

## Extraction automatique d'informations à partir de micro-textes non structurés

Cédric Vidrequin<sup>1</sup>, Juan-Manuel Torres-Moreno<sup>1</sup>,  
Jean-Jacques Schneider<sup>2</sup>, Marc El-Bèze<sup>1</sup>

(1) Laboratoire Informatique d'Avignon, Agroparc  
BP1228, 84911 Avignon CEDEX 9, France

[{cedric.vidrequin, marc.elbeze, juan-manuel.torres} @univ-avignon.fr](mailto:{cedric.vidrequin, marc.elbeze, juan-manuel.torres}@univ-avignon.fr)

(2) Société SEMANTIA, Parc d'activité de Gémenos,  
30 avenue du château de Jouques, 13420 Gémenos, France  
[jjschneider@semantia.com](mailto:jjschneider@semantia.com)

**Résumé** Nous présentons dans cet article une méthode d'extraction automatique d'informations sur des textes de très petite taille, faiblement structurés. Nous travaillons sur des textes dont la rédaction n'est pas normalisée, avec très peu de mots pour caractériser chaque information. Les textes ne contiennent pas ou très peu de phrases. Il s'agit le plus souvent de morceaux de phrases ou d'expressions composées de quelques mots. Nous comparons plusieurs méthodes d'extraction, dont certaines sont entièrement automatiques. D'autres utilisent en partie une connaissance du domaine que nous voulons réduite au minimum, de façon à minimiser le travail manuel en amont. Enfin, nous présentons nos résultats qui dépassent ce dont il est fait état dans la littérature, avec une précision équivalente et un rappel supérieur.

**Abstract** In this article, we present a method of automatic extraction of informations on very small-sized and weakly structured texts. We work on texts whose drafting is not normalised, with very few words to characterize each information. Texts does not contain sentences, or only few. There are mostly about fragments of sentences or about expressions of some words. We compare several extracting methods, some completely automatic and others using an small domain knowledge. We want this knowledge to be minimalistic to reduce as much as possible any manual work. Then, we present our results, witch are better than those published in the literature, with an equivalent precision and a greater recall.

**Mots-clés** extraction automatique, micro-texte, texte non structuré, petites annonces.

**Keywords** automatique extraction, micro-text, unstructured text, adds.

## 1 Introduction

L'extraction automatique d'informations sur des textes de petite taille peut se révéler relativement simple ou très compliquée, selon les caractéristiques des documents traités et le type d'informations cherchées. Le domaine des petites annonces est souvent utilisé pour expérimenter ce type d'extraction automatique car on peut travailler sur des petits, des très petits voire des micro-textes. De plus, on y rencontre à la fois des données spécifiques à chaque annonce, et des données communes liées au domaine. Deux approches sont utilisées pour rédiger une petite annonce : la saisie de texte libre ou la saisie par formulaire. La première permet au vendeur de décrire son bien aussi précisément qu'il le désire. La seconde accorde plus de facilité à l'acheteur lors de sa recherche, en proposant de filtrer les annonces par critères (marque, prix, garantie ...). Si la saisie par formulaire prévoit assez souvent une zone de description libre, celle-ci n'est, en général, pas traitée lors de la recherche par critère. On souhaiterait donc concilier la liberté apportée par la saisie en texte libre, et la simplicité de la recherche par critères. C'est dans cette optique que nous développons une méthode d'extraction automatique d'informations, à partir de petites annonces. Les méthodes employées dans la littérature sont en général satisfaisantes, mais ne font état que de l'extraction d'une partie des informations. C'est pourquoi nous souhaitons développer un outil capable de détecter le plus de critères possibles, avec un taux d'erreur minimal.

En section 2, nous recensons les méthodes existantes dont nous nous sommes inspirés. Dans la section 3, est décrite la méthode que nous proposons. Enfin, nous rapportons, en section 4, les résultats que nous obtenons avec notre système. Nous concluons en section 5 et donnons les perspectives que nous envisageons l'avenir.

## 2 Méthodes pour l'extraction d'informations dans des petites annonces

Une des approches utilisées pour extraire automatiquement des informations à partir de petites annonces est de réaliser un adaptateur (*wrapper*). Celui-ci parcourt le document et en extrait une *carte hiérarchique* des données. Cette méthode a besoin de documents structurés ou semi structurés, comme des documents HTML, et implique le plus souvent la rédaction manuelle de règles d'extraction pour chaque critère. (Gao et Sterling, 1999) extraient des informations à partir de sites web hétérogènes en suivant deux étapes successives. Ils mettent tout d'abord en évidence les critères caractérisant chaque objet de l'annonce, puis ils regroupent les connaissances en concepts hiérarchiques. La première étape est proche du problème qui nous intéresse : les critères recherchés sont par exemple le prix, la superficie ou le type d'un bien immobilier. Ceux-ci sont mis en évidence grâce à des fonctions d'extraction adaptées, réalisées de façon manuelle, en utilisant des listes de termes et des expressions régulières. Les listes peuvent caractériser par exemple les marques et les modèles de véhicule automobile. Les expressions régulières, quant à elles, peuvent modéliser des informations comme le prix ou la surface d'une habitation. La détection des critères se fait à l'aide d'une liste de prédicats de priorité entre concepts et balisage HTML.

