Extraction d'information à partir de modèles de Markov cachés

Frédéric Kerloch , Patrick Gallinari

LIP6, Equipe Connexionniste – Université Pierre et Marie Curie
8 rue du Capitaine Scott, 75015 Paris
{kerloch/gallinari}@poleia.lip6.fr

Mots-clés : Extraction d'Information, Modèles de Markov Cachés
Keywords: Information Extraction, Hidden Markov Models

Résumé La quantité gigantesque de données textuelles produites sous forme numérique a créé d'énormes besoins en outils capables d'exploiter l'information contenue dans ces textes. L'hétérogénéité et la taille des corpus condamnent par avance toute approche basée sur des techniques purement manuelles. Il faut donc développer des méthodes automatiques capables de structurer les textes, ainsi que d'extraire l'information pertinente. Dans ce cadre, nous avons développé un système d'extraction basé sur des modèles de Markov cachés (MMC). Nous abordons l'apprentissage de la structure ainsi que celui des paramètres d'émission. Nous présentons ensuite des résultats obtenus sur une instance de problème proche de l'extraction, le défi DEFT05.

Abstract The huge quantity of textual data now available has created a need of tools able to exploit the information present in these texts. Manual approaches are inappropriate, due to the heterogeneity and the size of the corpora. Automatic methods must be developed in order to structure the texts and to extract relevant information. In this context, we have developed an extraction system based on Hidden Markov Models. We consider the structure training as well as the estimation of the parameters. We present some results obtained in DEFT05, a data mining challenge close to extraction.

1 Introduction

L'abondance de données disponibles sous forme numérique a rendu cruciales les techniques permettant de faciliter l'accès à l'information pertinente. La conception de ce type de systèmes doit mettre l'accent sur leur caractère automatique, tout travail purement manuel (sans intervention d'un processus automatique) étant voué à l'échec face à la quantité gigantesque d'informations à traiter.

Plusieurs types de tâches peuvent être définies à l'intérieur de cette problématique, selon le niveau de granularité avec lequel les corpus sont abordés:

- Aide à la navigation : au niveau de granularité le plus élevé, un système d'aide à la recherche d'information peut permettre de simplifier la navigation dans les données. Cette aide peut se faire par exemple en structurant hiérarchiquement le corpus (Njike-Fotzo, 2004), ou encore en apprenant automatiquement à détecter des comportements de navigation face à un corpus, permettant ainsi de proposer un système de liens dynamiques s'adaptant à l'utilisateur (Blanchard, 2004)

- Recherche d'information : se plaçant au niveau du document, de nombreuses techniques permettent l'interrogation de larges corpus de données via une indexation du corpus, puis le calcul d'une mesure de similarité performante permettant renvoi des documents les plus pertinents pour une requête donnée.

- Extraction d'information : enfin, au niveau de granularité le plus fin, il est possible d'envisager des systèmes permettant l'identification précise de l'information pertinente. Ceci peut être fait à l'intérieur de corpus homogènes, où chaque texte est censé contenir des informations dont le type est prédéfini (problématique typique de l'extraction d'information), ou de manière plus large dans des corpus hétérogènes directement interrogés à partir de questions posées par l'utilisateur, le système devant renvoyer un passage de texte répondant précisément à la question (problématique de type question / réponse)

Dans la suite, nous nous intéresserons plus particulièrement aux problèmes d'extraction d'information. Dans la deuxième partie, nous donnons la définition précise d'une tâche d'extraction, et faisons un bref état de l'art. Nous détaillons ensuite l'utilisation de modèles de Markov cachés en extraction d'information. La quatrième partie décrit le modèle retenu, ainsi que les méthodes d'apprentissage des paramètres. La cinquième partie présente quelques résultats obtenus sur le corpus DEFT05. La sixième partie présente les conclusions et perspectives.

2 Extraction d'Information

2.1 Définition d'une tâche d'extraction

La définition précise des objectifs et des termes employés en extraction d'information sont issus des campagnes MUC (Messages Understanding Conferences, Grishman 1996). Ces campagnes avaient pour but l'évaluation de méthodes permettant d'extraire de l'information à partir de textes. Elles ont introduit la notion de patron d'extraction, qui contient une description de l'ensemble des champs à extraire dans chaque texte. Une tâche d'extraction consistait pour chaque texte à remplir un patron d'extraction associé. L'évaluation se faisait ensuite en regardant pour chaque champ du patron si les bons extraits de texte étaient présents.

