Stratégies de sélection des exemples pour l'apprentissage actif avec des champs aléatoires conditionnels

Vincent Claveau Ewa Kijak
IRISA – CNRS – Univ. Rennes 1, Campus de Beaulieu, 35042 Rennes cedex
Vincent.Claveau@irisa.fr, Ewa.Kijak@irisa.fr

Résumé. Beaucoup de problèmes de TAL sont désormais modélisés comme des tâches d'apprentissage supervisé. De ce fait, le coût des annotations des exemples par l'expert représente un problème important. L'apprentissage actif (*active learning*) apporte un cadre à ce problème, permettant de contrôler le coût d'annotation tout en maximisant, on l'espère, la performance de la tâche visée, mais repose sur le choix difficile des exemples à soumettre à l'expert.
Dans cet article, nous examinons et proposons des stratégies de sélection des exemples pour le cas spécifique des champs aléatoires conditionnels (*Conditional Random Fields*, CRF), outil largement utilisé en TAL. Nous proposons d'une part une méthode simple corrigeant un biais de certaines méthodes de l'état de l'art. D'autre part, nous détaillons une méthode originale de sélection s'appuyant sur un critère de respect des proportions dans les jeux de données manipulés. Le bien-fondé de ces propositions est vérifié au travers de plusieurs tâches et jeux de données, incluant reconnaissance d'entités nommées, *chunking*, phonétisation, désambiguïsation de sens.

Abstract.

Strategies to select examples for Active Learning with Conditional Random Fields
Nowadays, many NLP problems are modelized as supervised machine learning tasks. Consequently, the cost of the expertise needed to annotate the examples is a widespread issue. Active learning offers a framework to that issue, allowing to control the annotation cost while maximizing the classifier performance, but it relies on the key step of choosing which example will be proposed to the expert.
In this paper, we examine and propose such selection strategies in the specific case of Conditional Random Fields (CRF) which are largely used in NLP. On the one hand, we propose a simple method to correct a bias of certain state-of-the-art selection techniques. On the other hand, we detail an original approach to select the examples, based on the respect of proportions in the datasets. These contributions are validated over a large range of experiments implying several tasks and datasets, including named entity recognition, chunking, phonetization, word sens disambiguation.

Mots-clés : CRF, champs aléatoires conditionnels, apprentissage actif, apprentissage semi-supervisé, test statistique de proportion.

Keywords: CRF, conditional random fields, active learning, semi-supervised learning, statistical test of proportion.

1 Introduction

De nombreuses tâches de TAL reposent désormais sur des approches d'apprentissage artificiel supervisé. Parmi les techniques couramment employées, les champs aléatoires conditionnels (*Conditional Random Fields*, CRF) ont montré d'excellentes performances pour tout ce qui relève de l'annotation de séquences (*tagging*, reconnaissance d'entités nommées et extraction d'information, translittération...). Cependant, comme pour tous les problèmes supervisés, le coût d'annotation des séquences pour entraîner les modèles est un critère important à considérer. Pour des problèmes simples, comme l'étiquetage en parties-du-discours, des études on montré que ce coût est relativement faible (Garrette & Baldridge, 2013), mais la plupart des problèmes cités précédemment nécessitent au contraire un très grand nombre d'annotations (cf. section 5.2).

Pour limiter ce coût, les approches semi-supervisées exploitent, en plus des exemples annotés, des exemples non-annotés qui sont eux plus facilement disponibles. Parmi ces approches, l'apprentissage actif (*Active learning*) permet à l'expert d'annoter des exemples supplémentaires de manière itérative, contrôlant ainsi le compromis coût d'annotation/performance du classifieur. Un classifieur peut ainsi être appris ou amélioré à chaque itération, et peut servir à guider le choix des prochains exemples à annoter. Dans cet article, nous nous intéressons à ce problème d'apprentissage actif, et plus précisément au problème de la sélection des exemples qui sont proposés à l'expert, dans le cas particulier des CRF.

Il existe bien sûr déjà de nombreuses méthodes de sélection, soit génériques, soit propres aux CRF. Dans cet article, nous montrons que certaines méthodes très classiques de l'état de l'art comportent un biais tendant à favoriser le choix d'exemples longs, et donc coûteux à annoter. Nous proposons une technique simple pour lever ce biais. Mais notre contribution principale porte sur la proposition d'une technique de sélection originale, utilisant la représentation qui est faite des données par les CRF, et s'appuyant sur un critère de respect des proportions d'attributs dans les jeux de données. Ces différentes propositions sont évaluées expérimentalement sur plusieurs jeux de données et tâches classiques des CRF.

L'article est structuré de la façon suivante. La section 2 rappelle quelques notions de bases autour des CRF et de l'apprentissage actif, et présente des travaux connexes ainsi que les données servant à nos expérimentations. Nous revisitons ensuite en section 3 certaines techniques habituellement utilisées pour en montrer les limites. Dans la section 4, nous présentons une nouvelle technique pour la sélection des exemples à annoter. La section 5 présente et commente les expérimentations menées, et la dernière section présente quelques perspectives ouvertes par ce travail.

2 Contexte et état de l'art

2.1 Notions de base

Les champs aléatoires conditionnels ou Conditional Random Fields (Lafferty *et al.*, 2001) sont des modèles graphiques non dirigés qui représentent la distribution de probabilités d'annotations y sur des observations x. Ils sont très employés en TAL ; x est alors une une séquence de lettres ou de mots et y la séquence correspondante de labels. Dans ce cadre, la probabilité conditionnelle $P(y|x)$ se définit à travers la somme pondérée de fonctions dites caractéristiques (*feature functions*) f_j :

$$P(y|x,\theta) = \frac{1}{Z_\lambda(x)} \exp\left(\sum_j \sum_t \lambda_j f_j(x, y_t, y_{t-1}, t)\right)$$

où $Z_\lambda(x)$ est un facteur de normalisation et θ est le vecteur des poids λ_j. Les fonctions caractéristiques sont souvent binaires, renvoyant 1 lorsqu'une certaine combinaison de labels et d'attributs des observations est satisfaite, 0 sinon. Elles sont appliquées à chaque position t de la séquence et leur poids λ_j reflète leur importance pour déterminer la classe. Il est important de noter qu'en pratique, ce n'est pas tout le vecteur x qui est considéré mais juste une certaine combinaison d'attributs sur les objets autour de la position t. Ces combinaisons sont définies par l'utilisateur, le plus souvent indirectement par un ensemble de patrons $\{\text{Pat}_i\}$ dont les réalisations à chaque position t de chaque séquence x ($\text{Pat}_i(x,t)$), ajoutées aux informations de labels correspondantes (y_{t-1} et y_t), définissent l'ensemble des fonctions possibles.

