Fouille de règles d'annotation partielles pour la reconnaissance des entités nommées

Damien Nouvel [1, 2] Jean-Yves Antoine [1] Nathalie.Friburger [1]
Arnaud.Soulet [1]
(1) LI, 3 place Jean Jaurès, 41000 Blois
(2) Alpage, INRIA & Université Paris-Diderot, 75013 Paris
{prenom.nom}@univ-tours.fr

RÉSUMÉ

Ces dernières décennies, l'accroissement des volumes de données a rendu disponible une diversité toujours plus importante de types de contenus échangés (texte, image, audio, vidéo, SMS, tweet, données statistiques, spatiales, etc.). En conséquence, de nouvelles problématiques ont vu le jour, dont la recherche d'information au sein de données potentiellement bruitées. Dans cet article, nous nous penchons sur la reconnaissance d'entités nommées au sein de transcriptions (manuelles ou automatiques) d'émissions radiodiffusées et télévisuelles. À cet effet, nous mettons en œuvre une approche originale par fouille de données afin d'extraire des motifs, que nous nommons règles d'annotation. Au sein d'un modèle, ces règles réalisent l'annotation automatique de transcriptions. Dans le cadre de la campagne d'évaluation Etape, nous mettons à l'épreuve le système implémenté, mXS, étudions les règles extraites et rapportons les performances du système. Il obtient de bonnes performances, en particulier lorsque les transcriptions sont bruitées.

ABSTRACT

Mining Partial Annotation Rules for Named Entity Recognition

During the last decades, the unremitting increase of numeric data available has led to a more and more urgent need for efficient solution of information retrieval (IR). This paper concerns a problematic of first importance for the IR on linguistic data : the recognition of named entities (NE) on speech transcripts issued from radio or TV broadcasts. We present an original approach for named entity recognition which is based on data mining techniques. More precisely, we propose to adapt hierarchical sequence mining techniques to extract automatically from annotated corpora intelligible rules of NE detection. This research was carried out in the framework of the Etape NER evaluation campaign, where mXS, our text-mining based system has shown good performances challenging the best symbolic or data-driven systems

MOTS-CLÉS : Entités nommées, Fouille de données, Règles d'annotation.

KEYWORDS: Named Entities, Data Mining, Annotation Rules.

1 Introduction

Ces dernières décennies, le développement considérable des technologies de l'information et de la communication a modifié la manière dont nous accédons et manipulons les connaissances. Nous constatons une diversité toujours plus importante des types de contenus échangés (texte, image, audio, vidéo, SMS, tweet, données statistiques, spatiales, etc.), ce qui nécessite de résoudre de nombreuses problématiques, parmi lesquelles la recherche d'information, qui a intéressé la communauté du TALN dès les années 90 avec les campagnes d'évaluation MUC (Grishman et Sundheim, 1996). Les travaux sur le sujet ont porté une attention particulière aux noms propres de personnes, de lieux et d'organisations, appelés entités nommées (EN). Au gré des besoins, celles-ci ont été étendues aux dates, aux expressions numériques, aux marques ou aux fonctions, avant de recouvrir un large spectre d'expressions linguistiques.

De nombreux systèmes ont été élaborés pour réaliser la reconnaissance d'entités nommées (REN), selon des approches orientées connaissances ou orientées données. Les premières ont généralement une grande précision mais nécessitent un coup humain de développement important, ce qui se traduit généralement par une couverture (et donc un rappel) perfectible. Les approches orientées données, par ajustement automatique de paramètres d'un modèle numérique, permettent d'obtenir de bonnes performances, avec un coup d'entrée limité, du moment où l'on dispose d'une base d'apprentissage de taille suffisante. Ils sont également réputés présenter une dégradation graduelle de leurs performances sur des données bruitées. Cependant, l'aspect "boîte noire" des algorithmes d'apprentissage rend difficile l'amélioration ciblée de leurs performances.

Ces constats ont été vérifiés par de nombreuses campagnes d'évaluation. À titre d'exemple, lors de la campagne d'évaluation francophone Ester2 (Galliano *et al.*, 2009), portant sur le traitement de transcriptions de parole radio ou télédiffusée, les deux meilleurs systèmes travaillant sur des transcriptions manuelles étaient des systèmes à base de connaissance, tandis que les tests effectués sur des sorties de reconnaissance de la parole ont été dominés par un système orienté données.

Les travaux que nous présentons dans cet article ont été menés dans le cadre de la campagne Etape (qui a fait suite à Ester2) qui visait notamment à évaluer des systèmes de REN sur des flux de parole conversationnelle. Nous y proposons une approche novatrice pour la REN : l'utilisation de méthodes de fouille de données séquentielle hiérarchique. À nos yeux, ces travaux présentent plusieurs originalités du point de vue du TALN :

(i) nous élaborons un moyen-terme entre les approches orientées données et orientées connaissances reposant sur la recherche, à partir de données d'apprentissage, de motifs pour la REN : cette technique centrée données permet l'extraction de connaissances interprétables ;

(ii) la stratégie de détection des entités nommées est originale, par la recherche séparée du début et de la fin des entités, en nous appuyant sur le contexte immédiat pour placer les balises d'annotation : cela présente l'intérêt de conserver une certaine robustesse en cas de disfluence ou d'erreur de reconnaissance au sein de l'entité nommée.

Cet article porte sur l'élaboration, l'implémentation et l'évaluation d'une telle approche. En partie 2, nous faisons un état de l'art des approches pour la REN. La partie 3 présente le formalisme de fouille pour l'extraction de règles d'annotation et leur utilisation pour reconnaître des entités nommées. En partie 4, nous décrivons le jeu de données utilisé et les résultats obtenus lors de l'évaluation dans le cadre de la campagne Etape.

