Fiche de présentation de
Active Learning for Natural Language Parsing and Information Extraction,
de Cynthia A. Thompson, Mary Elaine Califf et Raymond J. Mooney

Philippe Gambette

Cet article présente l'utilisation de l'apprentissage actif pour deux problèmes de traitement automatique des langues : le parsing de langages naturels, appliqué à un problème de question-réponse, avec question en langue naturelle, sur la base de données géographiques GeoBase[1] ; et l'extraction d'information, c'est à dire la transformation automatique de posts d'offres d'emplois sur des serveurs de news en formulaires avec divers champs remplis : type d'emploi, rémunération, localisation...

L'introduction insiste sur l'intérêt de l'apprentissage actif qui consiste à effectuer l'apprentissage tout en étiquetant les exemples, afin de minimiser le nombre d'exemples à étiqueter. Cette problématique est particulièrement intéressante dans le domaine des langues naturelles où l'étiquetage demande un gros travail humain, et où les données non étiquetées sont en revanche abondantes. Les auteurs citent donc de nombreux exemples d'applications d'apprentissage actif à divers problèmes d'étiquetage, et annoncent qu'ils vont l'utiliser pour des problèmes plus complexes que l'étiquetage.

Ils présentent alors les bases de l'apprentissage actif, qui consiste à partir d'un ensemble *L* de *n* exemples annotés, et de répéter plusieurs fois la boucle suivante : évaluer pour tous les exemples non annotés l'incertitude s'ils sont annotés avec l'algorithme d'apprentissage entraîné sur *L*, annoter les *k* exemples les plus incertains, et les ajouter à l'ensemble *T*, puis réentraîner l'algorithme d'apprentissage sur *T*. Cette boucle s'arrête... un jour, soit parce qu'il n'y a plus d'exemples, soit après un nombre de boucles défini préalablement. Ainsi, il y a 3 paramètres : le nombre d'exemples annotés initialement *n*, le nombre de ceux annotés à chaque boucle *k*. et le nombre d'exemples à annoter pour considérer que l'ensemble d'entraînement sera assez robuste pour entraîner optimalement l'algorithme d'apprentissage. La mesure de l'incertitude dépend du problème traité. Les auteurs mentionnent en passant la méthode des comités qui consiste à utiliser plusieurs programmes d'apprentissage (ou plutôt un même algorithme entraîné sur des données différentes, variées), ce qui permet de définir l'incertitude comme la discordance entre les avis de ces programmes.

Le système CHILL est alors présenté, c'est un programme qui permet de déduire d'un couple (phrase,parsing sémantique) (par exemple *("What is the capital of Texas ?",answer(A, (capital(B,A), equal(B,stateid(texas)))))*), un parser qui fera automatiquement la traduction de la langue naturelle vers la question en langage utilisable par l'ordinateur pour interroger la base de données géographiques et récupérer la réponse à la question. La partie induction, comme ça n'est pas précisé dans l'article, est gérée par un algorithme d'apprentissage relationnel mêlant les approches de bas en haut et de haut en bas, CHILLIN, développé pour l'occasion[2].

L'extraction d'information est gérée par RAPIER[3], un algorithme d'apprentissage relationnel de bas en haut, c'est à dire qui trouve des règles de plus en plus générales, les règles consistant en fait en motifs, c'est à dire des contraintes, syntaxiques ou sémantiques, sur les mots précédant et suivant le mot-cible (l'information qui sera ajoutée à un des champs du formulaire finalement).

L'utilisation de l'apprentissage actif est alors détaillée pour CHILL, suivie par les résultats expérimentaux : il s'agit de trouver un moyen d'évaluer l'incertitude, puis de comparer l'apprentissage actif à l'apprentissage passif. Pour évaluer l'incertitude sur le parsing sémantique d'une phrase, soit celle-ci n'a pas pu être parsée, et alors l'exemple est très incertain. Si on a plusieurs exemples dans ce cas, c'est la « profondeur » du parsing, c'est à dire le nombre d'étapes de parsing qu'on a pu effectuer en appliquant les règles, qui est utilisée. S'il y a moins de *k* exemples non parsables à une boucle de l'algorithme actif, il faut compléter en prenant les exemples parsables les plus incertains : on évalue la certitude comme la moyenne de la certitude des règles utilisées lors du parsing, une règle étant d'autant plus certaine que beaucoup d'exemples la vérifient. Pour comparer les résultats à l'apprentissage passif, on effectue le même algorithme pour l'apprentissage passif en choisissant juste les *k* exemples à annoter au hasard. Enfin, pour évaluer les performances, on regarde le nombre de réponses correctes sur 250 questions, parsées par un expert. L'apprentissage automatique donne de meilleures règles que celles fixées par les experts pour GeoBase, et l'apprentissage actif montre une légère amélioration par rapport à l'apprentissage passif : on atteint 68% de justesse avec 29% d'exemples en moins environ