De la même façon, (Seo *et al.*, 2001) mettent en évidence des paires [étiquette, valeur] en utilisant des connaissances du domaine de l'immobilier. Ces connaissances, modélisées dans un fichier XML, représentent les différentes façon de formuler chaque critère : à un même niveau de l'arbre XML se trouvent les différentes formulations pour la valeur du critère concerné. Cette représentation constitue une alternative à l'utilisation d'expressions régulières. Les paires détectées peuvent avoir différentes formes : \$395 000 ou 5 BR ou encore 2 BA. Ces

paires correspondent au critère *prix* qui a pour valeur 395 000, au critère *nombre de chambres* qui a pour valeur 5, ou au critère *nombre de salles de bain* qui a pour valeur 2.

Notre façon d'aborder la problématique est différente de l'approche des auteurs. Ceux-ci s'attachent plus à réaliser une segmentation des pages Web en annonces, qu'à un découpage d'annonces. D'autre part, les critères extraits sont très limités, et ne représentent pas l'ensemble de l'annonce. Ce nombre n'est d'ailleurs pas donné, mais les exemples illustrant leurs résultats en comportent moins de dix pour chaque type d'annonce. L'évaluation de leur performance porte uniquement sur les critères qu'ils visent à détecter.

(Embley *et al.*, 1998) réalisent une extraction semblable à celle de (Seo *et al.*, 2001), sur des textes non structurés, en s'appuyant sur une ontologie. Cette ontologie est obtenue à partir d'un modèle sémantique de données et a pour but de décrire la vue désirée du domaine retenu. Les auteurs appliquent un parseur, un outil de reconnaissance de constantes et mots clés, puis un générateur de texte structuré sur le texte non structuré. Des couples [clé, valeur] sont générés, en respectant les contraintes et les règles de l'ontologie. Cette méthode n'a pas pour prétention de fonctionner sur tout type de documents, mais vise les documents riches en données constantes - dates, noms, identifiants, quantités - ou dont le domaine d'application est décrit par une partie d'ontologie restreinte. C'est à partir de l'ontologie que sont générées les expressions régulières qui sont utilisées par l'outil de reconnaissance de paires [clé, valeur], pour extraire les critères. Le moteur de génération de texte structuré est ensuite appliqué, suivant une liste d'heuristiques prédéfinies. (Embley *et al.*, 1998) évaluent leur système sur un ensemble de 216 annonces automobiles et 100 annonces d'offres d'emploi. Tout comme (Gao et Sterling, 1999), ils n'extraient pas la totalité des critères des annonces. Cette fois, la limitation vient des contraintes imposées par l'ontologie. Par exemple, le numéro de téléphone des annonces automobiles n'accepte qu'une seule valeur, même si l'annonce en comporte plusieurs. Les calculs de rappel et de précision sont effectués sur la base d'une réponse unique et non sur l'ensemble des informations contenues dans l'annonce, et seuls les critères définis par l'ontologie sont pris en compte. Nous nous différencions des auteurs sur ce point, comme nous le verrons en section 4.

(Peleato *et al.*, 2000) étiquettent des données à partir de lexiques, d'expressions régulières et d'analyses de position des mots. Les étiquettes correspondent au nom des critères, les données à leur valeur. Les expressions et noms sont tout d'abord détectés grâce à des lexiques, établis à partir d'une étude de fréquence d'apparition des mots, dans un corpus de 10 000 annonces. On rencontre dans ces listes des termes tels "*camion*", "*airbag*", ou encore "*libre de suite*". Des listes publiques, obtenues sur le Web, sont utilisées pour détecter certaines informations comme les noms de villes ou de pays, ainsi que les abréviations. Un jeu d'expressions régulières est ensuite appliqué pour les informations telles que les dates, les prix ou les numéros de téléphone. Ce jeu d'expressions a été enrichi manuellement à partir de tests basés sur un corpus d'entraînement. Un second processus a pour but d'identifier la nature des informations restantes, à partir de l'annonce partiellement étiquetée. Cette dernière est segmentée suivant la ponctuation, et des prépositions dans le cas d'annonces d'offre d'emploi. Les mots mis en évidence sont comparés à des listes de mots clés décrivant chaque critère. Ces listes ont été constituées manuellement, suite à l'étude précédemment citée. La différence entre ces listes et les lexiques est que ces listes ne définissent pas une information précise, mais plutôt la catégorie d'information concernée. Par exemple, "*climatisation*" ou "*alarme*" appartiennent à la catégorie "*options*" ; "*spacieux*" ou "*charmant*" appartiennent à la catégorie "*qualité*". L'étiquetage du segment se fait en fonction de la liste à laquelle correspondent le plus de mots du segment. Les positions relatives des mots sont utilisées pour étiqueter les segments qui n'ont pu l'être jusque-là. Ainsi, un segment est rattaché à celui qui le précède directement, suivant un jeu prédéfini de règles. Par exemple, un mot non étiqueté suivant directement la marque d'un véhicule est étiqueté comme modèle. L'étude précédente montre