2 Approches pour la reconnaissance d'EN structurées

2.1 Approches orientées connaissances

Les approches orientées connaissances sont basées sur la description de règles décrivant les entités nommées et leur contexte à l'aide d'indices linguistiques fournis par le texte lui-même et des ressources externes (dictionnaires). Généralement, les textes sont étiquetés syntaxiquement (éventuellement sémantiquement) grâce aux dictionnaires, puis un ensemble de règles, qui prennent en compte les indices morphologiques (présence de majuscule, ponctuation), morpho-syntaxiques et sémantique, permettent de repérer les ENs. Les règles utilisent ces éléments, soit comme preuves internes de la présence d'une entité nommée, soit par description de son contexte d'apparition (McDonald, 1996; Friburger et Maurel, 2004). Une preuve interne sera, par exemple, la présence d'un prénom avant un mot commençant par une majuscule ; ce prénom indiquera un nom de personne (ex : 'François Hollande'). Nous voyons que c'est la "connaissance" qui guide cette approche, celle de l'expert qui créé les règles, selon les informations à sa disposition (dont les ressources externes).

Dès les années 1990, un certain nombre de systèmes (Stephens, 1993; Hobbs J. R. et Tyson, 1996) mettent en œuvre cette approche orientée connaissances. Les automates sont particulièrement adaptés à l'élaboration et l'utilisation des règles. De plus, l'utilisation de transducteurs [1] permet de produire très intuitivement une annotation à l'aide de *balises* ('<pers>', '</pers>', '<org>', '</org>', etc.), ils sont donc largement utilisés pour ce type de tâche (Friburger et Maurel, 2004; Brun et Ehrmann, 2010; Béchet *et al.*, 2011). Enfin, les transducteurs peuvent être organisés sous forme de cascades, chaque transducteur permettant de lever des ambiguïtés et de mettre à disposition des reconnaissances pour les transducteurs suivants (ce qui permet de reconnaître des imbrications). L'ordre dans lequel sont appliqués les transducteurs a alors une grande importance.

Étant donné les traitements qu'elles mettent en œuvre, les approches orientées connaissances insèrent au sein des *séquences de mots* ce que nous appelons des *marqueurs*, comme le montre la figure 1 pour l'expression `'fondation Cartier'`.

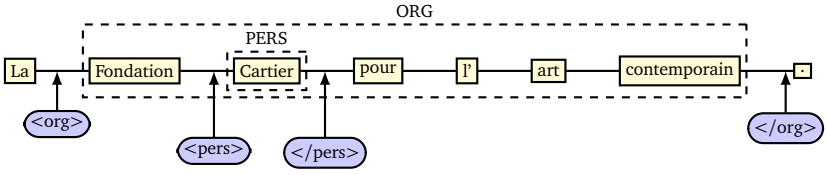


FIGURE 1 – Annotation par balises

Les approches orientées connaissances peuvent être utilisées et adaptées à des textes sans apprentissage préalable. Leur limitation est liée au fait que les ressources utilisées sont rarement exhaustives (par exemple, les noms propres forment une classe "ouverte") : il semble illusoire de bâtir ce type d'approche sur l'hypothèse d'un lexique complet des entités nommées existantes.

1. Automates qui modifient le texte fourni en entrée par insertion de balises

2.2 Approches orientées données

Les approches orientées données paramètrent un modèle automatiquement grâce à un apprentissage sur un corpus d'entraînement. Ce corpus d'entrainement, créé par des experts, fournit de nombreux exemples de données : le système apprend sur ces exemples puis prédit l'étiquette d'une nouvelle donnée, selon son modèle. Le corpus d'entrainement est constitué d'un ensemble de textes annotés en entités nommées par des experts. L'apprentissage automatique sera chargé d'ajuster les paramètres disponibles, cette procédure étant guidée à chaque itération par les erreurs que commet le système sur les jeux de données disponibles. Une fois l'apprentissage réalisé, le système est en mesure d'annoter de nouveaux textes en entités nommées selon les paramètres de son modèle. Traditionnellement, l'apprentissage automatique se rapproche plutôt d'une classification (attribution d'une classe à un mot) que d'une annotation (délimitation d'une expression linguistique).

Pour la REN, le format BIO [2] s'est imposé. La figure 2 présente la classification par mots réalisée pour l'énoncé '`<org> fondation <pers> Cartier </pers> </org>`'. Signalons qu'en partie 3, nous présentons une approche orientée donnée, mais qui est apparentée à un mécanisme de transduction (à l'aide d'indices locaux) plutôt que de classification.

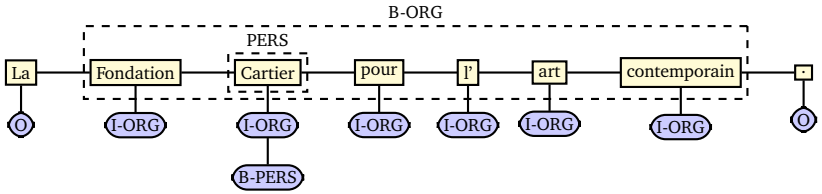


FIGURE 2 – Annotation par classification

Généralement, ces approches estiment la probabilité des classes selon les tokens et les informations qui y sont associées. Parmi les modèles numériques adaptés, figurent les modèles bayésiens, la régression logistique (ou maximum d'entropie), les machines à vecteur de support (SVM) , etc. La régression logistique a démontré son efficacité pour la reconnaissance d'entités nommées (Mikheev *et al.*, 1999; Ekbala *et al.*, 2010), permettant de prendre en compte de multiples traits discriminants (morphologiques, morpho-syntaxiques, lexicaux) interdépendants. D'autres modèles tirent parti de la séquentialité, comme les HMM (Bikel *et al.*, 1999), par modélisation des transitions entre états (types d'entités nommées) et des générations d'observations (mots).

Pour prendre en compte simultanément la multiplicité des indices locaux et les aspects séquentiels au sein d'un modèle unifié, les MEMM [3] (McCallum *et al.*, 2000) puis les CRF [4] (Raymond et Fayolle, 2010; Zidouni *et al.*, 2010) sont les modèles réputés les plus adéquats à ce jour. L'inconvénient est qu'ils restent difficiles à interpréter : les traits découverts sont généralement composites et exhibent des dépendances complexes dont il est difficile d'affirmer qu'elles sont nécessaires ou suffisantes pour déterminer les entités nommées.