L'apprentissage d'un CRF consiste à estimer les poids λ_j à partir de données dont les labels sont connus. On cherche alors

le vecteur θ qui maximise la log-vraisemblance $\mathcal{L}$ du modèle sur les m séquences annotées :

$$\mathcal{L}(\theta) = \sum_m \log P_\theta(y^{(m)}|x^{(m)}, \theta)$$

En pratique, on ajoute souvent des contraintes sur la taille du vecteur θ pour éviter le sur-apprentissage. Ce problème d'optimisation peut être résolu en utilisant des algorithmes de type quasi-Newton, comme L-BFGS (Schraudolph *et al.*, 2007). Une fois le CRF appris, l'application du CRF à des nouvelles données consiste à trouver, pour une séquence d'observations x, la séquence de labels la plus probable, notée y^* dans la suite de cet article, par exemple avec un algorithme de Viterbi.

Grâce à leur capacités à prendre en compte l'aspect séquentiel et les descriptions riches des textes, les CRF ont été utilisés avec succès dans de nombreuses tâches s'exprimant comme des problèmes d'annotation. Ils sont ainsi devenus des outils standard pour l'extraction d'information, la reconnaissance d'entités nommées, le tagging, etc. (Wang *et al.*, 2006; Pranjal *et al.*, 2006; Constant *et al.*, 2011; Raymond & Fayolle, 2010, inter alia).

2.2 Apprentissage semi-supervisé

L'apprentissage semi-supervisé consiste à utiliser conjointement des données annotées (notées $\mathcal{T}$ dans la suite de l'article) et des données non-annotées ($\mathcal{N}$). Son but est de réduire le nombre d'annotations et donc le coût de l'annotation, et/ou d'améliorer les performances du classifieur à coût d'annotation identique. Différentes approches d'apprentissage semi-supervisé ont déjà été explorées pour les CRF. Plusieurs travaux utilisent les données non étiquetées directement dans l'apprentissage du modèle en modifiant l'expression de l'entropie. Cette modification rend la fonction objectif non-concave et nécessite donc d'adapter la procédure d'apprentissage.

Une autre famille de travaux a consisté à adapter les procédures d'apprentissage et de décodage pour que les CRF soient capables d'exploiter des connaissances sur les séquences autres que l'annotation complète de la séquence. Il peut s'agir par exemple d'annotations partielles des séquences, c'est-à-dire dont les étiquettes ne portent que sur quelques mots (Salakhutdinov *et al.*, 2003). Il peut également s'agir de connaissances a priori sur la distribution des étiquettes sachant certains attributs (Mann & McCallum, 2008).

Bien que cela ne relève pas strictement de l'apprentissage semi-supervisé, il convient également d'évoquer les travaux utilisant des techniques annexes sur les données non annotées pour améliorer l'apprentissage sur les données annotées. Par exemple, (Miller *et al.*, 2004) et (Freitag, 2004) font du *clustering* sur les données non-annotées pour proposer de nouveaux attributs – en l'occurrence, des classes de mots – ensuite utilisés pour mieux décrire les données annotées. Dans cette veine, il convient également de citer les travaux de (Ando & Zhang, 2005) et ceux de (Smith & Eisner, 2005). Ces derniers exploitent une proximité entre une séquence annotée et d'autres séquences pour influer sur l'estimation des paramètres du CRF. Bien que là encore ces travaux ne se placent pas dans le même cadre que nos travaux, ceux-ci partagent néanmoins l'idée d'exploiter la ressemblance des séquences vues comme des ensembles d'attributs.

2.3 Apprentissage actif

Dans notre cas, nous nous plaçons dans un cadre spécifique d'apprentissage semi-supervisé qualifié d'apprentissage actif (*active learning*). Son principe est que la supervision est effectuée par l'expert de manière itérative et interactive (Settles, 2010). Cela est souvent mis en œuvre dans un algorithme dont les grandes lignes sont les suivantes :

1. apprendre un classifieur à partir de $\mathcal{T}$
2. appliquer le classifieur à $\mathcal{N}$
3. sélectionner des exemples de $\mathcal{N}$
4. annoter ces exemples et les ajouter à $\mathcal{T}$
5. retourner en 1

Ce processus est ainsi répété jusqu'à ce qu'un critère d'arrêt soit atteint. Ce critère peut être que le coût de l'annotation maximal est atteint, que la performance du classifieur minimale est atteinte, ou que $\mathcal{N}$ est vide.

Le point crucial de ces algorithmes d'apprentissage actif est l'étape 3 de sélection des exemples à faire annoter à l'expert. On cherche à choisir les exemples les plus bénéfiques pour l'apprentissage, ceux permettant d'obtenir les meilleures performances de classification, et pour ce faire, on s'appuie souvent sur l'étape 2. Beaucoup de travaux ont été proposés sur ce

point, notamment dans le domaine du TAL (Olsson, 2009) où ces problèmes d'annotation sont courants. Indépendamment des classifieurs utilisés, plusieurs familles de stratégies ont été proposées. La plus courante est la sélection par incertitude dans laquelle on utilise le résultat de l'étape de 2 pour choisir les exemples pour lesquels le classifieur courant est le moins sûr (cf. section 3). Un défaut connu de cette approche est qu'au début du processus, quand il y a peu d'exemples annotés, les mesures d'incertitude du classifieur ne sont pas fiables.

Une autre famille très usuelle est la sélection par comité. Son principe est d'apprendre non pas un mais plusieurs classifieurs à l'étape 1, de les appliquer à $\mathcal{N}$, et de sélectionner les exemples sur lesquels ils sont le plus en désaccord. Cette approche est souvent mise en œuvre par des techniques de *bagging* et/ou de *boosting* (Abe & Mamitsuka, 1998), ou par des représentations complémentaires des données sur lesquelles sont appris des classifieurs différents (Pierce & Cardie, 2001). En plus du coût calculatoire plus important généré par ces apprentissage multiples, ces techniques souffrent du même problème que la sélection par incertitude : les classifieurs sont peu fiables dans les premiers tours de l'itération avec $|\mathcal{T}|$ petit.