Pour Rapier, la certitude d'un formulaire est la somme des certitudes de ses champs. Si un champ est vide, sa certitude dépend du temps de remplissage de ce champ : basse si le champ est souvent rempli. S'il est plein, avec éventuellement plusieurs valeurs trouvées par plusieurs règles, on prend le minimum, pour chaque règle, de la certitude définie comme le *nombre d'exemples vérifiés par la règle – 5 fois le*

1 http://www.cs.utexas.edu/users/ml/geo-demo.html
2 Thèse de John Zelle : *Using Inductive Logic Programming to Automate the Construction of Natural Language Parsers*, (1995)
3 Mary Elaine Califf, Raymond J. Mooney : *Bottom-Up Relational Learning of Pattern Matching Rules for Information Extraction*, (2003)

nombre d'exemples non vérifiés. L'évaluation se fait sur 300 posts annotés, avec un système de validation croisée à 10 plis, et évaluation du bruit et de la couverture sur tous les champs des formulaires, et calcul de la classique F-measure. L'efficacité est un peu meilleure qu'avec l'algorithme précédent, mais pas impressionnante : 44% d'économie d'annotation d'exemples pour une F-measure de 70%.

Enfin, les auteurs admettent avoir échoué à utiliser la méthode d'apprentissage actif par comités, insistent sur le fait de trouver des méthodes pour choisir le paramètre *k* et de trouver des mesures de certitude plus subtiles, ainsi que sur la nécessité d'essayer d'appliquer l'apprentissage actif dans d'autres domaines.

Un paragraphe sur les travaux reliés permet de compléter un peu l'état de l'art sur le domaine, et les auteurs concluent sur les gains apportés par l'apprentissage actif.

Mon avis :

L'article était bien écrit, les protocoles expérimentaux bien détaillés. La clarté était permise grâce aux deux exemples pour bien situer les problèmes, mais les algorithmes utilisés, CHILL et RAPIER, n'étaient pas décrits en détail. Un exemple des règles générées par ces deux programmes aurait été la bienvenue, d'autant que même dans les articles de description des ces programmes, cités plus haut, ces exemples ne sont pas très nombreux.

Si les performances de 20 ou 40% peuvent intéresser le patron désireux de réduire les frais d'annotation de corpus, la performance semble toutefois maigre par rapport à certaines méthodes d'apprentissage actif qui permettent d'économiser 90% de l'annotation des exemples et de réaliser de vrais changements d'échelle en matière de coût. D'autant que ces améliorations sont celles obtenues pour un certain point de l'expérience (5% de moins que le résultat final) sur lequel on n'a aucune précision en pratique : si je veux utiliser les algorithmes présentés sur mes données, quel sera mon point d'arrêt de l'apprentissage actif ? De plus, on n'a aucune indication sur la variabilité des performances en fonction du choix des paramètres *n* et *k*, ce qui est assez inquiétant : peut-être ces performances correspondent-elles aux choix optimaux des paramètres déterminés expérimentalement par les auteurs ?

On n'a pas non plus d'indication de temps de calculs, ne serait-ce que des ordres de grandeur, pour les algorithmes CHILL et RAPIER.

Enfin, on peut se demander où se place cette recherche dans le cadre de celle sur le parsing sémantique en linguistique : les travaux de la théorie sens-texte[4] (Mel'čuck), avec la (lente) création de dictionnaires explicatifs et combinatoires permettent justement de déterminer ces règles de parsing de façon experte. Quand elle sera aboutie, peut-être permettra-t-elle une interprétation directe des questions de GeoBase en langage machine ? L'approche présentée dans l'article semble être restreinte à des corpus très spécialisés, puisque c'est un expert qui doit effectuer le parsing de base. Ce parsing doit donc être très spécifique.

4 Alain Polguère, *La Théorie Sens-Texte* (1998)