On peut distinguer plusieurs manières d'aborder une tâche d'extraction. La première consiste à effectuer l'extraction de chaque champ séparément : on parle alors d'extraction single slot. Mais il peut être utile de considérer des dépendances entre les champs, que l'on extraira alors simultanément : on parle alors d'extraction multi-slot.

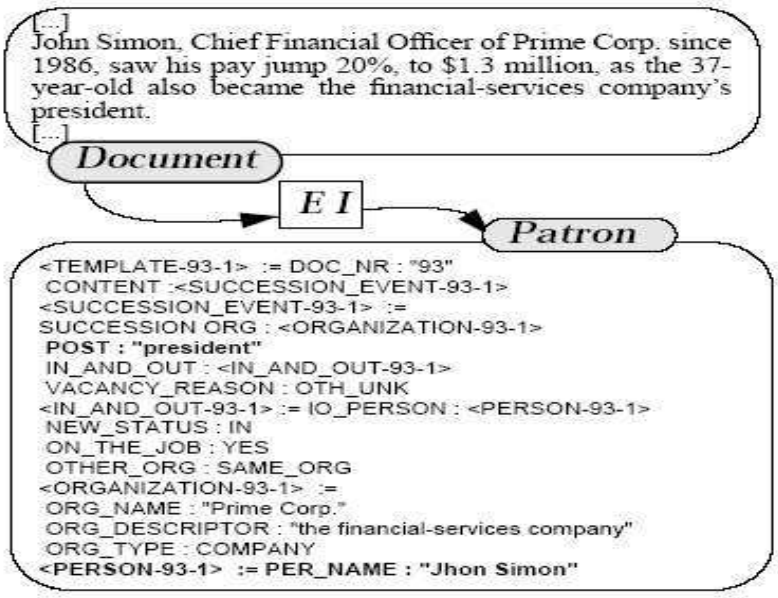

Figure 1 : Un extrait de texte issus de MUC5, et son patron associé

2.2 Etat de l'art

Dans les premières campagnes d'évaluation MUC, la majorité des systèmes était construite à partir d'analyses linguistiques complexes effectuées sur le corpus d'apprentissage. Ces analyses permettaient la construction manuelle de motifs d'extraction. Lors du passage à un corpus différent, tous les motifs étaient à réécrire. Le besoin d'utiliser des méthodes automatiques, ou au moins semi-automatiques, se fit donc rapidement sentir. (Cardie 1993), (Riloff 1993), (Soderland 1995) présentent des systèmes permettant l'acquisition automatique de ressources linguistiques utiles à la construction de motifs d'extraction. Par la suite, trois directions principales de recherche peuvent être identifiées.

La première consiste en l'utilisation de méthodes issues de l'inférence grammaticale pour apprendre des expressions régulières permettant de caractériser les différents champs à extraire. Ces systèmes étaient utilisés initialement pour effectuer de l'extraction dans des pages HTML très structurées qui permettaient une identification simple des champs. L'algorithme BWI (Kushmerick, Freitag 2000) apprend des délimiteurs de champs peu précis, puis améliore l'apprentissage par des techniques de Boosting. (Kosala et al. 2002) construisent des automates d'arbres pour retrouver les noeuds pertinents dans l'arbre de dérivation HTML.

La seconde s'apparente à de l'apprentissage de règles, essentiellement à base d'apprentissage relationnel. (LP)² (Ciravegna 2003) est à ce jour l'algorithme d'extraction le plus performant sur les tâches classiques d'extraction. Il construit ses motifs d'extraction en partant d'une règle vide, puis en lui ajoutant progressivement des contraintes. Seules les meilleures règles générées sont conservées. Deux règles différentes sont générées pour chaque début et fin de champ, ce qui permet d'atteindre un haut niveau de précision. Le rappel est ensuite augmenté en réintroduisant des règles moins bonnes mais qui, en association avec une règle apprise dans l'étape précédente, permettent de finir la délimitation d'un champ.