A ce jour, les approches orientées données se basent majoritairement sur une représentation "plate" des entités nommées. Comme nous le verrons en partie 4, nous cherchons à réaliser la

2. Begin, Inside, Outside
3. Modèles markoviens à maximum d'entropie
4. Champs aléatoires conditionnels

REN structurée (avec imbrications). Notons que quelques travaux (Finkel et Manning, 2005; Dinarelli et Rosset, 2011) ont adapté avec un certain succès des méthodes orientées données à la reconnaissance de structure.

De manière générale, nous remarquons que les approches automatiques nécessitent un travail préalable conséquent (préparation des jeux de données, implémentation du modèle, des procédures d'apprentissage et d'estimation, sélection des traits et dépendances pertinents, etc.) avant d'être en mesure de paramétrer les modèles, et qu'il reste difficile de les utiliser pour extraire des connaissances ou pour étudier des phénomènes particuliers.

2.3 Proposition : les marqueurs d'annotation

Nous le voyons, les approches guidées par les données s'appuient sur des indices locaux variés. La nature "locale" de la structuration en entités nommées est alors un atout. Les systèmes orientés connaissances ont l'avantage de modéliser la structure interne des entités nommées. Ainsi un système à base de connaissances aura plus de facilité à analyser l'encapsulation d'entités nommées comme dans l'exemple suivant (issu d'Etape) : '*le député UMP de Haute-Saône*' où l'entité nommée globale est construite à l'aide de l'entité '*UMP*', de type organisation, et de l'entité '*Haute-Saône*', de type division géographique administrative.

Cependant, ces dernières approches utilisent une connaissance dont la construction est coûteuse et délicate. Aussi avons-nous souhaité développer une approche permettant l'extraction automatique sur corpus de motifs se rapprochant des règles de reconnaissance mises en œuvre par la REN symbolique. La fouille hiérarchique séquentielle de données est adéquate à cet effet.

Par ailleurs, les systèmes orientés connaissances sont aujourd'hui contraints à modéliser intégralement la structure des entités, voire de ses contextes d'introduction. Ce choix est discutable et met à l'épreuve la robustesse des systèmes lorsqu'ils traitent de la parole spontanée. Une erreur de reconnaissance sur un seul mot de l'entité (dûe par exemple à une disfluence) empêche l'application de la règle de détection.

Afin de répondre à cette insuffisance, nous proposons de **séparer la détection du début et de la fin de l'entité**, pour ensuite chercher à associer une marque de début et de fin d'entité. Notre hypothèse est que l'on dispose de suffisamment d'indices locaux pour caractériser précisément le début ou la fin d'une entité.

Considérons pas exemple l'énoncé annoté suivant '*En <date> <num> 1969 </num> </date> <pers> <prenom> Georges </prenom> <famille> Pompidou </famille> </pers> dirige la <org> <loc> France </loc> </org>*'. Notre hypothèse est que chacune des marques d'annotation ('*<pers>*', '*<prenom>*', '*</prenom>*', '*</pers>*', etc.) est détectable séparément. De plus, la détection d'une entité encapsulée telle que '*<prenom>*' peut guider la détection de l'entité englobante. Il s'agira, pour le système, d'extraire des règles d'annotation, d'estimer localement les marqueurs probables, puis de déterminer, par leurs combinaisons, l'annotation la plus vraisemblable. Nous implémentons un système de reconnaissance d'entités nommées, mXS, selon cette approche originale. Grâce à ce procédé, notre système reconnaît par exemple le montant '*deux cent ça compte mille*' (erreur de transcription pour *deux cent cinquante mille*), alors qu'un système symbolique sera mis en difficulté.

3 Extraction de règles d'annotation pour la REN

3.1 Enrichissement ambigu des données

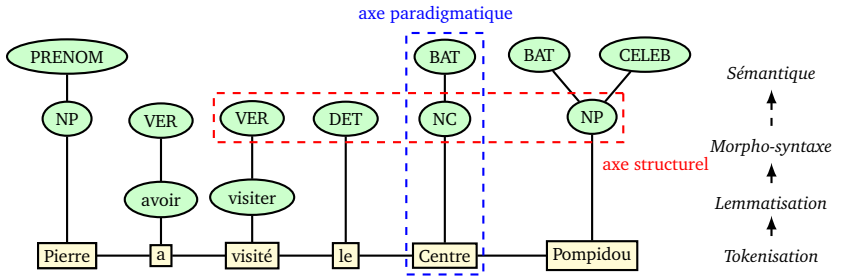


FIGURE 3 – Représentation des structures à fouiller

L'approche que nous mettons en œuvre repose sur des analyses fréquemment conduites pour traiter le langage naturel (morpho-syntaxe, lexiques). Pour la fouille, ces traitements sont interprétés comme autant d'*enrichissements* des données, à utiliser pour rechercher des motifs généralisés dans les données. La figure 3 présente de manière schématique, sur l'exemple *'Pierre a visité le Centre Georges Pompidou'*, la manière dont se superposent ces enrichissements.

La fouille de données devra alors tenir compte de deux axes : *paradigmatique*, pour la superposition d'enrichissements, et *structurel*, pour l'examen des contigüités entre items. Comme nous le verrons par la suite, ce processus est flexible : les enrichissements peuvent être plus ou moins profonds selon les éléments considérés. Nous pouvons moduler à volonté l'axe paradigmatique selon les éléments observés et la tâche d'annotation à réaliser.

3.1.1 Morpho-syntaxe

Nous réalisons conjointement la tokenisation, la lemmatisation et l'étiquetage morpho-syntaxique avec `TreeTagger` (Schmid, 1994). De surcroît, nous en adaptons la sortie comme suit :

- **Déterminants** : les déterminants définis (*'le'*, *'la'*, *'les'*, *'l'*') sont sous-catégorisés en `'DET/DEF'`.
- **Prépositions** : la sous-catégorie `'PRP:det'` (*'au'*, *'du'*, *'des'*) forme une catégorie `'PRPDET'`.
- **Nombres** : les nombres sont sous-catégorisés selon leur nombre de chiffres [5].
- **Noms propres et abréviations** : ces deux catégories se généralisent en `'NAMABR'`.
- **Nom propres, abréviations, noms, verbes** : ces éléments sont sous-catégorisés par le suffixe des trois derniers caractères (`'NOM/SUFF:ier'`, `'NAMABR/NAM/SUFF:ges'`, `'VER/SUFF:vre'`).
- **Verbes** : les sous-catégories relatives au mode et temps du verbe sont supprimées.

