



Matthew Michelson  
Craig A. Knoblock

## Au delà de la recherche par mot-clé pour les sources de données sur le World Wide Web

Une des plus importantes caractéristiques du World Wide Web est la capacité à donner aux utilisateurs un accès possible un nombre important d'information. La plupart de cette information est cependant encore mal organisée et inaccessible par une simple recherche par mot-clé. Par exemple, considérez le site d'enchères Ebay (<http://www.ebay.com>).

Si un utilisateur veut déterminer le prix moyen demandé pour un article, ou en compter les occurrences, il ou elle devra soumettre une recherche par mot-clé, récupérer toutes les listes, éliminer par filtre les pages qui ne correspondent pas, et déterminer article par article la réponse adéquate aux questions. En outre, notez qu'une recherche par mot-clé manquerait tous les articles dont le nom serait mal orthographié. En clair, la recherche par mot-clé n'est pas assez appropriée pour fournir des réponses intéressantes sur les données, surtout si nous voulons faire déterminer ces réponses par des programmes, plutôt que par des utilisateurs.

Il est préférable d'inclure dans les listes Ebay un mécanisme leur permettant d'être interrogées de façon structurée. Ainsi, déterminer le prix moyen ou le nombre d'un article consisterait en une simple et unique ligne de requête que même un programme pourrait exécuter. Cet article vous présente la méthode permettant aux sources de données, telles que Ebay, de supporter des requêtes structurées. Nous appelons chacune des listes un « envoi » le but étant d'extraire de chaque envoi les « attributs » inclus dans celui-ci et qui décrivent l'entité. Cette extraction est plus formellement connue sous le nom d'Extraction d'Information (IE).

Reprenons notre exemple d'EBay, en supposant que les envois sont des voitures. Dans ce cas, les attributs seraient des éléments descriptifs de la voiture, tel que la marque, le modèle ou l'année, et exécuter IE sur un envoi nous permettrait de sélectionner tous les attributs appropriés, même si ceux-ci sont mal orthographiés et indépendamment de leur place dans le texte de l'envoi. Une fois les attri-

buts extraits, nous pouvons alors leur affecter des étiquettes, puis interroger la source de données en utilisant ces étiquettes comme schéma d'interrogation. L'addition de ces étiquettes est connue en tant qu'« annotation ». L'approche globale d'annotation est montrée en Figure 1, à l'aide d'un exemple d'envoi d'EBay. Les différentes parties de cette figure sont décrites plus loin dans cet article, mais il est utile de remarquer l'exemple d'annotation du tout dernier encadré de la figure.

### Sources de données non structurées et non grammaticales

Dans cet article nous examinerons les sources de données annotées qui sont « non structurées » et « non grammaticales ». Par sources de données non structurées nous entendons les sources de données qui varient d'envoi en envoi dans l'ordre et inclusion des attributs. Un envoi Ebay sur une voiture pourrait inclure le modèle et l'année, dans cet ordre, pendant qu'un autre aurait l'année et la marque. Puisque l'ordre et l'inclusion des attributs varient aléatoirement dans un envoi, nous ne pouvons pas exploiter cette structure.

« Non grammaticales » se rapporte au fait que la plupart du temps ces envois ne sont pas conformes aux règles de la langue. Si elles l'étaient, nous pourrions exploiter des techniques de traitement de langage naturel pour découvrir plus facilement les attributs. Par exemple, cela pourrait nous aider à identifier les substantifs dans les envois. Cependant, puisque l'envoi n'est pas grammaticalement correct, il ne peut être analysé comme une phrase.

Nous nous concentrons sur les sources de données non grammaticales et non structurées car il y a eu déjà beaucoup de recherches sur les extractions d'attributs de données semi-structurées et structurées, telles que les pages Web, aussi bien que sur les données qui se conforment aux règles d'un langage, telles qu'extraire des entités à partir d'articles de nouvelles. L'IE sur des sources de données non grammaticales et non structurées pose une difficulté car sans structure ou grammaire il y a peu d'indices pour identifier quels éléments de l'envoi sont des attributs, et inversement, quels éléments de l'envoi sont des marques qui peuvent être ignorées, et que nous appelons « junk ».

