Reconnaissance d'entités nommées appliquée à la veille économique

Julien Gosme — jgo@sparklane.fr
équipe R&D, Sparklane, Nantes

Communication orale

Mots-clés : reconnaissance d'entités nommées, forêt d'arbres de décision, veille économique.
Keywords: named entity recognition, random forest, business intelligence.

Introduction

Sparklane propose une plateforme SaaS d'aide à la décision aux équipes commerciales de ses clients B2B. Celle-ci inclut un ensemble d'outils permettant le ciblage sur des dizaines de critères firmographiques telles que le chiffre d'affaires, le secteur d'activité ou la présence sur les réseaux sociaux. Ces critères sont complétés par une vingtaine de types d'évènement animant la vie de l'entreprise tels que le lancement de nouveaux produits, la présence dans un salon ou la fermeture d'un site. Ces événements sont issus de la veille de la presse économique, nationale, régionale ou de blogs spécialisés et offrent la possibilité aux clients de Sparklane d'adapter plus finement leurs démarches commerciales.

La veille, mise en place en 2014, est réalisée en quatre phases : la collecte, la thématisation, la reconnaissance d'entreprises, l'annotation manuelle. Aujourd'hui, plus de 200 000 articles avec les entreprises citées annotés manuellement ont été accumulé, ce qui permet d'envisager de faire évoluer le système.

Nous présentons ici les travaux en cours de réalisation sur la reconnaissance des noms d'entreprises. Dans un premier temps, nous décrivons la modularisation du système de reconnaissance permettant de faciliter les évaluations. Dans un second temps, nous rapportons les résultats d'une première expérience concernant le repérage des noms d'entreprises d'articles de presse. Enfin nous donnerons les perspectives de ces travaux.

Modularisation du système de reconnaissance de noms d'entreprises

Une des motivations de la modularisation part du constat de la difficulté à identifier la source du bruit de notre système actuel. Par exemple, un grand nombre de faux positifs concernant le mot « total » (pas le groupe pétrolier) sont produits par notre système. Hors ce dernier a été conçu de manière monolithique, nous n'avons pas accès aux résultats intermédiaires pour l'analyser puis le faire évoluer.

Pour cette raison nous sommes en train de redéfinir notre système en trois étapes suffisamment indépendantes pour permettre d'évaluer la qualité de chaque traitement :

locator Repérage de la position des noms correspondant à des entreprises, en s'appuyant sur des caractéristiques liées au contexte et à la forme de surface, l'expérience suivante décrit plus en détail son fonctionnement.

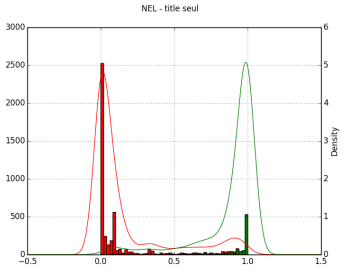
matcher Mise en correspondance de mots repérés préalablement avec le SIREN [1] de notre base de données de 9 millions d'entreprises françaises. Cette mise en correspondance est réalisée par une recherche floue (distance de Levenshtein ≤ 2) sur les formes minuscules et désaccentuées des noms d'entreprises de notre base. La recherche floue est implantée par un automate de Levenshtein [5] et l'indexation par une structure de trie [3]. Cette étape peut introduire une ambiguïté (même nom partagé par plusieurs SIREN), le score de mise en correspondance est réduit dans ce cas.

aggregator Fusionner les résultats des modules précédents du niveau mot au niveau article. Puisqu'il est fréquent que les noms d'entreprises soient à la fois répétés dans le corps de l'article et présent dans le titre, nous concevrons un système renforçant le score dans ces cas. Une première approche consiste simplement à retenir le meilleur score associé à chaque SIREN.