en effet que le modèle d'un véhicule suit le plus souvent la marque. Les segments restants sont étiquetés *indéfinis*. Le système est évalué sur 77 annonces dont 6 portent sur l'automobile, 30 sur les offres d'emploi et 41 sur l'immobilier. Ces annonces sont annotées manuellement par plusieurs personnes, les résultats donnant lieu à une mesure de kappa (1).

$$-1 \leq K = \frac{P(A) - P(E)}{1 - P(E)} \leq 1 \quad (1)$$

$P(A)$  est la proportion de correcteurs proposant la même étiquette, et  $P(E)$  la proportion d'accord aléatoire (cas où l'on met les étiquettes au hasard). Les auteurs obtiennent un kappa  $k=0,9$ , ce qui est au dessus du taux  $k=0,8$ , considéré comme très acceptable. Une fois de plus, tous les critères des annonces ne sont pas pris en compte lors de l'évaluation. Lorsque les correcteurs ne sont pas d'accord, le critère est considéré comme neutre et ne fait pas partie de l'évaluation. Si un critère n'est relevé ni par les correcteurs, ni par le système, il est également ignoré. Sur l'ensemble de test, 519 critères sont pris en compte sur 1 415 critères. Pour les critères évalués, si les correcteurs et le système s'accordent sur la détection du critère, celui-ci est compté comme correct, sinon il est compté comme incorrect. Seule la précision est évaluée, en comparant manuellement les réponses des correcteurs et celles du système.

### 3 Extraction automatique par le contenu

Nous parlons ici d'extraction plutôt que de repérage, car cette dernière notion laisse sous-entendre un chevauchement possible entre les critères. La société SEMANTIA nous fournit un corpus de 83 749 petites annonces, longues en moyenne de 35 termes, et dont voici un exemple : "206 1.6 16V ROLLAND GARROS 5 Portes VERT FONCE 24692 Km 2004 Essence Alarme Garantie 12 mois 12900€". Notre objectif est d'utiliser le moins possible de connaissances du domaine. Nous séparons pour cela les informations en informations à valeur variable et informations à caractère booléen.

#### 3.1 Les critères à valeur variable

Le premier type de critères correspond à ceux dont la valeur peut fortement varier d'une annonce à l'autre, et pour lesquels on ne peut pas opérer de détection automatique efficace. En effet, on peut considérer que la valeur de ces critères est presque unique pour chaque annonce. C'est le cas pour des informations comme un prix, une date ou une marque. Nous utilisons donc notre connaissance du domaine pour extraire ces critères, ce qui nécessite un certain travail manuel, en amont de l'extraction. Tout comme (Gao et Sterling, 1999) ou (Peleato *et al.*, 2000), nous utilisons des listes pour mettre en évidence une partie des informations. En revanche, notre objectif étant d'avoir la plus petite contribution manuelle en amorce, nous en limitons fortement l'utilisation. Nous n'utilisons donc qu'une liste fermée de valeurs et une liste d'expressions régulières. La première est obtenue sur Internet<sup>1</sup>, et contient uniquement les marques et les modèles des véhicules. Pour les autres informations, nous fabriquons manuellement une courte liste d'expressions régulières simples, modélisant chacune des informations recherchées. Par exemple, le nombre de portes est extrait par l'expression suivante : " $[1-7][ ]*[Pp](or/OR)?(te[s]?/TE[S]?)$ ". Contrairement à (Embley *et al.*, 1998) qui utilisent 165 expressions régulières pour l'automobile, notre liste contient une 15<sup>ème</sup> d'expressions régulières. Ces dernières permettent d'extraire des critères comme le prix du véhicule, le nombre de portes, le millésime ou encore le kilométrage. À la différence de l'étude de fréquence de (Peleato *et al.*, 2000), dont la mise en place a duré 45 jours, notre amorce manuelle peut être réalisée en moins d'une journée, et peut être, si besoin, rapidement adaptée en fonction des résultats obtenus.

