DEA MVA 2006-2007

Introduction aux modèles graphiques

# Classification de textes par les algorithmes de Bayes naïfs

Aurélien Boffy

janvier 2007

## 1 Présentation du problème

### 1.1 La classification supervisée de textes

L'objectif de la classification automatique de textes est d'apprendre à une machine à attribuer à un document $d$ une catégorie $c_i$ où $i \in \{1 \dots K\}$, en examinant son contenu. Un outil comme *Google News* doit ainsi par exemple être capable de dire si un article traite de politique, de sport ou de sciences. Il existe de multiples autres applications, comme la détection du spam ou l'identification automatique de la langue d'un document.

Pour ce faire, on dispose d'une base de données de documents $(d_i)_{1 \leq i \leq N}$ qui ont déjà été classés manuellement (c'est la raison pour laquelle on parle de classification supervisée).

De nombreuses méthodes d'apprentissage automatique ont été utilisées ces dernières années, comme les arbres de décision, les réseaux de neurones ou les SVM.

Dans leur article de 1998, *A comparison of event models for Naive Bayes text classification*, A. McCallum et K. Nigam rappellent le principe de l'algorithme de Bayes naïf, qui est un des algorithmes qui a été le plus utilisé, certainement car il est très efficace pour de nombreuses tâches de Data Mining. Les auteurs comparent notamment deux variantes de l'algorithme que nous allons détailler.

### 1.2 Les hypothèses de l'algorithme de Bayes naïf

L'algorithme de Bayes naïf présuppose qu'il existe un modèle génératif de textes : les documents sont générés par un modèle de mélange, dont les composantes (i.e. les variables cachées) sont les catégories de documents. Ainsi, dans un premier temps, la classe du document est choisie, puis le texte est généré en choisissant les mots qui le composent dans un dictionnaire $D = \{m_1 \dots m_M\}$ composé de $M$ mots, avec des paramètres qui diffèrent selon la classe de documents qui a été sélectionnée.

L'algorithme est dit « naïf » car il s'appuie sur une hypothèse forte : la présence d'un mot dans un texte est indépendante de la présence des autres mots, c'est-à-dire par exemple que la présence du mot « football » dans un article n'influe pas sur la présence du mot « ballon » ou du mot « marguerite », ce qui est clairement faux dans les faits.

Notons aussi que la position des mots dans le texte et leur agencement les uns par rapport aux autres ne sont absolument pas pris en compte.

## 2 Deux algorithmes de Bayes naïfs distincts

Lorsqu'on parle d'algorithmes de Bayes naïfs pour la classification de texte, il faut faire attention au fait qu'il en existe en fait deux variantes, selon le modèle génératif de textes que l'on utilise.

### 2.1 Modèle de Bernoulli multivarié

Lorsqu'un document est généré par le modèle de Bernoulli multivarié, il est en fait caractérisé par la présence ou l'absence de chacun des mots $m_i$ du dictionnaire : un document peut donc être caractérisé par un vecteur binaire $b = (b_1, \dots, b_M)$ (où l'on rappelle que $M$ est le nombre de mots du dictionnaire), avec $b_i = 1$ si le mot $m_i$ est présent dans le texte et $b_i = 0$ sinon.

### 2.2 Modèle multinomial

On note que dans le modèle précédent, le nombre d'occurrences d'un mot donné n'intervient pas. Ainsi, un texte où le mot « football » apparaît 20 fois est traité exactement de la même façon qu'un texte où ce mot n'apparaît qu'une fois. Le modèle multinomial s'attache à intégrer cette information, et caractérise donc un document par les mots qui le composent, ainsi que par leur nombre d'occurrences.

# 3 Classification avec l'algorithme de Bayes naïf

Fort logiquement, l'algorithme de Bayes naïf est basé sur la loi de Bayes pour classifier un document. Ainsi, la probabilité $p(c_i|d)$ qu'un document $d$ appartienne à la classe $c_i$ est :

$$p(c_i|d) = \frac{p(c_i)p(d|c_i)}{p(d)} = \frac{p(c_i)p(d|c_i)}{\sum_{j=1}^{K} p(c_j)p(d|c_j)}$$

Les $p(c_i)$ sont facilement estimés lors de la phase d'apprentissage en calculant simplement la fréquence d'apparition de chaque classe. Reste donc à savoir comment calculer les membres $p(d|c_i)$ qui représentent la probabilité de générer le document $d$ considéré lorsqu'on est dans la classe $c_i$. C'est ici que les deux modèles diffèrent :

## 3.1 Modèle de Bernoulli multivarié

Dans le cas du modèle de Bernoulli multivarié où seule compte la présence ou l'absence dans le texte de chaque mot du dictionnaire, la probabilité de générer un document $d$ correspond à la probabilité de générer le vecteur binaire associé $b = (b_1, \ldots, b_M)$ :

$$\begin{aligned} p(d|c_i) &= \prod_{j=1}^{M} p(b_j|c_i) \\ &= \prod_{j=1}^{M} \Big[\delta(b_j = 1)p(m_j|c_i) + \delta(b_j = 0)\Big(1 - p(m_j|c_i)\Big)\Big] \end{aligned}$$

où la probabilité $p(m_j|c_i)$ qu'un mot donné $m_j$ apparaisse dans un texte d'une catégorie donnée $c_i$ est estimée durant la phase d'apprentissage en calculant simplement la proportion de documents de la classe $c_i$ qui contiennent le mot $m_j$ (en évitant grâce à une astuce d'avoir des probabilités nulles, car elles annuleraient tout le produit).