Une dernière famille usuelle est la sélection basée sur la modification attendue du modèle. Le principe est ici de sélectionner l'exemple qui impacterait le plus le modèle, en supposant que cet impact résulterait en une amélioration des performances. L'intuition sous-jacente est que l'exemple choisi couvre des cas non traités par les exemples de $\mathcal{T}$. La mise en œuvre de cette approche dépend beaucoup du classifieur utilisé. Settles & Craven (2008) a proposé plusieurs variantes de cette approche pour les CRF, dont seulement l'une, appelée *Information Density*, a donné quelques résultats positifs. Celle-ci repose simplement sur le choix de la séquence dans $\mathcal{N}$ la plus différente des séquences de $\mathcal{T}$. Pour évaluer cette différence, les auteurs représentent les séquences comme un vecteur des combinaisons d'attributs capturés par les fonctions caractéristiques. Les labels des séquences de $\mathcal{N}$ étant inconnus, il faut bien noter que ces sont les attributs sur x qui sont considérés. La séquence la plus dissimilaire est simplement définie comme celle ayant le cosinus moyen avec les séquences de $\mathcal{T}$ le plus faible.

Ces derniers travaux sont les plus proches de ceux que nous présentons dans cet article. Nous en reprenons d'ailleurs en partie la représentation des séquences, vues comme des ensembles d'attributs, bien que le critère que nous proposons se veut plus performant que celui proposé dans ces travaux (cf. section 4). Par ailleurs, la méthode d'évaluation utilisée par Settles & Craven (2008) ne rend pas compte correctement de l'effort d'annotation fourni à chaque itération : les auteurs évaluent les performances en fonction des séquences, sans considérer que certaines peuvent être beaucoup plus longues que d'autres. Pour notre part, l'effort d'annotation est mesuré en terme de mots annotés, ce qui a des conséquences sur les stratégies de sélection classiques testées par ces auteurs (cf. section suivante).

2.4 Contexte expérimental

Dans la suite de cet article, nous allons valider nos propositions de sélection des séquences sur différentes tâches pour lesquelles les CRF sont classiquement utilisés. Nous décrivons brièvement ces tâches et ces données ci-dessous ; pour plus de détails, le lecteur intéressé peut se reporter aux références indiquées.

Nous utilisons le jeu de données de la tâche de reconnaissance d'entités nommées de la campagne ESTER (Gravier *et al.*, 2005). Il contient des transcriptions d'émissions de radio en français, soit 55 000 groupes de souffle, dont les entités nommées sont annotées selon 8 classes (personne, lieu, temps...). Le jeu CoNLL2002 contient les données utilisées pour la tâche de reconnaissance d'entités nommées en néerlandais proposée dans le cadre de CoNLL 2002 (Tjong Kim Sang, 2002). Il contient 4 labels d'entités différents et nous utilisons 14 000 séquences (phrases) dans les expériences rapportées dans la suite de l'article. Le jeu CoNLL2000 est composé de textes de journaux en anglais annotés en chunks (Tjong Kim Sang & Buchholz, 2000). Il contient environ 11 000 phrases et 4 classes (3 types de chunks et un label 'autre'). Nous utilisons également les données de désambiguisation de sens de SensEval-2 (Edmonds & Cotton, 2001). Ce jeu de données porte sur la désambiguisation de *hard, line, serve, interest*, chacun des sens étant représenté par un label différent. Il contient environ 16 000 phrases. Une tâche un peu différente sur laquelle nous nous testons est celle de la phonétisation de mots isolés en anglais fournis par les données Nettalk. Le but est de transcrire ces mots dans un alphabet phonétique spécifique. Cette tâche est donc vue comme une tâche d'annotation lettre par lettre. On a ainsi 18 000 mots et 52 labels différents correspondant à l'alphabet phonétique. Une étape préliminaire des données a consisté à aligner les mots avec leur phonétisation et donc à introduire le cas échéant des symboles 'vide'.

Les données sont décrites de manière habituelle pour ces tâches, avec les parties du discours, lemmes, information sur la présence de majuscule, etc., et le schéma d'annotation BIO est adopté lorsque nécessaire (ESTER, CONLL2002, CONLL2000). Tous ces corpus de données ont été divisés en neuf dixièmes pour l'entraînement (ensemble $\mathcal{T}$ et $\mathcal{N}$) et

un dixième pour l'évaluation des performances. Dans la plupart des cas, la mesure de performance utilisée est le taux de précision par mot (label correct ou non), sauf pour la tâche de phonétisation, où c'est le taux de précision par séquence (le mot doit être entièrement et correctement phonétisé). Cette mesure est réalisée à chaque itération et rapportée à l'effort d'annotation, c'est-à-dire au nombre de mots auxquels l'expert a ajouté le label.

L'implémentation des CRF que nous utilisons est WAPITI (Lavergne *et al.*, 2010), avec ses paramètres par défaut sauf si indiqué autrement. Il convient de noter que des tests avec d'autres réglages (algorithmes d'optimisation, normalisation...), non rapportés dans l'article, ne modifient pas les conclusions présentées.

3 Sélection par incertitude

Comme nous l'avons vu, une solution classique pour la sélection des exemples à annoter à chaque itération est de proposer à l'oracle ceux pour lesquels le classifieur appris à l'issue de l'itération précédente est le moins sûr. Avec des CRF, cela se traduit par choisir la séquence x sur la base des probabilités $P(y|x;\theta)$.

3.1 Confiance minimale et entropie de séquence

Parmi les différentes façons de procéder, Settles & Craven (2008) montre que deux stratégies de cette famille obtiennent de bons résultats dans la plupart des cas. Il s'agit de la sélection par confiance minimale et de la sélection par entropie de séquence. La première consiste simplement à choisir dans $\mathcal{N}$ la séquence obtenant la probabilité minimale avec le modèle courant :

$$x = \underset{x \in \mathcal{N}}{\operatorname{argmin}} P(y^*|x,\theta)$$

La méthode par entropie consiste à choisir la séquence x de plus grande entropie sur l'ensemble des labels possibles y de cette séquence :

$$x = \underset{x \in \mathcal{N}}{\operatorname{argmax}} \left(-\sum_y P(y|x,\theta) \log P(y|x,\theta) \right)$$

3.2 Biais de longueur

L'un des problèmes de ces approches de l'état-de-l'art est qu'elles ont tendance à choisir les séquences les plus longues, celles-ci ayant des probabilités souvent plus faibles que des séquences courtes. Or, le coût d'annotation est proportionnel à la longueur des séquences, c'est donc un comportement potentiellement indésirable si l'on cherche à maximiser la performance pour un coût d'annotation minimal. Pour illustrer cela, nous reportons dans les figures 1 et 2 la longueur des séquences du jeu de données ESTER selon leur probabilité donnée par un modèle entraîné sur respectivement 20 et 10 000 séquences choisies aléatoirement. Dans les deux cas, on observe bien le biais attendu : en moyenne, la probabilité de la séquence donnée par le CRF est linéairement corrélée à sa longueur. Cela est notamment plus marqué quand le modèle est appris sur peu de séquences (figure de gauche). Or c'est justement le cas pour les premières itérations de l'apprentissage actif. Ce critère est donc particulièrement peu adapté en début d'apprentissage actif.