<sup>1</sup> [http://fr.wikipedia.org/wiki/Constructeur\\_automobile](http://fr.wikipedia.org/wiki/Constructeur_automobile) et liens sous-jacents

Un seul groupe d'informations s'est révélé à l'usage trop compliqué à extraire : il s'agit des informations *puissance-motorisation-cylindrée*. En effet, dans la plupart des annonces, on constate que ces critères ne sont pas nécessairement regroupés et ordonnés. Les informations peuvent être portées différemment d'une annonce l'autre, par un ou plusieurs mots. Certaines informations peuvent être absentes et les valeurs de ces critères sont suffisamment variables pour empêcher une détection automatique efficace. Les combinaisons de ces informations sont nombreuses, et si elles sont portées par un mot unique, il est alors peu probable de pouvoir les mettre en évidence de façon totalement automatique, tout en garantissant une extraction suffisamment fine. Pour ces raisons, le traitement de ces informations fait l'objet d'une méthode d'extraction automatique spécifique, comme l'ont fait (Gao et Sterling, 1999). Cette méthode combine un jeu de quatre expressions régulières : la première, la plus générale, permet d'extraire le groupe de critères de l'annonce, comme par exemple "*1.5DCI 70CV*". Les trois autres, plus spécialisées extraient chacun des trois critères individuellement : la puissance (*70CV*), la motorisation (*DCI*) et la cylindrée (*1.5*).

## **3.2 Les critères à caractère booléen**

Une fois les critères à valeur variable extraits, nous considérons que les informations restantes font partie du groupe des critères à caractère booléen. Pour les extraire automatiquement, nous tirons parti de la taille de notre corpus. Les sections suivantes décrivent deux méthodes d'extraction qui ne nécessitent pas ou très peu d'apport manuel.

### **3.2.1 Découpage suivant la ponctuation**

Nous définissons ici une liste fermée de séparateurs [,-./;] suivant lesquels on découpe tour à tour les annonces. On obtient ainsi plusieurs découpages possibles différents. Les critères à valeur variable déjà extraits sont remplacés par le séparateur courant, ce qui donne un pré-découpage fiable. Le texte restant est découpé en fonction du même séparateur, afin de mettre en évidence les critères potentiels. On trie les critères des différents découpages, de façon décroissante, et en fonction de leur nombre d'occurrences sur l'ensemble des annonces. De la liste obtenue, on élimine les critères automatiquement détectés qui sont trop courts ou trop longs. Nous gardons arbitrairement les critères longs de plus de trois lettres et comprenant au maximum sept mots. Ces valeurs ont été définies en observant le comportement du système durant la phase de développement. On élimine également les critères qui contiennent plusieurs fois le même séparateur, en partant du principe qu'il s'agit potentiellement d'un mauvais découpage. En effet, certaines annonces utilisent un type de séparateur, puis en changent en cours d'annonce. L'extraction automatique des critères se fait sur base de la liste ainsi mise en évidence.

Pour chaque annonce, on parcourt la liste des critères automatiquement mis en évidence. Si le critère est présent, on l'extrait de l'annonce et on passe au suivant, sinon on passe directement au critère suivant. Le traitement de l'annonce est terminé une fois que la liste des critères est entièrement parcourue. Si cette méthode a l'avantage de donner de très bons résultats en terme de précision, elle a aussi des inconvénients. Elle est limitée en rappel dans la mesure où elle ne cherche pas à extraire tous les critères des annonces, mais seulement ceux que la méthode a retenus. Ensuite, certains critères sont impossibles à extraire comme c'est le cas des critères contenant plusieurs ponctuations<sup>2</sup>. Enfin, les critères de plus de sept mots, comme par exemple "*livraison partout en France comprise dans le prix*", ne peuvent être extraits.

<sup>2</sup>

"*Banquette arrière 2/3 - 1/3*" ou "*retroviseurs électr. - dégivr. &ab*"

### 3.2.2 Découpage avec les collocations

Nous désirons à présent augmenter le rappel en essayant d'extraire l'ensemble des critères de chaque annonce. Nous souhaitons également utiliser une méthode qui soit moins liée aux caractères de ponctuation, puisque ces derniers peuvent se trouver dans l'expression de certains critères. Afin de mettre en évidence des groupes de mots ayant un sens particulier lorsqu'ils se suivent, nous avons développé une méthode d'extraction s'appuyant sur les collocations et utilisant un filtrage basé sur les ratios de vraisemblance.

Une collocation (Manning et Schütze, 1999) est une tournure de phrase pour laquelle le tout est perçu pour avoir un sens au-delà de la somme des parties. Les collocations comprennent les groupes nominaux, les groupes verbaux ou encore les locutions. Enfin, tout type de groupe de mots fréquemment répété est candidat au statut de collocation. Les collocations ont la double particularité que les termes qui la composent apparaissent fréquemment ensemble, tout en pouvant avoir une existence indépendante.