### Ensembles de références

Puisque nous ne pouvons pas compter sur la structure ou la grammaire des sources de don-

#### Sur les auteurs :

Matthew Michelson est étudiant à l'université de Californie du Sud. Craig A. Knoblock est chef de projet sénior à l'institut USC des Sciences de l'Information et maître de conférence en recherches à l'USC.  
Contacts: {michelso,knoblock}@isi.edu

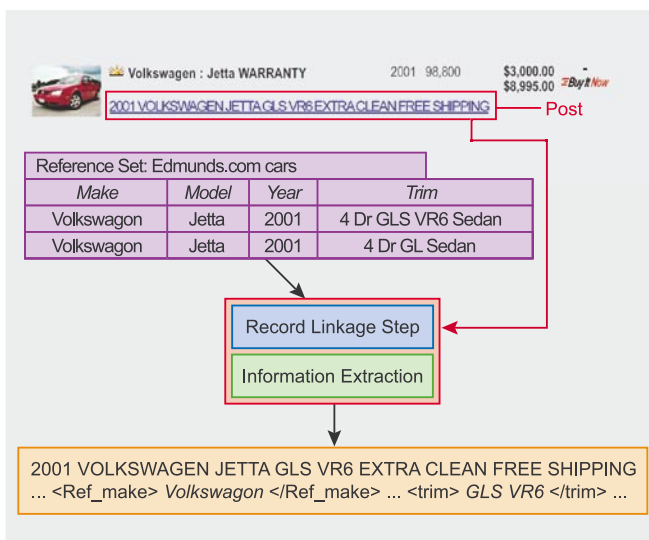


Figure 1. Les points du niveau de similitude d'enregistrement

nées, nous devons mélanger IE avec des connaissances externes pour lui donner des indices sur quels morceaux de l'envoi sont des attributs et ceux qui sont des « junk ».

Cette connaissance extérieure vient sous forme d'« ensembles de référence ». Un ensemble de référence est un ensemble d'entités avec des attributs associés. Ce peut être une base de données connectée ou déconnectée ou un ensemble de documents connectée ou déconnectée. En utilisant toujours notre exemple de voitures sur Ebay, un ensemble de référence pourrait être le site Web automobile de Edmunds (<http://www.edmunds.com>) qui fournit les attributs pour les entités de voiture. Les attributs sont dans ce cas des données telles que la marque, le modèle, l'année et le type de véhicule. L'information de ce site peut être organisée pour ressembler au mieux à une base de données, où chaque entité possède ses attributs associés, et qu'ainsi cette base de données de voitures constitue un ensemble de référence. La Figure 1 inclut une partie de l'exemple d'ensemble de référence du site Web d'Edmunds.

Un ensemble de référence est utilisé en alignant chaque envoi au membre de l'ensemble de référence qui correspond au mieux. Cette procédure d'alignement est connue en tant que Liend'Enregistrement, et est représentée par le cadre bleu sur la Figure 1 avec l'étiquette Lien d'enregistrement. L'étape de liaison d'enregistrement prend en entrée l'ensemble de référence et l'envoi, et renvoie la meilleure correspondance possible de l'ensemble de référence, s'il en existe une. Cet envoi correspondant au mieux fournit alors les indices nécessaires à l'exécution de l'IE puisque nous pouvons rechercher dans l'envoi les valeurs des attributs de l'ensemble de référence. Comment est réalisé exactement le lien d'enregistrement est le thème de la prochaine section.

## Étape de lien d'enregistrement

Jusqu'à maintenant nous avons décrit ce que sont les envois et ce qui les rend unique, à savoir leur nature non structurée et non grammaticale. Cependant, en raison de ce manque de structure et de grammaire, nous nous rendons compte que nous avons besoin d'une certaine sorte

de connaissance extérieure afin d'exécuter IE. Cette connaissance extérieure vient sous forme d'ensembles de référence, et nous avons mentionné que nous employons les ensembles de référence en découvrant l'enregistrement de l'ensemble de référence qui correspond au mieux à l'envoi que nous annotons.

La découverte de cette correspondance adéquate est connue en tant que lien d'enregistrement. Le lien d'enregistrement a été étudié pendant longtemps dans les communautés d'Intelligence Artificielle et de base de données. Le lien d'enregistrement traditionnel prend un enregistrement d'une source de données et cherche son correspondant dans une autre. Ceci se fait en examinant les attributs de chaque source de données et en décidant quels enregistrements contiennent les attributs les plus ressemblants et qui correspondraient probablement. La Figure 2 montre un bon exemple de lien d'enregistrement traditionnel sur deux sources de données sur des hôtels.