Pour le processus de fouille de données, nous omettons les variations surfaciques (majuscules) et flexionnelles (déclinaisons et conjugaisons) : nous ne conservons pas les items lexicaux eux-mêmes et faisons reposer la recherche de motifs sur les lemmes proposés par `TreeTagger`. Par exemple, le *'En 1970 les socialistes [...]'* donnera la séquence :

`'PRP/en NUM/DIGITS:4/PREF:19/1970 DET/DEF/le NOM/SUFF:ste/socialiste'`.

5. Ce nombre est précisé s'il est inférieur ou égal à quatre le préfixe est utilisé dans ce dernier cas : `'NUM/DIGITS:MANY'`, `'NUM/DIGITS:4/PREF:20'` ..., `'NUM/DIGITS:1'`)

3.1.2 Lexiques

Les lexiques nous permettent d'ajouter un niveau sémantique aux hiérarchies. Nous exploitons des ressources diverses, dont certaines sont importées à partir des dictionnaires et motifs du système CasEN [6]. Nous y ajoutons quelques listes, constituées manuellement, en particulier pour les fonctions, lieux, organisations, quantités et dates. Ces ressources contiennent 221 547 expressions distinctes qui produisent 443 112 catégorisations sémantiques [7]. Une large part est dédiée à la reconnaissance des personnes et des lieux. Signalons qu'une partie de ces ressources est générée à partir d'automates (transducteurs CasEN) qui reconnaissent des expressions linguistiques utiles à la REN.

Ces ressources sont utilisées telles quelles pour produire les enrichissements. Ceux-ci peuvent alors être sémantiquement ambigus, ce que nous notons comme une disjonction exclusive $\oplus$. Par exemple, au nom propre *Washington* seront affectées les catégories sémantiques `'CELEB⊕TOPO⊕ORG-LOC-GOV⊕PREN⊕VILLE'`. Notons ici que nous considérons que les noms propres forment une classe ouverte et qu'ils n'ont pas vocation à être utilisés lexicalisés au sein des motifs extraits : lorsqu'ils ont donné lieu à des enrichissements sémantiques, les items lexicaux sont omis afin que la fouille de données ne repose que sur les catégories sémantiques.

3.2 Exploration de règles d'annotation de segments

Les données ainsi enrichies forment le langage $\mathscr{L}_r$ et ont vocation a être fouillées afin d'y rechercher des motifs séquentiels d'intérêt (Fischer *et al.*, 2005; Cellier et Charnois, 2010) pour la REN.

Le langage des motifs $\mathscr{L}_{p^+}$ comprend celui des données enrichies et toutes leurs généralisations. Un élément de motif (item) couvre une donnée, notée $\leq_{ci}$, lorsqu'il s'y trouve en tenant compte des disjonctions $\oplus$. Par exemple, l'item `'TOPO/Washington'` couvre la donnée enrichie `'CELEB/Washington⊕TOPO/Washington'`. Dès lors, nous nous inspirons de travaux intégrant des hiérarchies aux séquences (Srikant et Agrawal, 1996), en y ajoutant la notion de segment [8] particulièrement adaptée au traitement de structures au sein desquelles des items se répètent (comme des syntagmes sémantiquement catégorisés).

Couverture d'un motif de segments sur des données : soient un motif de segments $P = p_1 p_2 \dots p_n \in \mathscr{L}_r$ et une séquence de la base de données enrichie $I = i_1 i_2 \dots i_p \in \mathscr{L}_{p^+}$, alors P *couvre les segments de* I, noté $P \leq_{c^+} I$, s'il existe une fonction *discrète croissante* $S()$ définie de $[1,p]$ vers $[1,n]$ telle que, pour tout $j \in [1,p]$, alors $p_j \leq_{ci} i_{S(j)}$

Ce même mécanisme sera pris en compte lorsqu'il s'agit de généraliser selon l'axe paradigmatique : l'objectif est que, par exemple, `'CELEB'` couvre indifféremment *'Pompidou'* et *'Valery Giscard d'Estaing'*. Plus généralement, nous définissons trois relations de généralisation entre motifs :

– **Généralisation hiérarchique entre motifs de segments** : soient deux motifs de segments $P = p_1 p_2 \dots p_n \in \mathscr{L}_{p^+}$ et $Q = q_1 q_2 \dots q_p \in \mathscr{L}_{p^+}$, alors P *généralise hiérarchiquement les segments de* Q, noté $P \leq_g Q$, s'il existe une une fonction *discrète croissante* $S()$ définie de $[1,p]$ vers $[1,n]$ telle que, pour tout $j \in [1,p]$, alors $p_j \leq_{ci} q_{S(j)}$.

6. *http ://tln.li.univ-tours.fr/Tln_CasEN.html*
7. Il est fréquent que plusieurs catégories sémantiques soient associées aux entrées
8. Pour respecter l'anti-monotonie, deux items contigus ne peuvent être identiques ou parents l'un de l'autre

– **Généralisation par affixation entre motifs** : soient deux motifs $P = p_1 p_2 \dots p_n \in \mathscr{L}_{p^+}$ et $Q = q_1 q_2 \dots q_p \in \mathscr{L}_{p^+}$, alors *P généralise par affixation Q*, noté $P \leq_g Q$, si $p \geq n$ et s'il existe au moins un $k \in [0, p-n]$ tel que, pour tout $j \in [1, n]$, alors $q_{j+k} = p_j$.
– **Généralisation sur marqueurs entre motifs** : soient deux motifs $P = p_1 p_2 \dots p_n \in \mathscr{L}_{p^+}$ et $Q = q_1 q_2 \dots q_p \in \mathscr{L}_{p^+}$, alors *P généralise sur marqueurs Q*, noté $P \leq_g Q$, si $p \geq n$ et s'il existe une fonction *discrète strictement croissante* $C()$ définie de $[1, n]$ vers $[1, p]$ telle que, pour tout $j \in [1, n]$, alors $p_j = q_{C(j)}$ et, pour tout $k \in [1, p]$ tel que $k \notin \{C(j), j \in [1, n]\}$, alors $q_k \in \Sigma_m$.