À l'inverse, une normalisation brutale par la longueur des séquences a tendance à privilégier les séquences très courtes n'apportant pas d'informations utiles à l'apprentissage.

3.3 Normalisation

Sur la base des constatations précédentes, il semble important de normaliser selon la longueur des séquences. Il serait possible d'utiliser les coefficients des droites de régression, mais comme on peut le constater sur les figures précédentes, celles-ci décrivent un comportement globale du nuage de points mais ne sont pas forcément adaptées pour décrire les points autour d'une longueur de séquence fixée.

Nous proposons à la place une méthode de normalisation adaptative plus locale, s'appuyant sur l'observation de la probabilité moyenne des séquences pour une longueur donnée. Cela revient à étudier le comportement du nuage de point, c'est-à-dire la répartition des probabilités, sur une tranche verticale des figures précédentes. Nous proposons pour cela

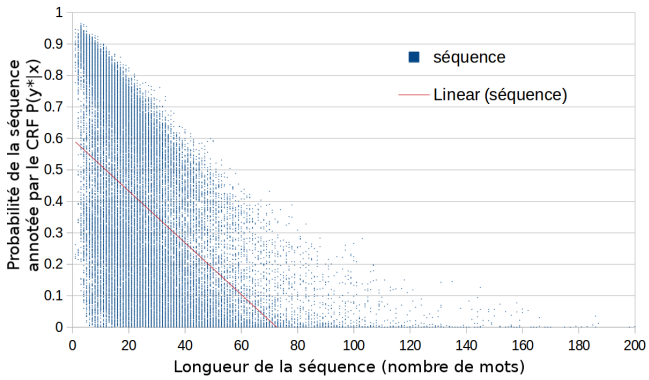

FIGURE 1 – Probabilités des séquences ($P(y^*|x)$) du jeu de données ESTER selon leur longueur obtenues avec un modèle appris sur 20 séquences, et droite de régression linéaire.

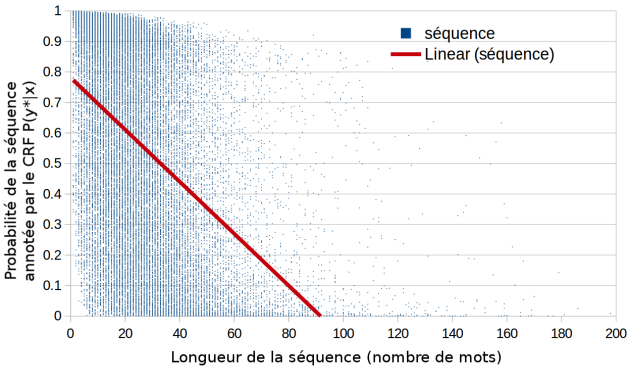

FIGURE 2 – Probabilités des séquences ($P(y^*|x)$) du jeu de données ESTER selon leur longueur, obtenues avec un modèle appris sur 10 000 séquences, et droite de régression linéaire.

une méthode de normalisation s'inspirant des méthodes d'estimation par fenêtres de Parzen (Parzen, 1962; Wasserman, 2005). L'idée sous-jacente est que pour une longueur de séquence fixée (ou à plus ou moins ϵ), les scores de probabilités devraient être distribués uniformément entre 0 et 1. Pour une séquence x de $\mathcal{N}$ de longueur l, nous estimons la moyenne $\hat{\mu}_l$ et l'écart-type $\hat{\sigma}_l$ des probabilités obtenues à cette itération sur toutes les séquences de $\mathcal{N}$ de longueur $l \pm \epsilon$, c'est-à-dire de l'ensemble $\{P(y'^*|x') \mid x' \in \mathcal{N}, |x'| = |x| \pm \epsilon\}$. Ces valeurs sont alors utilisées pour centrer et réduire les probabilités utilisées dans les stratégies de sélection précédentes. Par exemple, pour la sélection par confiance minimale, on a :

$$x = \underset{x \in \mathcal{N}}{\operatorname{argmin}} \left(\frac{P(y^*|x, \theta) - \hat{\mu}_l}{\hat{\sigma}_l} \right)$$

Cela doit ainsi permettre, pour chaque longueur de clause considérée, d'améliorer la dispersion des probabilités des séquences de cette longueur, et donc d'annuler le biais de longueur des séquences observé précédemment.

En pratique, dans les expériences rapportées en section 5, les séquences de longueur comparables ne sont pas trouvées à ϵ près mais par voisinage : la moyenne est calculée sur un nombre fixé de séquences dont la longueur s'approche le plus de celle visée. Cette approche inspirée de l'estimation par k-plus-proches voisins permet de traiter les cas de séquences *outlier* aux longueurs très différentes pour lesquelles un voisinage défini à ϵ près ne couvrirait aucune autre séquence.

4 Représentativité des fonctions caractéristiques

La proposition principale de cet article est de considérer que la distribution des attributs, tels que capturés par les fonctions caractéristiques, peut guider la sélection des exemples à faire annoter dans un cycle d'apprentissage actif. Pour étayer cette intuition, nous étudions tout d'abord dans la sous-section 4.1 comment ces attributs sont distribués en terme de fréquence et en terme d'utilisation dans les modèles. La sous-section 4.2 propose une méthode originale pour sélectionner les séquences à faire annoter en se basant sur ces considérations.

4.1 Étude préliminaire

Les fonctions caractéristiques encodent les relations entre la description des séquences et les classes. Il est intéressant d'en observer les fréquences d'apparition dans les données, mais aussi de voir quelles sont parmi elles les fonctions effectivement utilisées pour la prédiction. Pour cela, nous calculons la distribution des occurrences de toutes les fonctions caractéristiques constructibles sur les données ESTER, soit plus formellement :

$$occ(f_j) = |\{f_j(x^{(m)}, y_{t-1}^{(m)}, y_t^{(m)}, t) = 1| \text{ tout exemple } m, \text{ toute position } t\}|$$

Nous entraînons d'autre part également deux modèles sur l'ensemble des données ESTER (supervision complète) afin d'étudier les fonctions effectivement utilisées pour la prédiction dans les modèles. Pour chaque modèle, nous extrayons

donc de l'ensemble des fonctions caractéristiques celles ayant un poids $|\lambda_j| > 0$. Les paramètres d'apprentissage, notamment la stratégie de normalisation en L1 ou L2, influent beaucoup sur le nombre de fonctions caractéristiques de poids non nuls. Nous entraînons donc un modèle avec une normalisation L1 et un autre avec une normalisation *elastic-net* (mixant également L1 et L2). Nous rapportons dans la figure 3 ces trois distributions : celle de l'ensemble des fonctions caractéristiques possibles à partir des données et celles effectivement utilisées dans les modèles appris L1/L2 et L1 sur ces données.