## Notre différence sur le lien d'enregistrement

Cependant, notre lien d'enregistrement est légèrement différent et exige une nouvelle approche. Un envoi n'est pas encore décomposé en attributs, donc les approches traditionnelles de lien d'enregistrement ne s'appliquent pas. C'est à dire, les attributs à examiner pour similitude sont inclus dans l'envoi, donc une comparaison attribut par attribut n'est pas possible.

Au lieu de cela, notre méthode implique la création d'un vecteur de points de similitude entre l'envoi et tous les attributs de l'ensemble de référence concaténés ensembles. De cette façon, nous pouvons rapprocher l'attribut par similitude d'attribut pour tous ceux qui donnent immédiatement une similitude entre l'enregistrement entier et l'envoi. Ceci s'appelle « niveau de similitude d'enregistrement ». Ainsi, par exemple, en utilisant le premier membre de l'ensemble de référence de la Figure 1, les points du niveau de similitude d'enregistrement seraient les points de similitude entre l'envoi, « 2001 VOLKSWAGEN JETTA GLS VR6 EXTRA CLEAN FREE SHIPPING » et les attributs concaténés « Volkswagen Jetta 2001 4 Dr GLS VR6 Sedan ».

Cependant, il n'y a pas assez d'informations dans le niveau de similitude d'enregistrement pour discerner une

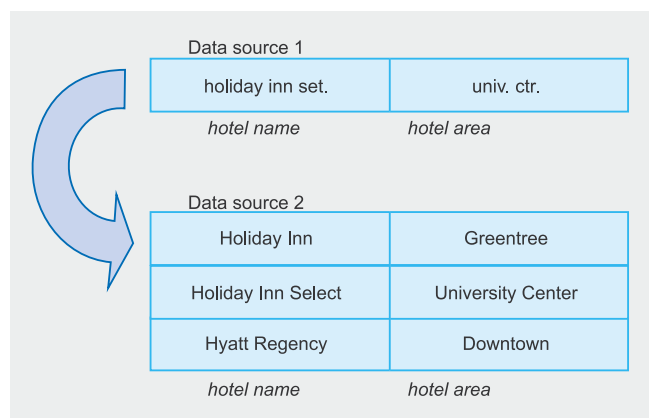


Figure 2. L'exemple de lien d'enregistrement traditionnel sur deux sources de données sur des hôtels

correspondance. En particulier, il est possible que deux enregistrements puisse partager le même niveau de similitude d'enregistrement avec un envoi, tout en différant sur les attributs de l'enregistrement ont causé cette similitude.

Par exemple, considérez la Figure 3, qui montre le lien d'enregistrement entre un envoi sur des hôtels et un ensemble de référence d'hôtels. Sur la figure, chaque membre de l'ensemble de référence correspond à l'envoi sur deux attributs, avec en commun le nom de l'hôtel. Cependant, le premier correspond à l'environnement de l'hôtel alors que le second correspond aux étoiles. L'environnement de l'hôtel est plus distinctif qu'un classement par étoiles, ainsi nous avons besoin de « someway » pour refléter les similitudes entre l'envoi et chaque attribut individuellement. Ceci est fait en ajoutant les points de similitude entre l'envoi et chaque attribut de l'ensemble de référence définissant ainsi le vecteur de points de similitude. Ces comparaisons individuelles d'attributs donnent une approximation d'une comparaison attribut par attribut, et sont appelés "niveau de similitude de champ". Avec notre exemple courant, nous allons produire les points de similitude entre l'envoi et « Volkswagen », l'envoi et « Jetta » et ainsi de suite.

Ainsi, notre vecteur total de similitudes inclut à la fois les notions de niveau de similitude d'enregistrement, avec la concaténation d'attribut, et de niveau de similitude de champ avec chaque attribut individuellement. Il convient de noter que ce vecteur de similitudes n'est pas créé pour tous les enregistrements de l'ensemble de référence. En fait, dans le lien d'enregistrement en général, les membres d'un ensemble de données ne sont jamais comparés à tous les membres des autres ensembles de données. Au lieu de cela, seul un sous-ensemble d'enregistrements, appelés « candidats », est d'abord choisi, puis ces candidats sont utilisés pour le lien d'enregistrement.

