

Quelles connaissances linguistiques permettent d'améliorer la classification de blogs avec les k-ppv ?

Nicolas Béchet, Inès Bayoudh

► **To cite this version:**

Nicolas Béchet, Inès Bayoudh. Quelles connaissances linguistiques permettent d'améliorer la classification de blogs avec les k-ppv ?. 2008, pp.63-72. lirmm-00336566

HAL Id: lirmm-00336566

<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00336566>

Submitted on 4 Nov 2008

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Quelles connaissances linguistiques permettent d'améliorer la classification de blogs avec les k-ppv ?

Nicolas Bechet*, Ines Bayouhd*,**

*Équipe TAL, LIRMM - UMR 5506, CNRS
Université Montpellier 2, 34392 Montpellier Cedex 5 - France

**INSAT, Université du 7 Novembre à Carthage
Centre Urbain Nord B.P. 676 Tunis Cedex 1080 - Tunisie

Résumé. Les blogs sont des sites web interactifs pouvant s'apparenter à un journal personnel, mis à jour régulièrement. De tels sites sont constitués d'articles pouvant être de thèmes distincts. Cette disparité pose le problème de recherche d'information dans ces nouvelles sources d'informations. Il est donc essentiel de proposer une classification thématique de ces articles. Ce papier propose d'évaluer différentes approches utilisant des connaissances linguistiques afin de classer automatiquement de tels articles issus de blogs avec l'utilisation de l'algorithme des k-ppv, répondant ainsi aux besoins des utilisateurs.

1 Introduction

Les travaux présentés dans cet article sont issus d'une collaboration avec la Société PaperBlog (<http://www.paperblog.fr/>). Cette Société héberge un site web proposant un référencement de blogs (ou weblog) issus de sites web partenaires. Les blogs s'apparentent à des sites web constitués d'articles souvent ordonnés chronologiquement ou ante-chronologiquement. Chaque article est écrit à la manière d'un journal de bord, pour lequel des commentaires peuvent être apportés. Ce nouveau type de site web, illustrant les concepts du web 2.0, s'est popularisé ces dernières années du fait de sa facilité de publication, de son interactivité et pour finir d'une grande liberté d'expression. Ce dernier point pose le problème de la recherche d'information dans de tels articles.

L'idée du site web de PaperBlog est de répondre à la question : comment trouver des articles d'une thématique précise issue de blogs ? Pour cela, les articles des blogs sont évalués suivant leur pertinence puis associés à une catégorie thématique (comme *culture*, *informatique*, *insolite* etc.). Cette approche permet de retrouver des informations d'une thématique précise contenues dans les blogs. L'objectif de nos travaux est d'apporter une méthode qui effectue cette classification thématique de manière automatique (celle-ci étant actuellement réalisée manuellement).

Pour cette tâche, nous avons choisi d'implémenter un algorithme classique de classification de données textuelles, les k plus proches voisins (k-ppv). Ce classificateur va tout d'abord être appliqué d'une manière standard, puis en l'associant à diverses approches utilisant des informations grammaticales. Nous pouvons ainsi évaluer les différentes représentations de données qui

s'appuient sur des connaissances linguistiques afin de déterminer quelles sont les plus adaptées. L'algorithme des k-ppv offre en effet de bons résultats. Cependant, un facteur essentiel pour optimiser ceux-ci est la qualité des données utilisées. Par nos différentes approches, nous proposons d'étudier en quelle mesure l'optimisation des données peut influencer sur la qualité des résultats. Pour cela, nous travaillons sur un corpus issu du site de PaperBlog, d'une taille de 3,4 Mo contenant 2520 articles et composé de plus de 400 000 mots. Celui-ci est réparti en cinq classes : alimentation, talents, people, cuisine et bourse.

La section suivante décrira les différents classificateurs – autres que les k-ppv – permettant de répondre à nos besoins. Nous présenterons dans la section 3 le classificateur choisi : les k-ppv et décrirons ensuite (section 4) les approches exploitant des informations grammaticales. Nous présenterons finalement les résultats obtenus (section 5).

