Module - Apprentissage

Master Informatique, mention PRIM, 2005-2006

8 mars 2006

1 Analyse de texte

1.1 Position du problème

Un trigramme est une succession de trois lettres. Par exemple `abc`, `dek`, `eek` sont des trigrammes. Lorsqu'on se donne un mot, on peut identifier les trigrammes qui le constituent. Par exemple, le mot `examen` est constitué des 4 trigrammes `exa`, `xam`, `ame`, `men`.

Pour analyser un texte, on décide d'éliminer tous les articles et les mots de liaison, pour ne garder que les mots vraiment significatifs. L'idée est alors de compter, dans ce texte, le nombre de fois que chacun des trigrammes possibles apparaît (par exemple, disons que pour ce texte, le trigramme `aaa` apparaît 0 fois, le trigramme `aab` apparaît 0 fois, ..., le trigramme `lle` apparaît 43 fois, ..., le trigramme `ous` apparaît 22 fois, ... le trigramme `zzy` apparaît 1 fois, le trigramme `zzz` apparaît 0 fois).

On souhaite représenter un texte par un vecteur, dont la i-ème composante est le nombre d'occurrence du i-ème trigramme possible (la composante 1 est le nombre de `aaa` du texte, la composante 2 le nombre de `aab`, la 3 le nombre de `aac`, ... jusqu'à la dernière composante, le nombre de `zzz`).

Question 1.1 : Quelle est la dimension de l'espace vectoriel dans lequel on représente les textes ?

On supposera que le sens du texte influence la distribution des trigrammes issus de ce texte. Ainsi, deux textes de 500 mots parlant de l'entretien d'un jardin devraient avoir des vecteurs associés similaires.

Question 1.2 : Le codage que nous avons choisi est il influencé par la longueur du texte ? Si oui, proposer un traitement des vecteurs pour y remédier, si non, justifiez par un exemple de texte court, moyen, et très long.

1.2 Articles de journaux sur une semaine

Toutes les semaines, on regroupe dans notre base tous les articles des grands quotidiens français, ce qui fournit une collection de vecteurs (un par article, suivant la procédure décrite précédemment). Bien sûr, sachant que le nombre de sujets d'actualité est relativement faible, plusieurs de ces textes parlent de la même chose, et doivent donc avoir des vecteurs associés relativement proches.

Question 1.3 : En quoi un apprentissage non-supervisé permettrait d'isoler les sujets prédominants de la semaine ? Illustrez votre propos en considérant la technique des k-means.
Question 1.4 : Déclinez les grandes étapes communes à toutes les techniques d'apprentissage non supervisé.
Question 1.5 : A quoi correspondent, dans notre problème, les vecteurs prototypes ?
Question 1.6 : Qu'est-ce que la notion de topologie dans le cadre de l'apprentissage non supervisé ? Qu'est-ce que la topologie représente dans notre problème ?
Question 1.7 : On souhaite présenter les informations relatives aux différents sujets de la semaine sur un écran. Quel algorithme d'apprentissage non supervisé vous semble particulièrement adapté ? Quel avantage y a-t-il à l'utiliser ?

Dans les journaux, il y a toujours des articles de remplissage, qui ne concernent pas directement l'actualité, mais un sujet spécifique, qui aurait très bien pu être proposé à un autre moment.
Question 1.8 : Comment repérer ces articles, une fois que l'apprentissage non-supervisé est réalisé ?

1.3 Articles de journaux de la semaine suivante

La semaine suivante, on souhaite faire la même analyse.
Question 1.9 : Comment peut-on injecter dans notre traitement la connaissance issue de la semaine dernière ? Est-ce pertinent ?

1.4 Suivi de l'information

Il est un peu arbitraire de découper l'analyse semaine par semaine, et l'on se propose de faire un suivi « en continu » de l'information. Pour ce faire, on alimente quotidiennement un algorithme d'apprentissage non supervisé avec les articles du jour (que l'on supposera très nombreux, de l'ordre de 1000). On souhaite que chaque article soit **un seul** exemple, présenté **une seule fois** à l'algorithme. L'algorithme est ainsi nourri par un flux d'articles, et l'état courant des prototypes devrait nous donner une image instantanée des sujets d'actualité.
Question 1.10 : Peut-on utiliser l'algorithme de Kohonen dans ce cas ? Si oui, justifiez, si non, proposer une modification pour que cela soit possible.

On se propose d'utiliser l'algorithme Growing Neural Gas.
Question 1.11 : Décrivez le principe (pas le détail) de cet algorithme, et précisez comment vous l'adapteriez à notre problème (distances, condition d'arrêt, etc...).

1.5 Détection de sujet

On souhaite maintenant détecter les textes traitant de la violence à l'école. On construit une base de 1 million d'articles, dont 500000 parlent de violence à l'école, alors que les autres non. Sachant quels sont les textes qui parlement de violence à l'école, et quels sont ceux qui n'en parle pas, on se propose d'utiliser une machine à vecteur support (SVM) pour apprendre à reconnaître les textes

qui parlement de violence à l'école. On associe à un texte un vecteur, comme précédemment à l'aide des trigrammes.

Question 1.12 : Qu'est-ce que la marge d'un séparateur linéaire ? En quoi ce concept est central pour les SVM ?

Question 1.13 : Qu'est-ce qu'un noyau pour une SVM ? En quoi changer de noyau influence-t-il les résultats de la SVM ?

Question 1.14 : Dans notre cas, justifiez mathématiquement que l'on peut utiliser comme noyau $K(x, y) = \sum_i x_i y_i$, où z_i désigne la i-ième composante du vecteur z. Les vecteurs considérés sont bien entendu ceux construits à partir des trigrammes.

Question 1.15 : Que pensez-vous de la répartition des 1000000 de textes dans l'espace des vecteurs ? Sont ils regroupés de façon compacte, occupent-ils cet espace de façon dense ? Les textes parlant de violence à l'école et les autres ont-il une chance d'être linéairement séparables ?

Question 1.16 : Pensez vous que le noyau K des questions précédente est adapté ? Justifiez.