Les ratios de vraisemblance sont une approche par test d'hypothèse. Ils sont appropriés pour des données éparées et sont faciles à interpréter, grâce à un nombre qui exprime la vraisemblance entre les hypothèses testées. Les hypothèses  $H_0$  et  $H_1$  considèrent respectivement l'indépendance et la dépendance entre deux mots. Le ratio de vraisemblance correspond à la probabilité de réunir les deux conditions de l'hypothèse. Pour comparer les collocations potentielles, le logarithme de vraisemblance est utilisé. Celui-ci a l'avantage de mettre en évidence les collocations dont les occurrences sont rares. La variable aléatoire utilisée est asymptotique<sup>3</sup> à une distribution de type  $\chi^2$  et peut être utilisée pour réaliser des tests d'hypothèse. Ainsi, en utilisant les tables de valeurs de référence d'une  $\chi^2$ , on peut valider ou refuser une hypothèse, en associant un score de confiance à la décision prise.

La mise en évidence des critères se fait de façon itérative en utilisant le corpus d'annonces. Lors de la première itération, des couples de termes sont construits à partir de mots, tandis que pour les itérations suivantes, les termes peuvent être encore des mots ou des groupements de mots. Nous recensons le nombre d'occurrences de chaque couple de termes, ainsi que celui des termes pris séparément. À partir de ces informations, nous calculons le ratio de vraisemblance des couples de termes et nous les classons par ordre décroissant. Le corpus étant suffisamment grand, nous approximations la variable aléatoire qui définit le ratio de vraisemblance par une distribution  $\chi^2$ . Puisque nous travaillons à chaque itération sur des couples de termes, le degré de liberté est de un. Nous décidons de tolérer un taux d'erreur de 0,001, ce qui nous amène à supprimer tous les couples dont le score est inférieur<sup>4</sup> à 10,83. Nous supprimons les couples pour lesquels le premier terme se termine par une ponctuation, ou pour lesquels le second terme commence par une ponctuation<sup>5</sup>, modélisant par là un mauvais découpage potentiel de l'annonce. Nous avons expérimenté différents ensembles de ponctuations concernées, y compris l'ensemble vide (section 4.2). Les couples conservés dans la liste sont regroupés en une seule entité et représentent les critères potentiels.

Des critères pouvant se chevaucher, il est important de réaliser les choix qui favorisent le découpage optimal. L'annonce "(...)10000 km hifi système alarme garantie or ESP (...)" présente une ambiguïté forte pouvant amener aux découpages suivants : [10000 km, hifi, **système alarme, garantie or, ESP**] ou [10000 km, **hifi système, alarme garantie, or, ESP**]. Nous développons donc une méthode inspirée de l'algorithme de Viterbi (Viterbi 1967) projeté sur une seule dimension. Parmi les découpages possibles, cette méthode cherche un maximum global de la somme des ratios de vraisemblance correspondant aux critères

<sup>3</sup> Quand le corpus est suffisamment grand.

<sup>4</sup> Tirée des tables des distributions  $\chi^2$

<sup>5</sup> Exemple : "(...) *Immobiliseur, Vitres électriques, Volant multifonctions. Garantie. (...)*"

découpés. Soient  $n$  le nombre de mots du plus long critère,  $m$  le nombre de mots de l'annonce traitée,  $S$  un tableau de  $m$  cases qui fait correspondre les mots de la phrase et le score du meilleur découpage de l'annonce jusqu'à ce mot, et  $I$  un tableau de  $m$  cases qui fait correspondre à chaque mot l'indice du début du critère correspondant au meilleur découpage. Nous faisons glisser une fenêtre de mots sur le texte de l'annonce, en commençant par le premier mot et en allant vers le dernier. À chaque décalage, nous prenons une fenêtre de  $n$  mots que nous diminuons progressivement d'un mot, jusqu'à arriver au singleton. Pour chaque groupe de mots de la fenêtre, si le  $n$ -gramme est présent dans la liste de critères proposés :

- dans  $S$  : on calcule le score de la case correspondant au dernier mot de la fenêtre :
  - on normalise le ratio de vraisemblance du critère en le multipliant par le nombre de mots du critère. Les ratios ont des valeurs positives, supérieure à 1. On favorise ainsi la présence de critères longs, plus difficiles à obtenir que des critères courts ;
  - on ajoute à ce score celui de la case précédant le premier mot du  $n$ -gramme (et correspondant donc au dernier mot du  $n$ -gramme précédent). Cela permet de cumuler les scores des critères en fonction du découpage ;
- dans  $I$  : on enregistre l'indice correspondant à la première case de la fenêtre dans la case correspondant au dernier mot de la fenêtre. Cela nous permet de remonter par la suite du dernier  $n$ -gramme vers le premier.