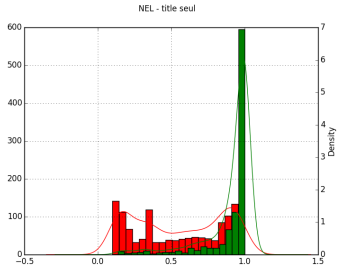
Repérage de la position des noms correspondant à des entreprises

Nous disposons de données de référence, plus de 200 000 articles avec au moins une entreprise française annotée manuellement. Nous pouvons donc employer une méthode d'apprentissage supervisé. Parmi les approches disponibles, nous avons évalué l'opportunité d'utiliser un modèle de perceptron multi-couches et un modèle de forêt d'arbres de décisions[2]. Ces approches sont connues pour donner des résultats pertinents pour la reconnaissance d'entités nommées.[6] Nous utilisons les mêmes caractéristiques que l'analyseur syntaxique *PerceptronTagger* de la bibliothèque NLTK[1], à savoir les suffixes, préfixes et formes de surface des mots adjacents. Une étude préliminaire a montré que les forêts d'arbres de décision obtiennent généralement de meilleurs résultats et sont moins sensibles au bruit présent dans notre corpus de référence.

1. Identifiant unique d'une entreprise française.



(a) Réponse du modèle



(b) Réponse du modèle $p > 0.1$

FIGURE 1: Réponse du module de classification binaire (échantillons positifs en vert, négatifs en rouge)

		Notre système	
		Nég.	Pos.
Réf.	Nég.	8 755	350
	Pos.	340	8 765

(a) Entraînement (seuil $p \geq 0,5$).

		Notre système	
		Nég.	Pos.
Réf.	Nég.	4 295	673
	Pos.	61	912

(b) Test (seuil $p \geq 0,5$).

FIGURE 2: Matrices de confusions.

KSB	présente	sa	vanne	en	acier	moulé
0,78	0,24	0,04	0,12	0,00	0,00	0,24

Finalement	Taittinger	renonce	à	la	présidentielle
0,38	0,9	0,16	0,02	0,08	0,02

FIGURE 3: Exemples de sorties du système provenant du corpus de test

Nous avons évalué la performance d'un modèle de forêt d'arbres de décision à l'aide d'un échantillon de 1 000 articles pour l'entraînement et 250 autres pour le test. Après application d'heuristiques, pour déduire l'annotation au niveau mot à partir de l'annotation au niveau article, nous avons extrait 9 105 échantillons positifs (mots inclus dans un nom d'entreprise) pour plus de 1,8 million d'échantillons négatifs. Nous avons rééquilibré le corpus d'entraînement en retenant autant d'échantillons négatifs que d'échantillons positifs. Finalement nous entraînons un système de 200 arbres de décision. [2] Avec un seuil à 0,5, la précision est de 0,58, le rappel est de 0,94 et le score f_1 de 0,71 (voir la matrice de confusion Fig. 2). La réponse du modèle nous permet d'arbitrer facilement entre précision et rappel en faisant varier le seuil.

Conclusions et perspectives

La modularisation du système initial permet de se concentrer plus efficacement sur chaque aspect du problème d'identification de noms d'entreprises dans les articles de presse. La première expérience sur le repérage de noms d'entreprises nous invite à continuer dans la direction de l'apprentissage supervisé à l'aide de forêts d'arbres de décision. En effet la réponse du modèle discrimine suffisamment les échantillons positifs des échantillons négatifs pour permettre d'arbitrer entre précision et rappel.

Le module *matcher* mérite également des évolutions, par exemple lever l'ambiguïté due à l'association du nom avec le SIREN. Dans le cas où plusieurs entreprises partagent le même nom ou des noms très similaires. Si on fait l'hypothèse qu'elles apparaissent dans des articles aux thèmes distincts (secteur d'activité, produit, localisation de l'entreprise, …), alors on peut exploiter des techniques de thématisation pour lever l'ambiguïté de ces noms.

Références

[1] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python: analyzing text with the natural language toolkit.* " O'Reilly Media, Inc.", 2009.

[2] Leo Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[3] Edward Fredkin. Trie memory. *Communications of the ACM*, 3(9):490–499, 1960.

[4] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[5] Klaus U Schulz and Stoyan Mihov. Fast string correction with levenshtein automata. *International Journal on Document Analysis and Recognition*, 5(1):67–85, 2002.

[6] René Speck and Axel-Cyrille Ngonga Ngomo. Ensemble learning for named entity recognition. In *International Semantic Web Conference*, pages 519–534. Springer, 2014.

2. Nous employons l'implantation *RandomForestClassifier* de Scikit-Learn[4]