Ces généralisations nous permettent de rechercher des motifs dans lesquels apparaissent les marqueurs d'entités nommées. Par exemple, au sein de l'énoncé *'Le <fonc> président </fonc> <pers> Georges Pompidou </pers> débattait souvent.'*, nous relevons, par relations de couverture et de généralisation, une occurrence pour les motifs `'NOM/président <pers> CELEB </pers>'` ou `'NOM/président CELEB </pers> VERB/débattre'`, par exemple.

Finalement, La notion de règle d'annotation partielle découle de celle de motif de segments :

Règle d'annotation partielle une règle d'annotation partielle est un motif de segments $P \in \mathscr{L}_{p^+}$ contenant au moins un élément de Σ_r et un élément de Σ_m.

Notons qu'à ce stade les règles d'annotation contiennent un nombre indéterminé de marqueurs. Il conviendra de filtrer au besoin lors de l'extraction des motifs et de s'assurer que l'on utilise ces règles de manière adéquate afin de produire une annotation.

3.3 Filtrage et extraction de règles d'annotations partielles

La combinatoire du langage $\mathscr{L}_{p^+}$ étant importante, il est nécessaire de filtrer les règles. Pour cela, nous déterminons la fréquence et la confiance des règles, afin d'éliminer celles qui n'ont que peu d'intérêt. À l'aide de la couverture et des généralisations définies ci-dessus, nous déterminons la fréquence $Freq(P, \mathscr{D})$ d'une règle P comme son nombre d'occurrences au sein du corpus $\mathscr{D}$. La confiance d'une règle d'annotation P estime la proportion de phrases où la règle est appliquée avec justesse :

$$Conf(P, \mathscr{D}) = \frac{Freq(P, \mathscr{D})}{Freq(Ret_m(P), \mathscr{D})} \quad \text{(la fonction } Ret_m(P) \text{ retire les marqueurs de } P)$$

Même en fixant des seuils de support et confiance sélectifs, les règles d'annotation peuvent être trop nombreuses à cause des combinaisons possibles au travers de la hiérarchie. Afin de contenir cette abondance de règles, nous proposons de grouper les règles, puis d'éliminer celles qui ne sont pas informatives, à l'instar de (Pasquier *et al.*, 1999). L'idée forte est que deux motifs qui couvrent les mêmes exemples sont redondants car ils appartiennent à la même classe d'équivalence :

Équivalence de motifs au regard d'une base de données : soient P et Q deux motifs et $\mathscr{D}$ une base de données, alors P est *équivalent à Q au regard de* $\mathscr{D}$, notée $P \equiv_{\mathscr{D}} Q$, si $P \leq_g Q$ ou $Q \leq_g P$ et $Freq(P, \mathscr{D}) = Freq(Q, \mathscr{D})$

Dans la suite, plutôt que d'extraire toutes les règles d'une même classe d'équivalence, nous nous contentrerons des motifs les plus spécifiques car ils sont porteurs de plus de corrélations. Par ailleurs, nous étendons cette équivalence par une marge de tolérance lors de la comparaison des fréquences à δ%, ce que nous appellons alors filtrage δ.

3.4 Annotation automatique à partir des règles d'annotation

Les règles d'annotation sont utilisées par mXS pour réaliser l'annotation en entités nommées. Pour une position j d'un texte, de nombreuses règles peuvent proposer des marqueurs. Nous estimons la probabilité d'insérer des marqueurs en M_j (transductions) par régression logistique, ce qui nous permet de tenir compte de la multiplicité des règles $P \in \mathscr{P}_j$ selon la formule :

$$P(m \in M_j|\mathscr{P}_j) = \frac{1}{Z(\mathscr{P}_j)} \cdot \exp \sum_{P \in \mathscr{P}_j} \lambda_{P,m}$$

Dans une annotation (et plus particulièrement si elle est structurée), plusieurs marqueurs peuvent se trouver à une position donnée. Il nous faut être en mesure de faire le lien entre la probabilité d'insérer un marqueur individuel et celle d'insérer une séquence de marqueurs. Pour cela, nous tenons compte des statistiques issues du corpus sous forme de probabilités conditionnelles [9] :

$$P(M_j = m_1 m_2 \dots m_p) = \frac{1}{p} \cdot \sum_{k=1}^{p} P(m_k \in M_k|\mathscr{P}_k) P(m_1 \dots m_p|m_k)$$

Lorsque les probabilités de séquences de marqueurs $P(M_i)$ sont estimées, nous les utilisons afin de déterminer quelle est, pour un énoncé donné, l'annotation la plus vraisemblable parmi les annotations valides. Une hypothèse d'indépendance entre marqueurs au sein d'un énoncé nous permet de résoudre la recherche de l'annotation par programmation dynamique.

4 Expériences sur le corpus Etape

4.1 Données

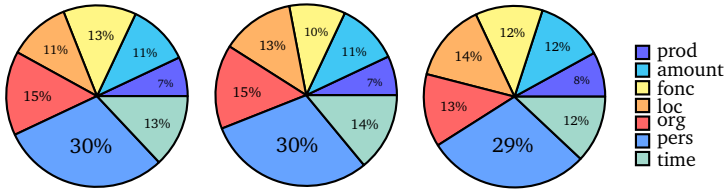
Corpus	Sources(nombre de fichiers)	Tokens	Énoncés	EN
Etape-Train	BFMTV (5), France Inter (16), LCP (23)	355 975	14 989	46 259
Etape-Dev	BFMTV (1), France Inter (6), LCP(6), TV8 (2)	115 530	5 724	14 112
Etape-Test	BFMTV (1), France Inter (6), LCP (5), TV8 (2)	123 221	6 770	13 055
Total	74 enregistrements	594 726	27 483	73 426
Etape-Quaero	France Classique (1), France Culture (1), France Inter (62), France Info (13), RFI (14), RTM (97)	1 596 427	43 828	279 797

TABLE 1 – Caractéristiques du corpus Etape

Le travail a été réalisé dans le contexte de la campagne d'évaluation Etape [10], en interaction avec le programme Quaero [11]. Cette campagne a porté sur le traitement d'émissions radiodiffusées et télévisuelles, donc orales et en partie spontanées. L'objectif est d'annoter les entités nommées structurées, tant sur les transcriptions manuelles qu'en sortie de systèmes de reconnaissance de la parole. La table 1 indique les parties à disposition. Le corpus Etape-Test étant en cours d'adjudication, nous ne l'utilisons pas pour mener nos expériences. Etape-Quaero [12] est volumineux et reste difficile à exploiter par la fouille. En conséquence, nous n'utilisons que Etape-Train (extraction des règles et paramétrage du modèle) et Etape-Dev (évaluation).

