seront substituées aux arguments correspondants figurant dans les littéraux linguistiques bruts.

- Tout au long de l'algorithme, nous associons à chaque référent une information sémantique sur sa nature. Par défaut, il s'agit du trait *"Véhicule"*, mais cette information peut évoluer pour désigner une nature plus spécifique qui peut être dans la liste : *"Vélo", "Moto", "Voiture"* et "*Camion"*. On définie une matrice de compatibilité entre natures exprimant que la nature *"Véhicule"* est compatible avec toutes les autres natures mais que ces dernières sont deux à deux incompatibles. Cette matrice servira de filtre dans la recherche d'un antécédent parmi plusieurs et pour décider éventuellement de créer une nouvelle référence (voir l'algorithme plus loin).

- Avec ces conventions, un certain nombre d'anaphores peuvent être résolues immédiatement (*"je", "ma", "mon"*… désignent la référence *"auteur"*, *"A", "B"*, … désignent respectivement *"veh_A", "veh_B"*, …).

- La résolution d'une nouvelle anaphore consiste à choisir la référence de l'un des antécédents déjà résolus ou de décider d'introduire une nouvelle référence (un agent qui n'a pas été encore évoqué dans le texte) en présence de certains indices linguistiques, ou si l'ensemble des antécédents compatibles est vide. Nous ne considérons pas les cataphores qui sont très rares dans le corpus, ce qui nous a permis de proposer un algorithme itératif qui parcourt la liste des anaphores déjà traitées et, à chaque itération, résout la première anaphore non encore résolue. Nous avons de ce fait la garantie que l'algorithme se termine toujours.

- Nous exploitons les spécificités du domaine dont la métonymie évoquée ci-dessus entre personnes et véhicules pour effectuer deux sortes de propagation à travers nos relations syntaxiques de surface : une propagation des références permettant d'affecter une référence nouvellement calculée *R* pour une anaphore *X* à une autre anaphore *Y* non encore résolue si *X* et *Y* sont reliés par certaines relations[4]. Parallèlement à ce premier type de propagation, un deuxième type permet de

---

[2] les principales relations syntaxiques extraites du textes sont : verbe-sujet (sujet(X,Y)), verbe-objet (objet(X,Y)), préposition-nom-complément (compl_n(X,Y,Z)), préposition-verbe-complément (compl_v(X,Y,Z)), nom-qualification (qualif_n(X,Y)) et verbe-qualification (qualif_v(X,Y)) .

[3] Normalement, les protagoniste utilisent les symboles : A, B, ... pour désigner les différents véhicules, mais cette recommandation est rarement respectée. On trouve ainsi l'utilisation des pronoms *je* et *nous* et l'utilisation des noms des véhicules et des personnes. En cas ou un même référent peut être désignés par plus d'une référence à la fois, nous utilisons l'ordre (arbitraire) des références dans lequel elles sont évoquées dans ce paragraphe.

[4] Les relations pouvant véhiculer des informations à travers la propagation sont : qualif_n, compl_n et compl_v. Supposons, par exemple que nous disposons de la relation : compl_n(de, chauffeur, véhicule) (chaffeur de

mettre à jour la nature sémantique d'une anaphore si elle est associée à une référence ayant une nature plus spécifique, ce qui permet un meilleur filtrage sémantique dans la suite de l'algorithme.

## 4. Le pseudo-algorithme

```
Entrée : liste de référents
Sortie : liste de références associées
Debut
 Résoudre les cas triviaux
 Propager les références et les natures
 Pour (tout référent X non résolu) Faire
 Si (un indice d'ajout d'une nouvelle référence se présente) Alors
      Ref(X) = nouvelle référence
 Sinon
  construire la liste ANT des antécédents compatibles
  Si (card(ANT)=0) Alors Ref(X) = nouvelle référence FinSi
  Si (card(ANT)=1) Alors Ref(X) = le seul élément de ANT FinSi
  Si (card(ANT) > 1) Alors
       Ref(X) = la référence la plus proche avec
       moins de priorité à la référence auteur.       FinSi
 FinSi
 FinPour
Fin
```

Figure 1. **Pseudo-algorithme de résolution d'anaphores route**

L'algorithme est amorcé par une résolution des cas triviaux suivie d'une première propagation des références et des natures. Il parcourt ensuite la liste des anaphores non résolues et associe à chacune soit une nouvelle référence soit une référence déjà introduite. À la suite de la résolution de chaque anaphore, la procédure de propagation des références et des natures est à nouveau déclenchée. La décision d'ajouter une nouvelle référence est basée sur des indices tels que :

- Si le référent est un nom introduit par un déterminant indéfini (*"un", "une"*) p.ex. *"un véhicule"*, *"une voiture"*, il s'agit très vraisemblablement d'un nouvel élément dans la scène du texte.
- Pour un nom introduit par un déterminant défini (*"le", "la"*), on introduit une nouvelle référence :
  - si ce nom est suivi de *"qui"*, ou qu'il est le sujet d'un verbe au participe présent ;
  - s'il s'agit du premier référent introduit dans le texte ;
  - les noms comme *"adversaire"* et *"voisin"* conduisent souvent à de nouvelles références si on ne dispose que d'un seul antécédent. De même pour les noms associés à des adjectifs comme *"autre"* et *"premier"*.