Le choix des candidats pour le lien d'enregistrement est appelé « blocage ». Le but du blocage est de limiter la taille de l'ensemble des candidats, sans y enlever les vraies correspondances, aussi vite que possible. Notre technique d'annotation est indépendante de l'algorithme de blocage choisi. Par exemple nous pouvons simplement dire que n'importe quel membre de l'ensemble de référence qui partage une marque commune avec l'envoi est candidate.

Une fois que tous les enregistrements candidats de l'ensemble de référence ont un vecteur de points de similitude, ils sont tous recalculés en mode binaire. En particulier, à chaque index du vecteur de similitude, les candidats ayant la valeur maximale de cet index changent leur valeur à 1, et le reste des candidats changent leur valeur à 0. Par exemple, donnons deux vecteurs correspondants candidats :

V1 = (0.1, 0.2, 0.5, ..., 2.1)

V2 = (0.2, 0.1, 0.5, ..., 0.8)

Après recalcul binaire, ils deviennent :

V1 = (0, 1, 1, ..., 1)

V2 = (1, 0, 1, ..., 0)

Ce recalcul binaire est fait pour souligner la différence entre les points du vecteur. Certains vecteurs auront des index avec

des scores proches, mais ceux-ci seront exagérés par le recalcul binaire, par conséquent la meilleure correspondance sera plus facilement détectée.

La construction et le marquage des vecteurs de points de similitude est toute une étape de prétraitement pour la machine apprenant la partie de l'étape de lien d'enregistrement qui nous formulera le candidat présentant la meilleure correspondance à l'envoi.

La machine apprenant la partie de l'étape de lien d'enregistrement prend l'ensemble de candidats (maintenant un ensemble de vecteurs binaires) et les étiquettes comme correspondant ou non correspondant. Il peut également retourner un score de confiance associé à la correspondance. Ainsi, nous pouvons simplement prendre le candidat étiqueté comme correspondant, et avec le score de confiance le plus élevé, en tant que meilleure correspondance. Si nous voulons imposer l'idée qu'il n'y a qu'un membre de l'ensemble de référence pour chaque envoi, alors nous rejetons tous les candidats qui ont le même score maximal de confiance. De cette façon, nous nous assurons qu'un seul candidat peut toujours correspondre à l'envoi. Si nous voulons relâcher cette restriction, nous pouvons sélectionner n'importe quel des candidats ayant le score maximal comme meilleure correspondance.

La technique actuelle utilisée pour l'apprentissage machine est connue sous le nom de Support Vector Machine (SVM). Les SVMs sont largement utilisés dans la recherche sur l'Intelligence Artificielle comme technique efficace d'apprentissage machine. L'idée maîtresse est de considérer les vecteurs dans un espace à n dimensions. Si le problème de l'apprentissage était facile, nous pourrions adapter un plan à n dimensions, appelé hyperplan, entre les vecteurs qui correspondent et ceux qui ne correspondent pas. Puis, selon l'emplacement d'un vecteur, nous pourrions déterminer sa classe. Puisque nous ne pouvons pas le faire, SVM trace chacun des vecteurs dans un nouvel espace à n dimensions, où il est possible d'adapter un hyperplan qui distingue les ensembles. Il détermine alors si un vecteur correspond ou non, selon son emplacement par rapport à ce nouvel hyperplan dans le nouvel espace de caractéristique.

Les SVMs sont des techniques surveillées d'apprentissage machine, ce qui signifie qu'elles exigent des données d'entraînement étiquetées. Ces dernières consistent en des couples de membres d'ensemble de référence et d'envois qui sont déjà marqués comme correspondantes ou non correspondantes. Celles-ci permettent au SVM d'apprendre

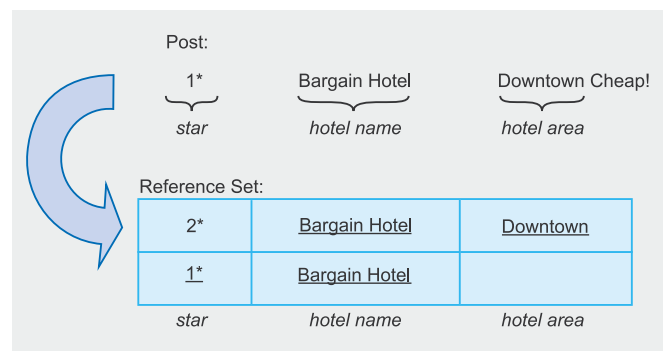


Figure 3. Le lien d'enregistrement entre un envoi sur des hôtels et un ensemble de référence d'hôtels

comment adapter l'hyperplan aux données afin qu'il puisse classifier les vecteurs précédemment invisibles.