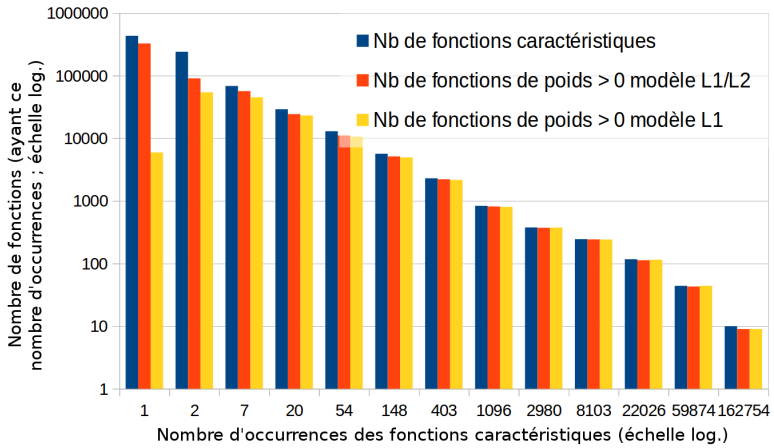

FIGURE 3 – Distribution des fonctions caractéristiques (nombre de fonctions selon leur nombre d'occurrences ; échelle logarithmique sur les deux axes) sur les données ESTER et distribution des fonctions utilisées dans le modèle.

On observe que ces trois distributions sont très similaires sauf pour les fonctions caractéristiques les plus rares, notamment avec le modèle L1. La plupart des combinaisons d'attributs/labels des données apparaissent donc comme utiles (car de poids $|\lambda_j| > 0$) pour les prédictions dans nos deux modèles. Cela signifie que les modèles CRF que nous utilisons exploitent une très grande majorité des combinaisons attributs/labels présentes dans les données, que ces combinaisons soient très fréquentes ou plus rares (à l'exception des très rares configurations pour les modèles L1), et de manière proportionnelle à leur fréquence dans les données. Pour construire un jeu d'entraînement plus petit mais menant à des modèles ayant des caractéristiques similaires, il semble important d'offrir le maximum de variété de configurations en respectant ces proportions, c'est-à-dire en respectant au mieux la distributions des combinaisons d'attributs/labels du jeu d'entraînement complet.

Dans le cas semi-supervisé, la majorité des données n'est pas annotée. Il est donc important de vérifier si les conclusions précédentes sont également vraies en ne regardant pas les labels. On examine donc la distribution des fonctions caractéristiques sans considération des labels, c'est-à-dire uniquement en regardant les attributs relevant de x. La figure 4 illustre ainsi la distribution non plus exactement des fonctions caractéristiques, mais de leurs occurrences quel que soit leur label, ce que l'on note f_j^*. Formellement, on a donc :

$$\text{occ}(f_j^*) = |\{f_j(x^{(m)}, y_1, y_2, t) = 1| \text{ tout exemple } m, \text{ toute position } t, \text{ toutes classes } y_1 \ \ y_2\}|$$

On observe les mêmes tendances que précédemment. Ces différentes expériences suggèrent l'importance d'avoir un jeu d'entraînement varié et représentatif de l'ensemble des combinaisons d'attributs définis par les fonctions caractéristiques. C'est sur la base de ce critère que nous proposons la stratégie de sélection présentée ci-après.

4.2 Test de proportion

À chaque itération de l'apprentissage actif, on souhaite avoir l'ensemble d'entraînement le plus représentatif des données que l'on va traiter. Par représentatif, on veut dire dont la distribution des séquences, telles que vues par le CRF via les fonctions caractéristiques, soit la plus proche de celles de $\mathcal{T} \cup \mathcal{N}$. On se place donc comme précédemment dans un cadre où chaque séquence est vue comme l'ensemble des fonctions caractéristiques qu'elle permet de générer, labels non compris.

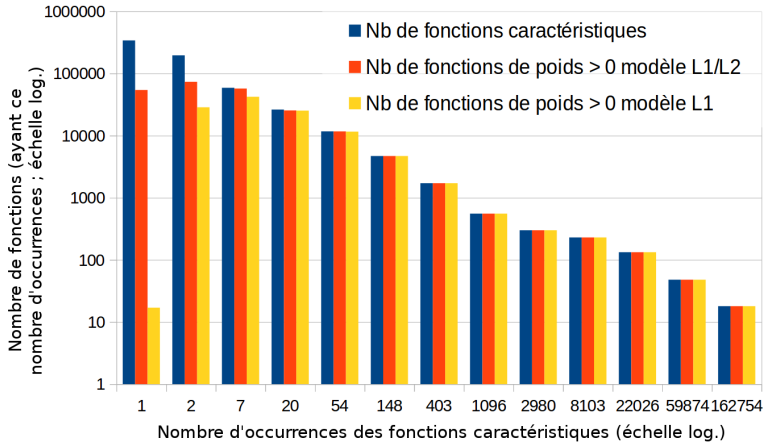

FIGURE 4 – Distribution des fonctions caractéristiques sans indication de label (nombre de fonctions selon leur nombre d'occurrences ; échelle logarithmique sur les deux axes) sur les données ESTER et distribution des fonctions utilisées dans le modèle sans indication de classe.

Pour choisir la séquence x à ajouter à l'ensemble d'entraînement à chaque itération (et donc à faire annoter par l'oracle), nous voulons donc mesurer combien le jeu d'entraînement résultant $\mathcal{T} \cup \{x\}$ se rapproche de l'ensemble des données à notre disposition (annotées ou non, i.e. $\mathcal{T} \cup \mathcal{N}$). Pour cela, pour chaque fonction caractéristique possible, nous proposons d'examiner simplement si la proportion de cette fonction observée dans l'échantillon $\mathcal{T} \cup \{x\}$ est comparable à celle de l'échantillon $\mathcal{T} \cup \mathcal{N}$. Ces échantillons ne sont pas indépendants, mais peuvent être considérés comme tel dès lors que $|\mathcal{N}| \gg |\mathcal{T}|$, ce qui est assuré dans tous les cas aux premières itérations de l'apprentissage actif.