## 3.2 Modèle multinomial

Avec le modèle multinomial, un document est généré en tirant dans un premier temps une longueur de texte $|d|$, puis en tirant de façon indépendante $|d|$ mots dans le dictionnaire, chaque mot ayant bien-sûr une probabilité différente d'être choisie. Comme le nombre d'occurrences est pris en compte, on autorise naturellement de tirer plusieurs fois le même mot. On reconnaît ainsi la réalisation d'un tirage avec remise qui se fait selon une loi multinomiale : la probabilité de générer le texte $d$ lorsque la classe est $c_i$ est donc :

$$p(d|c_i) = p(|d|)|d|! \prod_{j=1}^{M} \frac{p(m_j|c_i)^{N_j}}{N_j!}$$

où :
- $p(|d|)$ est la probabilité de générer un document de longueur $|d|$ et est calculée facilement pendant la phase d'apprentissage
- $N_j$ est le nombre d'occurrences du mot $m_j$ dans le document $d$
- comme précédemment, les probabilités $p(m_j|c_i)$ sont estimées pendant la phase d'apprentissage, en calculant cette fois-ci la proportion du mot $m_j$ parmi tous les autres mots du dictionnaire dans les documents de la classe $c_i$.

# 4 Création d'un « bon » dictionnaire

Le choix d'un « bon » dictionnaire est primordial dans la classification automatique de documents. En effet, selon le nombre de mots qui le composent, les résultats peuvent varier très significativement et certaines méthodes de classification marcheront mieux avec de petits dictionnaires alors que d'autres seront plus efficaces avec des gros (nous aurons un aperçu de ceci dans la partie sur les résultats expérimentaux).

La taille du dictionnaire doit aussi dépendre des catégories que l'on recherche : si tous les documents d'une catégorie donnée concerne un même sujet restreint avec un vocabulaire limité, les résultats seront meilleurs avec un petit dictionnaire, alors que si une même catégorie peut contenir divers thèmes avec des vocabulaires variés, il conviendra d'utiliser un dictionnaire plus conséquent.

De plus, si l'on ne s'attache pas à réduire la taille du dictionnaire et que l'on considère tous les mots présents dans la base de données d'apprentissage, la dimension de l'espace dans lequel on travaille est très élevée et les algorithmes peuvent devenir particulièrement gourmands en mémoire et en temps de calcul.

Il est important de savoir réduire la taille du dictionnaire de façon intelligente. Il existe différentes techniques :

**Suppression des mots vides de sens (*stop words*) :** la première étape consiste généralement à supprimer tous les mots qui ne contiennent aucune information sémantique, souvent des pronoms, des articles, des adverbes, etc. Notons néanmoins qu'il convient toujours de faire attention car ces mots peuvent parfois avoir un certain intérêt selon les catégories de documents que l'on recherche. Par exemple, si l'on souhaite rechercher des blogs sur internet, les mots tels que « je » ou « moi » peuvent être intéressants.

**Recherche de radical (*stemming*) :** cette technique permet de réduire la taille du vocabulaire en regroupant les mots qui ont la même racine sémantique (« étudier », « études », « étudiant », « étudions », ...). Il existe pour ce faire des algorithmes connus, comme la méthode de Porter.

**Sélection grâce à l'information mutuelle :** Aux deux méthodes précédentes, des algorithmes plus évolués, généralement issus de la théorie de l'information, sont utilisés pour ne conserver que les mots ayant un vrai pouvoir discriminant. L'*information mutuelle* est souvent utilisée pour déterminer quels mots sont particulièrement caractéristiques d'une ou de plusieurs catégories. Pour chaque mot, on regarde s'il est corrélé avec les classes : si c'est le cas, cela signifie qu'il est porteur de sens et donc qu'il faut le conserver. Plus précisément, si l'entropie de la distribution des classes est significativement diminuée lorsqu'on sait si le mot est présent ou non, cela signifie que le mot est porteur d'information et qu'il doit être conservé. On calcule donc pour chaque mot $m_i$ l'information mutuelle :

$$\begin{aligned} IM(C, m_i) &= H(C) - H(C|m_i) \\ &= -\sum_{j=1}^{K} p(c_j)\log(p(c_j)) + \sum_{f_i \in \{0;1\}} p(f_i) \sum_{j=1}^{K} p(c_j|f_i)\log(p(c_j|f_i)) \\ &= \sum_{j=1}^{K} \sum_{f_i \in \{0;1\}} p(c_j, f_i) \log\left(\frac{p(c_j, f_i)}{p(c_j)p(f_i)}\right) \end{aligned}$$

où $f_i = 1$ quand le mot $m_i$ est présent et $f_i = 0$ sinon, et où toutes les probabilités sont calculées en prenant en compte tous les documents de la base de données d'apprentissage.