Une fois que le SVM a déterminé la meilleure correspondance à l'envoi, il y a une procédure finale à l'étape de lien d'enregistrement. Nous ajoutons l'annotation à l'envoi qui inclut les attributs du membre de référence correspondant le mieux. En Figure 1, un de ces attributs est indiqué avec les marques <Ref\_make>. C'est ainsi pour plusieurs raisons. Premièrement, cela donne des valeurs standard pour lesquelles il faut interroger les données. Si nous utilisons les valeurs réelles extraites pour interroger les données, nous rencontrons le même problème qu'avec la recherche par mot-clé, à savoir que les différentes fautes d'orthographe laissent certains envois en dehors de l'interrogation. Deuxièmement, cela inclut les attributs qui peuvent ne pas avoir été inclus par l'utilisateur. Dans notre exemple de voiture Ebay en cours, un envoi peut inclure un modèle, un type et l'année. Si nous interrogeons les envois sur la marque cet envoi sera laissé derrière puisqu'il ne possède pas explicitement de marque. Cependant, avec les attributs de l'ensemble de référence qui lui sont annotés, il possède dorénavant une marque et sera retourné par l'interrogation. Enfin, il y a certains attributs qui sont extrêmement durs pour l'IE. L'extraction pourrait assez mal s'exécuter sur ces attributs et ne pas être utile lors des requêtes parce qu'elle retournerait beaucoup de résultats inexacts ou pas assez d'envois. Toutefois, si l'étape de lien d'enregistrement est bien réalisée, alors nous aurions à la place des attributs de l'ensemble de référence à utiliser, de sorte que cet attribut soit encore utile pour interroger les données.

## La méthode d'extraction

Une fois que nous avons trouvé le membre de l'ensemble de référence correspondant au mieux, nous pouvons exploiter ce membre dans l'IE. L'intuition est de prendre chaque marque de l'envoi et de regarder s'il correspond à un

des attributs de l'ensemble de référence. Ceci est appelé l'étape d'Extraction d'Information et est indiqué dans le cadre vert de la Figure 1.

Spécifiquement, nous prenons chaque marque de l'envoi et créons un vecteur de points de similitude entre cette marque et chaque attribut du membre correspondant de l'ensemble de référence. C'est semblable à la création du vecteur de points de similitude du lien d'enregistrement, excepté que nous n'exécutons pas de recalcul binaire sur le vecteur, puisque nous n'avons pas besoin de choisir un gagnant d'un ensemble de candidats, comme précédemment. En outre, ce vecteur contient un ensemble unique de points pour comparer les similitudes de cette marque avec des attributs spéciaux appelés attributs « communs ».

En général, ces attributs communs sont des données qui ne sont pas faciles à représenter par des ensembles de référence, alors qu'ils exposent suffisamment de caractéristiques d'identification exploitables. Des exemples d'attributs communs seraient des prix et des dates. Ces caractéristiques particulières prennent en compte pour extraire ces types de données l'utilisation de techniques plus traditionnelles, telles que les expressions régulières. Ainsi, nous incluons des points faire correspondre un prix à une expression régulière, par exemple. Ces points donneraient un score positif pour une correspondance et 0 autrement. Ceci tient compte de l'extraction des attributs utiles de l'envoi qui sont facilement extraits au moyen de quelques règles experts.

Une fois que nous avons créé le vecteur de points de similitude, nous tentons d'identifier le type d'attribut du vecteur de la marque en passant le vecteur à une multi-classe SVM. Semblable au SVM entraîné à étiqueter les correspondances ou non correspondances, une multi-classe SVM est capable d'identifier une marque comme un membre de n classes. Dans notre cas, n-1 des classes sont les types d'attributs, comme la marque de la voiture ou le modèle, et la nième classe est un « junk », ce qui signifie que la marque peut être ignorée.