Plus formellement, nous effectuons un test statistique de proportion entre les deux échantillons $\mathcal{T} \cup \{x\}$ et $\mathcal{T} \cup \mathcal{N}$, respectivement notés 1 et 2 et de taille n_1 et n_2. Soit $\hat{p}_1^j = r_1^j/n_1$ l'estimateur de proportion d'occurrences d'une certaine fonction caractéristique f_j apparaissant r_1^j fois dans l'échantillon 1, et $\hat{p}_2^j = r_2^j/n_2$ l'estimateur de proportion de la même fonction pour l'échantillon 2. On peut alors calculer le z-score suivant :

$$z_{j,x} = \frac{\hat{p}_1^j(f_j) - \hat{p}_2^j(f_j)}{\sqrt{\hat{p}^j * (1-\hat{p}^j) * (1/n_1 + 1/n_2)}} \quad \text{avec} \quad \hat{p}^j = \frac{r_1^j + r_2^j}{n_1 + n_2}$$

Ce z-score suit une loi normale centrée réduite, ce qui nous permet de calculer la probabilité $P(z_{j,x})$ d'observer une telle différence de proportion entre nos deux échantillons. Dans notre cadre, une probabilité élevée traduit intuitivement que l'échantillon 1 contient une proportion comparable à celle de l'échantillon 2 de la fonction caractéristique visée f_j.

Il faut bien sûr combiner ces probabilités pour toutes les fonctions caractéristiques. Pour cela, nous faisons une hypothèse simplificatrice qui est de considérer que les observations des fonctions caractéristiques sont indépendantes. Même s'il est évident que cette hypothèse est invalidée dans la plupart des cas, elle nous permet d'estimer simplement la probabilité globale de l'échantillon sur l'ensemble des fonctions caractéristiques comme le produit des $P(z_{j,x})$ pour toutes les fonctions caractéristiques f_j. Finalement, le choix de la séquence à ajouter à l'ensemble des séquences annotées est donc celle maximisant cette probabilité :

$$x^* = \underset{x \in \mathcal{N}}{\operatorname{argmax}} \prod_j P(z_{j,x})$$

5 Expérimentations

Dans cette section, nous comparons expérimentalement les différentes stratégies évoquées de sélection des exemples pour l'apprentissage actif. Celles-ci sont rappelées ci-dessous, et les courbes d'apprentissage obtenues sont présentées dans la sous-section 5.2.

5.1 Contexte

Les stratégies de sélection de que nous testons sont d'une part celles de la littérature, qui nous servent ainsi de point de comparaison. Il s'agit de la sélection par confiance minimale, entropie, et *information density*. Nous ajoutons également une stratégie *baseline* consistant à choisir les séquences au hasard (*random*). Nous testons d'autre part également notre stratégie de normalisation sur la confiance minimale (confiance minimale normalisée) et la méthode basée sur la proportion. Nous ne rapportons pas de résultats avec des techniques de sélection par comité, celles-ci obtenant des résultats plus faibles que les précédentes dans la quasi-totalité des cas (Settles & Craven, 2008).

Toutes ces méthodes sont testées dans les mêmes conditions (paramètres du CRF, patrons...). À l'initialisation, une séquence est tirée au hasard pour servir de premier exemple (le même pour toutes les méthodes de sélection). À chaque itération, un unique exemple est choisi pour être annoté et le classifieur est ré-entraîné sur l'ensemble des données annotées (il ne s'agit donc pas d'une mise à jour du CRF).

5.2 Résultats

Les figures 5, 6, 7 et 8 présentent les courbes d'apprentissage sur nos différents jeux de données. La performance des classifieurs appris à chaque itération est donc exprimée en fonction du coût de l'annotation cumulé de l'ensemble $\mathcal{T}$. Dans les figures, ce coût est rapporté sur une échelle logarithmique qui permet de bien apprécier les différents cas (peu d'annotations vs. beaucoup d'annotations).

Plusieurs observations en ressortent. D'une part, ces courbes ont des allures très différentes d'un jeu de données à l'autre. Cela s'explique par les caractéristiques des tâches et des données, impliquant que certaines soient plus facilement faisables avec de bonnes performances en peu d'annotations (CoNLL2000) ou non (CoNLL2002). On note au passage que pour certaines tâches, l'allure des courbes laisse penser que plus d'exemples améliorerait encore les performances ; les conclusions de (Garrette & Baldridge, 2013) ne sont onc pas à généraliser. Pour tous les jeux de données sauf Nettalk, les différences observées, notamment lorsque le coût d'annotation est petit, sont sensibles. Concernant Nettalk, il est plus difficile de faire ressortir une méthode de sélection meilleure que les autres. Cela s'explique certainement par la difficulté de la tâche due notamment au très grand nombre de labels possibles, et donc au très grand nombre de configurations attributs/labels possibles qui nécessite dans tous les cas un nombre extrêmement important d'exemples pour couvrir toutes ces configurations.

Deuxièmement, on observe que les trois stratégies de la littérature offrent des performances moyennes, parfois peu éloignées de la stratégie *random*. Les stratégies de confiance minimale et entropie sont même parfois nettement en deçà du hasard (SenseEval-2), visiblement pénalisées par leurs biais discutés en section 3. Ce point est important à noter puisqu'il est souvent occulté par les évaluations ne prenant en compte que le nombre de séquences, comme nous l'avons déjà souligné pour le travail de (Settles & Craven, 2008).

Troisièmement, on constate le bien fondé de notre proposition de normalisation puisque cette stratégie nous permet d'obtenir des résultats meilleurs ou identiques à la version non normalisée. Elle obtient notamment les meilleurs résultats lorsque le nombre d'annotation est important (ESTER, CoNLL2002, SensEval-2), même si l'échelle logarithmique cache ici un peu cette longue domination.

Enfin, notre proposition de sélection basée sur le respect des proportions obtient de très bons résultats dans nos différents cas d'étude. Elle se comporte globalement mieux que les autres techniques de sélection, y compris l'*information density* dont elle se rapproche conceptuellement. On peut noter que notre stratégie apporte un gain notable lorsque le coût d'annotation considéré est petit. Ce résultat attendu s'explique par le fait que la méthode ne repose pas sur les prédictions, peu fiables à ce stade, du classifieur courant. En revanche, ce gain est moindre voire nul par rapport aux autres méthodes lorsque la quantité de données annotées devient très importante. Cela montre la limite de notre approche qui n'exploite aucune information issue du classifieur, mais permet aussi d'imaginer des stratégies mixtes dans lesquelles ces informations de classification seraient également exploitées à d'un certain nombre d'annotations.