# 5 Expériences

## 5.1 Protocole expérimental

Il est aujourd'hui assez facile grâce à Internet et aux annuaires de sites de se procurer de grosses bases de données d'apprentissage dont les documents ont été classés manuellement. Après avoir procédé à la construction du dictionnaire avec les méthodes décrites plus haut et à la phase d'apprentissage en utilisant une partie des données, on utilise le reste des documents pour tester les algorithmes.

Il existe alors différents critères pour estimer la performance des méthodes. On utilise le plus souvent l'*exactitude* qui est une mesure du nombre de documents bien classés sur le nombre de documents total. Lorsque l'on considère des classifications binaires où le but est alors de déterminer si un document appartient ou non à une catégorie précise, on utilise plutôt d'autres mesures statistiques comme la *précision*, qui ne considère que les textes que l'on a choisi d'assigner à la catégorie et qui correspond à la proportion de ces textes qui y appartiennent effectivement, ou encore le *rappel* qui, parmi tous les textes de la catégorie, renvoie la proportion de ceux qui y ont vraiment été assignés. On peut aussi calculer le « break-even point » qui est le point

où la précision et le rappel sont égaux. Plus ce point se rapproche de 100 %, plus le classificateur est performant.

### 5.2 Résultats

Comme nous l'avons évoqué plus haut, certains algorithmes fonctionnent mieux avec de petits dictionnaires alors que d'autres démontrent de meilleurs résultats lorsque le vocabulaire est plus grand. C'est exactement le cas lorsqu'on compare le modèle de Bernoulli multivarié et le modèle multinomial. Pour la plupart des bases de données considérées dans l'article de McCallum et Nigam, le premier atteint une exactitude maximale lorsque le nombre de mots du dictionnaire est de l'ordre de 100 ou 200, alors que le second fonctionne mieux lorsque le nombre de mots se compte en milliers voire en dizaines de milliers.

D'une façon générale, le maximum d'exactitude atteint est plus élevé (souvent de 20 à 50 % supérieur) avec le modèle multinomial où le nombre d'occurrences des mots est pris en compte. Les « break-even points » sont aussi généralement plus grands (d'environ 5 %).

### 5.3 Analyse des résultats

Deux raisons principales peuvent expliquer la supériorité du modèle multinomial sur le modèle de Bernoulli multivarié :

- L'explication la plus évidente est bien-sûr le fait que prendre en compte le nombre d'occurrences des mots ne peut être que bénéfique. À titre d'exemple, si l'on souhaite trouver les articles d'actualité qui parlent de finance, la présence de nombreux chiffres pourraient être un indice très bénéfique. Cependant, si l'on ne considère que la présence ou l'absence de chiffres dans le texte, il suffit que chacun des articles soit daté pour que l'information « présence d'un chiffre » perde tout son pouvoir discriminant.
- Un autre point noir du modèle de Bernoulli multivarié est sa dépendance avec la taille des documents. Tous les mots ont bien évidemment une probabilité plus élevé d'apparaître lorsque le texte est plus long, et ceci peut fausser tous les résultats. Comme le modèle multinomial ne considère finalement que des proportions car chaque nouveau mot est pris en compte dans les probabilités, il est relativement indépendant de la taille des documents.

# 6 Conclusion

Nous avons expliqué la problématique de la classification de documents et présenté une méthode parmi d'autres, l'algorithme de Bayes naïf. Cette méthode fut introduite pour la classification de textes à la fin des années 1990 et a aujourd'hui été surpassée par d'autres méthodes (en lisant la littérature sur le sujet, il semble que les machines à vecteurs de support (SVM) et l'algorithme des k plus proches voisins soient les techniques qui donnent les meilleurs résultats (cf. *An Evaluation of Statistical Approaches to Text Categorization* de Yang et Liu (Sigir'99) et l'article de Kim *et al.*, *Some Effective Techniques for Naive Bayes Text Classification* (TKDE'06)).

L'algorithme de Bayes naïf est néanmoins toujours très utilisé car il est bien connu et simple à mettre en œuvre. De plus, il permet d'utiliser sans peine des vocabulaires de très grande taille, alors que des méthodes comme SVM n'y sont pas adaptées.

Enfin, la présentation d'une méthode comme celle-là permet de mettre le doigt sur les différentes problématiques qui apparaissent lors de la classification automatique de documents, comme la construction d'un bon dictionnaire ayant une taille optimale, les problèmes liés à la longueur relative des textes, etc.