En l'absence de ces indices, l'algorithme construit la liste des références de tous les antécédents compatibles. Si la liste est vide, il ajoute une nouvelle référence, si elle contient un seul élément, celui-ci est considéré comme étant la référence recherchée et enfin si la liste content plusieurs éléments, on applique une heuristique qui consiste à choisir celle qui est la plus proche de l'anaphore avec moins de priorité à la référence *auteur* (on considère qu'il y a peu de chance que l'auteur se désigne par un référent qui reste non encore résolu à ce stade)

## 5. Résultats et discussions

Nous avons travaillé sur un corpus de 160 textes d'accidents de la route. Nous avons utilisé 73 textes au cours du développement du système (corpus d'entraînement). Les 87 textes qui restent ont été utilisés pour la validation. Le tableau suivant résume les résultats obtenus :

---

véhicule) : si on résout le référent *chauffeur* en lui associant une référence R, celle-ci sera propagée au référent *véhicule*. Grâce à la métonymie entre personnes et véhicules les deux référents désignent le même objet.

| | Textes d'entraînement | Textes de validation |
|---|---|---|
| **Nombre d'anaphores** | 428 | 592 |
| **Nombre de références correctes** | 424 | 564 |
| **Pourcentage des références correctes** | **99 %** | **95 %** |

Tableau 1. **Résultats du test de l'heuristique de résolution d'anaphores**

Le système réussit à atteindre un pourcentage de succès très encourageant (95%) pour les nouveaux textes qui n'ont pas servi à son développement. Ceci montre clairement que l'heuristique que nous avons proposée fonctionne de façon satisfaisante. Nous soulignons que l'objectif que nous nous sommes fixé pour notre système rend indiscutable l'importance de la détermination des bonnes références pour assurer le succès du raisonnement : s'agissant de déterminer la cause d'un accident comme une violation d'une norme, en évoquant bien entendu l'agent qui a commis cette violation, il est évident qu'une partie importante de l'information pertinente que le système vise à inférer dépend de la détermination correcte des liens entre les anaphores et les agents correspondant dans la scène décrite. La limitation de l'espace nous empêche de détailler l'impact des 5% de référents non correctement résolues (dans les textes de validation) sur les étapes suivantes du système. Nous notons, par contre, que la plupart de ces erreurs ont "résisté" par la suite. Elles se manifestaient, dans les *"littéraux sémantiques"* (exprimant ce que le texte évoque explicitement) et dans les littéraux exprimant les causes recherchées, comme erreurs dans les paramètres attendus pour ces littéraux. Cependant, le nombre de ces erreurs étant limité, celles-ci n'ont pas dégradé significativement les performances du système qui a réussi à trouver les bonnes causes pour 80 % des nouveaux textes (et même 85% en prenant en compte les cas où cette cause a été trouvée, mais où d'autres causes moins pertinentes ont également été données par notre système).

## 6. Conclusion et perspectives

Ce papier décrit une heuristique de résolution d'anaphores dans le domaine restreint mais non trivial des accidents de la route. L'heuristique proposée tire profit des spécificités du domaine afin de proposer une mise en œuvre de la contrainte bien connue de compatibilité sémantique que nous utilisons comme moyen principal de détermination des références. Nous avons montré l'intérêt des connaissances sémantiques pour la tâche de résolution d'anaphores ainsi que la pertinence de cette tâche dans un système complet de compréhension automatique. Nous voulons, dans le futur, vérifier si de telles heuristiques restent possibles et efficaces sur d'autres domaines. Si c'est le cas, nous pouvons compter sur une diversification des domaines d'étude pour généraliser notre approche.

## Références

R. Mitkov, (2002). Anaphora resolution, Longman, Studies in Language and Linguistics.

TJ., Hobbs, (1978). Resolving pronoun references. Lingua, 44 : 311-338 .

S. Lappin & H.J. Leass, (1994). An algorithm for pronominal anaphora resolution. Computational Linguistics, 20(4) : 535-561.

R. Stuckardt, (2001). Design and Enhanced Evaluation of a Robust Anaphor Resolution Algorithm. Computational Linguistics, 27(4) : 479-506.

C. Kennedy & B. Boguraev, (1996). Anaphora for everyone : pronominal anaphora resolution without a parser. Proceedings of the 16th International Conference on Computational Linguistics (COLING'96), pp. 113-118.

F. Nouioua, (2007). Extraction et utilisation des normes pour un raisonnement causal dans un corpus textuel. Thèse de doctorat de l'université Paris 13, Avril.

F. Nouioua & D. Kayser, (2006). Une expérience de sémantique inférentielle. Actes de la conférence : Traitement automatique de la langue naturelle (TALN'06), pp. 246-255.