2 Résumé de l'état de l'art sur la classification de texte

Le domaine de la classification automatique se compose de deux approches distinctes : la classification supervisée et la classification non supervisée. La distinction entre ces deux approches vient de la connaissance ou non des classes. En effet, pour une approche non supervisée, les classes sont définies de manière automatique (Cormack (1971); Johnson (1967)) alors qu'une approche supervisée part du principe que les classes sont connues, ayant été préalablement définies par un expert (Borko et Bernick (1963); Yang et Liu (1999)). Cette seconde approche est appelée catégorisation.

Nous proposons dans cet article de classer automatiquement des articles de blogs dans des classes définies au préalable. Nous disposons pour cela d'un ensemble d'articles classés manuellement par la Société PaperBlog, c'est donc naturellement que nous nous tournons vers une tâche de catégorisation automatique de textes. Cette Société souhaiterait en effet pouvoir classer les nouveaux articles de blogs de manière automatique en utilisant les connaissances provenant des articles précédemment classés manuellement. La classification de textes propose de regrouper des textes de thématiques proches dans un même ensemble appelé classe ou catégorie. Cette tâche induit la notion d'apprentissage dont deux principales approches sont définies : l'approche symbolique et l'approche numérique (se référer aux travaux de (Moulinier et al. (1996)) qui présentent un exemple d'apprentissage symbolique appliqué à la classification de textes). Nous nous intéressons dans nos travaux à l'approche numérique.

L'apprentissage consiste à construire un classificateur de manière automatique en "apprenant" les caractéristiques des exemples déjà classés. Le classificateur généré permet dès lors, avec l'ajout d'un nouvel objet, de déterminer sa catégorie d'appartenance. Nous proposons par la suite de présenter deux méthodes couramment utilisées pour des tâches de catégorisation.

– Les machines à vecteurs support (SVM).

Le principe des SVM défini par (Vapnik (1995)) suppose que l'on peut séparer linéairement l'espace de représentation des objets à classer. En d'autres termes, l'objectif est de trouver une surface linéaire de séparation, appelée hyperplan, maximisant la marge entre les exemples positifs et négatifs d'un corpus d'apprentissage. La distance séparant

les vecteurs les plus proches de l'hyperplan doit donc être maximale. Ces vecteurs sont appelés des vecteurs supports. Un nouvel objet est classé en fonction de sa position par rapport à l'hyperplan. Cette approche reste par ailleurs limitée par son caractère binaire. Il existe en effet des méthodes appliquant le concept des SVM sur des problèmes multi-classes mais ils supposent plusieurs étapes, en créant une nouvelle classification binaire pour chaque étape. L'ordre dans lequel les classes sont traitées influence ainsi les résultats du classificateur. Il s'avère également que la méthode SVM est plus coûteuse en temps d'apprentissage (Joachims (1998)) que les NaiveBayes ou k-ppv qui sont décrites plus loin. Les SVM donnent cependant de très bons résultats appliqués à une tâche de classification de textes (Lewis et al. (2004)). Une description détaillée des SVM est présentée par (Burges (1998)).

– **Les classificateurs bayésiens naïfs (NaiveBayes).**

Ces classificateurs se fondent sur le théorème de Bayes défini comme suit :

$$P(h|D) = \frac{P(D|h) \times P(h)}{P(D)} \text{ avec}$$

- $P(h|D)$ = probabilité de l'hypothèse h sachant D (probabilité *a posteriori*)
- $P(h)$ = probabilité que h soit vérifiée indépendamment des données D (probabilité *a priori*)
- $P(D)$ = probabilité d'observer des données D indépendamment de h
- $P(D|h)$ = probabilité d'observer des données D sachant que h est vérifiée.