6 Conclusions

À l'heure où la plupart des problèmes du TAL sont exprimés en tâches d'apprentissage supervisé, le coût des annotations des exemples par l'expert représente un problème important. L'apprentissage actif apporte un cadre à ce problème, per-

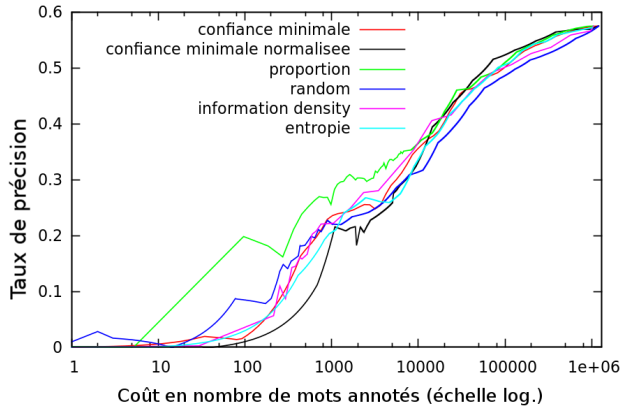

FIGURE 5 – Courbe d'apprentissage sur les données ESTER : taux de précision selon le coût d'annotation en mots (échelle log)

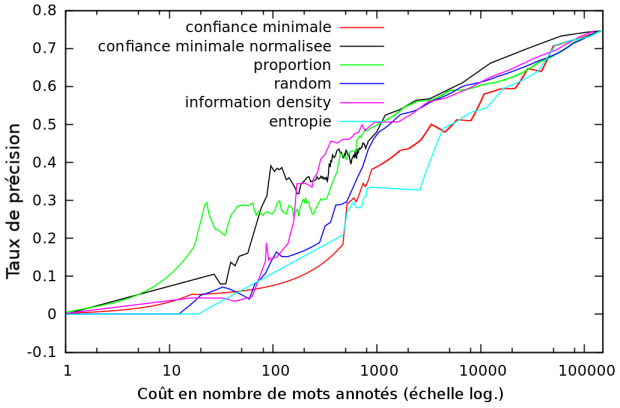

FIGURE 6 – Courbe d'apprentissage sur les données CoNLL2002 : taux de précision selon le coût d'annotation en mots (échelle log)

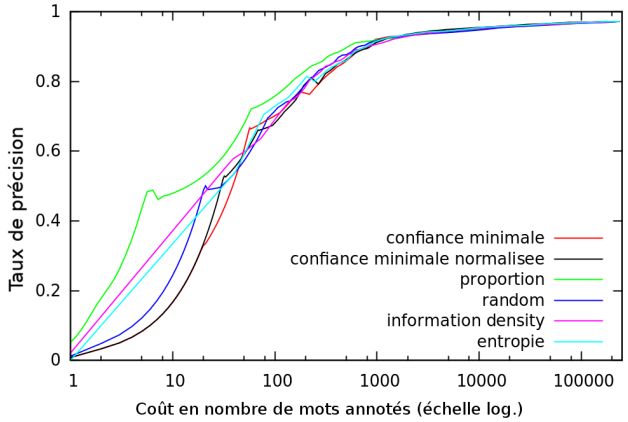

FIGURE 7 – Courbe d'apprentissage sur les données CoNLL2000 : taux de précision selon le coût d'annotation en mots (échelle log)

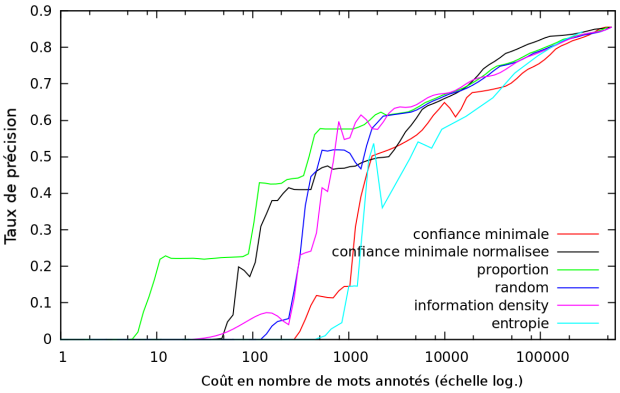

FIGURE 8 – Courbe d'apprentissage sur les données SensEval-2 : taux de précision selon le coût d'annotation en mots (échelle log)

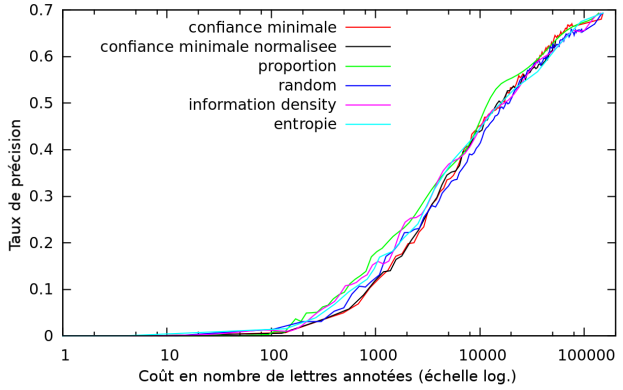

FIGURE 9 – Courbe d'apprentissage sur les données Nettalk : taux de précision (mots correctement phonétisés) selon le coût d'annotation en nombre de lettres (échelle log)

mettant de contrôler le coût d'annotation tout en maximisant, on l'espère, la performance de la tâche visée. Comme nous l'avons vu, cela est en fait largement dépendant de la stratégie de sélection des exemples. Dans cet article, nous avons examiné quelques unes de ces stratégies pour lesquelles nous avons mis en évidence un biais dégradant le ratio coût d'annotation/performance. La normalisation que nous avons proposée permet de lever ce biais de manière très simple tout en offrant un gain de performance notable. Lorsque les coûts d'annotation sont limités, la nouvelle stratégie que nous avons proposée, s'appuyant sur un critère original de proportionalité, s'avère la plus avantageuse.