Le troisième type d'approche utilise des méthodes d'apprentissage statistique pour définir des modèles stochastiques capables d'effectuer l'extraction. On distinguera tout d'abord des méthodes plutôt « génératives » : (Freitag 1999) utilise des Modèles de Markov cachés (MMC) pour effectuer l'extraction. (Peshkin, Pfeffer 2003) étendent ce type de modèle en utilisant des réseaux bayésiens dynamiques (RBD), qui leur permettent d'enrichir simplement leur représentation des token des textes, et d'utiliser des relations entre les différents champs. Il obtient des performances équivalentes à (LP)² sur le corpus classique d'annonce de séminaires [1]. Enfin, (McCallum et al. 2004) introduit un modèle stochastique légèrement plus souple que les MMC, les Conditional Random Fields (CRF), qui leur permettent là aussi d'enrichir leur représentation des tokens. D'autres méthodes utilisent des approches plus « discriminantes », comme (Kushmerick et Finn 2004) qui utilisent des modèles discriminants pour apprendre les frontières de délimitation des champs.

3 Modèles de Markov cachés : application à l'Extraction d'Information

Deux approches sont possibles pour l'utilisation de MMC dans des tâches d'extraction : Une approche single slot, où un HMM est construit par slot. Un état ou plusieurs états sont associés à l'information à extraire. Les tokens générés par ces états seront considérés comme appartenant au type d'information associé. Cette approche est simple et robuste, mais ne permet pas de prendre en compte l'agencement des slots les uns par rapport aux autres. Il peut donc être utile de passer à une approche multi slot, ou un seul MMC est utilisé pour extraire tous les champs. Comme dans l'approche single-slot, on a un ou plusieurs états par champ, plus des états non pertinents, mais ici tous les champs sont représentés. A cause du plus grand nombre de paramètres, l'estimation des probabilités du modèle est moins robuste, mais permet la prise en compte de relations entre les champs.

(Leek, 1997) a le premier introduit les MMC pour effectuer de l'extraction. Il construit à la main une architecture spécifique de MMC pour l'extraction d'information dans des corpus biomédicaux.

(Zaragoza 1998) utilise un modèle de MMC pour de l'extraction dans un corpus de journaux financiers. L'extraction se fait en deux étapes. Dans un premier temps, un classifieur à base de réseau de neurones multicouches est appliqué aux phrases afin de repérer celles susceptibles de contenir de l'information pertinente. Dans la seconde étape, un MMC permet l'identification précise de l'information à l'intérieur des phrases sélectionnées.

(Freitag 1999) utilise des MMC pour de l'extraction dans un corpus d'annonce de séminaires. (Freitag 2000) tente d'apprendre la structure du modèle en partant d'un MMC de base, qu'il complexifie à partir de règles heuristiques. Le MMC obtenant les meilleurs résultats sur le corpus d'apprentissage est conserve.

(Skounakis 2003) utilise des modèles de Markov hiérarchiques. L'idée de base se rapproche de celle de (Zaragoza 1998), la sélection des phrases pertinentes se faisant de manière implicite dans le modèle : le premier niveau hiérarchique du MMC effectue la classification, le deuxième l'extraction dans les phrases pertinentes.

[1] Voir : http://www.isi.edu/info-agents/RISE/

4 Un modèle à base de MMC

4.1 Motivation du choix du modèle

Nous souhaitions utiliser un modèle robuste, dont les probabilités sont simples à estimer, et qui permette une utilisation avec des variables cachées. Notre idée était de pouvoir utiliser notre modèle avec des données non étiquetées, et de pouvoir descendre à un niveau de granularité plus fin dans l'étiquetage (séparer les champs en début / milieu /fin), sans utiliser d'heuristique a priori. Les MMC, RBD et les CRF respectaient les deux premiers critères. Les MMC permettent une utilisation avec des variables cachées et un temps d'apprentissage raisonnable dans le cas de données partiellement étiquetées.

4.2 Description générale du modèle

Le modèle est constitué d'un MMC possédant 3 états par champs à extraire : un état pour les tokens précédant les champs, un état pour les champs, et un état pour les tokens succédant aux champs. Les deux états préfixes et suffixes doivent permettre de capter des régularités dans les séquences annonçant et terminant les champs. Le modèle possède aussi un état « non pertinent » générant tous les autres tokens.