Ce théorème repose sur l'hypothèse que des solutions recherchées peuvent être trouvées à partir de distributions de probabilité dans les hypothèses et dans les données. Un classificateur bayésien naïf, dans le cadre de la classification de textes, permet de déterminer la classe d'un document spécifié en supposant que les documents sont indépendants. Cette hypothèse d'indépendance ne reflète pas la réalité d'où l'appellation *naïf*. La classe la plus probable d'un nouvel objet est déterminée en combinant les prédictions de toutes les hypothèses en les pondérant par leurs probabilités *a priori*. Pour un ensemble de classes C et une instance spécifiée par un ensemble d'attributs A , la valeur de classification bayésienne naïve c est définie comme suit :

$$c = \operatorname{argmax}_{c_j \in C} \prod_{a_i \in A} P(a_i|c_j)$$

Ce classificateur s'est montré moins performant pour des tâches de classification de textes que d'autres méthodes (Weiss et al. (2005)). Il reste néanmoins capable de bien fonctionner avec des données incomplètes et peut être appliqué à de nombreux secteurs d'activités (juridique, médicale, économique, etc.). Cette approche est détaillée par (Cornuéjols et Miclet (2002)).

Ces deux méthodes sont couramment employés à des tâches de classification de textes comme (Chen et al. (2006)) qui présentent une comparaison de celles-ci (avec des commentaires d'opinions).

Classification automatique d'articles de blogs

Il existe d'autres approches permettant la catégorisation de texte. Citons les algorithmes fondés sur les arbres de décision ou DTree (Quinlan (1986)) comme les C4.5 (Quinlan (1993)). Le principe de construction de ces arbres consiste à déterminer des règles (termes) permettant de séparer des textes (pour une tâche de classification de textes) en fonction d'attributs communs.

Citons également les réseaux de neurones artificiels (NNet) dont l'idée est de simuler le fonctionnement des neurones humains (McCulloch et Pitts (1943)). Le principal défaut de cette approche est son temps de calcul considérable, compte tenu de sa dépendance d'un corpus d'apprentissage de taille conséquente.

Citons pour finir la méthode des k plus proches voisins (k-ppv ou k-NN). C'est cette méthode que nous avons utilisée dans notre approche. Les k-ppv sont en effet très simples à mettre en œuvre et permettent une implémentation rapide pour également fournir des résultats satisfaisants (Yang (1999)) ce qui a en partie motivé notre choix pour cet algorithme. De plus, cette méthode reste robuste sur des cas de données incomplètes, ce qui est assez fréquent pour des articles de blogs. Cette approche sera détaillée dans la section suivante.

3 L'algorithme des k plus proches voisins

Le principe de l'algorithme des k-ppv (Cover et Hart (1967)) est de mesurer la similarité entre un nouveau document et l'ensemble des documents ayant été préalablement classés. Ces documents peuvent être considérés comme un jeu d'apprentissage, bien qu'il n'y ait pas de réelle phase d'apprentissage avec les k-ppv.

Cet algorithme revient à constituer un espace vectoriel dans lequel chaque document est modélisé par un vecteur de mots. Un tel vecteur a pour dimension le nombre de mots de la base d'apprentissage. Chaque élément de ce vecteur est en effet constitué du nombre d'occurrences d'un mot issu de la base d'apprentissage. Les documents classés sont ordonnés de manière décroissante afin que le premier document soit celui ayant obtenu le meilleur score de similarité avec le document devant être classé. Suivant la valeur de k , il est ainsi effectué un classement des k documents les plus proches. La mesure de similarité la plus couramment utilisée est le calcul du cosinus de l'angle formé par les deux vecteurs de documents. Le cosinus entre deux vecteurs A et B vaut le produit scalaire de ces vecteurs A et B divisés par le produit de la norme de A et de B.

Après avoir déterminé quels étaient les k plus proches voisins, il faut définir une méthodologie afin d'attribuer une classe au nouveau document. Il existe dans la littérature deux approches classiques décrites spécifiquement dans (Bergo (2001)) afin de répondre à cette problématique :

- soit proposer de classer le document dans la même catégorie que celui ayant obtenu le meilleur score de similarité parmi le jeu d'apprentissage,
- soit, si $k > 1$ de considérer les k documents les mieux classés. Alors nous pouvons attribuer la classe suivant plusieurs options. Une première méthode peut être de calculer parmi les k documents les plus proches, pour chaque catégorie, le nombre de documents appartenant à cette catégorie (1). Une seconde propose de prendre en compte le rang des k documents (2). Il s'agit pour toutes les catégories, d'effectuer la somme des occurrences d'une catégorie multipliée par l'inverse de son rang.