Une fois que toutes les marques de l'envoi ont été identifiées comme appartenant à un attribut ou un junk, nous nettoyons chaque ensemble, extrait d'attributs de l'envoi. L'ensemble, extrait d'attributs est juste la concaténation de chacune des marques de l'envoi qui possèdent la même étiquette. Dans l'exemple en cours, les marques « GLS » et « VR6 » serait chacune identifiée comme le type de voiture, ainsi le type de voiture de l'ensemble extrait serait « GLS VR6 ». De manière intuitive, nous avons besoin d'étiqueter des marques à isoler, parce que les données peuvent être complètement déstructurées, et ainsi empêcher l'exploitation de la structure de l'envoi. Toutefois, ceci introduit des marques bruyantes, c'est à dire, des marques qui devraient être étiquetées en tant que « junk », mais ne le sont pas. Pour corriger ce problème, nous prenons chaque ensemble, extrait d'attributs le comparons à son attribut correspondant membre de l'ensemble de référence. Nous enlevons une marque à la fois de l'extrait d'attributs, puis si ce nouvel attribut correspondant mieux à l'attribut de l'ensemble de référence que l'ancien, cette marque devient candidate à la suppression. Après avoir traité chaque marque de cette manière soit nous enlevons le candidat à la

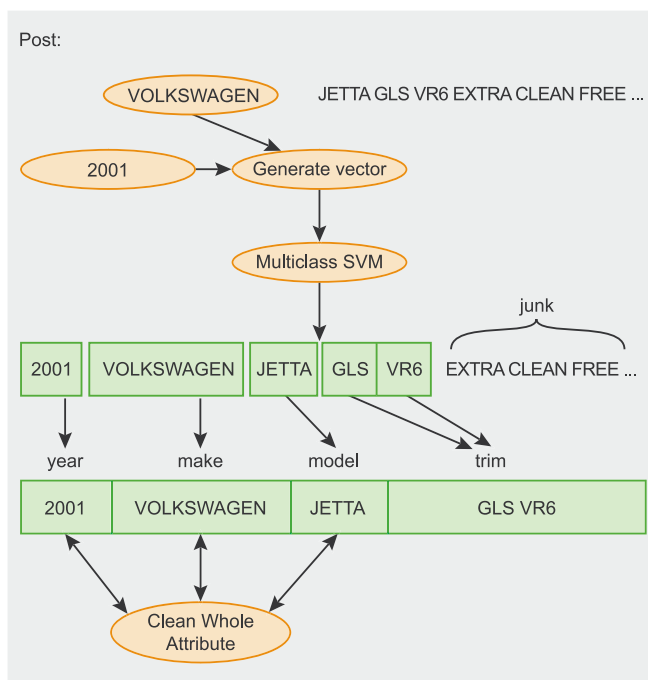


Figure 4. Le processus d'extraction dans son ensemble

suppression qui a rapporté la meilleure correspondance à l'attribut de l'ensemble de référence, soit nous terminons le traitement parce qu'aucune marque n'a rapporté la bonne correspondance. Si nous ne terminons pas, alors nous démarrons à nouveau un cycle.

Ce traitement enlève non seulement les marques bruyantes de l'extrait d'attribut, mais il a l'avantage supplémentaire de lever les ambiguïtés. Si une marque peut facilement appartenir à plus d'un type d'attribut, nous pouvons étiqueter les deux et laisser la procédure de nettoyage enlever la marque de l'attribut auquel il ne devrait pas appartenir.

Ceci est l'intégralité de la technique d'extraction des attributs d'un envoi en vue de construire les annotations permettant des interrogations structurées. La Figure 4 dépeint le processus d'extraction dans son ensemble.

Quelques observations sont à noter au sujet de la procédure d'extraction. Tout d'abord, puisque habituellement les attributs se chevauchent dans un ensemble de référence, cette approche d'extraction est robuste face aux erreurs faites en étape de lien d'enregistrement. Si l'étape de lien d'enregistrement n'identifie pas correctement un membre correspondant dans l'ensemble de référence, la correspondance qu'il identifie d'habitude a une information suffisamment similaire pour la rendre utile. Par exemple, avec les voitures sur Ebay, si l'étape de lien d'enregistrement retourne une Volkswagen Jetta de 2001, mais avec un type incorrect, il y aura tout de même assez d'information utile pour extraire de l'envoi la marque, le modèle et l'année de la voiture. C'est une raison pour laquelle l'algorithme ne doit pas s'arrêter après l'étape de lien d'enregistrement, bien que cela semble tentant, puisqu'à ce stade la plupart des données sont annotées avec des valeurs standard.