9. Ces probabilités sont normalisées a posteriori
10. Évaluations en Traitement Automatique de la Parole (2011-2012)
11. http://www.quaero.org (2008-2013)
12. Adaptation du corpus Ester au format Etape

Les types principaux d'entités nommées sont les personnes (`pers`)), fonctions (`fonc`), organisations (`org`), lieux (`loc`), productions humaines (`prod`), points dans le temps (`time`), quantités (`amount`) et évènements (`event`). À granularité fine (sur laquelle est réalisée l'évaluation), ils sont répartis en 34 sous-types. La figure 4 indique leur répartition au sein du corpus Etape. Notons que les entités nommées sont étendues à des expressions construites à partir de noms communs, ce qui amène à considérer une large gamme d'expressions linguistiques.



(a) Etape-Train (b) Etape-Dev (c) Etape-Test

FIGURE 4 – Répartition des types principaux d'entités

En plus des entités nommées, leurs *composants* sont annotés, soit spécifiques à certains types (jour, mois, etc. pour une date) ou transverses (valeur, unité, qualificateur, etc.). Ces éléments permettent de mieux décrire les entités lors de leur annotation (Rosset *et al.*, 2011).

Le nombre d'entités nommées rapporté au nombre de tokens du corpus est de 12,3%, dont 4,8% pour les entités et 7,5% pour les composants. Globalement, ce corpus, quoiqu'assez volumineux, est bien équilibré pour les types principaux d'entités et de composants. Notons que nous réalisons l'exploration des données sur un corpus qui contient des disfluences, répétitions, etc.

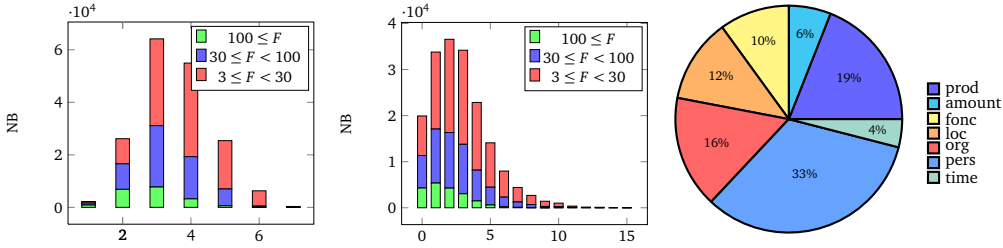
4.2 Extraction de règles d'annotation

Pour implémenter la fouille de données, nous construisons un arbre des préfixes communs *par niveaux*, le processus est optimisé en exploitant la propriété d'*anti-monotonie* (Agrawal et Srikant, 1995) et les hiérarchies (Wang et Han, 2004). De plus, nous poussons deux contraintes supplémentaires pour l'extraction des règles d'annotation :

– **Nombre de marqueurs** : une règle d'annotation partielle ne contient qu'un marqueur.
– **Niveaux** : le nombre d'itérations de l'algorithme par niveaux est limité à 7.

L'approche que nous adoptons nous permet d'explorer exhaustivement les motifs fréquents et confiants. Les seuils minimaux sont fixés à 3 en fréquence et 5% en confiance. Le système extrait alors 143 205 règles d'annotation partielles [13]. La figure 5 montre que la longueur des règles varie autour de trois éléments, et leur profondeur d'items [14] se situe autour de quatre. Ces statistiques confirment que les règles d'annotation sont explorées sur les deux axes que nous avons définis. Nous voyons aussi que la répartition des règles d'annotation par types d'EN est diversement corrélée au corpus. Les types `time` et `amount` sont moins représentés : il y a moins de descripteurs pour ces types, il pourrait alors être assez homogène dans les données. Inversement, le type `prod`, est sur-représenté et nous faisons l'hypothèse qu'il est assez hétérogène.

13. En 15 minutes, sur un seul cœur, en consommant 1,5Go de RAM
14. Somme sur les items des spécialisations au delà de la racine

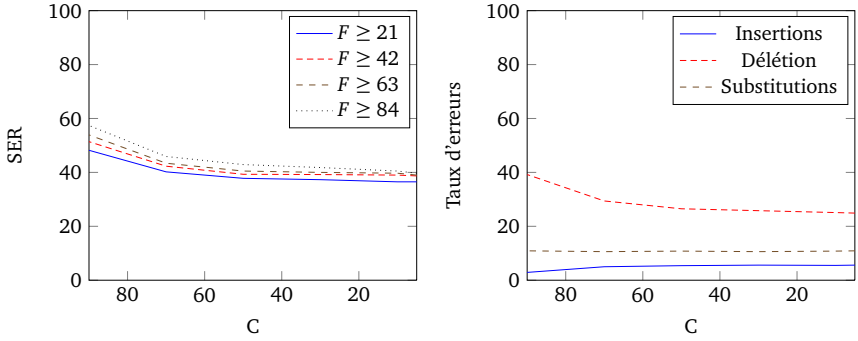


(a) Longueur des règles (b) Profondeur des règles (c) Types d'entités des règles

FIGURE 5 – Caractéristiques des règles d'annotation extraites

4.3 Reconnaissance d'entités nommées

Nous utilisons l'outil *scikit-learn* [15] (Pedregosa *et al.*, 2011) pour réaliser la régression logistique. La figure 6 présente les résultats obtenus en SER [16] et les taux par types d'erreurs (Galibert *et al.*, 2011). Ces graphiques confirment que le système réduit graduellement ses erreurs à mesure que les seuils de fréquence et de confiance sont abaissés.