Prenons par exemple un document d à classer parmi quatre classes, C1, C2, C3 et C4. Définissons $k = 6$. Considérons le classement suivant de d_{new} avec le jeu d'apprentissage D contenant les documents d_i :

documents	classe des documents	rang
d1	C2	1
d2	C2	2
d3	C4	3
d4	C4	4
d5	C1	5
d6	C4	6

En utilisant la première approche (1), nous aurions attribué la classe C4 à d_{new} . En effet la classe C4 est celle qui possède le plus de documents parmi les k plus proches voisins (trois documents). La seconde approche (2) aurait quant à elle classé d_{new} dans C2. Nous obtenons en effet avec cette mesure par exemple pour la classe C1 : un seul document dans la classe au cinquième rang soit $C1 = 1/5 = 0,2$. Nous obtenons pour les autres classes $C2 = 1,5$, $C3 = 0$ et $C4 = 0,75$.

Nous utiliserons dans nos expérimentations la première approche (1), celle-ci étant la forme la plus répandue comme décrite dans (Yang et Liu (1999)) en utilisant deux paramètres :

- le seuil de classe, fixant un nombre minimal de termes devant appartenir à une classe pour qu'un nouveau document soit attribué à cette classe,
- le seuil de similarité en dessous duquel, les candidats ne seront plus admis parmi les k plus proches voisins car étant jugés d'une similarité trop éloignée.

4 Les différentes approches utilisées

Nous proposons dans ce papier des approches constituant des nouvelles représentations du corpus original en utilisant des connaissances grammaticales. Afin d'obtenir de telles connaissances, nous utilisons un étiqueteur grammatical.

4.1 L'étiqueteur grammatical TreeTagger

Nous avons fait le choix de l'étiqueteur grammatical TreeTagger (Schmid (1995)), qui permet d'étiqueter des textes dans plusieurs langues dont le français. Il utilise des probabilités conditionnelles d'apparition d'un terme en fonction des termes précédents. Les probabilités sont construites à partir d'un ensemble de tri-grammes (constitués de trois étiquettes grammaticales consécutives). Le TreeTagger propose par exemple les résultats suivants pour la phrase : *Les étiquettes grammaticales apportent une information supplémentaire.*

Classification automatique d'articles de blogs

Les	DET :ART	le
étiquettes	NOM	étiquette
grammaticales	ADJ	grammatical
apportent	VER :pres	apporter
une	DET :ART	un
information	NOM	information
supplémentaire	ADJ	supplémentaire
.	SENT	.

La première colonne correspond au terme traité (forme fléchie), la seconde nous renseigne sur la catégorie grammaticale de ce terme et la dernière nous donne sa forme lemmatisée. Nous proposons, avec ses informations, différentes approches présentées dans la section suivante.

4.2 Les méthodes expérimentales

Nous proposons d'utiliser des combinaisons de mots avec les catégories *Nom* (noté N), *Verbe* (V) et *Adjectif* (A). Cette approche consiste à reconstituer un corpus ne contenant que les mots appartenant à la combinaison définie. Prenons par exemple la combinaison V_N. Le corpus reconstitué ne contiendra que des verbes et des noms. Nous noterons ces méthodes M1, M2, ..., M7 pour les combinaisons N, V, A, N_V, N_A, V_A et N_V_A. Nous définissons également les méthodes F et L pour respectivement le corpus avec des formes fléchies et le corpus sous forme lemmatisée¹.

La section suivante propose de présenter le protocole d'évaluation suivi et les résultats obtenus avec nos différentes approches.

5 Experimentations

Afin de mener nos expérimentations, nous proposons de comparer les performances de l'algorithme des k-ppv en utilisant diverses méthodes. Nous utiliserons les appellations définies dans la section 4.2. Cette évaluation comprend plusieurs étapes :

- Suppression des balises html et des mots outils (mots génériques revenant souvent dans le texte comme "donc", "certain", etc.) du corpus.
- Application d'une des méthodes présentées.
- Application d'un processus de validation croisée en segmentant les données en cinq sous-ensembles et utilisation des k-ppv pour catégoriser les articles.
- Obtention d'une matrice de confusion et calcul du taux d'erreur.