Le modèle d'émission des observations prend en compte d'éventuelles informations venant s'ajouter à la simple donnée du token (tags morpho-syntaxiques, étiquettes sémantiques, capitalisation …). Les symboles d'émission sont donc des vecteurs d'attributs.

4.3 Apprentissage de la structure du modèle

En partant d'un modèle ergodique, la structure est apprise implicitement en estimant directement les probabilités de transition sur la base d'apprentissage. D'autres techniques d'apprentissage de la structure sont envisageables : des techniques de type bottom-up, la structure est construite à partir d'un modèle simple que l'on enrichit progressivement, ou bien top-down, où une structure très complexe est successivement élaguée. Dans les deux cas, une recherche exhaustive dans l'espace des structures est impossible (à cause de l'explosion combinatoire du nombre de modèles à tester). La solution se trouve alors dans des algorithmes de type Hill Climbing, mais qui ne peuvent garantir l'optimalité de la solution trouvée. L'approche proposée ici permet de résoudre simplement le problème de la structure, mais en fixant à priori le nombre d'états et leur sémantique.

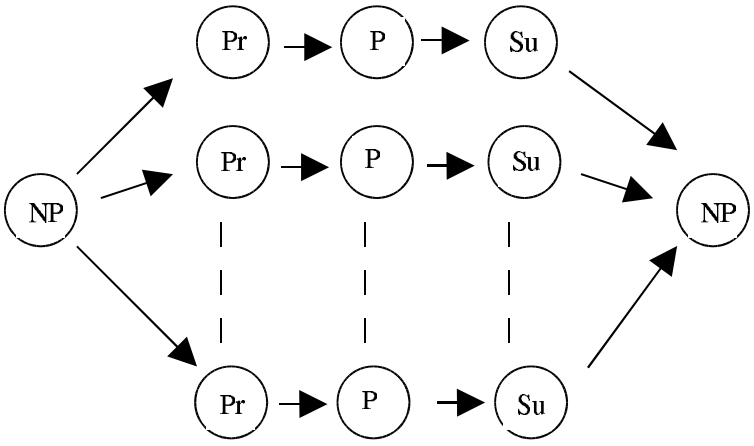

Figure 2 : Modèle après apprentissage de la structure[2], les auto-transitions ayant été supprimées pour des raison de lisibilité

4.4 Apprentissage des paramètres du modèle

Les émissions de notre modèle sont formées par un vecteur contenant les différents attributs de chaque token (informations morpho-syntaxiques, pré-étiquetage d'entités nommées, informations de type majuscule/minuscule, lettre/chiffre).

Dans la phase d'apprentissage, chaque token est associé à un et un seul état : les tokens appartenant à un champ sont associés à l'état correspondant, les tokens dans une fenêtre de *n* mots avant (resp. après) les champs sont associés aux états préfixes (resp. suffixes) respectifs, le reste étant associé à l 'état non-pertinent. Le méta paramètre *n* a été fixé de manière heuristique à 7 pour tous les champs.

Plusieurs types sont d'estimation possibles : on peut estimer directement les probabilités d'émission par maximum de vraisemblance. On se trouve confronté au problème de la taille de l'espace de représentation (on doit estimer un nombre de paramètres égal au produit des tailles des espaces de représentation de chaque attribut). Les estimateurs obtenus ne sont pas assez robustes.

Pour éviter ce problème, il est possible de traiter les attributs comme indépendants sachant l'état. La probabilité d'émission d'un vecteur devient alors le produit des probabilités de chacun de ses attributs, qui sont là aussi estimées par maximum de vraisemblance. Mais cette approche fait apparaître un problème de normalisation dans le produit des probabilités des attributs, ceux ayant le moins de valeurs possibles voyant leur importance relative surévaluée.