Un  $n$ -gramme peut ne pas apparaître dans la liste des critères, parce qu'il a été filtré, ou parce qu'il s'agit d'un mot seul. Nous n'attribuons alors pas de poids supplémentaire au  $n$ -gramme, et propageons simplement celui du découpage précédent. Dans le premier cas, cela permet de conserver la possibilité de découper ce critère suivant le  $n$ -gramme, si on ne trouve pas de meilleur découpage en réduisant ou/et en décalant la fenêtre de mots. Dans le second cas, la notion de collocation n'a pas vraiment de sens pour un mot seul, et nous n'avons pas à disposition de probabilité associée à cet événement. Malgré cela, la propagation du score sans ajout de poids permet de considérer le mot seul comme un critère potentiel.

Une fois toutes les fenêtres évaluées, nous prenons successivement les maxima de  $S$  pour extraire le meilleur  $n$ -gramme courant. Une fois un  $n$ -gramme extrait, nous supprimons tous les scores des mots lui appartenant. Nous réalisons le découpage du dernier critère trouvé vers le premier. Les annonces découpées servent de base à l'itération suivante, où les termes ne sont plus des mots (ou des  $n$ -grammes) mais des bi-grammes (ou des  $n+1$  grammes).

## 4 Expériences

L'évaluation de la performance du système nécessite une référence qui doit être construite manuellement. Ce processus étant long et fastidieux, seules 200 petites annonces prises au hasard sont actuellement vérifiées<sup>6</sup>. Cette évaluation comprend deux étapes, l'une portant sur la méthode d'extraction par découpage en fonction des ponctuations, l'autre sur la méthode des collocations. Nous utilisons les annonces pour lesquelles nous avons relevé manuellement l'ensemble des critères de chaque annonce. Nous comptabilisons ceux correctement détectés, ceux mal détectés et ceux qui ne sont pas détectés. Contrairement à (Peleato *et al.*, 2000), nous évaluons tous les critères de chaque annonce. Ainsi, nos mesures de précisions et de rappel sont calculées sur un éventail plus large, et sont donc soumises à de plus fortes contraintes. Durant cette phase d'évaluation, nous séparons l'évaluation des critères à valeur variable de celle des critères à caractère booléen. Nous conjuguons, dans un second temps, ces informations dans une moyenne globale. Cela nous permet de nous situer par rapport aux résultats de la littérature. Les méthodes utilisent des principes similaires, mais les données des

<sup>6</sup> Nous en préparons actuellement 200 supplémentaires.

Les systèmes sont différents, le nombre ainsi que le type de critères détectés ne sont pas les mêmes. (Peleato *et al.*, 2000) obtiennent 73% de précision en moyenne sur les trois domaines testés<sup>7</sup>. Ils obtiennent leur meilleure précision pour les annonces automobile avec une valeur de 88%. Souvenons nous qu'ici, le rappel n'est pas évalué. (Seo *et al.*, 2001) comparent les mots qu'ils ont extraits aux concepts auxquels ils correspondent. Ils obtiennent 100% de précision et 80% de rappel, sur les 4 critères qu'ils extraient. (Gao et Sterling, 1999) évaluent seulement les critères sur lesquels ils travaillent. Ils présentent des résultats allant de 90 à 99% pour la précision, et de 76 à 99% pour le rappel, suivant les pages web testées. Rappelons tout de même que leurs expériences sont basées sur des pages semi-structurées, ce qui n'est pas notre cas. (Embley *et al.*, 1998) obtiennent les meilleurs résultats en précision-rappel avec des scores respectif de 99% et 94%. Cependant, dans le cas de valeurs multiples pour un critère, une seule valeur est prise en compte.