(a) Performances (b) Erreurs, $F \geq 21$

FIGURE 6 – Performances (SER) et erreurs selon la Fréquence (F) et la Confiance (C)

Nous menons des expériences supplémentaires , dont les résultats sont reportés dans le tableau 2 pour les configurations suivantes :

- `Logit` : système par défaut
- `Logit-Dicos` : désactivation des ressources lexicales
- `Logit+Test` : apprentissage en fusionnant les corpus `Etape-Train` et `Etape-Dev`
- `Logit-D25` : filtrage δ à 25%,
- `Logit-D50` : filtrage δ à 50%,
- `Logit-D75` : filtrage δ à 75%,

15. `http://scikit-learn.org`
16. Slot Error Rate, taux d'erreur pondéré

Le système donne des résultats satisfaisants, étant donné la difficulté de la tâche. Sans surprise, la désactivation des dictionnaires dégrade considérablement les performances. Lorsque les données comportent les données d'évaluation (`Logit+Test`), le surapprentissage est modéré, ce qui est lié au fait que les règles d'annotation ne sont pas lexicalisées. Les expériences `Logit-DXX` nous montrent clairement que le système obtient encore des performances très acceptables lorsque l'on réduit significativement le nombre de règles extraites à l'aide du filtrage δ.

Approche	Règles	SER	I	D	S	P	R	Fm
`Logit`	143 205	**35,9**	5,6	24,2	10,8	79,8	64,9	71,6
`Logit-Dicos`	80 231	**45,2**	5,9	30,2	16,3	70,7	53,5	60,9
`Logit+Test`	141 550	**26,3**	3,2	18,6	8,1	86,6	73,3	79,4
`Logit-D25`	100 027	**36,2**	5,6	24,6	10,9	79,7	64,6	71,3
`Logit-D50`	73 332	**36,7**	5,4	25,2	11,0	79,5	63,8	70,8
`Logit-D75`	50 408	**39,0**	5,4	27,0	11,7	78,2	61,3	68,7

TABLE 2 – Performances (SER), erreurs d'Insertion (I), de Délétion (D), de Substitution (S), Précision (P), Rappel (R), F-mesure (Fm) des approches

Nous menons des évaluations séparées des types primaires (sans sous-types) d'entités nommées et de composants. La figure 7 en donne les résultats. Les entités nommées sont moins bien reconnues que les composants et plusieurs types (en particulier les expressions de temps) posent encore problème. Ceci dit, le système équilibre relativement bien sa précision et son rappel et la reconnaissance d'entités nommées selon l'approche présentée donne des résultats.

Types	SER	P	R	Fm
Entités	**38,9**	76,4	62,3	68,6
Composants	**33,0**	86,4	68,5	76,4
Tous	**35,9**	79,8	64,9	71,6

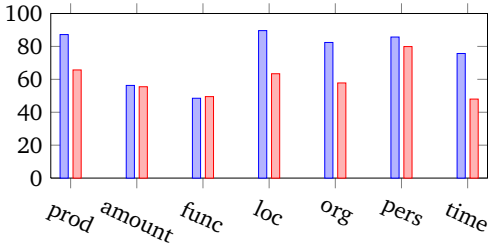


FIGURE 7 – SER, précision (gauche) et rappel (droite) par types primaires et composants

La phase d'adjudication de la campagne d'évaluation ETAPE n'est pas achevée à l'heure de la rédaction de cet article. Nous avons cependant été autorisés à reporter en table 3 les performances anonymes des systèmes avant adjudication. Les SER présentés sont donnés sur les transcriptions manuelles et sur les sorties de différents systèmes de reconnaissance, pour lesquels sont mentionnés les WER [17].

Parmi les autres systèmes participants, le système 3 utilise des CRF (binarisés, un par type), le système 6/7/8 utilise un CRF pour les composants et un PCFG pour reconstituer les entités, CasEN utilise des transducteurs. De manière générale, `mXS` affiche de bonnes performances (entre la 1 ère et la 3 ème position). Les taux d'erreurs élevés sont liés à la difficulté de la tâche (parole spontanée, imbrications, typologie fine). Sans surprise, les performances sont dégradées sur les données bruitées par la reconnaissance de parole. Nous voyons que `mXS` et résiste bien aux erreurs de reconnaissance de la parole.

17. Word Error Rate

Part.	Type	Man	Rover	WER23	WER24	WER25	WER30	WER35
1	OC	84.8	98,1	100,7	94,2	98,9	98,4	100,9
2	OC	172.0	147,4	178,8	160,4	168,0	163,9	168,2
3	CRF	**33.8**	**57,2**	**59,3**	64,7	**62,0**	**61,7**	**71,8**
4	OC	55.6	88,0	98,8	76,8	92,8	94,9	99,6
5	CRF	43.6	69,7	73,8	72,1	73,7	74,8	86,0
6	CRF+PCFG	na	79,2	79,5	66,8	80,8	80,0	87,0
7	CRF+PCFG	na	67,8	68,4	67,6	70,9	69,9	85,2
8	CRF+PCFG	36.4	na	na	na	na	na	na
9	CRF	62.8	75,8	79,2	76,9	79,8	80,5	90,5
10	OC	42.9	65,0	69,9	66,3	70,5	69,9	87,0
CasEN	OC	49.3	na	na	68,4	na	na	na
mXS	Règles	41.0	63,7	67,5	**64,1**	69,1	68,6	80,4

TABLE 3 – SER de la campagne Etape par système (OC=Orienté Connaissances) sur les transcriptions avant adjudication (manuel : Man, transcription automatiques : Rover et WERXX, dont WER24 avec majuscules)

5 Conclusion

La reconnaissance d'entités nommées structurées sur de la parole spontanée nécessite de mettre au point des systèmes robustes. Dans cet article, nous présentons une approche originale à base de fouille de données, qui extrait des règles d'annotation partielles et paramètre un modèle numérique les utilisant.