Une autre raison d'exécuter l'extraction, au delà du désir évident de voir les valeurs réelles des attributs saisies par les utilisateurs, est que l'entraînement du système à extraire tous les attributs l'aidera à extraire les attributs « communs » de deux façons. Premièrement, il fait du junk une classification plus rare, ce qui améliore la précision de classification des junk. Deuxièmement, il est utile de pouvoir classifier quelque chose qui ne l'est pas. Considérez l'exemple suivant. Un modèle de voiture peut être appelé « \$35 ». C'est un cas étrange en effet, mais si le système n'est pas entraîné à extraire des modèles de voiture, cette marque serait certainement classifiée comme un prix. Cependant, puisque le système peut apprendre que c'est un modèle de voiture, ce ne sera pas un prix.

## Références :

- Matthew Michelson et Craig A. Knoblock, « Semantic Annotation of Unstructured and Ungrammatical Text », Dans les procédés de la 19ième Conférence Internationale sur l'Intelligence Artificielle (IJCAI), Edinburgh, Scotland, 2005
- Matthew Michelson, « Building Queryable Datasets from Ungrammatical and Unstructured Sources », Thèse de doctorat, Université de Californie du Sud, 2005
- Ces articles peuvent être obtenus sur [www.isi.edu/~michelso](http://www.isi.edu/~michelso)

## L'exécution

Cet algorithme a été montré empiriquement pour surpasser d'autres méthodes de lien d'enregistrement et d'extraction pour la tâche d'annotation sémantique. Plus spécifiquement, l'exemple de cet article est simple, et l'algorithme a été montré pour s'exécuter sur des données beaucoup plus complexes, comme par exemple, des envois avec de nombreux attributs manquants et diverses fautes d'orthographe sur les valeurs des attributs. Les lecteurs intéressés peuvent lire les publications scientifiques présentées à la fin de cet article, dans la section Références, pour voir le nombre réel d'exécutions. En outre, les publications de recherche donnent un traitement plus avancé et détaillé des algorithmes.

Un autre aspect d'exécution à considérer avec n'importe quelle méthode d'apprentissage contrôlée est la quantité de données d'entraînement que le système exige pour bien s'exécuter. De plus, les publications de recherche montrent qu'un entraînement sur seulement 10% des données (moins de 80 exemples dans certains cas) ne dégrade pas l'exécution de l'algorithme. C'est un avantage, puisque l'étiquetage des données d'entraînement est coûteux et fastidieux.

Une dernière observation en ce qui concerne l'exécution se réfère aux ensembles de référence. Il semblerait que si nous devions utiliser deux ensembles de référence séparés (puisque l'algorithme n'est pas lié à un seul ensemble), alors nous devrions utiliser le produit en croix des enregistrements de ces deux ensembles de référence comme un seul et grand ensemble de référence. Ceci rendrait l'exécution globale beaucoup plus fastidieuse en raison de la très grande taille de ce nouvel ensemble de référence. Ce n'est pas le cas. L'algorithme supporte facilement l'utilisation d'ensembles de référence en mode itératif, façon à ce que plusieurs ensembles de référence sont nécessaires ils peuvent être utilisés. C'est, en fait, une optimisation utile puisque de petits ensembles de référence d'attributs de référence unique peuvent être formés et utilisés à la place de simples grands ensembles de référence.

## Conclusion

Cet article a présenté une méthode grâce à laquelle les données non structurées et non grammaticales du World Wide Web sont rendues utilisables par l'annotation des données en vue de requêtes structurées. Ces requêtes structurées vont au delà de simples recherches par mot-clé pour transformer des données en information. À l'avenir, lorsque des agents Web agiront en notre nom, sur des arrangements de réservation de voyage ou sur des échanges de marchandises, ces types de requêtes deviendront cruciaux du point de vue du processus décisionnel. La recherche présentée ici est une première étape, et nous espérons que des améliorations futures aideront à établir le lien entre le temps où les utilisateurs doivent accomplir de pénibles tâches sur le World Wide Web et celui où leurs agents le feront pour eux. ■