Le taux d'erreur, permettant de mesurer le taux d'articles mal classés, est défini ainsi :

$$\text{taux d'erreur} = \frac{\text{nombre d'articles mal classés}}{\text{nombre total d'articles}}$$

Nous définissons également le Tf-Idf (Term Frequency x Inverse Document Frequency) qui servira à réaliser une normalisation de nos données lors de nos expérimentations : $W_{ij} = tf_{ij} \cdot \log_2(N/n)$ avec :

¹La forme lemmatisée du corpus a été obtenue avec le TreeTagger

- w_{ij} = poids du terme T_j dans le document D_i ,
- tf_{ij} = fréquence du terme T_j dans le document D_i ,
- N = nombre de documents dans la collection,
- n = nombre de documents où le terme T_j apparaît au moins une fois.

Nous utilisons, dans le cadre de l'application des k-ppv, une valeur de 2 pour le seuil de classe et de 0.2 pour le seuil de similarité, valeurs jugées comme les plus appropriées à nos travaux de manière expérimentale. Rappelons que ces mesures peuvent impliquer que certains articles soient considérés comme non classés.

Nous proposons tout d'abord de mesurer l'apport d'une normalisation (le Tf-Idf) et d'une lemmatisation sur notre corpus en utilisant les approches L (forme lemmatisée) et F (forme fléchie). Le tableau 1 présente le taux d'erreur obtenu avec l'application de ces approches. Il montre que la lemmatisation du corpus a tendance à dégrader les résultats en termes de taux d'erreur. Cependant, en appliquant le Tf-Idf, cette tendance s'inverse avec de meilleurs résultats pour la forme lemmatisée (méthode L), cette approche obtenant le plus faible taux d'erreur de ce tableau. Les tableaux 2 et 3 présentent les matrices de confusions obtenues en utilisant

TAB. 1 – *Tableau évaluant l'apport de la normalisation et de la lemmatisation*

l'approche L et F avec le Tf-Idf dont les taux d'erreurs du tableau 1 sont issus. Ces tableaux montrent que l'approche utilisant les lemmes est meilleure que celle conservant les formes fléchies pour toutes les classes exceptée la classe *alimentation*. Nous constatons de plus que le nombre d'articles non classés est significativement plus important pour la méthode F (135 pour la méthode L contre 256 pour la méthode F). Ces résultats s'expliquent par le fait qu'une lemmatisation lève certaines ambiguïtés pouvant par conséquent influencer le classement établi par l'algorithme des k-ppv.

TAB. 2 – *Matrice de confusion obtenue en utilisant l'approche L avec normalisation*

TAB. 3 – *Matrice de confusion obtenue en utilisant l'approche F avec normalisation*

Nous comparons ensuite dans le tableau 4 la méthode L avec normalisation, ayant obtenu le plus faible taux d'erreur, avec les méthodes M1 à M7 décrites dans la section 4.2. Nous constatons tout d'abord que sans l'utilisation du Tf-Idf, les méthodes M1, M4, M5, M6 et M7 réduisent le taux d'erreur obtenu par la méthode L, la méthode M4 minimisant ce taux. Nous montrons par ces résultats que les mots sélectionnés par ces méthodes sont plus porteurs de

sens que ceux sélectionnés par les autres méthodes. Les méthodes M2 et M3, respectivement les méthodes contenant les verbes et les adjectifs, possèdent en effet moins d'informations que les noms (M1) ou les diverses combinaisons de catégories grammaticales (M4 à M7). La méthode M4 (les noms et les verbes) confirme ces bons résultats en tenant compte de l'application du Tf-Idf. Elle ne parvient cependant qu'à égaler la méthode L, là où toutes les autres approches augmentent le taux d'erreur. Ces expérimentations montrent que les verbes et les adjectifs contiennent moins d'informations utiles comparé aux noms. Elles permettent aussi de montrer que l'association des noms_verbes, verbes_adjectifs et noms_verbes_adjectifs se révèlent être assez équivalente en termes d'informations alors que l'association noms_verbes permet une classification plus fine.