Les résultats obtenus dans le cadre de la campagne Etape indiquent que notre approche novatrice fait jeu égal avec les systèmes état de l'art. Pour éviter tout biais méthodologique, nous restons toutefois en attente d'une référence débarrassée de toute erreur d'annotation : c'est l'objectif de la phase d'adjudication en cours. Notre objectif à court terme est de mieux caractériser les points forts et limitations du modèle (détection séparée du début et de la fin des annotations). Nous comptons également mettre à l'épreuve le système sur d'autres tâches qui pourraient bénéficier de l'extraction de motifs de segments.

Remerciements

Ces travaux ont été réalisés dans le cadre du projet ANR Etape. Merci en particulier à Olivier Galibert (LNE), Matthieu Carré (ELDA) et Guillaume Gravier (IRISA).

Références

AGRAWAL, R. et SRIKANT, R. (1995). Mining sequential patterns. *In International Conference on Data Engineering (ICDE'95)*, pages 3–14.

BIKEL, D., SCHWARTZ, R. et WEISCHEDEL, R. M. (1999). An algorithm that learns what's in a name. *Machine Learning*, 34:211–231.

BRUN, C. et EHRMANN, M. (2010). Un système de détection d'entités nommées adapté pour la campagne d'évaluation ester 2. *In Traitement Automatique du Langage Naturel (TALN'10)*.

BÉCHET, F., SAGOT, B. et STERN, R. (2011). Coopération de méthodes statistiques et symboliques pour l'adaptation non-supervisée d'un système d'étiquetage en entités nommées. *In Traitement Automatique des Langues Naturelles (TALN'11)*.

CELLIER, P. et CHARNOIS, T. (2010). Fouille de données séquentielles d'itemsets pour l'apprentissage de patrons linguistiques. *In Traitement Automatique des Langues Naturelles (TALN'10)*.

DINARELLI, M. et ROSSET, S. (2011). Models cascade for tree-structured named entity detection. *In International Joint Conference on Natural Language Processing (IJCNLP'11)*.

EKBALA, A., SOURJIKOVA, E., FRANK, A. et PONZETTO, S. P. (2010). Assessing the challenge of fine-grained named entity recognition and classification. *In Annual Meeting of the Association for Computational Linguistics (ACL'10) - Named Entities Workshop*, pages 93–101, Uppsala, Sweden.

FINKEL, J. R. et MANNING, C. D. (2005). Nested named entity recognition. *In Conference on Empirical Methods in Natural Language Processing (EMNLP'09)*.

FISCHER, J., HEUN, V. et KRAMER, S. (2005). Fast frequent string mining using suffix arrays. *In 5th IEEE International Conference on Data Mining (ICDM'05)*, pages 609–612.

FRIBURGER, N. et MAUREL, D. (2004). Finite-state transducer cascades to extract named entities in texts. *Theoretical Computer Sciences (TCS)*, 313:93–104.

GALIBERT, O., ROSSET, S., GROUIN, C., ZWEIGENBAUM, P. et QUINTARD, L. (2011). Structured and extended named entity evaluation in automatic speech transcriptions. *In International Joint Conference on Natural Language Processing (IJCNLP'11)*.

GALLIANO, S., GRAVIER, G. et CHAUBARD, L. (2009). The ester 2 evaluation campaign for the rich transcription of french radio broadcasts. *In International Speech Communication Association (INTERSPEECH'09)*.

GRISHMAN, R. et SUNDHEIM, B. (1996). Message undersrtanding conference - 6 : A brief history. *In International Conference on Computational Linguistics (COLING'96)*, pages 466–471, Copenhagen, Denmark.

HOBBS J. R., Appelt D., B. J. I. D. K. M. S. M. et TYSON, M. (1996). *FASTUS : A Cascaded Finite-State Transducer for Extracting Information from Natural-Language Text*, pages 383–406.

MCCALLUM, A., FREITAG, D. et PEREIRA, F. (2000). Maximum entropy markov models for information extraction and segmentation. *In International Conference on Machine Learning (ICML'00)*, pages 591–598.

MCDONALD, D. D. (1996). *Internal and External Evidence in the Identification and Semantic Categorisation of Proper Names*, pages 32–43.

MIKHEEV, A., MOENS, M. et GROVER, C. (1999). Named entity recognition without gazetteers. *In Proc. of the Ninth Conference of the European Chapter of the Association for Computational Linguistics*, pages 1–8.

PASQUIER, N., BASTIDE, Y., TAOUIL, R. et LAKHAL, L. (1999). Efficient mining of association rules using closed itemset lattices. *INF. SYST.*, 24(1):25–46.

PEDREGOSA, F., VAROQUAUX, G., GRAMFORT, A., MICHEL, V., THIRION, B., GRISEL, O., BLONDEL, M., PRETTENHOFER, P., WEISS, R., DUBOURG, V., VANDERPLAS, J., PASSOS, A., COURNAPEAU, D., BRUCHER, M., PERROT, M. et Édouard DUCHESNAY (2011). Scikit-learn : Machine learning in python. *Journal of Machine Learning Research*, 12:2825–2830.

RAYMOND, C. et FAYOLLE, J. (2010). Reconnaissance robuste d'entités nommées sur de la parole transcrite automatiquement. *In Traitement Automatique des Langues Naturelles (TALN'10)*.

ROSSET, S., GROUIN, C. et ZWEIGENBAUM, P. (2011). Entité nommées structurées : guide d'annotation quaero. Rapport technique, LIMSI (2011-04).

SCHMID, H. (1994). Probabilistic pos tagging using decision trees. *In New Meth. in Lang. Proc. (NEMLP'94)*.

SRIKANT, R. et AGRAWAL, R. (1996). Mining sequential patterns : Generalizations and performance improvements. *In International Conference on Extending Database Technology (EDBT'96)*, pages 3–17.

STEPHENS, C. S. (1993). The analysis and acquisition of proper names for the understanding of free text. *Computers and the Humanities*, 26:441–456.

WANG, J. et HAN, J. (2004). Bide : Efficient mining of frequent closed sequences. *In International Conference on Data Engineering (ICDE'04)*.

ZIDOUNI, A., ROSSET, S. et GLOTIN, H. (2010). Efficient combined approach for named entity recognition in spoken language. *In Conference of the International Speech Communication Association (INTERSPEECH'10)*.