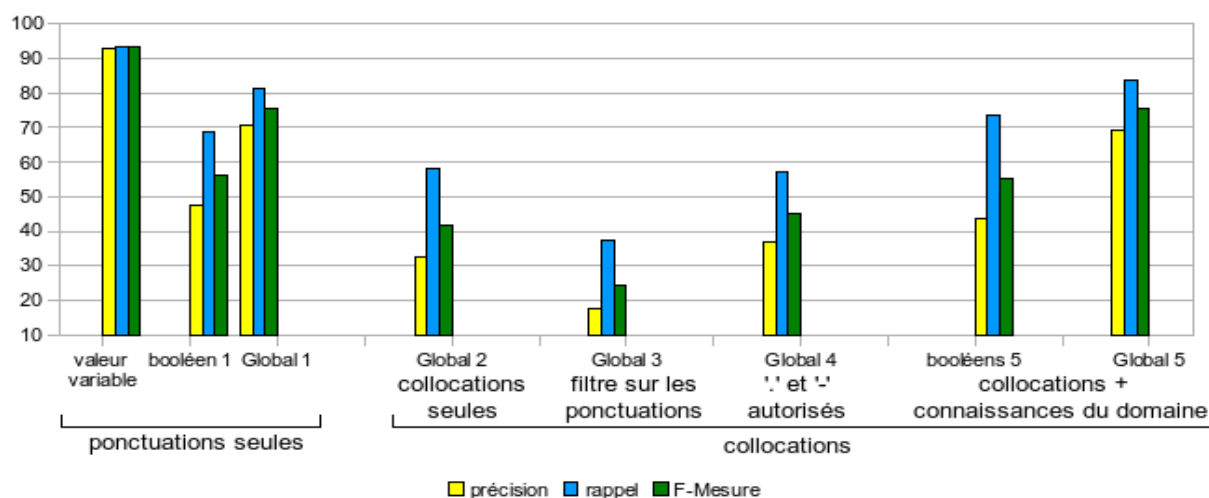


Figure 1 : Résultats des méthodes proposées

#### 4.1 Evaluation du découpage suivant les caractères de ponctuation

Notre système extrait une quinzaine de critères à valeur variable. De plus, il tient compte des valeurs multiples pour un critère. Ainsi il est pénalisé s'il manque une valeur pouvant être extraite par un correcteur humain. Il est, de ce point de vue, jugé plus sévèrement que ses concurrents. Enfin, notre système n'utilise que très peu de connaissances du domaine, comparativement aux autres systèmes. Malgré ces contraintes, nos performances se situent dans une bonne moyenne avec une F-mesure de 0,93 (Fig. 1 - *valeur variable*).

Mais cette évaluation n'est que partielle : elle ne prend pas en compte l'ensemble des critères des annonces. Nous la réalisons uniquement dans le but de nous positionner par rapport aux travaux déjà mis en œuvre dans le domaine. Pour les critères automatiquement détectés (*booléens 1*), nous obtenons une F-mesure de 0,56. Les performances de notre système restent donc acceptables, avec une F-mesure globale de 0,75, soient 70,6% de précision et 81,3% de rappel (*Global 1*). Cette évaluation prend en compte les critères comportant des erreurs récurrentes, des fautes d'orthographe ou des abréviations.

#### 4.2 Evaluation du découpage basé sur les collocations

La première expérience que nous avons réalisée a pour but d'étudier l'impact de l'utilisation des méthodes numériques seules. Nous n'utilisons donc que les critères mis en évidence par

<sup>7</sup>

Automobile, immobilier et offres d'emploi



les collocations, et nous ne réalisons pas de filtrage sur les caractères de ponctuation (*Global 2*). N'utilisant pas de connaissances du domaine, nous perdons beaucoup en performance et obtenons seulement 0,42 lors du calcul de la F-mesure. De nouveaux critères, impossibles à détecter avec la méthode précédente, sont à présent extraits. C'est notamment le cas de "*Banquette arrière 2/3 - 1/3*". En revanche, malgré une grande variabilité de l'ordre des critères, certains d'entre eux apparaissent suffisamment côte à côte pour être regroupés. C'est le cas par exemple de "*vitres électriques vitres teintées*".

Si l'on filtre les critères par rapport à la ponctuation (voir 3.2.2), la F-mesure chute alors à 0,24 (*Global 3*). De nombreux mots sont en effet regroupés de façon injustifiée du point de vue du sens. C'est le cas par exemple des mots "€" et "Tél." qui sont rapprochés dans le même bi-gramme. En effet, nous empêchons la constitution d'un  $n$ -gramme dont l'un des mots est suivi par un caractère de ponctuation. On ne peut donc pas regrouper le mot "Tél." et le numéro de téléphone qui lui correspond. Ensuite, le numéro de téléphone du contact suit très fréquemment le prix de l'objet de la vente. Le regroupement de ces deux informations est donc facilité. De plus, le numéro de téléphone est souvent découpé en cinq mots, un espace séparant chacun des chiffres. Il est donc plus difficile de recomposer les 6-grammes que les bi-grammes (lorsque les chiffres sont, par exemple, collés les uns aux autres). En permettant l'apparition d'un point ou d'un tiret entre deux mots d'un critère, on améliore légèrement les résultats et l'on obtient 0,45 de F-mesure (*Global 4*). Si on se réfère à la première méthode, l'une des raisons de la perte de performances, est liée au regroupement de critères proches du point de vue de l'information qu'ils portent. C'est notamment le cas de la marque et du modèle. Ces deux informations étant très souvent regroupées au niveau de l'annonce, la méthode des collocations en fait presque systématiquement un  $n$ -gramme. Le découpage n'est donc pas aussi fin que ce que l'on attend. On retrouve le même genre de problème avec des informations beaucoup plus complexes à extraire comme la cylindrée, la motorisation et la puissance. Là encore, ces informations sont le plus souvent collées les unes aux autres, et donc impossibles à extraire automatiquement.