Pour remédier à ces problèmes nous avons utilisé un algorithme d'apprentissage discriminant pour estimer les probabilités d'émissions. Nous avons testé plusieurs modèles de classifieurs probabilistes discriminants. Nous avons retenu un modèle à base de machine à vecteurs support à noyau linéaire, appris par la méthode SMO (Platt 1998)

[2] Pour des raisons de lisibilité, les transitions autres que « gauche –droite » n'ont pas été représentées. Elles disparaissant en générale presque toutes lors de la phase d'estimation des probabilités de transition, sauf dans le cas de champs souvent proches les uns des autres.

En l'état, le MMC appris extrait de manière erronée beaucoup de passages introduits par des token fréquents. Il faut donc forcer artificiellement le modèle à ne considérer que des motifs introductifs suffisamment longs. De plus, le modèle a tendance à ne pas rester suffisamment longtemps dans les états pertinents, tronquant très souvent les fins de passages pertinents. Une solution est d'introduire des modèles de durée, afin de maîtriser plus finement le temps passé dans chaque état. Nous avons opté pour un modèle de durée simple consistant à dupliquer tous les états après l'apprentissage des paramètres. Les passages extraits sont ainsi plus longs, et l'apparition dans une séquence d'un token très courant ne suffit plus à basculer dans un état annonciateur (états Pr de la figure 2, correspondant par exmple à des mots introductifs – article…).

5 Evaluation

Le défi DEFT05 ne correspond pas exactement à la tâche d'extraction générique pour laquelle le système a été conçu et utilisé initialement. Les notions de motifs introducteurs et terminaux usuels en extraction n'ont pas de sens ici, de façon plus générale, la notion de séquence et de succession d'évènements n'est pas présente dans la tâche DEFT. Cette dernière est plus proche d'une tâche de classification de phrases ou de segmentation thématique – le style apportant à première vue peu d'information. Nous avons toutefois utilisé une version simplifiée de notre modèle sur les différentes instances de la tâche DEFT pour évaluer si cette classe de méthode était apte à résoudre ce type de problème.

Les phrases de J. Chirac précédant et suivant les discours de F. Mitterrand ne sont ni annonciatrices, ni terminales, car les discours n'ont aucun lien entre eux. Les états préfixes et suffixes de notre modèle n'ont donc pas d'intérêt, et ont été supprimés. Nous obtenons donc un MMC à deux états modélisant les discours de J. Chirac et de F. Mitterrand. Le modèle d'extraction se réduit à un classifieur simple avec des probabilités de transition entre états.

Description du corpus

Le corpus du défi DEFT05 est constitué de discours de J. Chirac, dans lesquels ont été insérées des portions de discours de F. Mitterrand. L'objectif est de retrouver les passages issus d'allocutions de Mitterrand.

Résultats

Les résultats obtenus sont décrits ci-après. Dans la tâche 1, les noms de personnes et les dates ont été remplacés par des balises, dans la tâche 2, seuls les noms ont été remplacés.

	F1 de notre système	F1 moyen	Ecart type
Tâche 1	0.731591	0.629217	0.25852
Tâche 2	0.793889	0.673813	0.22447
Tâche 3	0.788093	0.690224	0.20492

Figure 4 : résultats pour les différentes tâches de DEFT05.

Notre système se situe au dessus de la moyenne, mais nous ne connaissons pas à ce jour les meilleurs résultats obtenus pour cette tâche. Il faut noter que la présence des noms de personnes et des dates améliore les performances de manière significative. Le modèle et la représentation utilisés sont particulièrement simples et devraient pouvoir être facilement améliorés.

6 Conclusion et perspectives

Nous avons développé un modèle général d'extraction d'information à base de modèles de Markov cachés, pouvant prendre en compte des observations de type vectoriel grâce à notre méthode d'apprentissage des probabilités d'émission à partir d'approches discriminantes. Comme (McCallum 2004) et (Peshkin, Pfeffer 2003), il permet l'intégration rapide de plusieurs attributs pouvant être utiles à l'extraction. L'utilisation d'un modèle simple et robuste comme les MMC permet de plus d'envisager l'utilisation de données non étiquetées pour améliorer l'apprentissage, voire de raffiner le niveau de granularité en redécoupant chaque état pour obtenir une modélisation plus fine des champs à extraire.

Nous envisageons plusieurs améliorations du modèle présenté.