TAB. 4 – Tableau évaluant l'utilisation d'outils grammaticaux

6 Conclusions

Nous avons présenté dans cet article une catégorisation automatique d'articles de blogs afin de répondre aux besoins de la Société PaperBlog dans ce domaine. Nous avons pour cela utilisé l'algorithme des k plus proches voisins que nous avons confronté à diverses approches utilisant des informations grammaticales.

Celles-ci ont montré des résultats satisfaisants sans effectuer de normalisation, mais restent limitées dans le cas contraire. Nous avons également identifié quelles catégories grammaticales étaient les plus porteuses de sens. Cela nous permet d'envisager de futures approches permettant d'effectuer des pondérations suivant les catégories grammaticales. Nous avons par exemple établi que les noms sont les plus porteurs de sens et pourraient se voir attribuer un poids plus important dans le cadre de l'utilisation de l'approche des k-ppv. Nous envisageons pour finir d'expérimenter nos méthodes avec d'autres méthodes de catégorisations.

Remerciements

Nous remercions la Société PaperBlog, en particulier Nicolas Verdier et Maxime Biais, pour la participation à ces travaux ainsi que pour le partage des données qui ont pu être expérimentées.

Références

- Bergo, A. (2001). Text categorization and prototypes. Technical report.
- Borko, H. et M. Bernick (1963). Automatic document classification. *J. ACM* 10(2), 151–162.
- Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery* 2(2), 121–167.

- Chen, C., F. Ibekwe-SanJuan, E. SanJuan, et C. Weaver (2006). Visual analysis of conflicting opinions. *vast 0*, 59–66.
- Cormack, R. M. (1971). A review of classification (with discussion). *the Royal Statistical Society 3*, 321–367.
- Cornuéjols, A. et L. Miclet (2002). *Apprentissage artificiel, Concepts et algorithmes*. Eyrolles.
- Cover, T. et P. Hart (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on 13*(1), 21–27.
- Joachims, T. (1998). Text categorization with support vector machines : learning with many relevant features. In *Proc. 10th European Conference on Machine Learning ECML-98*, pp. 137–142.
- Johnson, S. C. (1967). Hierarchical clustering schemes. *Psychometrika 32*, 241–254.
- Lewis, D. D., Y. Yang, T. G. Rose, et F. Li (2004). Rcv1 : A new benchmark collection for text categorization research. *Journal of Machine Learning Research 5*(Apr), 361–397.
- Mcculloch, W. et W. Pitts (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics 5*, 115–133.
- Moulinier, I., G. Raskinis, et J. Ganascia (1996). Text categorization : a symbolic approach. In *In Proceedings of the Fifth Annual Symposium on Document Analysis and Information Retrieval*, pp. 87–99.
- Quinlan, J. R. (1986). Induction of decision trees. *Mach. Learn. 1*(1), 81–106.
- Quinlan, J. R. (1993). *C4.5 : programs for machine learning*. San Francisco, CA, USA : Morgan Kaufmann Publishers Inc.
- Schmid, H. (1995). Improvements in part-of-speech tagging with an application to german. In *Proceedings of the ACL SIGDAT-Workshop, Dublin*.
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*. Springer, N.Y.
- Weiss, S. M., N. Indurkha, T. Zhang, et F. Damerau (2005). *Text Mining : Predictive Methods for Analyzing Unstructured Information*. Springer.
- Yang, Y. (1999). An Evaluation of Statistical Approaches to Text Categorization. *Information Retrieval 1*(1-2), 69–90.
- Yang, Y. et X. Liu (1999). A re-examination of text categorization methods. In *SIGIR '99 : Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, New York, NY, USA, pp. 42–49. ACM Press.

Summary

Weblogs are interactive and regularly updated websites which can be seen as diaries. These websites are composed by articles based on distinct topics. Thus, it is necessary to develop Information Retrieval approaches for this new web knowledge. The first important step of this process is the categorization of the articles. The paper above compares several methods using linguistic knowledge with k-NN algorithm for automatic categorization of weblogs articles.