Bien sûr, beaucoup de variantes, d'améliorations et de pistes de recherche sont envisageables. Parmi celles-ci, nous souhaitons essayer de prendre en compte la dépendance entre les fonctions caractéristiques. Dans notre proposition actuelle, elles sont abusivement considérées comme indépendantes, ce qui n'est jamais le cas en pratique. Ces dépendances peuvent même être très importantes puisque les patrons permettant de construire les fonctions caractéristiques font souvent appel plusieurs fois aux mêmes éléments (lemme du mot courant, PoS du mot courant...) et que ces éléments sont eux-mêmes en relation de dépendance. Tout ceci peut donc fortement impacter l'estimation de nos probabilités et fausser finalement le choix de l'exemple.

Une autre piste prometteuse est de mélanger ces différentes techniques de sélection pour en combiner les avantages. Il est bien sûr possible de les intégrer simplement (vote, produit des scores ou des rangs...), mais il nous semble plus intéressant de viser des combinaisons plus complexes, que l'on pourrait par exemple obtenir avec des techniques d'apprentissage d'ordre (*learning to rank*) (Liu, 2009).

Enfin, dans notre cadre actuel, les séquences sélectionnées sont annotées complètement. Il serait intéressant d'étudier le cas des annotations partielles, avec les mêmes contraintes d'optimisation du ratio coût/performance, en nous inspirant par exemple des travaux de Salakhutdinov *et al.* (2003).

Références

ABE N. & MAMITSUKA H. (1998). Query learning strategies using boosting and bagging. In *Proceedings of the Fifteenth International Conference on Machine Learning*, Madison, Wisconsin, USA : Morgan Kaufmann Publishers Inc.

ANDO R. K. & ZHANG T. (2005). A high-performance semi-supervised learning method for text chunking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, p. 1–9, Stroudsburg, PA, USA : Association for Computational Linguistics.

CONSTANT M., TELLIER I., DUCHIER D., DUPONT Y., SIGOGNE A. & BILLOT S. (2011). Intégrer des connaissances linguistiques dans un CRF : Application à l'apprentissage d'un segmenteur-étiqueteur du français. In *Traitement Automatique du Langage Naturel (TALN'11)*, Montpellier, France.

EDMONDS P. & COTTON S. (2001). Senseval-2 : Overview. In *Proceedings of SENSEVAL-2 Second International Workshop on Evaluating Word Sense Disambiguation Systems*, p. 1–5 : Association for Computational Linguistics.

FREITAG D. (2004). Trained named entity recognition using distributional clusters. In *Proceedings of the conference EMNLP*.

GARRETTE D. & BALDRIDGE J. (2013). Learning a part-of-speech tagger from two hours of annotation. p. 138–147.

GRAVIER G., BONASTRE J.-F., GEOFFROIS E., GALLIANO S., TAIT K. M. & CHOUKRI K. (2005). ESTER, une campagne d'évaluation des systèmes d'indexation automatique. In *Actes des Journées d'Étude sur la Parole, JEP, Atelier ESTER2*.

LAFFERTY J., MCCALLUM A. & PEREIRA F. (2001). Conditional random fields : Probabilistic models for segmenting and labeling sequence data. In *International Conference on Machine Learning (ICML)*.

LAVERGNE T., CAPPÉ O. & YVON F. (2010). Practical very large scale CRFs. In *Proceedings the 48th Annual Meeting of the Association for Computational Linguistics (ACL)*, p. 504–513 : Association for Computational Linguistics.

LIU T.-Y. (2009). Learning to rank for information retrieval. *Foundations and Trends in Information Retrieval*, **3**(3), 225–331.

MANN G. S. & MCCALLUM A. (2008). Generalized expectation criteria for semi-supervised learning of conditional random fields. In *Proceedings of ACL-08 : HLT*, p. 870–878, Colombus, Ohio, USA.

MILLER S., GUINNESS J. & ZAMANIAN A. (2004). Name tagging with word clusters and discriminative training. In *Procedings of the conference ACL*.

OLSSON F. (2009). *A literature survey of active machine learning in the context of natural language processing*. Rapport interne Swedish Institute of Computer Science, Swedish Institute of Computer Science.

PARZEN E. (1962). On estimation of a probability density function and mode. *Ann. Math. Stat.*, **33**, 1065–1076.

PIERCE D. & CARDIE C. (2001). Limitations of co-training for natural language learning from large datasets. In *Proceedings of the 2001 Conference on Empirical Methods in Natural Language Processing (EMNLP 2001)*, Pittsburgh, Pennsylvania, USA.

PRANJAL A., DELIP R. & BALARAMAN R. (2006). Part Of speech Tagging and Chunking with HMM and CRF. In *Proceedings of NLP Association of India (NLPAI) Machine Learning Contest*.

RAYMOND C. & FAYOLLE J. (2010). Reconnaissance robuste d'entités nommées sur de la parole transcrite automatiquement. In *Actes de la conférence Traitement Automatique des Langues Naturelles*, Montréal, Canada.

SALAKHUTDINOV R., ROWEIS S. & GHAHRAMANI Z. (2003). Optimization with EM and Expectation-Conjugate-Gradient. In *Proceedings of the conference ICML*.

SCHRAUDOLPH N. N., YU J. & GÜNTER S. (2007). A stochastic quasi-Newton method for online convex optimization. In *Proceedings of 11th International Conference on Artificial Intelligence and Statistics*, volume 2 of *Workshop and Conference Proceedings*, p. 436–443, San Juan, Puerto Rico.

SETTLES B. (2010). *Active Learning Literature Survey*. Computer sciences technical report 1648, University of Wisconsin–Madison.

SETTLES B. & CRAVEN M. (2008). An analysis of active learning strategies for sequence labeling tasks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, p. 1069–1078 : ACL Press.

SMITH N. & EISNER J. (2005). Contrastive estimation : Training log-linear models on unlabeled data. In *Proceedings of ACL*.

TJONG KIM SANG E. F. (2002). Introduction to the conll-2002 shared task : Language-independent named entity recognition. In *Proceedings of CoNLL-2002*, p. 155–158 : Taipei, Taiwan.

TJONG KIM SANG E. F. & BUCHHOLZ S. (2000). Introduction to the conll-2000 shared task : Chunking. In C. CARDIE, W. DAELEMANS, C. NEDELLEC & E. TJONG KIM SANG, Eds., *Proceedings of CoNLL-2000 and LLL-2000*, p. 127–132 : Lisbon, Portugal.

WANG T., LI J., DIAO Q., WEI HU Y. Z. & DULONG C. (2006). Semantic event detection using conditional random fields. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW '06)*.

WASSERMAN L. (2005). *All of Statistics : A Concise Course in Statistical Inference*. Springer Texts in Statistics.