- Nous souhaiterions intégrer une approche plus « discriminante » (au sens donné en 2.2) en incorporant dans le modèle des informations plus spécifiques aux frontières entre pertinent / non pertinent. Dans le cas de DEFT par exemple, il serait souhaitable d'intégrer à notre modèle des informations relatives aux transitions entre phrases (les cassures sémantiques semblant par exemple particulièrement pertinentes pour la tâche). Pour l'instant, cela est fait de manière implicite grâce au mécanisme d'états annonciateurs et terminaux. Une première approche possible serait de combiner notre modèle avec un classifieur à fenêtre.

- Beaucoup d'instances de problèmes d'extraction montrent que l'ordre d'apparition des champs est très régulier : même séparés par de longs passages non-pertinents, ils apparaissent dans le même ordre. Pour l'instant, notre modèle peut capter ce type de régularité uniquement pour des champs très proches via le mécanisme d'apprentissage des probabilités de transitions. Il faudrait par exemple introduire dans le modèle un attribut supplémentaire contenant le type du champ précédemment extrait.

Références

BLANCHARD J., PETITJEAN B., ARTIÈRES T., GALLINARI P.(2005), Un système d'aide à la navigation dans les documents hypermédia, Actes de *EGC 2005*

CIRAVEGNA F. (2001), Adaptive Information Extraction from Text by Rule Induction and Generalisation, in Proceedings of *17th International Joint Conference on Artificial Intelligence* (IJCAI 2001)

CARDIE C. (1993), A case-based approach to knowledge acquisition for domain-specific sentence analysis, in Proceedings of the *Eleventh National Conference on Artificial Intelligence*

GRISHMAN R. (1996), Design of the MUC-6 Evaluation. In Proceedings of the *Sixth Message Understanding Conference* (MUC6) , 13-33. Morgan Kauffman

FINN A., KUSHMERICK N. (2004). Information Extraction by Convergent Boundary Classification. In Proceedings of *AAAI-04 Workshop on Adaptive Text Extraction and Mining*

FREITAG D. , MCCALLUM A. (1999), Information extraction using HMMs and shrinkage, In Proceedings of *AAAI-99 Workshop on Machine Learning for Information Extraction*

FREITAG D., MCCALLUM A. (2000), Information extraction with HMM structures learned by stochastic optimisation, In Proceedings of *AAAI-2000*

FREITAG D., KUSHMERICK N. (2000), "Boosted Wrapper Induction". In Proceedings of the *Seventeenth National Conference on Artificial Intelligence* (AAAI-2000)

KOSALA R., V. DEN BUSSCHE J., BRUYNOOGHE M., AND BLOCKEEL H. (2002), Information extraction in structured documents using tree automata induction, In proceedings of *PKDD 02*

LAFFERTY J., MCCALLUM A. AND PEREIRA F. (2001), Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, In Proceedings of *the 18th International Conf. on Machine Learning* (ICML 01)

LEEK, T. R. (1997). Information extraction using hidden Markov models. Master's thesis, UC San Diego

NJIJE-FOTZO H., GALLINARI P. (2004), Apprentissage de relations de spécialisation/généralisation entre concepts – Application à la structuration hierachique automatique de corpus, Actes de *CORIA 04*

PESHKIN L., PFEFER A. (2003), Bayesian Information Extraction Network, In Proceedings of the *Eighteenth International Joint Conf. on Artificial Intelligence* (IJCAI03)

PLATT J. (1998), Fast training of Support Vector Machine using sequencial minimal optimization, *Advances in Kernal methods – Support Vector Learning*, MIT Presse

SODERLAND S., FISHER D., ASELTINE J. AND LEHNERT W. (1995). Crystal: Inducing a conceptual dictionary. In Proceedings of the *Fourteenth International Joint Conference on Artificial Intelligence* (IJCAI95)

SKOUNAKIS M., CRAVEN M. ET RAY S. (2003) Hierarchical Hidden Markov Models for Information Extraction In *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, (IJCAI03)

ZARAGOZA H. AND GALLINARI P. (1998), Coupled Hierarchical IR and Stochastic Models for Surface Information Extraction. In proceedings of *The 20th Annual Colloquium on IR Research, British Computer Society's Information Retrieval Specialist Group* (BCG-RSG'98)