Nous constatons donc que nous ne pouvons nous passer totalement de la connaissance du domaine pour assurer une extraction satisfaisante. C'est pourquoi nous testons l'extraction des collocations, après avoir extrait au préalable les critères à valeur variable. Nous obtenons ainsi des performances légèrement supérieures à la première méthode avec une F-mesure de 0,76 (*Global 5*). Les résultats de l'extraction des critères à valeur variable sont évidemment inchangés. Pour les critères à caractère booléen, nous perdons un peu en précision (43,9% contre 47,5% pour la première méthode) mais nous gagnons en rappel (73,8% contre 68,7%). Ces résultats s'expliquent par le fait que l'on essaye ici d'extraire tous les critères et non pas seulement ceux qui ont été mis en évidence automatiquement. On augmente donc le rappel, mais puisque l'on *prend plus de risques*, on diminue forcément la précision. On note également une amélioration de la qualité de certains critères que l'on extrayait de façon automatique. Ainsi les critères complexes comme "*coussins gonflables (4 & plus)*" ou "*garanti garantie or 12 mois*" sont correctement extraits, ce qui n'était pas le cas auparavant.

## **5 Conclusion**

Nous avons vu dans cet article que les méthodes que nous proposons donnent des résultats comparables à ce que l'on rencontre dans la littérature, mais en utilisant une connaissance du domaine réduite tout en détectant plus de critères. De plus, une partie des critères est extraite sans connaissances du domaine et notre méthode continue à donner des résultats acceptables. À présent, nos objectifs sont d'améliorer nos résultats en testant l'impact de l'utilisation d'un outil de racinisation. Nous générerons ainsi des critères plus généraux, et nous espérons

rapprocher des critères liés. Pour extraire les critères des petites annonces, nous utiliserons cet outil de racinisation pour faire correspondre  $n$ -grammes et critères. D'autre part, nous prévoyons de mettre à l'épreuve notre méthode en testant celle-ci sur des petites annonces écrites dans une autre langue, ou traitant d'un autre domaine. Nous comptons appliquer notre algorithme sur des petites annonces automobile en anglais, puis sur des petites annonces immobilières en français. Ces expériences nous permettront en outre d'évaluer le coût réel de l'adaptation des connaissances du domaine. Enfin, pour résoudre les problèmes de mots mal découpés à cause des ponctuations ou pour des mots abrégés de façon particulière (ex: "dégivr.", "airbags front. et lat." ), nous réfléchissons à l'utilisation d'un algorithme de réduction de bruit. En considérant les annonces comme un ensemble de données bruitées, ou contenant des erreurs, le but serait de détecter ces erreurs et de pouvoir les corriger automatiquement, à condition, bien sûr, que les formes correctes soient prédominantes sur les formes incorrectes.

**Remerciement** Ces travaux sont co-financés par l'ANRT - CIFRE n° 777/2004.

## Références

Y. CHOUEKA, T. KLEIN, E. NEUWITZ. Automatic retrieval of frequent idiomatic and collocational expressions in a large corpus. *Journal for literary and linguistic computing*, 4:34-38.

GAO X., STERLING L. (1999). Semi-Structured Data Extraction from Heterogeneous Sources. In *Proceedings of 2nd International Workshop on Innovative Internet Information Systems (IIS'99) in conjunction with the European Conference on Information Systems (ECIS'99)*, Copenhagen, Denmark.

EMBLEY D. W., CAMPBELL D. M., SMITH R. D. and LIDDLE S. W. (1998). Ontology-Based Extraction and Structuring of Information from Data-Rich Unstructured Documents. In G. Gardarin, J.C. French, N. Pissinou, K. Makki and L. Bouganim, editors, *In Proceedings of the International Conference on Knowledge Management*. ACM.

PELEATO R. A., CHAPPELIER J.-C., and RAJMAN M. (2000). Automated Information Extraction out of Classified Advertisements. In *Natural Language Processing and Information Systems : 5th International Conference on Applications of Natural Language to Information Systems, NLDB 2000*, Versailles, France.

SEO H., YANG J., and CHOI J. (2001), Knowledge-based Wrapper Generation by Using XML. In *IJCAI-2001 Workshop on Adaptive Text Extraction and Mining*, Seattle, Washington.

MANNING C. D., SCHÜTZE H. (1999). *Foundations of statistical natural language processing*. MIT Press, 620 p., Cambridge, MA.

VITERBI A. (1967). Error bounds for convolutional codes and an asymptotically optimum decoding algorithm. *IEEE Transactions on Information Theory*.